Logical Induction

Scott Garrabrant, Tsvi Benson-Tilsen, Andrew Critch, Nate Soares, and **Jessica Taylor**
{scott,tsvi,critch,nate,jessica}@intelligence.org
Machine Intelligence Research Institute

Abstract

We present a computable algorithm that assigns probabilities to every logical statement in a given formal language, and refines those probabilities over time. For instance, if the language is Peano arithmetic, it assigns probabilities to all arithmetical statements, including claims about the twin prime conjecture, the outputs of long-running computations, and its own probabilities. We show that our algorithm, an instance of what we call a *logical inductor*, satisfies a number of intuitive desiderata, including: (1) it learns to predict patterns of truth and falsehood in logical statements, often long before having the resources to evaluate the statements, so long as the patterns can be written down in polynomial time; (2) it learns to use appropriate statistical summaries to predict sequences of statements whose truth values appear pseudorandom; and (3) it learns to have accurate beliefs about its own current beliefs, in a manner that avoids the standard paradoxes of self-reference. For example, if a given computer program only ever produces outputs in a certain range, a logical inductor learns this fact in a timely manner; and if late digits in the decimal expansion of π are difficult to predict, then a logical inductor learns to assign $\approx 10\%$ probability to "the nth digit of π is a 7" for large n. Logical inductors also learn to trust their future beliefs more than their current beliefs, and their beliefs are coherent in the limit (whenever $\phi \to \psi$, $\mathbb{P}_\infty(\phi) \le \mathbb{P}_\infty(\psi)$, and so on); and logical inductors strictly dominate the universal semimeasure in the limit.

These properties and many others all follow from a single *logical induction criterion*, which is motivated by a series of stock trading analogies. Roughly speaking, each logical sentence ϕ is associated with a stock that is worth \$1 per share if ϕ is true and nothing otherwise, and we interpret the belief-state of a logically uncertain reasoner as a set of market prices, where $\mathbb{P}_n(\phi) = 50\%$ means that on day n, shares of ϕ may be bought or sold from the reasoner for 50¢. The logical induction criterion says (very roughly) that there should not be any polynomial-time computable trading strategy with finite risk tolerance that earns unbounded profits in that market over time. This criterion bears strong resemblance to the "no Dutch book" criteria that support both expected utility theory (von Neumann and Morgenstern 1944) and Bayesian probability theory (Ramsey 1931; de Finetti 1937).

Contents

See https://intelligence.org/files/LogicalInductionAbridged.pdf for an abridged version of this paper.

1   Introduction

Every student of mathematics has experienced uncertainty about conjectures for which there is "quite a bit of evidence", such as the Riemann hypothesis or the twin prime conjecture. Indeed, when Zhang (2014) proved a bound on the gap between primes, we were tempted to increase our credence in the twin prime conjecture. But how much evidence does this bound provide for the twin prime conjecture? Can we quantify the degree to which it should increase our confidence?

The natural impulse is to appeal to probability theory in general and Bayes' theorem in particular. Bayes' theorem gives rules for how to use observations to update empirical uncertainty about unknown events in the physical world. However, probability theory lacks the tools to manage uncertainty about logical facts.

Consider encountering a computer connected to an input wire and an output wire. If we know what algorithm the computer implements, then there are two distinct ways to be uncertain about the output. We could be uncertain about the input—maybe it's determined by a coin toss we didn't see. Alternatively, we could be uncertain because we haven't had the time to reason out what the program does—perhaps it computes the parity of the 87,653rd digit in the decimal expansion of π, and we don't personally know whether it's even or odd.

The first type of uncertainty is about *empirical* facts. No amount of thinking in isolation will tell us whether the coin came up heads. To resolve empirical uncertainty we must observe the coin, and then Bayes' theorem gives a principled account of how to update our beliefs.

The second type of uncertainty is about a *logical* fact, about what a known computation will output when evaluated. In this case, reasoning in isolation can and should change our beliefs: we can reduce our uncertainty by thinking more about π, without making any new observations of the external world.

In any given practical scenario, reasoners usually experience a mix of both empirical uncertainty (about how the world is) and logical uncertainty (about what that implies). In this paper, we focus entirely on the problem of managing logical uncertainty. Probability theory does not address this problem, because probability-theoretic reasoners cannot possess uncertainty about logical facts. For example, let ϕ stand for the claim that the 87,653rd digit of π is a 7. If this claim is true, then $(1 + 1 = 2) \Rightarrow \phi$. But the laws of probability theory say that if $A \Rightarrow B$ then $\Pr(A) \le \Pr(B)$. Thus, a perfect Bayesian must be at least as sure of ϕ as they are that $1 + 1 = 2$! Recognition of this problem dates at least back to Good (1950).

Many have proposed methods for relaxing the criterion $\Pr(A) \le \Pr(B)$ until such a time as the implication has been proven (see, e.g, the work of Hacking [1967] and Christiano [2014]). But this leaves open the question of how probabilities should be assigned before the implication is proven, and this brings us back to the search for a principled method for managing uncertainty about logical facts when relationships between them are suspected but unproven.

We propose a partial solution, which we call *logical induction*. Very roughly, our setup works as follows. We consider reasoners that assign probabilities to sentences written in some formal language and refine those probabilities over time. Assuming the language is sufficiently expressive, these sentences can say things like "Goldbach's conjecture is true" or "the computation `prg` on input `i` produces the output `prg(i)=0`". The reasoner is given access to a slow deductive process that emits theorems over time, and tasked with assigning probabilities in a manner that outpaces deduction, e.g., by assigning high probabilities to sentences that are eventually proven, and low probabilities to sentences that are eventually refuted, well before they can be verified deductively. Logical inductors carry out this task in a way that satisfies many desirable properties, including:

1. Their beliefs are logically consistent in the limit as time approaches infinity.
2. They learn to make their probabilities respect many different patterns in logic, at a rate that outpaces deduction.
3. They learn to know what they know, and trust their future beliefs, while avoiding paradoxes of self-reference.

These claims (and many others) will be made precise in Section 4.

A logical inductor is any sequence of probabilities that satisfies our *logical induction criterion*, which works roughly as follows. We interpret a reasoner's probabilities as prices in a stock market, where the probability of ϕ is interpreted as the price of a share that is worth \$1 if ϕ is true, and \$0 otherwise (similar to Beygelzimer, Langford, and Pennock [2012]). We consider a collection of stock traders who buy and sell shares at the market prices, and define a sense in which traders can exploit markets that have irrational beliefs. The logical induction criterion then says that it should not be possible to exploit the market prices using any trading strategy that can be generated in polynomial-time.

Our main finding is a computable algorithm which satisfies the logical induction criterion, plus proofs that a variety of different desiderata follow from this criterion.

The logical induction criterion can be seen as a weakening of the "no Dutch book" criterion that Ramsey (1931) and de Finetti (1937) used to support standard probability theory, which is analogous to the "no Dutch book" criterion that von Neumann and Morgenstern (1944) used to support expected utility theory. Under this interpretation, our criterion says (roughly) that a rational deductively limited reasoner should have beliefs that can't be exploited by any Dutch book strategy constructed by an efficient (polynomial-time) algorithm. Because of the analogy, and the variety of desirable properties that follow immediately from this one criterion, we believe that the logical induction criterion captures a portion of what it means to do good reasoning about logical facts in the face of deductive limitations. That said, there are clear drawbacks to our algorithm: it does not use its resources efficiently; it is not a decision-making algorithm (i.e., it does not "think about what to think about"); and the properties above hold either asymptotically (with poor convergence bounds) or in the limit. In other words, our algorithm gives a theoretically interesting but ultimately impractical account of how to manage logical uncertainty.

1.1 Desiderata for Reasoning under Logical Uncertainty

For historical context, we now review a number of desiderata that have been proposed in the literature as desirable features of "good reasoning" in the face of logical uncertainty. A major obstacle in the study of logical uncertainty is that it's not clear what would count as a satisfactory solution. In lieu of a solution, a common tactic is to list desiderata that intuition says a good reasoner should meet. One can then examine them for patterns, relationships, and incompatibilities. A multitude of desiderata have been proposed throughout the years; below, we have collected a variety of them. Each is stated in its colloquial form; many will be stated formally and studied thoroughly later in this paper.

Desideratum 1 (Computable Approximability). *The method for assigning probabilities to logical claims (and refining them over time) should be computable.*

(See Section 5 for our algorithm.)

A good method for refining beliefs about logic can never be entirely finished, because a reasoner can always learn additional logical facts by thinking for longer. Nevertheless, if the algorithm refining beliefs is going to have any hope of practicality, it should at least be computable. This idea dates at least back at least to Good (1950), and has been discussed in depth by Hacking (1967) and Eells (1990), among others.

Desideratum 1 may seem obvious, but it is not without its teeth. It rules out certain proposals, such as that of Hutter et al. (2013), which has no computable approximation (Sawin and Demski 2013).

Desideratum 2 (Coherence in the Limit). *The belief state that the reasoner is approximating better and better over time should be logically consistent.*

(Discussed in Section 4.1.)

First formalized by Gaifman (1964), the idea of Desideratum 2 is that the belief state that the reasoner is approximating—the beliefs they would have if they had infinite time to think—should be internally consistent. This means that, in the limit

of reasoning, a reasoner should assign $\Pr(\phi) \leq \Pr(\psi)$ whenever $\phi \Rightarrow \psi$, and they should assign probability 1 to all theorems and 0 to all contradictions, and so on.

Desideratum 3 (Approximate Coherence). *The belief state of the reasoner should be approximately coherent. For example, if the reasoner knows that two statements are mutually exclusive, then it should assign probabilities to those sentences that sum to no more than 1, even if it cannot yet prove either sentence.*
(Discussed in sections 4.2 and 4.5.)

Being coherent in the limit is desirable, but good deductively limited reasoning requires approximate coherence at finite times. Consider two claims about a particular computation `prg`, which takes a number `n` as input and produces a number `prg(n)` as output. Assume the first claim says `prg(7)=0`, and the second says `prg(7)=1`. Clearly, these claims are mutually exclusive, and once a reasoner realizes this fact, they should assign probabilities to the two claims that sum to at most 1, even before they can evaluate `prg(7)`. Limit coherence does not guarantee this: a reasoner could assign bad probabilities (say, 100% to both claims) right up until they can evaluate `prg(7)`, at which point they start assigning the correct probabilities. Intuitively, a good reasoner should be able to recognize the mutual exclusivity *before* they've proven either claim. In other words, a good reasoner's beliefs should be approximately coherent.

Desideratum 3 dates back to at least Good (1950), who proposes a weakening of the condition of coherence that could apply to the belief states of limited reasoners. Hacking (1967) proposes an alternative weakening, as do Garrabrant, Fallenstein, et al. (2016).

Desideratum 4 (Learning of Statistical Patterns). *In lieu of knowledge that bears on a logical fact, a good reasoner should assign probabilities to that fact in accordance with the rate at which similar claims are true.* (Discussed in Section 4.4.)

For example, a good reasoner should assign probability $\approx 10\%$ to the claim "the nth digit of π is a 7" for large n (assuming there is no efficient way for a reasoner to guess the digits of π for large n). This desideratum dates at least back to Savage (1967), and seems clearly desirable. If a reasoner thought the 10^{100}th digit of π was almost surely a 9, but had no reason for believing this this, we would be suspicious of their reasoning methods. Desideratum 4 is difficult to state formally; for two attempts, refer to Garrabrant, Benson-Tilsen, et al. (2016) and Garrabrant, Soares, and Taylor (2016).

Desideratum 5 (Calibration). *Good reasoners should be well-calibrated. That is, among events that a reasoner says should occur with probability p, they should in fact occur about p proportion of the time.* (Discussed in Section 4.3.)

Calibration as a desirable property dates back to Pascal, and perhaps farther. If things that a reasoner says should happen 30% of the time actually wind up happening 80% of the time, then they aren't particularly reliable.

Desideratum 6 (Non-Dogmatism). *A good reasoner should not have extreme beliefs about mathematical facts, unless those beliefs have a basis in proof.*
(Discussed in Section 4.6.)

It would be worrying to see a mathematical reasoner place extreme confidence in a mathematical proposition, without any proof to back up their belief. The virtue of skepticism is particularly apparent in probability theory, where Bayes' theorem says that a probabilistic reasoner can never update away from "extreme" (0 or 1) probabilities. Accordingly, Cromwell's law (so named by the statistician Lindley [1991]) says that a reasonable person should avoid extreme probabilities except when applied to statements that are logically true or false. We are dealing with logical uncertainty, so it is natural to extend Cromwell's law to say that extreme probabilities should also be avoided on logical statements, except in cases where the statements have been *proven* true or false. In settings where reasoners are able

to update away from 0 or 1 probabilities, this means that a good reasoner's beliefs shouldn't be "stuck" at probability 1 or 0 on statements that lack proofs or disproofs.

In the domain of logical uncertainty, Desideratum 6 can be traced back to Carnap (1962, Sec. 53), and has been demanded by many, including Gaifman and Snir (1982) and Hutter et al. (2013).

Desideratum 7 (Uniform Non-Dogmatism). *A good reasoner should assign a non-zero probability to any computably enumerable consistent theory (viewed as a limit of finite conjunctions).* (Discussed in Section 4.6.)

For example the axioms of Peano arithmetic are computably enumerable, and if we construct an ever-growing conjunction of these axioms, we can ask that the limit of a reasoner's credence in these conjunctions converge to a value bounded above 0, even though there are infinitely many conjuncts. The first formal statement of Desideratum 7 that we know of is given by Demski (2012), though it is implicitly assumed whenever asking for a set of beliefs that can reason accurately about arbitrary arithmetical claims (as is done by, e.g., Savage [1967] and Hacking [1967]).

Desideratum 8 (Universal Inductivity). *Given enough time to think, the beliefs of a good reasoner should dominate the universal semimeasure.* (Discussed in Section 4.6.)

Good reasoning in general has been studied for quite some time, and reveals some lessons that are useful for the study of good reasoning under deductive limitation. Solomonoff (1964a, 1964b), Zvonkin and Levin (1970), and Li and Vitányi (1993) have given a compelling formal treatment of good reasoning assuming logical omniscience in the domain of sequence prediction, by describing an inductive process (known as a universal semimeasure) with a number of nice properties, including (1) it assigns non-zero prior probability to every computable sequence of observations; (2) it assigns higher prior probability to simpler hypotheses; and (3) it predicts as well or better than any computable predictor, modulo a constant amount of error. Alas, universal semimeasures are uncomputable; nevertheless, they provide a formal model of what it means to predict sequences well, and we can ask logically uncertain reasoners to copy those successes. For example, we can ask that they would perform as well as a universal semimeasure if given enough time to think.

Desideratum 9 (Approximate Bayesianism). *The reasoner's beliefs should admit of some notion of conditional probabilities, which approximately satisfy both Bayes' theorem and the other desiderata listed here.* (Discussed in Section 4.7.)

Bayes' rule gives a fairly satisfying account of how to manage empirical uncertainty in principle (as argued extensively by Jaynes [2003]), where beliefs are updated by conditioning a probability distribution. As discussed by Good (1950) and Glymour (1980), creating a distribution that satisfies both coherence and Bayes' theorem requires logical omniscience. Still, we can ask that the approximation schemes used by a limited agent be approximately Bayesian in some fashion, while retaining whatever good properties the unconditional probabilities have.

Desideratum 10 (Introspection). *If a good reasoner knows something, she should also know that she knows it.* (Discussed in Section 4.11.)

Proposed by Hintikka (1962), this desideratum is popular among epistemic logicians. It is not completely clear that this is a desirable property. For instance, reasoners should perhaps be allowed to have "implicit knowledge" (which they know without knowing that they know it), and it's not clear where the recursion should stop (do you know that you know that you know that you know that $1 = 1$?). This desideratum has been formalized in many different ways; see Christiano et al. (2013) and Campbell-Moore (2015) for a sample.

Desideratum 11 (Self-Trust). *A good reasoner thinking about a hard problem should expect that, in the future, her beliefs about the problem will be more accurate than her current beliefs.* (Discussed in Section 4.12.)

Stronger than self-knowledge is self-*trust*—a desideratum that dates at least back to Hilbert (1902), when mathematicians searched for logics that placed confidence in their own machinery. While Gödel, Kleene, and Rosser (1934) showed that strong forms of self-trust are impossible in a formal proof setting, experience demonstrates that human mathematicians are capable of trusting their future reasoning, relatively well, most of the time. A method for managing logical uncertainty that achieves this type of self-trust would be highly desirable.

Desideratum 12 (Approximate Inexploitability)**.** *It should not be possible to run a Dutch book against a good reasoner in practice.* (See Section 3 for our proposal.)

Expected utility theory and probability theory are both supported in part by "Dutch book" arguments which say that an agent is rational if (and only if) there is no way for a clever bookie to design a "Dutch book" which extracts arbitrary amounts of money from the reasoner (von Neumann and Morgenstern 1944; de Finetti 1937). As noted by Eells (1990), these constraints are implausibly strong: all it takes to run a Dutch book according to de Finetti's formulation is for the bookie to know a logical fact that the reasoner does not know. Thus, to avoid being Dutch booked by de Finetti's formulation, a reasoner must be logically omniscient.

Hacking (1967) and Eells (1990) call for weakenings of the Dutch book constraints, in the hopes that reasoners that are approximately inexploitable would do good approximate reasoning. This idea is the cornerstone of our framework—in particular, we consider reasoners that cannot be exploited in polynomial time, using a formalism defined below. See Definition 3.0.1 for details.

Desideratum 13 (Gaifman Inductivity)**.** *Given a Π_1 statement ϕ (i.e., a universal generalization of the form "for every x, ψ"), as the set of examples the reasoner has seen goes to "all examples", the reasoner's belief in ϕ should approach to certainty.* (Discussed below.)

Proposed by Gaifman (1964), Desideratum 13 states that a reasoner should "generalize well", in the sense that as they see more instances of a universal claim (such as "for every x, $\psi(x)$ is true") they should eventually believe the universal with probability 1. Desideratum 13 has been advocated by Hutter et al. (2013).

Desideratum 14 (Efficiency)**.** *The algorithm for assigning probabilities to logical claims should run efficiently, and be usable in practice.* (Discussed in Section 7.1.)

One goal of understanding "good reasoning" in the face of logical uncertainty is to design algorithms for reasoning using limited computational resources. For that, the algorithm for assigning probabilities to logical claims needs to be not only computable, but efficient. Aaronson (2013) gives a compelling argument that solutions to logical uncertainty require understanding complexity theory, and this idea is closely related to the study of bounded rationality (Simon 1982) and efficient meta-reasoning (Russell and Wefald 1991b).

Desideratum 15 (Decision Rationality)**.** *The algorithm for assigning probabilities to logical claims should be able to target specific, decision-relevant claims, and it should reason about those claims as efficiently as possible given the computing resources available.* (Discussed in Section 7.4.)

This desideratum dates at least back to Savage (1967), who asks for an extension to probability theory that takes into account the costs of thinking. For a method of reasoning under logical uncertainty to aid in the understanding of good bounded reasoning, it must be possible for an agent to use the reasoning system to reason efficiently about specific decision-relevant logical claims, using only enough resources to refine the probabilities well enough for the right decision to become clear. This desideratum blurs the line between decision-making and logical reasoning; see Russell and Wefald (1991a) and Hay et al. (2012) for a discussion.

Desideratum 16 (Answers Counterpossible Questions)**.** *When asked questions about contradictory states of affairs, a good reasoner should give reasonable answers.* (Discussed in Section 7.4.)

In logic, the principle of explosion say that from a contradiction, anything follows. By contrast, when human mathematicians are asked counterpossible questions, such as "what would follow from Fermat's last theorem being false?", they often give reasonable answers, such as "then there would exist non-modular elliptic curves", rather than just saying "anything follows from a contradiction". Soares and Fallenstein (2015) point out that some deterministic decision-making algorithms reason about counterpossible questions ("what would happen if my deterministic algorithm had the output a vs b vs c?"). The topic of counterpossibilities has been studied by philosophers including Cohen (1990), Vander Laan (2004), Brogaard and Salerno (2007), Krakauer (2012), and Bjerring (2014), and it is reasonable to hope that a good logically uncertain reasoner would give reasonable answers to counterpossible questions.

Desideratum 17 (Use of Old Evidence). *When a bounded reasoner comes up with a new theory that neatly describes anomalies in the old theory, that old evidence should count as evidence in favor of the new theory.*   (Discussed in Section 7.4.)

The problem of old evidence is a longstanding problem in probability theory (Glymour 1980). Roughly, the problem is that a perfect Bayesian reasoner always uses all available evidence, and keeps score for all possible hypotheses at all times, so no hypothesis ever gets a "boost" from old evidence. Human reasoners, by contrast, have trouble thinking up good hypotheses, and when they do, those new hypotheses often get a large boost by retrodicting old evidence. For example, the precession of the perihelion of Mercury was known for quite some time before the development of the theory of General Relativity, and could not be explained by Newtonian mechanics, so it was counted as strong evidence in favor of Einstein's theory. Garber (1983) and Jeffrey (1983) have speculated that a solution to the problem of logical omniscience would shed light on solutions to the problem of old evidence.

Our solution does not achieve all these desiderata. Doing so would be impossible; Desiderata 1, 2, and 13 cannot be satisfied simultaneously. Further, Sawin and Demski (2013) have shown that Desiderata 1, 6, 13, and a very weak form of 2 are incompatible; an ideal belief state that is non-dogmatic, Gaifman inductive, and approximately coherent in a weak sense has no computable approximation. Our algorithm is computably approximable, approximately coherent, and non-dogmatic, so it cannot satisfy 13. Our algorithm also fails to meet 14 and 15, because while our algorithm is computable, it is purely inductive, and so it does not touch upon the decision problem of thinking about what to think about and how to think about it with minimal resource usage. As for 16 and 17, the case is interesting but unclear; we give these topics some treatment in Section 7.

Our algorithm does satisfy desiderata 1 through 12. In fact, our algorithm is designed to meet only 1 and 12, from which 2-11 will all be shown to follow. This is evidence that our logical induction criterion captures a portion of what it means to manage uncertainty about logical claims, analogous to how Bayesian probability theory is supported in part by the fact that a host of good properties follow from a single criterion ("don't be exploitable by a Dutch book"). That said, there is ample room to disagree about how well our algorithm achieves certain desiderata, e.g. when the desiderata is met only in the asymptote, or with error terms that vanish only slowly.

1.2   Related Work

The study of logical uncertainty is an old topic. It can be traced all the way back to Bernoulli, who laid the foundations of statistics, and later Boole (1854), who was interested in the unification of logic with probability from the start. Refer to Hailperin (1996) for a historical account. Our algorithm assigns probabilities to sentences of logic directly; this thread can be traced back through Łoś (1955) and later Gaifman (1964), who developed the notion of coherence that we use in this paper. More recently, that thread has been followed by Demski (2012), whose framework we use, and Hutter et al. (2013), who define a probability distribution

on logical sentences that is quite desirable, but which admits of no computable approximation (Sawin and Demski 2013).

The objective of our algorithm is to manage uncertainty about logical facts (such as facts about mathematical conjectures or long-running computer programs). When it comes to the problem of developing formal tools for manipulating uncertainty, our methods are heavily inspired by Bayesian probability theory, and so can be traced back to Pascal, who was followed by Bayes, Laplace, Kolmogorov (1950), Savage (1954), Carnap (1962), and Jaynes (2003), and many others. Polya (1990) was among the first in the literature to explicitly study the way that mathematicians engage in plausible reasoning, which is tightly related to the object of our study.

We are interested in the subject of what it means to do "good reasoning" under logical uncertainty. In this, our approach is quite similar to the approach of Ramsey (1931), de Finetti (1937), von Neumann and Morgenstern (1944), Teller (1973), Lewis (1999), and Joyce (1999), who each developed axiomatizations of rational behavior and produced arguments supporting those axioms. In particular, they each supported their proposals with Dutch book arguments, and those Dutch book arguments were a key inspiration for our logical induction criterion.

The fact that using a coherent probability distribution requires logical omniscience (and is therefore unsatisfactory when it comes to managing logical uncertainty) dates at least back to Good (1950). Savage (1967) also recognized the problem, and stated a number of formal desiderata that our solution in fact meets. Hacking (1967) addressed the problem by discussing notions of approximate coherence and weakenings of the Dutch book criteria. While his methods are ultimately unsatisfactory, our approach is quite similar to his in spirit.

The flaw in Bayesian probability theory was also highlighted by Glymour (1980), and dubbed the "problem of old evidence" by Garber (1983) in response to Glymor's criticism. Eells (1990) gave a lucid discussion of the problem, revealed flaws in Garber's arguments and in Hacking's solution, and named a number of other desiderata which our algorithm manages to satisfy. Refer to Zynda (1995) and Sprenger (2015) for relevant philosophical discussion in the wake of Eells. Of note is the treatment of Adams (1996), who uses logical deduction to reason about an unknown probability distribution that satisfies certain logical axioms. Our approach works in precisely the opposite direction: we use probabilistic methods to create an approximate distribution where logical facts are the subject.

Straddling the boundary between philosophy and computer science, Aaronson (2013) has made a compelling case that computational complexity must play a role in answering questions about logical uncertainty. These arguments also provided some inspiration for our approach, and roughly speaking, we weaken the Dutch book criterion of standard probability theory by considering only exploitation strategies that can be constructed by a polynomial-time machine. The study of logical uncertainty is also tightly related to the study of bounded rationality (Simon 1982; Russell and Wefald 1991a; Rubinstein 1998; Russell 2016).

Fagin and Halpern (1987) also straddled the boundary between philosophy and computer science with early discussions of algorithms that manage uncertainty in the face of resource limitations. (See also their discussions of uncertainty and knowledge [Fagin et al. 1995; Halpern 2003].) This is a central topic in the field of artificial intelligence (AI), where scientists and engineers have pursued many different paths of research. The related work in this field is extensive, including (but not limited to) work on probabilistic programming (Vajda 1972; McCallum, Schultz, and Singh 2009; Wood, Meent, and Mansinghka 2014; De Raedt and Kimmig 2015); probabilistic inductive logic programming (Muggleton and Watanabe 2014; De Raedt and Kersting 2008; De Raedt 2008; Kersting and De Raedt 2007); and meta-reasoning (Russell and Wefald 1991b; Zilberstein 2008; Hay et al. 2012). The work most closely related to our own is perhaps the work of Thimm (2013a) and others on reasoning using inconsistent knowledge bases, a task which is analogous to constructing an approximately coherent probability distribution. (See also Muiño [2011], Thimm [2013b], Potyka and Thimm [2015], and Potyka [2015].) Our framework also bears some resemblance to the Markov logic network framework of Richardson and Domingos (2006), in that both algorithms are coherent in the limit. Where Markov logic networks are specialized

to individual restricted domains of discourse, our algorithm reasons about all logical sentences. (See also Kok and Domingos [2005], Singla and Domingos [2005], Tran and Davis [2008], Lowd and Domingos [2007], Mihalkova, Huynh, and Mooney [2007], Wang and Domingos [2008], and Khot et al. [2015].)

In that regard, our algorithm draws significant inspiration from Solomonoff's theory of inductive inference (Solomonoff 1964a, 1964b) and the developments on that theory made by Zvonkin and Levin (1970) and Li and Vitányi (1993). Indeed, we view our algorithm as a Solomonoff-style approach to the problem of reasoning under logical uncertainty, and as a result, our algorithm bears a strong resemblance to many algorithms that are popular methods for practical statistics and machine learning; refer to Opitz and Maclin (1999) and Dietterich (2000) for reviews of popular and successful ensemble methods. Our approach is also similar in spirit to the probabilistic numerics approach of Briol, Oates, Girolami, Osborne, and Sejdinovic (2015), but where probabilistic numerics is concerned with algorithms that give probabilistic answers to individual particular numerical questions, we are concerned with algorithms that assign probabilities to all queries in a given formal language. (See also [Briol, Oates, Girolami, and Osborne 2015; Hennig, Osborne, and Girolami 2015].)

Finally, our method of interpreting beliefs as prices and using prediction markets to generate reasonable beliefs bears heavy resemblance to the work of Beygelzimer, Langford, and Pennock (2012) who use similar mechanisms to design a learning algorithm that bets on events. Our results can be seen as an extension of that idea to the case where the events are every sentence written in some formal language, in a way that learns inductively to predict logical facts while avoiding the standard paradoxes of self-reference.

The work sampled here is only a small sample of the related work, and it neglects contributions from many other fields, including but not limited to epistemic logic (Gärdenfors 1988; Meyer and Van Der Hoek 1995; Schlesinger 1985; Sowa 1999; Guarino 1998), game theory (Rantala 1979; Hintikka 1979; Bacharach 1994; Lipman 1991; Battigalli and Bonanno 1999; Binmore 1992), paraconsistent logic (Blair and Subrahmanian 1989; Priest 2002; Mortensen 2013; Fuhrmann 2013; Akama and Costa 2016) and fuzzy logic (Klir and Yuan 1995; Yen and Langari 1999; Gerla 2013). The full history is too long and rich for us to do it justice here.

1.3   Overview

Our main result is a formalization of Desideratum 12 above, which we call the *logical induction criterion*, along with a computable algorithm that meets the criterion, plus proofs that formal versions of Desiderata 2-11 all follow from the criterion.

In Section 2 we define some notation. In Section 3 we state the logical induction criterion and our main theorem, which says that there exists a computable logical inductor. The logical induction criterion is motivated by a series of stock trading analogies, which are also introduced in Section 3.

In Section 4 we discuss a number of properties that follow from this criterion, including properties that hold in the limit, properties that relate to pattern-recognition, calibration properties, and properties that relate to self-knowledge and self-trust.

A computable logical inductor is described in Section 5. Very roughly, the idea is that given any trader, it's possible to construct market prices at which they make no trades (because they think the prices are right); and given an enumeration of traders, it's possible to aggregate their trades into one "supertrader" (which takes more and more traders into account each day); and thus it is possible to construct a series of prices which is not exploitable by any trader in the enumeration.

In Section 6 we give a few selected proofs. In Section 7 we conclude with a discussion of applications of logical inductiors, variations on the logical induction framework, speculation about what makes logical inductors tick, and directions for future research. The remaining proofs can be found in the appendix.

2 Notation

This section defines notation used throughout the paper. The reader is invited to skim it, or perhaps skip it entirely and use it only as a reference when needed.

Common sets and functions. The set of positive natural numbers is denoted by $\mathbb{N}^+$, where the superscript makes it clear that 0 is not included. We work with $\mathbb{N}^+$ instead of $\mathbb{N}^{\geq 0}$ because we regularly consider initial segments of infinite sequences up to and including the element at index n, and it will be convenient for those lists to have length n. Sums written $\sum_{i \leq n}(-)$ are understood to start at $i = 1$. We use $\mathbb{R}$ to denote the set of real numbers, and $\mathbb{Q}$ to denote the set of rational numbers. When considering continuous functions with range in $\mathbb{Q}$, we use the subspace topology on $\mathbb{Q}$ inherited from $\mathbb{R}$. We use $\mathbb{B}$ to denote the set $\{0, 1\}$ interpreted as Boolean values. In particular, Boolean operations like $\wedge$, $\vee$, $\neg$, $\to$ and $\leftrightarrow$ are defined on $\mathbb{B}$, for example, $(1 \wedge 1) = 1$, $\neg 1 = 0$, and so on.

We write $\mathrm{Fin}(X)$ for the set of all finite subsets of X, and $X^{\mathbb{N}^+}$ for all infinite sequences with elements in X. In general, we use B^A to denote the set of functions with domain A and codomain B. We treat the expression $f : A \to B$ as equivalent to $f \in B^A$, i.e., both state that f is a function that takes inputs from the set A and produces an output in the set B. We write $f : A \nrightarrow B$ to indicate that f is a partial function from A to B. We denote equivalence of expressions that represent functions by $\equiv$, e.g., $(x - 1)^2 \equiv x^2 - 2x + 1$. We write $\| - \|_1$ for the ℓ_1 norm. When A is an affine combination, $\|A\|_1$ includes the trailing coefficient.

Logical sentences. We generally use the symbols ϕ, ψ, χ to denote well-formed formulas in some language of propositional logic $\mathcal{L}$ (such as a theory of first order logic; see below), which includes the basic logical connectives $\neg$, $\wedge$, $\vee$, $\to$, $\leftrightarrow$, and uses modus ponens as its rule of inference. We assume that $\mathcal{L}$ has been chosen so that its sentences can be interpreted as claims about some class of mathematical objects, such as natural numbers or computer programs. We commonly write $\mathcal{S}$ for the set of all sentences in $\mathcal{L}$, and Γ for a set of axioms from which to write proofs in the language. We write $\Gamma \vdash \phi$ when ϕ can be proven from Γ via modus ponens.

We will write logical formulas inside quotes "$-$", such as $\phi := $ "$x = 3$". The exception is after $\vdash$, where we do not write quotes, in keeping with standard conventions. We sometimes define sentences such as $\phi := $ "Goldbach's conjecture", in which case it is understood that the English text could be expanded into a precise arithmetical claim.

We use underlines to indicate when a symbol in a formula should be replaced by the expression it stands for. For example, if $n := 3$, then $\phi := $ "$x > \underline{n}$" means $\phi = $ "$x > 3$", and $\psi := $ "$\underline{\phi} \to (x = \underline{n} + 1)$" means $\psi = $ "$x > 3 \to (x = 3 + 1)$". If ϕ and ψ denote formulas, then $\neg\phi$ denotes "$\neg(\underline{\phi})$" and $\phi \wedge \psi$ denotes "$(\underline{\phi}) \wedge (\underline{\psi})$" and so on. For instance, if $\phi := $ "$x > 3$" then $\neg\phi$ denotes "$\neg(x > 3)$".

First order theories and prime sentences. We consider any theory in first order logic (such as Peano Arithmetic, PA) as a set of axioms that includes the axioms of first order logic, so that modus ponens is the only rule of inference needed for proofs. As such, we view any first order theory as specified in a propositional calculus (following Enderton [2001]) whose atoms are the so-called "prime" sentences of first order logic, i.e., quantified sentences like "$\exists x \colon \cdots$", and atomic sentences like "$t_1 = t_2$" and "$R(t_1, \ldots, t_n)$" where the t_i are closed terms. Thus, every first-order sentence can be viewed as a Boolean combination of prime sentences with logical connectives (viewing "$\forall x \colon \cdots$" as shorthand for "$\neg\exists x \colon \neg \cdots$"). For example, the sentence

$$\phi := \text{``}((1 + 1 = 2) \wedge (\forall x \colon x > 0)) \to (\exists y \colon \forall z \colon (7 > 1 + 1) \to (y + z > 2))\text{''}$$

is decomposed into "$1 + 1 = 2$", "$\exists x \colon \neg(x > 0)$" and "$\exists y \colon \forall z \colon (7 > 1 + 1) \to (y + z > 2)$", where the leading "$\neg$" in front of the second statement is factored out as a Boolean operator. In particular, note that while $(7 > 1 + 1)$ is a prime sentence, it *does not* occur in the Boolean decomposition of ϕ into primes, since it occurs

within a quantifier. We choose this view because we will not always assume that the theories we manipulate include the quantifier axioms of first-order logic.

Defining values by formulas. We often view a formula that is free in one variable as a way of defining a particular number that satisfies that formula. For example, given the formula $X(\nu) =$ "$\nu^2 = 9 \ \wedge \ \nu > 0$", we would like to think of X as representing the unique value "3", in such a way that that we can then have "$5X + 1$" refer to the number 16.

To formalize this, we use the following notational convention. Let X be a formula free in one variable. We write $X(x)$ for the formula resulting from substituting x for the free variable of X. If

$$\Gamma \vdash \exists x \forall y \colon X(y) \to y = x,$$

then we say that X defines a unique value (via Γ), and we refer to that value as "the value" of X. We will be careful in distinguishing between what Γ can prove about $X(\nu)$ on the one hand, and the values of $X(\nu)$ in different models of Γ on the other.

If $X_1, \ldots, X_k$ are all formulas free in one variable that define a unique value (via Γ), then for any k-place relationship R, we write "$R(X_1, X_2, \ldots, X_k)$" as an abbreviation for

"$\forall x_1 x_2 \ldots x_k \colon X_1(x_1) \wedge X_2(x_2) \wedge \ldots \wedge X_k(x_k) \to R(x_1, x_2, \ldots, x_k)$".

For example, "$Z = 2X + Y$" is shorthand for

"$\forall xyz \colon X(x) \wedge Y(y) \wedge Z(z) \to z = 2x + y$".

This convention allows us to write concise expressions that describe relationships between well-defined values, even when those values may be difficult or impossible to determine via computation.

Representing computations. When we say a theory Γ in first order logic "can represent computable functions", we mean that its language is used to refer to computer programs in such a way that Γ satisfies the representability theorem for computable functions. This means that for every (total) computable function $f : \mathbb{N}^+ \to \mathbb{N}^+$, there exists a Γ-formula γ_f with two free variables such that for all $n, y \in \mathbb{N}^+$,

$$y = f(n) \ \text{ if and only if } \ \Gamma \vdash \forall \nu \colon \gamma_f(\underline{n}, \nu) \leftrightarrow \nu = \underline{y},$$

where "$\gamma_f(\underline{n}, \nu)$" stands, in the usual way, for the formula resulting from substituting an encoding of n and the symbol ν for its free variables. In particular, note that this condition requires Γ to be consistent.

When Γ can represent computable functions, we use "$\underline{f}(\underline{n})$" as shorthand for the formula "$\gamma_f(\underline{n}, \nu)$". In particular, since "$\gamma_f(\underline{n}, \nu)$" is free in a single variable ν and defines a unique value, we use "$\underline{f}(\underline{n})$" by the above convention to write, e.g.,

"$\underline{f}(3) < \underline{g}(3)$"

as shorthand for

"$\forall xy \colon \gamma_f(3, x) \wedge \gamma_g(3, y) \to x < y$".

In particular, note that writing down a sentence like "$\underline{f}(3) > 4$" does not involve computing the value $f(3)$; it merely requires writing out the definition of γ_f. This distinction is important when f has a very slow runtime.

Sequences. We denote infinite sequences using overlines, like $\overline{x} := (x_1, x_2, \ldots)$, where it is understood that x_i denotes the ith element of $\overline{x}$, for $i \in \mathbb{N}^+$. To define sequences of sentences compactly, we use parenthetical expressions such as $\overline{\phi} := ("\underline{n} > 7")_{n \in \mathbb{N}^+}$, which defines the sequence

$$("1 > 7", "2 > 7", "3 > 7", \ldots).$$

We define $x_{\leq n} := (x_1, \ldots, x_n)$. Given another element y, we abuse notation in the usual way and define $(x_{\leq n}, y) = (x_1, \ldots, x_n, y)$ to be the list $x_{\leq n}$ with y appended at the end. We write $()$ for the empty sequence.

A sequence $\overline{x}$ is called *computable* if there is a computable function f such that $f(n) = x_n$ for all $n \in \mathbb{N}^+$, in which case we say f computes $\overline{x}$.

Asymptotics. Given any sequences $\overline{x}$ and $\overline{y}$, we write

$$x_n \eqsim_n y_n \quad \text{for} \quad \lim_{n \to \infty} x_n - y_n = 0,$$

$$x_n \gtrsim_n y_n \quad \text{for} \quad \liminf_{n \to \infty} x_n - y_n \geq 0, \text{ and}$$

$$x_n \lesssim_n y_n \quad \text{for} \quad \limsup_{n \to \infty} x_n - y_n \leq 0.$$

3 The Logical Induction Criterion

In this section, we will develop a framework in which we can state the logical induction criterion and a number of properties possessed by logical inductors. The framework will culminate in the following definition, and a theorem saying that computable logical inductors exist for every deductive process.

Definition 3.0.1 (The Logical Induction Criterion). *A market $\overline{\mathbb{P}}$ is said to satisfy the **logical induction criterion** relative to a deductive process $\overline{D}$ if there is no efficiently computable trader $\overline{T}$ that exploits $\overline{\mathbb{P}}$ relative to $\overline{D}$. A market $\overline{\mathbb{P}}$ meeting this criterion is called a **logical inductor over $\overline{D}$**.*

We will now define markets, deductive processes, efficient computability, traders, and exploitation.

3.1 Markets

We will be concerned with methods for assigning values in the interval $[0, 1]$ to sentences of logic. We will variously interpret those values as prices, probabilities, and truth values, depending on the context. Let $\mathcal{L}$ be a language of propositional logic, and let $\mathcal{S}$ be the set of all sentences written in $\mathcal{L}$. We then define:

Definition 3.1.1 (Valuation). *A **valuation** is any function $\mathbb{V} : \mathcal{S} \to [0, 1]$. We refer to $\mathbb{V}(\phi)$ as the value of ϕ according to $\mathbb{V}$. A valuation is called rational if its image is in $\mathbb{Q}$.*

First let us treat the case where we interpret the values as prices.

Definition 3.1.2 (Pricing). *A **pricing** $\mathbb{P} : \mathcal{S} \to \mathbb{Q} \cap [0, 1]$ is any computable rational valuation. If $\mathbb{P}(\phi) = p$ we say that the price of a ϕ-share according to $\mathbb{P}$ is p, where the intended interpretation is that a ϕ-share is worth \$1 if ϕ is true.*

Definition 3.1.3 (Market). *A **market** $\overline{\mathbb{P}} = (\mathbb{P}_1, \mathbb{P}_2, \ldots)$ is a computable sequence of pricings $\mathbb{P}_i : \mathcal{S} \to \mathbb{Q} \cap [0, 1]$.*

We can visualize a market as a series of pricings that may change day by day. The properties proven in Section 4 will apply to any market that satisfies the logical induction criterion. Theorem 4.1.2 (Limit Coherence) will show that the prices of a logical inductor can reasonably be interpreted as probabilities, so we will often speak as if the prices in a market represent the beliefs of a reasoner, where $\mathbb{P}_n(\phi) = 0.75$ is interpreted as saying that on day n, the reasoner assigns 75% probability to ϕ.

In fact, the logical inductor that we construct in Section 5 has the additional property of being finite at every timestep, which means we can visualize it as a series of finite belief states that a reasoner of interest writes down each day.

Definition 3.1.4 (Belief State)**.** *A **belief state** $\mathbb{P} : \mathcal{S} \to \mathbb{Q} \cap [0,1]$ is a computable rational valuation with finite support, where $\mathbb{P}(\phi)$ is interpreted as the probability of ϕ (which is 0 for all but finitely many ϕ).*

We can visualize a belief state as a finite list of (ϕ, p) pairs, where the ϕ are unique sentences and the p are rational-number probabilities, and $\mathbb{P}(\phi)$ is defined to be p if (ϕ, p) occurs in the list, and 0 otherwise.

Definition 3.1.5 (Computable Belief Sequence)**.** *A **computable belief sequence** $\overline{\mathbb{P}} = (\mathbb{P}_1, \mathbb{P}_2, \ldots)$ is a computable sequence of belief states, interpreted as a reasoner's explicit beliefs about logic as they are refined over time.*

We can visualize a computable belief sequence as a large spreadsheet where each column is a belief state, and the rows are labeled by an enumeration of all logical sentences. We can then imagine a reasoner of interest working on this spreadsheet, by working on one column per day.

Philosophically, the reason for this setup is as follows. Most people know that the sentence "$1 + 1$ is even" is true, and that the sentence "$1 + 1 + 1 + 1$ is even" is true. But consider, is the following sentence true?

$$\text{``}1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 \text{ is even''}$$

To answer, we must pause and count the ones. Since we wish to separate the question of what a reasoner already knows from what they could infer using further computing resources, we require that the reasoner write out their beliefs about logic explicitly, and refine them day by day.

In this framework, we can visualize a reasoner as a person who computes the belief sequence by filling in a large spreadsheet, always working on the nth column on the nth day, by refining and extending her previous work as she learns new facts and takes more sentences into account, while perhaps making use of computer assistance. For example, a reasoner who has noticed that "$1 + \cdots + 1$ is even" is true iff the sentence has an even number of ones, might program her computer to write 1 into as many of the true "$1 + \cdots + 1$ is even" cells per day as it can before resources run out. As another example, a reasoner who finds a bound on the prime gap might go back and update her probability on the twin prime conjecture. In our algorithm, the reasoner will have more and more computing power each day, with which to construct her next belief state.

3.2 Deductive Processes

We are interested in the question of what it means for reasoners to assign "reasonable probabilities" to statements of logic. Roughly speaking, we will imagine reasoners that have access to some formal deductive process, such as a community of mathematicians who submit machine-checked proofs to an official curated database. We will study reasoners that "outpace" this deductive process, e.g., by assigning high probabilities to conjectures that will eventually be proven, and low probabilities to conjectures that will eventually be disproven, well before the relevant proofs are actually found.

Definition 3.2.1 (Deductive Process)**.** *A **deductive process** $\overline{D} : \mathbb{N}^+ \to$ $\mathrm{Fin}(\mathcal{S})$ is a computable nested sequence $D_1 \subseteq D_2 \subseteq D_3 \ldots$ of finite sets of sentences. We write D_∞ for the union $\bigcup_n D_n$.*

This is a rather barren notion of "deduction". We will consider cases where we fix some theory Γ, and D_n is interpreted as the theorems proven up to and including day n. In this case, $\overline{D}$ can be visualized as a slow process that reveals the knowledge of Γ over time. Roughly speaking, we will mainly concern ourselves with the case where $\overline{D}$ eventually rules out all and only the worlds that are inconsistent with Γ.

Definition 3.2.2 (World)*. A **world** is any truth assignment $\mathbb{W} : \mathcal{S} \to \mathbb{B}$. If $\mathbb{W}(\phi) = 1$ we say that ϕ is **true in** $\mathbb{W}$. If $\mathbb{W}(\phi) = 0$ we say that ϕ is **false in** $\mathbb{W}$. We write $\mathcal{W}$ for the set of all worlds.*

Observe that worlds are valuations, and that they are not necessarily consistent. This terminology is nonstandard; the term "world" is usually reserved for consistent truth assignments. Logically uncertain reasoners cannot immediately tell which truth assignments are inconsistent, because revealing inconsistencies requires time and effort. We use the following notion of consistency:

Definition 3.2.3 (Propositional Consistency)*. A world $\mathbb{W}$ is called **proposition- ally consistent**, abbreviated **p.c.**, if for all $\phi \in \mathcal{S}$, $\mathbb{W}(\phi)$ is determined by Boolean algebra from the truth values that $\mathbb{W}$ assigns to the prime sentences of ϕ. In other words, $\mathbb{W}$ is p.c. if $\mathbb{W}(\phi \wedge \psi) = \mathbb{W}(\phi) \wedge \mathbb{W}(\psi)$, $\mathbb{W}(\phi \vee \psi) = \mathbb{W}(\phi) \vee \mathbb{W}(\psi)$, and so on.*

*Given a set of sentences D, we define $\mathcal{PC}(D)$ to be the set of all p.c. worlds where $\mathbb{W}(\phi) = 1$ for all $\phi \in D$. We refer to $\mathcal{PC}(D)$ as the set of worlds **propositionally consistent with D**.*

*Given a set of sentences Γ interpreted as a theory, we will refer to $\mathcal{PC}(\Gamma)$ as the set of worlds **consistent with Γ**, because in this case $\mathcal{PC}(\Gamma)$ is equal to the set of all worlds $\mathbb{W}$ such that*

$$\Gamma \cup \{\phi \mid \mathbb{W}(\phi) = 1\} \cup \{\neg\phi \mid \mathbb{W}(\phi) = 0\} \nvdash \bot.$$

Note that a limited reasoner won't be able to tell whether a given world $\mathbb{W}$ is in $\mathcal{PC}(\Gamma)$. A reasoner can computably check whether a restriction of $\mathbb{W}$ to a finite domain is propositionally consistent with a finite set of sentences, but that's about it. Roughly speaking, the definition of exploitation (below) will say that a good reasoner should perform well when measured on day n by worlds propositionally consistent with D_n, and we ourselves will be interested in deductive processes that pin down a particular theory Γ by propositional consistency:

Definition 3.2.4 (Γ-Complete)*. Given a theory Γ, we say that a deductive process $\overline{D}$ is **Γ-complete** if*

$$\mathcal{PC}(D_\infty) = \mathcal{PC}(\Gamma).$$

As a canonical example, let D_n be the set of all theorems of PA provable in at most n characters.[1] Then $\overline{D}$ is PA-complete, and a reasoner with access to $\overline{D}$ can be interpreted as someone who on day n knows all PA-theorems provable in $\leq n$ characters, who must manage her uncertainty about other mathematical facts.

3.3 Efficient Computability

We use the following notion of efficiency throughout the paper:

Definition 3.3.1 (Efficiently Computable)*. An infinite sequence $\overline{x}$ is called **efficiently computable**, abbreviated **e.c.**, if there is a computable function f that outputs x_n on input n, with runtime polynomial in n (i.e. in the length of n written in unary).*

Our framework is not wedded to this definition; stricter notions of efficiency (e.g., sequences that can be computed in $\mathcal{O}(n^2)$ time) would yield "dumber" inductors with better runtimes, and vice versa. We use the set of polynomial-time computable functions because it has some closure properties that are convenient for our purposes.

---

1. Because PA is a first-order theory, and the only assumption we made about $\mathcal{L}$ is that it is a propositional logic, note that the axioms of first-order logic—namely, specialization and distribution—must be included as theorems in $\overline{D}$.

3.4 Traders

Roughly speaking, traders are functions that see the day n and the history of market prices up to and including day n, and then produce a series of buy and sell orders, by executing a strategy that is continuous as a function of the market history.

A linear combination of sentences can be interpreted as a "market order", where $3\phi - 2\psi$ says to buy 3 shares of ϕ and sell 2 shares of ψ. Very roughly, a trading strategy for day n will be a method for producing market orders where the coefficients are not numbers but *functions* which depend (continuously) on the market prices up to and including day n.

Definition 3.4.1 (Valuation Feature). *A valuation **feature** $\alpha : [0,1]^{\mathcal{S} \times \mathbb{N}^+} \to \mathbb{R}$ is a continuous function from valuation sequences to real numbers such that $\alpha(\overline{\mathbb{V}})$ depends only on the initial sequence $\mathbb{V}_{\leq n}$ for some $n \in \mathbb{N}^+$ called the rank of the feature, $\mathrm{rank}(\alpha)$. For any $m \geq n$, we define $\alpha(\mathbb{V}_{\leq m})$ in the natural way. We will often deal with features that have range in $[0,1]$; we call these $[0,1]$-features.*

*We write $\mathcal{F}$ for the set of all features, $\mathcal{F}_n$ for the set of valuation features of rank $\leq n$, and define an **$\mathcal{F}$-progression** $\overline{\alpha}$ to be a sequence of features such that $\alpha_n \in \mathcal{F}_n$.*

The following valuation features find the price of a sentence on a particular day:

Definition 3.4.2 (Price Feature). *For each $\phi \in \mathcal{S}$ and $n \in \mathbb{N}^+$, we define a **price feature** $\phi^{*n} \in \mathcal{F}_n$ by the formula*

$$\phi^{*n}(\overline{\mathbb{V}}) := \mathbb{V}_n(\phi).$$

We call these "price features" because they will almost always be applied to a market $\overline{\mathbb{P}}$, in which case ϕ^{*n} gives the price $\mathbb{P}_n(\phi)$ of ϕ on day n as a function of $\overline{\mathbb{P}}$.

Very roughly, trading strategies will be linear combinations of sentences where the coefficients are valuation features. The set of all valuation features is not computably enumerable, so we define an expressible subset:

Definition 3.4.3 (Expressible Feature). *An **expressible feature** $\xi \in \mathcal{F}$ is a valuation feature expressible by an algebraic expression built from price features ϕ^{*n} for each $n \in \mathbb{N}^+$ and $\phi \in \mathcal{S}$, rational numbers, addition, multiplication, $\max(-,-)$, and a "safe reciprocation" function $\max(1,-)^{-1}$. See Appendix A.2 for more details and examples. [2]*

*We write $\mathcal{EF}$ for the set of all expressible features, $\mathcal{EF}_n$ for the set of expressible features of rank $\leq n$, and define an **$\mathcal{EF}$-progression** to be a sequence $\overline{\xi}$ such that $\xi_n \in \mathcal{EF}_n$.*

For those familiar with abstract algebra, note that for each n, $\mathcal{EF}_n$ is a commutative ring. We will write $2 - \phi^{*6}$ for the function $\overline{\mathbb{V}} \mapsto 2 - \phi^{*6}(\overline{\mathbb{V}})$ and so on, in the usual way. For example, the feature

$$\xi := \max(0, \phi^{*6} - \psi^{*7})$$

checks whether the value of ϕ on day 6 is higher than the value of ψ on day 7. If so, it returns the difference; otherwise, it returns 0. If ξ is applied to a market $\overline{\mathbb{P}}$, and $\mathbb{P}_6(\phi) = 0.5$ and $\mathbb{P}_7(\psi) = 0.2$, then $\xi(\overline{\mathbb{P}}) = 0.3$. Observe that $\mathrm{rank}(\xi) = 7$, and that ξ is continuous.

The reason for the continuity constraint on valuation features is as follows. Traders will be allowed to use valuation features (which depend on the price history) to decide how many shares of different sentences to buy and sell. This creates a delicate situation, because we'll be constructing a market that has prices which

---

2. In particular, expressible features are a generalization of arithmetic circuits. The specific definition is somewhat arbitrary; what matters is that expressible features be (1) continuous; (2) compactly specifiable in polynomial time; and (3) expressive enough to identify a variety of inefficiencies in a market.

depend on the behavior of certain traders, creating a circular dependency where the prices depend on trades that depend on the prices.

This circularity is related to classic paradoxes of self-trust. What should be the price on a paradoxical sentence χ that says "I am true iff my price is less than 50 cents in this market"? If the price is less than 50¢, then χ pays out \$1, and traders can make a fortune buying χ. If the price is 50¢ or higher, then χ pays out \$0, and traders can make a fortune selling χ. If traders are allowed to have a discontinuous trading strategy—buy χ if $\mathbb{P}(\chi) < 0.5$, sell χ otherwise—then there is no way to find prices that clear the market.

Continuity breaks the circularity, by ensuring that if there's a price where a trader buys χ and a price where they sell χ then there's a price in between where they neither buy nor sell. In Section 5 we will see that this is sufficient to allow stable prices to be found, and in Section 4.11 we will see that it is sufficient to subvert the standard paradoxes of self-reference. The continuity constraint can be interpreted as saying that the trader has only finite-precision access to the market prices—they can see the prices, but there is some $\varepsilon > 0$ such that their behavior is insensitive to an ε shift in prices.

We are almost ready to define trading strategies as a linear combination of sentences with expressible features as coefficients. However, there is one more complication. It will be convenient to record not only the amount of shares bought and sold, but also the amount of cash spent or received. For example, consider again the market order $3\phi - 2\psi$. If it is executed on day 7 in a market $\overline{\overline{\mathbb{P}}}$, and $\mathbb{P}_7(\phi) = 0.4$ and $\mathbb{P}_7(\psi) = 0.3$, then the cost is $3 \cdot 40¢ - 2 \cdot 30¢ = 60¢$. We can record the whole trade as an affine combination $-0.6 + 3\phi - 2\psi$, which can be read as "the trader spent 60 cents to buy 3 shares of ϕ and sell 2 shares of ψ". Extending this idea to the case where the coefficients are expressible features, we get the following notion:

Definition 3.4.4 (Trading Strategy). *A **trading strategy for day** n, also called an **n-strategy**, is an affine combination of the form*

$$T = c + \xi_1\phi_1 + \cdots + \xi_k\phi_k,$$

where $\phi_1, \ldots, \phi_k$ are sentences, $\xi_1, \ldots, \xi_k$ are expressible features of rank $\leq n$, and

$$c = -\sum_i \xi_i\phi_i^{*n}$$

is a "cash term" recording the net cash flow when executing a transaction that buys ξ_i shares of ϕ_i for each i at the prevailing market price. (Buying negative shares is called "selling".) We define $T[1]$ to be c, and $T[\phi]$ to be the coefficient of ϕ in T, which is 0 if $\phi \notin (\phi_1, \ldots, \phi_k)$.

An n-strategy T can be encoded by the tuples $(\phi_1, \ldots \phi_k)$ and $(\xi_1, \ldots \xi_k)$ because the c term is determined by them. Explicitly, by linearity we have

$$T = \xi_1 \cdot (\phi_1 - \phi_1^{*n}) + \cdots + \xi_k \cdot (\phi_k - \phi_k^{*n}),$$

which means any n-strategy can be written as a linear combination of $(\phi_i - \phi_i^{*n})$ terms, each of which means "buy one share of ϕ_i at the prevailing price".

As an example, consider the following trading strategy for day 5:

$$\left[(\neg\neg\phi)^{*5} - \phi^{*5}\right] \cdot \left(\phi - \phi^{*5}\right) + \left[\phi^{*5} - (\neg\neg\phi)^{*5}\right] \cdot \left(\neg\neg\phi - (\neg\neg\phi)^{*5}\right).$$

This strategy compares the price of ϕ on day 5 to the price of $\neg\neg\phi$ on day 5. If the former is less expensive by δ, it purchase δ shares of ϕ at the prevailing prices, and sells δ shares of $\neg\neg\phi$ at the prevailing prices. Otherwise, it does the opposite. In short, this strategy arbitrages ϕ against $\neg\neg\phi$, by buying the cheaper one and selling the more expensive one.

We can now state the key definition of this section:

18

Definition 3.4.5 (Trader)**.** *A **trader** $\overline{T}$ is a sequence $(T_1, T_2, \ldots)$ where each T_n is a trading strategy for day n.*

We can visualize a trader as a person who gets to see the day n, think for a while, and then produce a trading strategy for day n, which will observe the history of market prices up to and including day n and execute a market order to buy and sell different sentences at the prevailing market prices.

We will often consider the set of efficiently computable traders, which have to produce their trading strategy in a time polynomial in n. We can visualize e.c. traders as traders who are computationally limited: each day they get to think for longer and longer—we can imagine them writing computer programs each morning that assist them in their analysis of the market prices—but their total runtime may only grow polynomially in n.

If $s := T_n[\phi] > 0$, we say that $\overline{T}$ buys s shares of ϕ on day n, and if $s < 0$, we say that $\overline{T}$ sells $|s|$ shares of ϕ on day n. Similarly, if $d := T_n[1] > 0$, we say that $\overline{T}$ receives d dollars on day n, and if $d < 0$, we say that $\overline{T}$ pays out $|d|$ dollars on day n.

Each trade T_n has value zero according to $\mathbb{P}_n$, regardless of what market $\overline{\mathbb{P}}$ it is executed in. Clever traders are the ones who make trades that are later revealed by a deductive process $\overline{D}$ to have a high worth (e.g., by purchasing shares of provable sentences when the price is low). As an example, a trader $\overline{T}$ with a basic grasp of arithmetic and skepticism about some of the market $\overline{\mathbb{P}}$'s confident conjectures might execute the following trade orders on day n:

Table 1: Visualizing markets and trades

Sentence	Market prices	Trade
$\phi :\leftrightarrow 1+1 = 2$	$\mathbb{P}_n(\phi) = 90¢$	$T_n[\phi] = 4$ shares
$\psi :\leftrightarrow 1+1 \neq 2$	$\mathbb{P}_n(\psi) = 5¢$	$T_n[\psi] = -3$ shares
$\chi :\leftrightarrow$ "Goldbach's conjecture"	$\mathbb{P}_n(\chi) = 98¢$	$T_n[\chi] = -1$ share

The net value of the shares bought and sold at these prices would be

$$4 \cdot 90¢ - 3 \cdot 5¢ - 1 \cdot 98¢ = \$2.47,$$

so if those three sentences were the only sentences bought and sold by T_n, $T_n[1]$ would be -2.47.

Trade strategies are a special case of affine combinations of sentences:

Definition 3.4.6 (Affine Combination)**.** *An **$\mathcal{F}$-combination** $A : \mathcal{S} \cup \{1\} \to \mathcal{F}_n$ is an affine expression of the form*

$$A := c + \alpha_1 \phi_1 + \cdots + \alpha_k \phi_k,$$

*where $(\phi_1, \ldots, \phi_k)$ are sentences and $(c, \alpha_1, \ldots, \alpha_k)$ are in $\mathcal{F}$. We define $\mathbb{R}$-**combinations**, $\mathbb{Q}$-**combinations**, and **$\mathcal{EF}$-combinations** analogously.*

*We write $A[1]$ for the trailing coefficient c, and $A[\phi]$ for the coefficient of ϕ, which is 0 if $\phi \notin (\phi_1, \ldots, \phi_k)$. The **rank** of A is defined to be the maximum rank among all its coefficients. Given any valuation $\mathbb{V}$, we abuse notation in the usual way and define the **value** of A (according to $\mathbb{V}$) linearly by:*

$$\mathbb{V}(A) := c + \alpha_1 \mathbb{V}(\phi_1) + \cdots + \alpha_k \mathbb{V}(\phi_k).$$

*An **$\mathcal{F}$-combination progression** is a sequence $\overline{A}$ of affine combinations where A_n has rank $\leq n$. An **$\mathcal{EF}$-combination progression** is defined similarly.*

Note that a trade T is an $\mathcal{F}$-combination, and the holdings $T(\overline{\mathbb{P}})$ from T against $\overline{\mathbb{P}}$ is a $\mathbb{Q}$-combination. We will use affine combinations to encode the net holdings $\sum_{i \leq n} T_i(\overline{\mathbb{P}})$ of a trader after interacting with a market $\overline{\mathbb{P}}$, and later to encode linear inequalities that hold between the truth values of different sentences.

3.5 Exploitation

We will now define exploitation, beginning with an example. Let $\mathcal{L}$ be the language of PA, and $\overline{D}$ be a PA-complete deductive process. Consider a market $\overline{\mathbb{P}}$ that assigns $\mathbb{P}_n(\text{"}1+1=2\text{"}) = 0.5$ for all n, and a trader who buys one share of "$1+1=2$" each day. Imagine a reasoner behind the market obligated to buy and sell shares at the listed prices, who is also obligated to pay out \$1 to holders of ϕ-shares if and when $\overline{D}$ says ϕ. Let t be the first day when "$1+1=2$" $\in D_t$. On each day, the reasoner receives 50¢ from $\overline{T}$, but after day t, the reasoner must pay \$1 every day thereafter. They lose 50¢ each day, and $\overline{T}$ gains 50¢ each day, despite the fact that $\overline{T}$ never risked more than \$$t/2$. In cases like these, we say that $\overline{T}$ exploits $\overline{\mathbb{P}}$.

With this example in mind, we define exploitation as follows:

Definition 3.5.1 (Exploitation)**.** *A trader $\overline{T}$ is said to **exploit** a valuation sequence $\overline{\mathbb{V}}$ relative to a deductive process $\overline{D}$ if the set of values*

$$\left\{ \mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{V}}) \right) \,\Big|\, n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n) \right\}$$

is bounded below, but not bounded above.

Given a world $\mathbb{W}$, the number $\mathbb{W}(\sum_{i \leq n} T_i(\overline{\mathbb{P}}))$ is the value of the trader's net holdings after interacting with the market $\overline{\mathbb{P}}$, where a share of ϕ is valued at \$1 if ϕ is true in $\mathbb{W}$ and \$0 otherwise. The set $\{ \mathbb{W}(\sum_{i \leq n} T_i(\overline{\mathbb{P}})) \mid n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n) \}$ is the set of all assessments of $\overline{T}$'s net worth, across all time, according to worlds that were propositionally consistent with $\overline{D}$ at the time. We informally call these *plausible assessments* of the trader's net worth. Using this terminology, Definition 3.5.1 says that a trader exploits the market if their plausible net worth is bounded below, but not above.

Roughly speaking, we can imagine that there is a person behind the market who acts as a market maker, obligated to buy and sell shares at the listed prices. We can imagine that anyone who sold a ϕ-share is obligated to pay \$1 if and when $\overline{D}$ says ϕ. Then, very roughly, a trader exploits the market if they are able to make unbounded returns off of a finite investment.

This analogy is illustrative but incomplete—traders can exploit the market even if they never purchase a sentence that appears in $\overline{D}$. For example, let ϕ and ψ be two sentences such that $(\phi \vee \psi)$ is provable in PA, but such that neither ϕ nor ψ is provable in PA. Consider a trader that bought 10 ϕ-shares at a price of 20¢ each, and 10 ψ-shares at a price of 30¢ each. Once $\overline{D}$ says $(\phi \vee \psi)$, all remaining p.c. worlds will agree that the portfolio $-5 + 10\phi + 10\psi$ has a value of at least $+5$, despite the fact that neither ϕ nor ψ is ever proven. If the trader is allowed to keep buying ϕ and ψ shares at those prices, they would exploit the market, despite the fact that they never buy decidable sentences. In other words, our notion of exploitation rewards traders for arbitrage, even if they arbitrage between sentences that never "pay out".

3.6 Main Result

Recall the logical induction criterion:

Definition 3.0.1 (The Logical Induction Criterion)**.** *A market $\overline{\mathbb{P}}$ is said to satisfy the **logical induction criterion** relative to a deductive process $\overline{D}$ if there is no efficiently computable trader $\overline{T}$ that exploits $\overline{\mathbb{P}}$ relative to $\overline{D}$. A market $\overline{\mathbb{P}}$ meeting this criterion is called a **logical inductor over $\overline{D}$**.*

We may now state our main result:

Theorem 3.6.1. *For any deductive process $\overline{D}$, there exists a computable belief sequence $\overline{\mathbb{P}}$ satisfying the logical induction criterion relative to $\overline{D}$.*

Proof. In Section 5, we show how to take an arbitrary deductive process $\overline{D}$ and construct a computable belief sequence $\overline{\mathtt{LIA}}$. Theorem 5.4.2 shows that $\overline{\mathtt{LIA}}$ is a logical inductor relative to the given $\overline{D}$. □

Definition 3.6.2 (Logical Inductor over Γ)**.** *Given a theory Γ, a logical inductor over a Γ-complete deductive process $\overline{D}$ is called a **logical inductor over Γ**.*

Corollary 3.6.3. *For any recursively axiomatizable theory Γ, there exists a computable belief sequence that is a logical inductor over Γ.*

4 Properties of Logical Inductors

Here is an intuitive argument that logical inductors perform good reasoning under logical uncertainty:

> Consider any polynomial-time method for efficiently identifying patterns in logic. If the market prices don't learn to reflect that pattern, a clever trader can use that pattern to exploit the market. Thus, a logical inductor must learn to identify those patterns.

In this section, we will provide evidence supporting this intuitive argument, by demonstrating a number of desirable properties possessed by logical inductors. The properties that we demonstrate are broken into twelve categories:

1. **Convergence and Coherence:** In the limit, the prices of a logical inductor describe a belief state which is fully logically consistent, and represents a probability distribution over all consistent worlds.

2. **Timely Learning:** For any efficiently computable sequence of theorems, a logical inductor learns to assign them high probability in a timely manner, regardless of how difficult they are to prove. (And similarly for assigning low probabilities to refutable statements.)

3. **Learning Statistical Patterns:** If a sequence of sentences appears pseudo-random to all reasoners with the same runtime as the logical inductor, it learns the appropriate statistical summary (assigning, e.g., 10% probability to the claim "the nth digit of π is a 7" for large n, if digits of π are actually hard to predict).

4. **Calibration and Unbiasedness:** Logical inductors are well-calibrated and, given good feedback, unbiased.

5. **Learning Logical Relationships:** Logical inductors inductively learn to respect logical constraints that hold between different types of claims, such as by ensuring that mutually exclusive sentences have probabilities summing to at most 1.

6. **Non-Dogmatism:** The probability that a logical inductor assigns to an independent sentence ϕ is bounded away from 0 and 1 in the limit, by an amount dependent on the complexity of ϕ. In fact, logical inductors strictly dominate the universal semimeasure in the limit. This means that we can condition logical inductors on independent sentences, and when we do, they perform empirical induction.

7. **Conditionals:** Given a logical inductor $\overline{\mathbb{P}}$, the market given by the conditional probabilities $\overline{\mathbb{P}}(- \mid \psi)$ is a logical inductor over $\overline{D}$ extended to include ψ. Thus, when we condition logical inductors on new axioms, they continue to perform logical induction.

8. **Expectations:** Logical inductors give rise to a well-behaved notion of the expected value of a logically uncertain variable.

9. **Trust in Consistency:** If the theory Γ underlying a logical inductor's deductive process is expressive enough to talk about itself, then the logical inductor learns inductively to trust Γ.

10. **Reasoning about Halting:** If there's an efficient method for generating programs that halt, a logical inductor will learn in a timely manner that those programs halt (often long before having the resources to evaluate them). If there's an efficient method for generating programs that don't halt, a logical inductor will at least learn not to expect them to halt for a very long time.

11. **Introspection:** Logical inductors "know what they know", in that their beliefs about their current probabilities and expectations are accurate.

12. **Self-Trust:** Logical inductors trust their future beliefs.

For the sake of brevity, proofs are deferred to Section 6 and the appendix. Some example proofs are sketched in this section, by outlining discontinuous traders that would exploit any market that lacked the desired property. The deferred proofs define polynomial-time continuous traders that approximate those discontinuous strategies.

In what follows, let $\mathcal{L}$ be a language of propositional logic; let $\mathcal{S}$ be the set of sentences written in $\mathcal{L}$; let $\Gamma \subset \mathcal{S}$ be a computably enumerable set of propositional formulas written in $\mathcal{L}$ (such as PA, where the propositional variables are prime sentences in first-order logic, as discussed in Section 2); and let $\overline{\mathbb{P}}$ be a computable logical inductor over Γ, i.e., a market satisfying the logical induction criterion relative to some Γ-complete deductive process $\overline{D}$. We assume in this section that Γ is consistent.

Note that while the computable belief sequence $\overline{\texttt{LIA}}$ that we define has finite support on each day, in this section we assume only that $\overline{\mathbb{P}}$ is a market. We do this because our results below hold in this more general case, and can be applied to $\overline{\texttt{LIA}}$ as a special case.

In sections 4.8-4.12 we will assume that Γ can represent computable functions. This assumption is not necessary until Section 4.8.

4.1 Convergence and Coherence

Firstly, the market prices of a logical inductor converge:

Theorem 4.1.1 (Convergence). *The limit $\mathbb{P}_\infty : \mathcal{S} \to [0,1]$ defined by*

$$\mathbb{P}_\infty(\phi) := \lim_{n \to \infty} \mathbb{P}_n(\phi)$$

exists for all ϕ.

Proof sketch. (Proof in: 6.1 or B.4.)

Roughly speaking, if $\overline{\mathbb{P}}$ never makes up its mind about ϕ, then it can be exploited by a trader arbitraging shares of ϕ across different days. More precisely, suppose by way of contradiction that the limit $\mathbb{P}_\infty(\phi)$ does not exist. Then for some $p \in [0,1]$ and $\varepsilon > 0$, we have $\mathbb{P}_n(\phi) < p - \varepsilon$ infinitely often and also $\mathbb{P}_n(\phi) > p + \varepsilon$ infinitely often. A trader can wait until $\mathbb{P}_n(\phi) < p - \varepsilon$ and then buy a share in ϕ at the low market price of $\mathbb{P}_n(\phi)$. Then the trader waits until some later m such that $\mathbb{P}_m(\phi) > p + \varepsilon$, and sells back the share in ϕ at the higher price. This trader makes a total profit of 2ε every time $\mathbb{P}_n(\phi)$ oscillates in this way, at no risk, and therefore exploits $\overline{\mathbb{P}}$. Since $\overline{\mathbb{P}}$ implements a logical inductor, this is not possible; therefore the limit $\mathbb{P}_\infty(\phi)$ must in fact exist.

This sketch showcases the main intuition for the convergence of $\overline{\mathbb{P}}$, but elides a number of crucial details. In particular, the trader we have sketched makes use of

discontinuous trading functions, and so is not a well-formed trader. These details are treated in Section 6.1.

Next, the limiting beliefs of a logical inductor represent a coherent probability distribution:

Theorem 4.1.2 (Limit Coherence). *$\mathbb{P}_\infty$ is coherent, i.e., it gives rise to an internally consistent probability measure* Pr *on the set $\mathcal{PC}(\Gamma)$ of all worlds consistent with Γ, defined by the formula*

$$\Pr(\mathbb{W}(\phi) = 1) := \mathbb{P}_\infty(\phi).$$

In particular, if Γ contains the axioms of first-order logic, then $\mathbb{P}_\infty$ defines a probability measure on the set of first-order completions of Γ.

Proof sketch. (Proof in: 6.2 or C.10.)

The limit $\mathbb{P}_\infty(\phi)$ exists by the convergence theorem, so Pr is well-defined. Gaifman (1964) shows that Pr defines a probability measure over $\mathcal{PC}(D_\infty)$ so long as the following three implications hold for all sentences ϕ an ψ:

- If $\Gamma \vdash \phi$, then $\mathbb{P}_\infty(\phi) = 1$,
- If $\Gamma \vdash \neg\phi$, then $\mathbb{P}_\infty(\phi) = 0$,
- If $\Gamma \vdash \neg(\phi \wedge \psi)$, then $\mathbb{P}_\infty(\phi \vee \psi) = \mathbb{P}_\infty(\phi) + \mathbb{P}_\infty(\psi)$.

Let us demonstrate each of these three properties.

First suppose that $\Gamma \vdash \phi$, but $\mathbb{P}_\infty(\phi) = 1 - \varepsilon$ for some $\varepsilon > 0$. Then shares of ϕ will be underpriced, as they are worth 1 in every consistent world, but only cost $1 - \varepsilon$. There is a trader who waits until ϕ is propositionally provable from D_n, and until $\mathbb{P}_n(\phi)$ has approximately converged, and then starts buying shares of ϕ every day at the price $\mathbb{P}_n(\phi)$. Since ϕ has appeared in $\overline{D}$, the shares immediately have a minimum plausible value of \$1. Thus the trader makes $1 - \mathbb{P}_n(\phi) \approx \varepsilon$ profit every day, earning an unbounded total value, contradicting the logical induction criterion. But $\overline{\mathbb{P}}$ cannot be exploited, so $\mathbb{P}_\infty(\phi)$ must be 1.

Similarly, if $\Gamma \vdash \neg\phi$ but $\mathbb{P}_\infty(\phi) = \varepsilon > 0$, then a trader could exploit $\overline{\mathbb{P}}$ by selling off shares in ϕ for a profit of $\mathbb{P}_n(\phi) \approx \varepsilon$ each day.

Finally, suppose that $\Gamma \vdash \neg(\phi \wedge \psi)$, but for some $\varepsilon > 0$,

$$\mathbb{P}_\infty(\phi \vee \psi) = \mathbb{P}_\infty(\phi) + \mathbb{P}_\infty(\psi) \pm \varepsilon.$$

Then there is a trader that waits until $\mathbb{P}_n$ has approximately converged on these sentences, and until $\neg(\phi \wedge \psi)$ is propositionally provable from D_n. At that point it's a good deal to sell (buy) a share in $\phi \vee \psi$, and buy (sell) a share in each of ϕ and ψ; the stocks will have values that cancel out in every plausible world. Thus this trader makes a profit of $\approx \varepsilon$ from the price differential, and can then repeat the process. Thus, they would exploit $\overline{\mathbb{P}}$. But this is impossible, so $\mathbb{P}_\infty$ must be coherent.

Theorem 4.1.2 says that if $\overline{\mathbb{P}}$ were allowed to run forever, and we interpreted its prices as probabilities, then we would find its beliefs to be perfectly consistent. In the limit, $\overline{\mathbb{P}}$ assigns probability 1 to every theorem and 0 to every contradiction. On independent sentences, its beliefs obey the constraints of probability theory; if ϕ provably implies ψ, then the probability of ψ converges to a point no lower than the limiting probability of ϕ, regardless of whether they are decidable. The resulting probabilities correspond to a probability distribution over all possible ways that Γ could be completed.

This justifies interpreting the market prices of a logical inductor as probabilities. Logical inductors are not the first computable procedure for assigning probabilities to sentences in a manner that is coherent in the limit; the algorithm of Demski (2012) also has this property. The main appeal of logical induction is that their beliefs become reasonable in a timely manner, outpacing the underlying deductive process.

4.2   Timely Learning

It is not too difficult to define a reasoner that assigns probability 1 to all (and only) the provable sentences, in the limit: simply assign probability 0 to all sentences, and then enumerate all logical proofs, and assign probability 1 to the proven sentences. The real trick is to recognize patterns in a timely manner, well before the sentences can be proven by slow deduction.

Logical inductors learn to outpace deduction on any efficiently computable sequence of provable statements.[3] To illustrate, consider our canonical example where $D_{\underline{n}}$ is the set of all theorems of PA provable in at most n characters, and suppose $\overline{\phi}$ is an e.c. sequence of theorems which are easy to generate but difficult to prove. Let $f(n)$ be the length of the shortest proof of ϕ_n, and assume that f is some fast-growing function. At any given time n, the statement ϕ_n is ever further out beyond D_n—it might take 1 day to prove ϕ_1, 10 days to prove ϕ_2, 100 days to prove ϕ_3, and so on. One might therefore expect that ϕ_n will also be "out of reach" for $\mathbb{P}_n$, and that we have to wait until a much later day close to $f(n)$ before expecting $\mathbb{P}_{f(n)}(\phi_n)$ to be accurate. However, this is not the case! After some finite time N, $\overline{\mathbb{P}}$ will recognize the pattern and begin assigning high probability to $\overline{\phi}$ in a timely manner.

Theorem 4.2.1 (Provability Induction). *Let $\overline{\phi}$ be an e.c. sequence of theorems. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n 1.$$

Furthermore, let $\overline{\psi}$ be an e.c. sequence of disprovable sentences. Then

$$\mathbb{P}_n(\psi_n) \eqsim_n 0.$$

Proof sketch.   (Proof in: 6.4 or C.3.)

> Consider a trader that acts as follows. First wait until the time a when $\mathbb{P}_a(\phi_a)$ drops below $1 - \varepsilon$ and buy a share of ϕ_a. Then wait until ϕ_a is worth 1 in all worlds plausible at time $f(a)$. Then repeat this process. If $\mathbb{P}_n(\phi_n)$ drops below $1 - \varepsilon$ infinitely often, then this trader makes ε profit infinitely often, off of an initial investment of \$1, and therefore exploits the market. $\overline{\mathbb{P}}$ is inexploitable, so $\mathbb{P}_n(\phi_n)$ must converge to 1. By a similar argument, $\mathbb{P}_n(\psi_n)$ must converge to 0.[4]

In other words, $\overline{\mathbb{P}}$ will learn to start believing ϕ_n by day n at the latest, despite the fact that ϕ_n won't be deductively confirmed until day $f(n)$, which is potentially much later. In colloquial terms, if $\overline{\phi}$ is a sequence of facts that can be generated efficiently, then $\overline{\mathbb{P}}$ inductively learns the pattern, and its belief in $\overline{\phi}$ becomes accurate faster than $\overline{D}$ can computationally verify the individual sentences.

For example, imagine that `prg(n)` is a program with fast-growing runtime, which always outputs either 0, 1, or 2 for all n, but such that there is no proof of this in the general case. Then

$$\text{``}\forall x \colon \texttt{prg}(x) = 0 \lor \texttt{prg}(x) = 1 \lor \texttt{prg}(x) = 2\text{''}$$

is *not* provable. Now consider the sequence of statements

$$\overline{\text{prg012}} := \big(\text{``}\texttt{prg}(\underline{n}) = 0 \lor \texttt{prg}(\underline{n}) = 1 \lor \texttt{prg}(\underline{n}) = 2\text{''}\big)_{n \in \mathbb{N}^+}$$

where each prg012$_n$ states that `prg` outputs a 0, 1, or 2 on that n in particular. Each individual prg012$_n$ is provable (it can be proven by running `prg` on input n), and $\overline{\text{prg012}}$ is efficiently computable (because the sentences themselves can be

---

3. Recall that a sequence $\overline{x}$ is efficiently computable iff there exists a computable function $n \mapsto x_n$ with runtime polynomial in n.

4. The traders sketched here are optimized for ease of proof, not for efficiency—a clever trader trying to profit from low prices on efficiently computable theorems would be able to exploit the market faster than this.

written down quickly, even if `prg` is very difficult to evaluate). Thus, provability induction says that any logical inductor will "learn the pattern" and start assigning high probabilities to each individual prg012_n no later than day n.

Imagine that $\overline{D}$ won't determine the output of `prg(n)` until the $f(n)$th day, by evaluating `prg(n)` in full. Provability induction says that $\overline{\mathbb{P}}$ will eventually recognize the pattern $\overline{\text{prg012}}$ and start assigning high probability to prg012_n no later than the nth day, $f(n) - n$ days before the evaluation finishes. This is true regardless of the size of $f(n)$, so if f is fast-growing, $\overline{\mathbb{P}}$ will outpace $\overline{D}$ by an ever-growing margin.

> **Analogy: Ramanujan and Hardy.** Imagine that the statements $\overline{\phi}$ are being output by an algorithm that uses heuristics to generate mathematical facts without proofs, playing a role similar to the famously brilliant, often-unrigorous mathematician Srinivas Ramanujan. Then $\overline{\mathbb{P}}$ plays the historical role of the beliefs of the rigorous G.H. Hardy who tries to verify those results according to a slow deductive process ($\overline{D}$). After Hardy ($\overline{\mathbb{P}}$) verifies enough of Ramanujan's claims ($\phi_{\leq n}$), he begins to trust Ramanujan, even if the proofs of Ramanujan's later conjectures are incredibly long, putting them ever-further beyond Hardy's current abilities to rigorously verify them. In this story, Hardy's inductive reasoning (and Ramanujan's also) outpaces his deductive reasoning.

This idiom of assigning the right probabilities to ϕ_n no later than day n will be common throughout the paper, so we give it a name.

Definition 4.2.2 (Timely Manner). *Let $\overline{\phi}$ be an e.c. sequence of sentences, and $\overline{p}$ be an e.c. sequence of rational numbers. We say that $\overline{\mathbb{P}}$ assigns $\overline{p}$ to $\overline{\phi}$ in a **timely manner** if for every $\varepsilon > 0$, there exists a time N such that for all $n > N$,*

$$|\mathbb{P}_n(\phi_n) - p_n| < \varepsilon.$$

In other words, $\overline{\mathbb{P}}$ assigns $\overline{p}$ to $\overline{\phi}$ in a timely manner if

$$\mathbb{P}_n(\phi_n) \eqsim_n p_n.$$

Note that there are no requirements on how large N gets as a function of ε. As such, when we say that $\overline{\mathbb{P}}$ assigns probabilities $\overline{p}$ to $\overline{\phi}$ in a timely manner, it may take a very long time for convergence to occur. (See Section 5.5 for a discussion.)

As an example, imagine the reasoner who recognizes that sentences of the form "$1 + 1 + \cdots + 1$ is even" are true iff the number of ones is even. Let $\overline{\phi}$ be the sequence where ϕ_n is the version of that sentence with $2n$ ones. If the reasoner starts writing a probability near 100% in the ϕ_n cell by day n at the latest, then intuitively, she has begun incorporating the pattern into her beliefs, and we say that she is assigning high probabilities to $\overline{\phi}$ in a timely manner.

We can visualize ourselves as taking $\overline{\mathbb{P}}$'s belief states, sorting them by $\overline{\phi}$ on one axis and days on another, and then looking at the main diagonal of cells, to check the probability of each ϕ_n on day n. Checking the nth sentence on the nth day is a rather arbitrary choice, and we might hope that a good reasoner would assign high probabilities to e.c. sequences of theorems at a faster rate than that. It is easy to show that this is the case, by the closure properties of efficient computability. For example, if $\overline{\phi}$ is an e.c. sequence of theorems, then so are $\overline{\phi}_{2n}$ and $\overline{\phi}_{2n+1}$, which each enumerate half of $\overline{\phi}$ at twice the speed, so by Theorem 4.2.1 (Provability Induction), $\overline{\mathbb{P}}$ will eventually learn to believe $\overline{\phi}$ at a rate of at least two per day. Similarly, $\overline{\mathbb{P}}$ will learn to believe $\overline{\phi}_{3n}$ and $\overline{\phi}_{n^2}$ and $\overline{\phi}_{10n^3+3}$ in a timely manner, and so on. Thus, up to polynomial transformations, it doesn't really matter which diagonal we check when checking whether a logical inductor has begun "noticing a pattern".

Furthermore, we will show that if $\overline{\mathbb{P}}$ assigns the correct probability on the main diagonal, then $\overline{\mathbb{P}}$ also learns to keep them there:

Theorem 4.2.3 (Persistence of Knowledge). *Let $\overline{\phi}$ be an e.c. sequence of sentences, and $\overline{p}$ be an e.c. sequence of rational-number probabilities. If $\mathbb{P}_\infty(\phi_n) \eqsim_n p_n$, then*

$$\sup_{m \geq n} |\mathbb{P}_m(\phi_n) - p_n| \eqsim_n 0.$$

Furthermore, if $\mathbb{P}_\infty(\phi_n) \lesssim_n p_n$, then

$$\sup_{m \geq n} \mathbb{P}_m(\phi_n) \lesssim_n p_n,$$

and if $\mathbb{P}_\infty(\phi_n) \gtrsim_n p_n$, then

$$\inf_{m \geq n} \mathbb{P}_m(\phi_n) \gtrsim_n p_n.$$

(Proof in: B.6.)

In other words, if $\overline{\mathbb{P}}$ assigns $\overline{p}$ to $\overline{\phi}$ in the limit, then $\overline{\mathbb{P}}$ learns to assign probability near p_n to ϕ_n at all times $m \geq n$. This theorem paired with the closure properties of the set of efficiently computable sequences means that checking the probability of ϕ_n on the nth day is a fine way to check whether $\overline{\mathbb{P}}$ has begun recognizing a pattern encoded by $\overline{\phi}$. As such, we invite the reader to be on the lookout for statements of the form $\mathbb{P}_n(\phi_n)$ as signs that $\overline{\mathbb{P}}$ is recognizing a pattern, often in a way that outpaces the underlying deductive process.

Theorems 4.2.1 (Provability Induction) and 4.2.3 (Persistence of Knowledge) only apply when the pattern of limiting probabilities is itself efficiently computable. For example, consider the sequence of sentences

$$\overline{\pi\mathrm{Aeq7}} := \left(\text{``}\underline{\pi}[\underline{\mathrm{Ack}}(\underline{n}, \underline{n})] = 7\text{''} \right)_{n \in \mathbb{N}^+}$$

where $\pi[i]$ is the ith digit in the decimal expansion of π and Ack is the Ackermann function. Each individual sentence is decidable, so the limiting probabilities are 0 for some $\pi\mathrm{Aeq7}_n$ and 1 for others. But that pattern of 1s and 0s is not efficiently computable (assuming there is no efficient way to predict the Ackermann digits of π), so provability induction has nothing to say on the topic.

In cases where the pattern of limiting probabilities are not e.c., we can still show that if $\overline{\mathbb{P}}$ is going to make its probabilities follow a certain pattern eventually, then it learns to make its probabilities follow that pattern in a timely manner. For instance, assume that each individual sentence $\pi\mathrm{Aeq7}_n$ (for $n > 4$) is going to spend a long time sitting at 10% probability before eventually being resolved to either 1 or 0. Then $\overline{\mathbb{P}}$ will learn to assign $\mathbb{P}_n(\pi\mathrm{Aeq7}_n) \approx 0.1$ in a timely manner:

Theorem 4.2.4 (Preemptive Learning). *Let $\overline{\phi}$ be an e.c. sequence of sentences. Then*

$$\liminf_{n \to \infty} \mathbb{P}_n(\phi_n) = \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{P}_m(\phi_n).$$

Furthermore,

$$\limsup_{n \to \infty} \mathbb{P}_n(\phi_n) = \limsup_{n \to \infty} \inf_{m \geq n} \mathbb{P}_m(\phi_n).$$

(Proof in: B.3.)

Let's unpack Theorem 4.2.4. The quantity $\sup_{m \geq n} \mathbb{P}_m(\phi_n)$ is an upper bound on the price $\mathbb{P}_m(\phi_n)$ on or after day n, which we can interpret as the highest price tag that that $\overline{\mathbb{P}}$ will ever put on ϕ_n after we first start checking it on n. We can imagine a sequence of these values: On day n, we start watching ϕ_n. As time goes on, its price travels up and down until eventually settling somewhere. This happens for each n. The limit infimum of $\sup_{m \geq n} \mathbb{P}_m(\phi_n)$ is the greatest lower bound p past which a generic ϕ_n (for n large) will definitely be pushed after we started watching it. Preemptive Learning says that if $\overline{\mathbb{P}}$ always eventually pushes ϕ_n up to a probability at least p, then it will learn to assign each ϕ_n a probability at least p in a timely manner (and similarly for least upper bounds).

For example, if each individual $\pi\mathrm{Aeq7}_n$ is *eventually* recognized as a claim about digits of π and placed at probability 10% for a long time before being resolved, then $\overline{\mathbb{P}}$ learns to assign it probability 10% on the main diagonal. In general, if $\overline{\mathbb{P}}$ is going to learn a pattern eventually, it learns it in a timely manner.

This leaves open the question of whether a logical inductor $\overline{\mathbb{P}}$ is smart enough to recognize that the $\overline{\pi\mathrm{Aeq7}}$ should each have probability 10% before they are settled (assuming the Ackermann digits of π are hard to predict). We will return to that question in Section 4.4, but first, we examine the reverse question.

4.3   Calibration and Unbiasedness

Theorem 4.2.1 (Provability Induction) shows that logical inductors are good at detecting patterns in what is provable. Next, we ask: when a logical inductor learns a pattern, when must that pattern be real? In common parlance, a source of probabilistic estimates is called *well calibrated* if among statements where it assigns a probability near p, the estimates are correct with frequency roughly p.

In the case of reasoning under logical uncertainty, measuring calibration is not easy. Consider the sequence clusters constructed from correlated clusters of size 1, 10, 100, 1000, ..., where the truth value of each cluster is determined by the parity of a late digit of π:

$$\mathrm{clusters}_1 :\leftrightarrow \text{``}\pi[\mathrm{Ack}(1,1)] \text{ is even''}$$
$$\mathrm{clusters}_2 :\leftrightarrow \cdots :\leftrightarrow \mathrm{clusters}_{11} :\leftrightarrow \text{``}\pi[\mathrm{Ack}(2,2)] \text{ is even''}$$
$$\mathrm{clusters}_{12} :\leftrightarrow \cdots :\leftrightarrow \mathrm{clusters}_{111} :\leftrightarrow \text{``}\pi[\mathrm{Ack}(3,3)] \text{ is even''}$$
$$\mathrm{clusters}_{112} :\leftrightarrow \cdots :\leftrightarrow \mathrm{clusters}_{1111} :\leftrightarrow \text{``}\pi[\mathrm{Ack}(4,4)] \text{ is even''}$$

and so on. A reasoner who can't predict the parity of the Ackermann digits of π should assign 50% (marginal) probability to any individual $\mathrm{clusters}_n$ for n large. But consider what happens if the 9th cluster turns out to be true, and the next billion sentences are all true. A reasoner who assigned 50% to those billion sentences was assigning the *right* probabilities, but their calibration is abysmal: on the billionth day, they have assigned 50% probability a billion sentences that were overwhelmingly true. And if the 12th cluster comes up false, then on the trillionth day, they have assigned 50% probability to a *trillion* sentences that were overwhelmingly false! In cases like these, the frequency of truth oscillates eternally, and the good reasoner only appears well-calibrated on the rare days where it crosses 50%.

The natural way to correct for correlations such as these is to check $\overline{\mathbb{P}}$'s conditional probabilities instead of its marginal probabilities. This doesn't work very well in our setting, because given a logical sentence ϕ, the quantity that we care about will almost always be the marginal probability of ϕ. The reason we deal with sequences is because that lets us show that ϕ has reasonable probabilities relative to various related sentences. For example, if $\phi := \text{``}\mathtt{prg}(32) = 17\text{''}$, then we can use our theorems to relate the probability of $\phi$ to the probability of the sequence $(\text{``}\mathtt{prg}(\underline{n}) = 17\text{''})_{n \in \mathbb{N}^+}$, and to the sequence $(\text{``}\mathtt{prg}(32) = \underline{n}\text{''})_{n \in \mathbb{N}^+}$, and to the sequence $(\text{``}\mathtt{prg}(\underline{n}) > \underline{n}\text{''})_{n \in \mathbb{N}^+}$, and so on, to show that ϕ eventually has reasonable beliefs about $\mathtt{prg}$ (hopefully before $\overline{\mathbb{P}}$ has the resources to simply evaluate $\mathtt{prg}$ on input 32). But at the end of the day, we'll want to reason about the marginal probability of ϕ itself. In this case, approximately-well-calibrated conditional probabilities wouldn't buy us much: there are 2^{n-1} possible truth assignments to the first $n-1$ elements of $\overline{\phi}$, so if we try to compute the marginal probability of ϕ_n from all the different conditional probabilities, exponentially many small errors would render the answer useless. Furthermore, intuitively, if $\overline{\phi}$ is utterly unpredictable to $\overline{\mathbb{P}}$, then the probabilities of all the different truth assignments to $\phi_{\leq n-1}$ will go to 0 as n gets large, which means the conditional probabilities won't necessarily be reasonable. (In Section 4.4 will formalize a notion of pseudorandomness.)

Despite these difficulties, we can recover some good calibration properties on the marginal probabilities if we either (a) restrict our consideration to sequences where the average frequency of truth converges; or (b) look at subsequences of $\overline{\phi}$ where $\overline{\mathbb{P}}$

27

has "good feedback" about the truth values of previous elements of the subsequence, in a manner defined below.

To state our first calibration property, we will define two different sorts of indicator functions that will prove useful in many different contexts.

Definition 4.3.1 (Theorem Indicator)**.** *Given a sentence ϕ, define* $\text{Thm}_\Gamma(\phi)$ *to be 1 if $\Gamma \vdash \phi$ and 0 otherwise.*

Definition 4.3.2 (Continuous Threshold Indicator)**.** *Let $\delta > 0$ be a rational number, and x and y be real numbers. We then define*

$$\text{Ind}_\delta(x > y) := \begin{cases} 0 & \text{if } x \leq y \\ \dfrac{x-y}{\delta} & \text{if} \quad y < x \leq y + \delta \\ 1 & \text{if} \qquad\qquad y + \delta < x. \end{cases}$$

Notice that $\text{Ind}_\delta(x > y)$ has no false positives, and that it is linear in the region between y and $y + \delta$. We define $\text{Ind}_\delta(x < y)$ analogously, and we define

$$\text{Ind}_\delta(a < x < b) := \min(\text{Ind}_\delta(x > a), \text{Ind}_\delta(x < b)).$$

Observe that we can generalize this definition to the case where x and y are expressible features, in which case $\text{Ind}_\delta(x > y)$ is an expressible $[0,1]$-feature.

Now we can state our calibration theorem.

Theorem 4.3.3 (Recurring Calibration)**.** *Let $\overline{\phi}$ be an e.c. sequence of decidable sentences, a and b be rational numbers, $\overline{\delta}$ be an e.c. sequence of positive rational numbers, and suppose that $\sum_n \left(\text{Ind}_{\delta_i}(a < \mathbb{P}_i(\phi_i) < b)\right)_{i \in \mathbb{N}^+} = \infty$. Then, if the sequence*

$$\left(\frac{\sum_{i \leq n} \text{Ind}_{\delta_i}(a < \mathbb{P}_i(\phi_i) < b) \cdot \text{Thm}_\Gamma(\phi_i)}{\sum_{i \leq n} \text{Ind}_{\delta_i}(a < \mathbb{P}_i(\phi_i) < b)} \right)_{n \in \mathbb{N}^+}$$

converges, it converges to a point in $[a, b]$. Furthermore, if it diverges, it has a limit point in $[a, b]$. (Proof in: D.3.)

Roughly, this says that if $\mathbb{P}_n(\phi_n) \approx 80\%$ infinitely often, then if we look at the subsequence where it's 80%, the limiting frequency of truth on that subsequence is 80% (if it converges).

In colloquial terms, on subsequences where $\overline{\mathbb{P}}$ says 80% and it makes sense to talk about the frequency of truth, the frequency of truth is 80%, i.e., $\overline{\mathbb{P}}$ isn't seeing shadows. If the frequency of truth diverges—as in the case with clusters—then $\overline{\mathbb{P}}$ is still well-calibrated infinitely often, but its calibration might still appear abysmal at times (if they can't predict the swings).

Note that calibration alone is not a very strong property: a reasoner can always cheat to improve their calibration (i.e., by assigning probability 80% to things that they're sure are true, in order to bring up the average truth of their "80%" predictions). What we really want is some notion of "unbiasedness", which says that there is no efficient method for detecting a predictable bias in a logical inductor's beliefs. This is something we can get on sequences where the limiting frequency of truth converges, though again, if the limiting frequency of truth diverges, all we can guarantee is a limit point.

Definition 4.3.4 (Divergent Weighting)**.** *A **divergent weighting** $\overline{w} \in [0,1]^{\mathbb{N}^+}$ is an infinite sequence of real numbers in $[0,1]$, such that $\sum_n w_n = \infty$.*

Note that divergent weightings have codomain $[0,1]$ as opposed to $\{0,1\}$, meaning the weightings may single out fuzzy subsets of the sequence. For purposes of intuition, imagine that $\overline{w}$ is a sequence of 0s and 1s, in which case each w can be interpreted as a subsequence. The constraint that the $w(n)$ sum to ∞ ensures that this subsequence is infinite.

Definition 4.3.5 (Generable From $\overline{\mathbb{P}}$)**.** *A sequence of rational numbers $\overline{q}$ is called* ***generable from*** $\overline{\mathbb{P}}$ *if there exists an e.c. $\mathcal{EF}$-progression $\overline{q^\dagger}$ such that $q_n^\dagger(\overline{\mathbb{P}}) = q_n$ for all n. In this case we say that $\overline{q}$ is $\overline{\mathbb{P}}$-**generable**. $\overline{\mathbb{P}}$-generable $\mathbb{R}$-sequences, $\mathbb{Q}$-combination sequences, and $\mathbb{R}$-combination sequences are defined analogously.*

Divergent weightings generable from $\overline{\mathbb{P}}$ are fuzzy subsequences that are allowed to depend continuously (via expressible market features) on the market history. For example, the sequence $(\mathrm{Ind}_{0.01}(\mathbb{P}_n(\phi_n) > 0.5))_{n \in \mathbb{N}^+}$ is a $\overline{\mathbb{P}}$-generable sequence that singles out all times n when $\mathbb{P}_n(\phi_n)$ is greater than 50%. Note that the set of $\overline{\mathbb{P}}$-generable divergent weightings is larger than the set of e.c. divergent weightings, as the $\overline{\mathbb{P}}$-generable weightings are allowed to vary continuously with the market prices.

Theorem 4.3.6 (Recurring Unbiasedness)**.** *Given an e.c. sequence of decidable sentences $\overline{\phi}$ and a $\overline{\mathbb{P}}$-generable divergent weighting $\overline{w}$, the sequence*

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(\phi_i) - \mathrm{Thm}_\Gamma(\phi_i))}{\sum_{i \leq n} w_i}$$

has 0 as a limit point. In particular, if it converges, it converges to 0.

(Proof in: D.2.)

Letting $\overline{w} = (1, 1, \ldots)$, this theorem says that the difference between the average probability $\mathbb{P}_n(\phi_n)$ and the average frequency of truth is 0 infinitely often (and 0 always, if the latter converges). Letting each w_n be $\mathrm{Ind}_\delta(a < \mathbb{P}_n(\phi_n) < b)$, we recover Theorem 4.3.3 (Recurring Calibration). In general, the fraction in Theorem 4.3.6 can be interpreted as a measure of the "bias" of $\overline{\mathbb{P}}$ on the fuzzy subsequence of $\overline{\phi}$ singled out by w. Then this theorem says that $\overline{\mathbb{P}}$ is unbiased on all $\overline{\mathbb{P}}$-generable subsequences where the frequency of truth converges (and unbiased infinitely often on subsequences where it diverges). Thus, if an e.c. sequence of sentences can be decomposed (by any $\overline{\mathbb{P}}$-generable weighting) into subsequences where the frequency of truth converges, then $\overline{\mathbb{P}}$ learns to assign probabilities such that there is no efficient method for detecting a predictable bias in its beliefs.

However, not every sequence can be broken down into well-behaved subsequences by a $\overline{\mathbb{P}}$-generable divergent weighting (if, for example, the truth values move "pseudorandomly" in correlated clusters, as in the case of $\overline{\mathrm{clusters}}$). In these cases, it is natural to wonder whether there are any conditions where $\overline{\mathbb{P}}$ will be unbiased anyway. Below, we show that the bias converges to zero whenever the weighting $\overline{w}$ is sparse enough that $\overline{\mathbb{P}}$ can gather sufficient feedback about ϕ_n in between guesses:

Definition 4.3.7 (Deferral Function)**.** *A function $f : \mathbb{N}^+ \to \mathbb{N}^+$ is called a **deferral function** if*

1. *$f(n) > n$ for all n, and*

2. *$f(n)$ can be computed in time polynomial in $f(n)$, i.e., if there is some algorithm and a polynomial function h such that for all n, the algorithm computes $f(n)$ within $h(f(n))$ steps.*

*If f is a deferral function, we say that f **defers** n to $f(n)$.*

Theorem 4.3.8 (Unbiasedness From Feedback)**.** *Let $\overline{\phi}$ be any e.c. sequence of decidable sentences, and $\overline{w}$ be any $\overline{\mathbb{P}}$-generable divergent weighting. If there exists a strictly increasing deferral function f such that the support of $\overline{w}$ is contained in the image of f and $\mathrm{Thm}_\Gamma(\phi_{f(n)})$ is computable in $\mathcal{O}(f(n+1))$ time, then*

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(\phi_i) - \mathrm{Thm}_\Gamma(\phi_i))}{\sum_{i \leq n} w_i} \approx_n 0.$$

In this case, we say "$\overline{w}$ allows good feedback on $\overline{\phi}$".

(Proof in: D.5.)

In other words, $\overline{\mathbb{P}}$ is unbiased on any subsequence of the data where a polynomial-time machine can figure out how the previous elements of the subsequence turned out before $\overline{\mathbb{P}}$ is forced to predict the next one. This is perhaps the best we can hope for: On ill-behaved sequences such as $\overline{\text{clusters}}$, where the frequency of truth diverges and (most likely) no polynomial-time algorithm can predict the jumps, the $\mathbb{P}_n(\phi_n)$ might be pure guesswork.

So how well does $\overline{\mathbb{P}}$ perform on sequences like $\overline{\text{clusters}}$? To answer, we turn to the question of how $\overline{\mathbb{P}}$ behaves in the face of sequences that it finds utterly unpredictable.

4.4   Learning Statistical Patterns

Consider the digits in the decimal expansion of π. A good reasoner thinking about the $10^{1,000,000}$th digit of π, in lieu of any efficient method for predicting the digit before they must make their prediction, should assign roughly 10% probability to that digit being a 7. We will now show that logical inductors learn statistical patterns of this form.

To formalize this claim, we need some way of formalizing the idea that a sequence is "apparently random" to a reasoner. Intuitively, this notion must be defined relative to a specific reasoner's computational limitations. After all, the digits of π are perfectly deterministic; they only appear random to a reasoner who lacks the resources to compute them. Roughly speaking, we will define a sequence to be pseudorandom (relative to $\overline{\mathbb{P}}$) if there is no e.c. way to single out any one subsequence that is more likely true than any other subsequence, not even using expressions written in terms of the market prices (by way of expressible features):

Definition 4.4.1 (Pseudorandom Sequence). *Given a set S of divergent weightings (Definition 4.3.4), a sequence $\overline{\phi}$ of decidable sentences is called **pseudorandom with frequency p** over S if, for all weightings $\overline{w} \in S$,*

$$\lim_{n \to \infty} \frac{\sum_{i \le n} w_i \cdot \mathrm{Thm}_\Gamma(\phi_i)}{\sum_{i \le n} w_i}$$

exists and is equal to p.

Note that if the sequence $\overline{\phi}$ is *actually* randomly generated (say, by adding $(c_1, c_2, \dots)$ to the language of Γ, and tossing a coin weighted with probability p towards heads for each i, to determine whether to add c_i or $\neg c_i$ as an axiom) then $\overline{\phi}$ is pseudorandom with frequency p almost surely.[5] Now:

Theorem 4.4.2 (Learning Pseudorandom Frequencies). *Let $\overline{\phi}$ be an e.c. sequence of decidable sentences. If $\overline{\phi}$ is pseudorandom with frequency p over the set of all $\overline{\mathbb{P}}$-generable divergent weightings, then*

$$\mathbb{P}_n(\phi_n) \eqsim_n p.$$

(Proof in: 6.5 or D.8.)

For example, consider again the sequence $\overline{\pi\mathrm{Aeq}7}$ where the nth element says that the $\mathrm{Ack}(n,n)$th decimal digit of π is a 7. The individual $\pi\mathrm{Aeq}7_n$ statements are easy to write down (i.e., efficiently computable), but each one is difficult to decide. Assuming there's no good way to predict the Ackermann digits of π using a $\overline{\mathbb{P}}$-generable divergent weighting, $\overline{\mathbb{P}}$ will assign probability 10% to each $\pi\mathrm{Aeq}7_n$ in a timely manner, while it waits for the resources to determine whether the sentence is true or false. Of course, on each individual $\pi\mathrm{Aeq}7_n$, $\overline{\mathbb{P}}$'s probability will go to 0 or 1 eventually, i.e., $\lim_{m \to \infty} \mathbb{P}_m(\pi\mathrm{Aeq}7_n) \in \{0, 1\}$.

---

5. Note that actually adding randomness to Γ in this fashion is not allowed, because we assumed that the axioms of Γ are recursively computable. It is possible to construct a logical inductors that have access to a source of randomness, by adding one bit of randomness to the market each day, but that topic is beyond the scope of this paper.

Theorem 4.4.2 still tells us nothing about how $\overline{\mathbb{P}}$ handles $\overline{\text{clusters}}$ (defined above), because the frequency of truth in that sequence diverges, so it does not count as pseudorandom by the above definition. To handle this case we will weaken our notion of pseudorandomness, so that it includes more sequences, yielding a stronger theorem. We will do this by allowing sequences to count as pseudorandom so long as the limiting frequency of truth converges on "independent subsequences" where the $n + 1$st element of the subsequence doesn't come until after the nth element can be decided, as described below. Refer to Garrabrant, Soares, and Taylor (2016) for a discussion of why this is a good way to broaden the set of sequences that count as pseudorandom.

Definition 4.4.3 (f-Patient Divergent Weighting)**.** *Let f be a deferral function. We say that a divergent weighting $\overline{w}$ is **f-patient** if there is some constant C such that, for all n,*

$$\sum_{i=n}^{f(n)} w_i(\overline{\mathbb{P}}) \leq C$$

In other words, $\overline{w}$ is f-patient if the weight it places between days n and $f(n)$ is bounded.

While we are at it, we will also strengthen Theorem 4.4.2 in three additional ways: we will allow the probabilities on the sentences to vary with time, and with the market prices, and we will generalize $\eqsim_n$ to $\gtrsim_n$ and $\lesssim_n$.

Definition 4.4.4 (Varied Pseudorandom Sequence)**.** *Given a deferral function f, a set S of f-patient divergent weightings, an e.c. sequence $\overline{p}$ of Γ-decidable sentences, and a $\overline{\mathbb{P}}$-generable sequence $\overline{p}$ of rational probabilities, $\overline{\phi}$ is called a **$\overline{p}$-varied pseudorandom sequence** (relative to S) if, for all $\overline{w} \in S$,*

$$\frac{\sum_{i \leq n} w_i \cdot (p_i - \text{Thm}_\Gamma(\phi_i))}{\sum_{i \leq n} w_i} \eqsim_n 0.$$

*Furthermore, we can replace $\eqsim_n$ with $\gtrsim_n$ or $\lesssim_n$, in which case we say $\overline{\phi}$ is **varied pseudorandom above $\overline{p}$** or **varied pseudorandom below $\overline{p}$**, respectively.*

Theorem 4.4.5 (Learning Varied Pseudorandom Frequencies)**.** *Given an e.c. sequence $\overline{\phi}$ of Γ-decidable sentences and a $\overline{\mathbb{P}}$-generable sequence $\overline{p}$ of rational probabilities, if there exists some f such that $\overline{\phi}$ is $\overline{p}$-varied pseudorandom (relative to all f-patient $\overline{\mathbb{P}}$-generable divergent weightings), then*

$$\mathbb{P}_n(\phi_n) \eqsim_n p_n.$$

Furthermore, if $\overline{\phi}$ is varied pseudorandom above or below $\overline{p}$, then the $\eqsim_n$ may be replaced with $\gtrsim_n$ or $\lesssim_n$ (respectively). (Proof in: D.7.)

Thus we see that $\overline{\mathbb{P}}$ does learn to assign marginal probabilities $\mathbb{P}_n(\text{clusters}_n) \approx 0.5$, assuming the Ackermann digits of π are actually difficult to predict. Note that while Theorem 4.4.5 requires each p_n to be rational, the fact that the theorem is generalized to varied pseudorandom above/below sequences means that Theorem 4.4.5 is a strict generalization of Theorem 4.4.2 (Learning Pseudorandom Frequencies).

In short, Theorem 4.4.5 shows that logical inductors reliably learn in a timely manner to recognize appropriate statistical patterns, whenever those patterns (which may vary over time and with the market prices) are the best available method for predicting the sequence using $\overline{\mathbb{P}}$-generable methods.

4.5   Learning Logical Relationships

Most of the above properties discuss the ability of a logical inductor to recognize patterns in a single sequence—for example, they recognize e.c. sequences of theorems

in a timely manner, and they fall back on the appropriate statistical summaries in the face of pseudorandomness. We will now examine the ability of logical inductors to learn relationships between sequences.

Let us return to the example of the computer program `prg` which outputs either 0, 1, or 2 on all inputs, but for which this cannot be proven in general by Γ. Theorem 4.2.1 (Provability Induction) says that the pattern

$$\overline{\text{prg012}} := \left(\text{``}\texttt{prg}(\underline{n}) = 0 \vee \texttt{prg}(\underline{n}) = 1 \vee \texttt{prg}(\underline{n}) = 2\text{''}\right)_{n \in \mathbb{N}^+}$$

will be learned, in the sense that $\overline{\mathbb{P}}$ will assign each prg012_n a probability near 1 in a timely manner. But what about the following three individual sequences?

$$\overline{\text{prg0}} := \left(\text{``}\texttt{prg}(\underline{n}) = 0\text{''}\right)_{n \in \mathbb{N}^+}$$
$$\overline{\text{prg1}} := \left(\text{``}\texttt{prg}(\underline{n}) = 1\text{''}\right)_{n \in \mathbb{N}^+}$$
$$\overline{\text{prg2}} := \left(\text{``}\texttt{prg}(\underline{n}) = 2\text{''}\right)_{n \in \mathbb{N}^+}$$

None of the three sequences is a sequence of only theorems, so provability induction does not have much to say. If they are utterly pseudorandom relative to r, then Theorem 4.4.5 (Learning Varied Pseudorandom Frequencies) says that $\overline{\mathbb{P}}$ will fall back on the appropriate statistical summary, but that tells us little in cases where there are predictable non-conclusive patterns (e.g., if `prg(i)` is more likely to output 2 when `helper(i)` outputs 17). In fact, if $\overline{\mathbb{P}}$ is doing good reasoning, the probabilities on the $(\text{prg0}_n, \text{prg1}_n, \text{prg2}_n)$ triplet ought to shift, as $\overline{\mathbb{P}}$ gains new knowledge about related facts and updates its beliefs. How could we tell if those intermediate beliefs were reasonable?

One way is to check their sum. If $\overline{\mathbb{P}}$ believes that `prg(i)` $\in \{0, 1, 2\}$ and it knows how disjunction works, then it should be the case that whenever $\mathbb{P}_n(\text{prg012}_t) \approx 1$, $\mathbb{P}_n(\text{prg0}_t) + \mathbb{P}_n(\text{prg1}_t) + \mathbb{P}_n(\text{prg2}_t) \approx 1$. And this is precisely the case. In fact, logical inductors recognize mutual exclusion between efficiently computable tuples of any size, in a timely manner:

Theorem 4.5.1 (Learning Exclusive-Exhaustive Relationships)**.** *Let $\overline{\phi^1}, \ldots, \overline{\phi^k}$ be k e.c. sequences of sentences, such that for all n, Γ proves that $\phi_n^1, \ldots, \phi_n^k$ are exclusive and exhaustive (i.e. exactly one of them is true). Then*

$$\mathbb{P}_n(\phi_n^1) + \cdots + \mathbb{P}_n(\phi_n^k) \eqsim_n 1.$$

Proof sketch. (Proof in: C.11.)

> Consider the trader that acts as follows. On day n, they check the prices of $\phi_n^1 \ldots \phi_n^k$. If the sum of the prices is higher (lower) than 1 by some fixed threshold $\varepsilon > 0$, they sell (buy) a share of each, wait until the values of the shares are the same in every plausible world, and make a profit of ε. (It is guaranteed that eventually, in every plausible world exactly one of the shares will be valued at 1.) If the sum goes above $1 + \varepsilon$ (below $1 - \varepsilon$) on the main diagonal infinitely often, this trader exploits $\overline{\mathbb{P}}$. Logical inductors are inexploitable, so it must be the case that the sum of the prices goes to 1 along the main diagonal.

This theorem suggests that logical inductors are good at learning to assign probabilities that respect logical relationships between related sentences. To show that this is true in full generality, we will generalize Theorem 4.5.1 to any linear inequalities that hold between the actual truth-values of different sentences.

First, we define the following convention:

Convention 4.5.2 (Constraint)**.** *An $\mathbb{R}$-combination A can be viewed as a **constraint**, in which case we say that a valuation $\mathbb{V}$ **satisfies** the constraint if $\mathbb{V}(A) \geq 0$.*

For example, the constraint

$$\text{AND} := -2 + \phi + \psi$$

says that both ϕ and ψ are true, and it is satisfied by $\mathbb{W}$ iff $\mathbb{W}(\phi) = \mathbb{W}(\psi) = 1$. As another example, the pair of constraints

$$\text{XOR} := (1 - \phi - \psi, \phi + \psi - 1)$$

say that exactly one of ϕ and ψ is true, and are satisfied by $\mathbb{P}_7$ iff $\mathbb{P}_7(\phi) + \mathbb{P}_7(\psi) = 1$.

Definition 4.5.3 (Bounded Combination Sequence). *By $\mathcal{BCS}(\overline{\mathbb{P}})$ (mnemonic: **bounded combination sequences**) we denote the set of all $\overline{\mathbb{P}}$-generable $\mathbb{R}$-combination sequences $\overline{A}$ that are bounded, in the sense that there exists some bound b such that $\|A_n\|_1 \leq b$ for all n, where $\|-\|_1$ includes the trailing coefficient.*

Theorem 4.5.4 (Affine Provability Induction). *Let $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$ and $b \in \mathbb{R}$. If, for all consistent worlds $\mathbb{W} \in \mathcal{PC}(\Gamma)$ and all $n \in \mathbb{N}^+$, it is the case that $\mathbb{W}(A_n) \geq b$, then*

$$\mathbb{P}_n(A_n) \gtrsim_n b,$$

and similarly for $=$ and $\eqsim_n$, and for $\leq$ and $\lesssim_n$. (Proof in: C.2.)

For example, consider the constraint sequence

$$\overline{A} := \big(1 - \text{prg0}_n - \text{prg1}_n - \text{prg2}_n\big)_{n \in \mathbb{N}^+}$$

For all n and all consistent worlds $\mathbb{W} \in \mathcal{PC}(\Gamma)$, the value $\mathbb{W}(A_n)$ is 0, so applying Theorem 4.5.5 to $\overline{A}$, we get that $\mathbb{P}_n(A_n) \eqsim_n 0$. By linearity, this means

$$\mathbb{P}_n(\text{prg0}_n) + \mathbb{P}_n(\text{prg1}_n) + \mathbb{P}_n(\text{prg2}_n) \eqsim_n 1,$$

i.e., $\overline{\mathbb{P}}$ learns that the three sequences are mutually exclusive and exhaustive in a timely manner, regardless of how difficult `prg` is to evaluate. Affine Provability Induction is a generalization of this idea, where the coefficients may vary (day by day, and with the market prices).

We can push this idea further, as follows:

Theorem 4.5.5 (Affine Coherence). *Let $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(A_n) \leq \liminf_{n \to \infty} \mathbb{P}_\infty(A_n) \leq \liminf_{n \to \infty} \mathbb{P}_n(A_n),$$

and

$$\limsup_{n \to \infty} \mathbb{P}_n(A_n) \leq \limsup_{n \to \infty} \mathbb{P}_\infty(A_n) \leq \limsup_{n \to \infty} \sup_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(A_n).$$

(Proof in: C.1.)

This theorem ties the ground truth on $\overline{A}$, to the value of $\overline{A}$ in the limit, to the value of $\overline{A}$ on the main diagonal. In words, it says that if all consistent worlds value A_n in (a, b) for n large, then $\mathbb{P}_\infty$ values A_n in $(c, d) \subseteq (a, b)$ for n large (because $\mathbb{P}_\infty$ is a weighted mixture of all consistent worlds), and $\overline{\mathbb{P}}$ learns to assign probabilities such that $\mathbb{P}_n(A_n) \in (c, d)$ in a timely manner. In colloquial terms, $\overline{\mathbb{P}}$ learns in a timely manner to respect *all* linear inequalities that actually hold between sentences, so long as those relationships can be enumerated in polynomial time.

For example, if `helper(i)=err` always implies `prg(i)=0`, $\overline{\mathbb{P}}$ will learn this pattern, and start assigning probabilities to $\mathbb{P}_n(\text{"}\texttt{prg(}\underline{n}\texttt{)=0}\text{"})$ which are no lower than those of $\mathbb{P}_n(\text{"}\texttt{helper(}\underline{n}\texttt{)=err}\text{"})$. In general, if a series of sentences obey some complicated linear inequalities, then so long as those constraints can be *written down* in polynomial time, $\overline{\mathbb{P}}$ will learn the pattern, and start assigning probabilities that respect those constraints in a timely manner.

This doesn't mean that $\overline{\mathbb{P}}$ will assign the *correct* values (0 or 1) to each sentence in a timely manner; that would be impossible for a deductively limited reasoner. Rather, $\overline{\mathbb{P}}$'s probabilities will start *satisfying the constraints* in a timely manner. For example, imagine a set of complex constraints holds between seven sequences, such that exactly three sentences in each septuplet are true, but it's difficult to tell which three. Then $\overline{\mathbb{P}}$ will learn this pattern, and start ensuring that its probabilities on each septuplet sum to 3, even if it can't yet assign particularly high probabilities to the correct three.

If we watch an individual septuplet as $\overline{\mathbb{P}}$ reasons, other constraints will push the probabilities on those seven sentences up and down. One sentence might be refuted and have its probability go to zero. Another might get a boost when $\overline{\mathbb{P}}$ discovers that it's likely implied by a high-probability sentence. Another might take a hit when $\overline{\mathbb{P}}$ discovers it likely implies a low-probability sentence. Throughout all this, Theorem 4.5.5 says that $\overline{\mathbb{P}}$ will ensure that the seven probabilities always sum to ≈ 3. $\overline{\mathbb{P}}$'s beliefs on any given day arise from this interplay of many constraints, inductively learned.

Observe that Affine Coherence is a direct generalization of Theorem 4.2.1 (Provability Induction). One way to interpret this theorem is that it says that $\overline{\mathbb{P}}$ is very good at learning inductively to predict long-running computations. Given any e.c. sequence of statements about the computation, if they are true then $\overline{\mathbb{P}}$ learns to believe them in a timely manner, and if they are false then $\overline{\mathbb{P}}$ learns to disbelieve them in a timely manner, and if they are related by logical constraints (such as by exclusivity or implication) to some other e.c. sequence of statements, then $\overline{\mathbb{P}}$ learns to make its probabilities respect those constraints in a timely manner. This is one of the main reasons why we think this class of algorithms deserves the name of "logical inductor".

Affine Coherence can also be interpreted as an approximate coherence condition on the finite belief-states of $\overline{\mathbb{P}}$. It says that if a certain relationship among truth values is going to hold in the future, then $\overline{\mathbb{P}}$ learns to make that relationship hold approximately in its probabilities, in a timely manner.[6]

In fact, we can use this idea to strengthen every theorem in sections 4.2-4.4, as below. (Readers without interest in the strengthened theorems are invited to skip to Section 4.6.)

Affine Strengthenings

Observe that Theorem 4.5.4 (Affine Provability Induction) is a strengthening of Theorem 4.2.1 (Provability Induction).

Theorem 4.5.6 (Persistence of Affine Knowledge)**.** *Let $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \inf_{m \geq n} \mathbb{P}_m(A_n) = \liminf_{n \to \infty} \mathbb{P}_\infty(A_n)$$

and

$$\limsup_{n \to \infty} \sup_{m \geq n} \mathbb{P}_m(A_n) = \limsup_{n \to \infty} \mathbb{P}_\infty(A_n).$$

(Proof in: B.5.)

To see that this is a generalization of Theorem 4.2.3 (Persistence of Knowledge), it might help to first replace $\overline{A}$ with a sequence $\overline{p}$ of rational probabilities.

---

6. Another notion of approximate coherence goes by the name of "inductive coherence" (Garrabrant, Fallenstein, et al. 2016). A reasoner is called inductively coherent if (1) $\mathbb{P}_n(\bot) \eqsim_n 0$; (2) $\mathbb{P}_n(\phi_n)$ converges whenever $\overline{\phi}$ is efficiently computable and each ϕ_n provably implies ϕ_{n+1}; and (3) for all efficiently computable sequences of provably mutually exclusive and exhaustive triplets (ϕ_n, ψ_n, χ_n), $\mathbb{P}_n(\phi_n) + \mathbb{P}_n(\psi_n) + \mathbb{P}_n(\chi_n) \eqsim_n 1$. Garrabrant, Fallenstein, et al. show that inductive coherence implies coherence in the limit, and argue that this is a good notion of approximate coherence. Theorems 4.1.2 (Limit Coherence) and 4.5.5 (Affine Coherence) imply inductive coherence, and indeed, logical induction is a much stronger notion.

Theorem 4.5.7 (Affine Preemptive Learning)**.** *Let $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \mathbb{P}_n(A_n) = \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{P}_m(A_n)$$

and

$$\limsup_{n \to \infty} \mathbb{P}_n(A_n) = \limsup_{n \to \infty} \inf_{m \geq n} \mathbb{P}_m(A_n) \ .$$

<div align="right">(Proof in: B.2.)</div>

Definition 4.5.8 (Determined via Γ)**.** *We say that a $\mathbb{R}$-combination A is **determined via Γ** if, in all worlds $\mathbb{W} \in \mathcal{PC}(\Gamma)$, the value $\mathbb{W}(A)$ is equal. Let $\mathrm{Val}_\Gamma(A)$ denote this value.*

Similarly, a sequence $\overline{A}$ of $\mathbb{R}$-combinations is said to be determined via Γ if A_n is determined via Γ for all n.

Theorem 4.5.9 (Affine Recurring Unbiasedness)**.** *If $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$ is determined via Γ, and $\overline{w}$ is a $\overline{\mathbb{P}}$-generable divergent weighting,*

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(A_i) - \mathrm{Val}_\Gamma(A_i))}{\sum_{i \leq n} w_i}$$

has 0 as a limit point. In particular, if it converges, it converges to 0. (Proof in: D.1.)

Theorem 4.5.10 (Affine Unbiasedness from Feedback)**.** *Given $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$ that is determined via Γ, a strictly increasing deferral function f such that $\mathrm{Val}_\Gamma(A_n)$ can be computed in time $\mathcal{O}(f(n+1))$, and a $\overline{\mathbb{P}}$-generable divergent weighting w,*

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(A_i) - \mathrm{Val}_\Gamma(A_i))}{\sum_{i \leq n} w_i} \eqsim_n 0.$$

In this case, we say "$\overline{w}$ allows good feedback on $\overline{A}$". <div align="right">(Proof in: D.4.)</div>

Theorem 4.5.11 (Learning Pseudorandom Affine Sequences)**.** *Given a $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$ which is determined via Γ, if there exists deferral function f such that for any $\overline{\mathbb{P}}$-generable f-patient divergent weighting $\overline{w}$,*

$$\frac{\sum_{i \leq n} w_i \cdot \mathrm{Val}_\Gamma(A_i)}{\sum_{i \leq n} w_i} \gtrsim_n 0,$$

then

$$\mathbb{P}_n(A_n) \gtrsim_n 0,$$

and similarly for $\eqsim_n$, and $\lesssim_n$. <div align="right">(Proof in: D.6.)</div>

4.6   Non-Dogmatism

Cromwell's rule says that a reasoner should not assign extreme probabilities (0 or 1) except when applied to statements that are logically true or false. The rule was named by Lindley (1991), in light of the fact that Bayes' theorem says that a Bayesian reasoner can never update away from probabilities 0 or 1, and in reference to the famous plea:

> I beseech you, in the bowels of Christ, think it possible that you may be mistaken. *– Oliver Cromwell*

The obvious generalization of Cromwell's rule to a setting where a reasoner is uncertain about logic is that they also should not assign extreme probabilities to sentences that have not yet been proven or disproven. Logical inductors *do not* satisfy this rule, as evidenced by the following theorem:

Theorem 4.6.1 (Closure under Finite Perturbations)**.** *Let $\overline{\mathbb{P}}$ and $\overline{\mathbb{P}'}$ be markets with $\mathbb{P}_n = \mathbb{P}'_n$ for all but finitely many n. Then $\overline{\mathbb{P}}$ is a logical inductor if and only if $\overline{\mathbb{P}'}$ is a logical inductor.* (Proof in: G.7.)

This means that we can take a logical inductor, completely ruin its beliefs on the 23rd day (e.g., by setting $\mathbb{P}_{23}(\phi) = 0$ for all ϕ), and it will still be a logical inductor. Nevertheless, there is still a sense in which logical inductors are non-dogmatic, and can "think it possible that they may be mistaken":

Theorem 4.6.2 (Non-Dogmatism)**.** *If $\Gamma \nvdash \phi$ then $\mathbb{P}_\infty(\phi) < 1$, and if $\Gamma \nvdash \neg\phi$ then $\mathbb{P}_\infty(\phi) > 0$.*

Proof sketch. (Proof in: G.4.)

> Consider a trader that watches ϕ and buys whenever it gets low, as follows. The trader starts with \$1. They spend their first 50 cents when $\mathbb{P}_n(\phi) < 1/2$, purchasing one share. They spend their next 25 cents when $\mathbb{P}_n(\phi) < 1/4$, purchasing another share. They keep waiting for $\mathbb{P}_n(\phi)$ to drop low enough that they can spend the next half of their initial wealth to buy one more share. Because ϕ is independent, there always remains at least one world $\mathbb{W}$ such that $\mathbb{W}(\phi) = 1$, so if $\mathbb{P}_n(\phi) \to 0$ as $n \to \infty$ then their maximum plausible profits are \$1 + \$1 + \$1 +... which diverges, and they exploit the market. Thus, $\mathbb{P}_\infty(\phi)$ must be bounded away from zero.

In other words, if ϕ is independent from Γ, then $\overline{\mathbb{P}}$'s beliefs about ϕ won't get stuck converging to 0 or 1. By Theorem 4.6.1 (Closure under Finite Perturbations), $\overline{\mathbb{P}}$ may occasionally jump to unwarranted conclusions—believing with "100% certainty", say, that Euclid's fifth postulate follows from the first four—but it always corrects these errors, and eventually develops conservative beliefs about independent sentences.

Theorem 4.6.2 guarantees that $\overline{\mathbb{P}}$ will be reasonable about independent sentences, but it doesn't guarantee reasonable beliefs about *theories*, because theories can require infinitely many axioms. For example, let Γ be a theory of pure first-order logic, and imagine that the language $\mathcal{L}$ has a free binary relation symbol "$\in$". Now consider the sequence $\overline{\mathrm{ZFCaxioms}}$ of first-order axioms of Zermelo-Fraenkel set theory (ZFC) which say to interpret "$\in$" in the set-theoretic way, and note that $\overline{\mathrm{ZFCaxioms}}$ is infinite. Each individual sentence $\mathrm{ZFCaxioms}_n$ is consistent with first-order logic, but if $\mathbb{P}_\infty$'s odds on each axiom were 50:50 and independent, then it would say that the probability of them all being true simultaneously was zero. Fortunately, for any computably enumerable sequence of sentences that are mutually consistent, $\mathbb{P}_\infty$ assigns positive probability to them all being simultaneously true.

Theorem 4.6.3 (Uniform Non-Dogmatism)**.** *For any computably enumerable sequence of sentences $\overline{\phi}$ such that $\Gamma \cup \overline{\phi}$ is consistent, there is a constant $\varepsilon > 0$ such that for all n,*

$$\mathbb{P}_\infty(\phi_n) \geq \varepsilon.$$

(Proof in: G.2.)

If ϕ_n is the conjunction of the first n axioms of ZFC, Theorem 4.6.3 shows that $\mathbb{P}_\infty$ assigns positive probability to theories in which the symbol "$\in$" satisfies all axioms of ZFC (assuming ZFC is consistent).

Reasoning about individual sentences again, we can put bounds on how far each sentence ϕ is bounded away from 0 and 1, in terms of the prefix complexity $\kappa(\phi)$ of ϕ, i.e., the length of the shortest prefix that causes a fixed universal Turing machine to output ϕ.[7]

---

7. We use prefix complexity (the length of the shortest prefix that causes a UTM to output ϕ) instead of Kolmogorov complexity (the length of the shortest complete program that causes a UTM to output ϕ) because it makes the proof slightly easier. (And, in the opinion of the authors, prefix complexity is the more natural concept.) Both types of complexity are defined relative to an arbitrary choice of universal Turing machine (UTM), but our theorems hold for every logical inductor regardless of the choice of UTM, because changing the UTM only amounts to changing the constant terms by some fixed amount.

Theorem 4.6.4 (Occam Bounds)**.** *There exists a fixed positive constant C such that for any sentence ϕ with prefix complexity $\kappa(\phi)$, if $\Gamma \nvdash \neg\phi$, then*

$$\mathbb{P}_\infty(\phi) \geq C2^{-\kappa(\phi)},$$

and if $\Gamma \nvdash \phi$, then

$$\mathbb{P}_\infty(\phi) \leq 1 - C2^{-\kappa(\phi)}.$$

(Proof in: G.3.)

This means that if we add a sequence of constant symbols $(c_1, c_2, \ldots)$ not mentioned in Γ to the language $\mathcal{L}$, then $\overline{\mathbb{P}}$'s beliefs about statements involving those constants will depend on the complexity of the claim. Roughly speaking, if you ask after the probability of a claim like "$c_1 = 10 \wedge c_2 = 7 \wedge \ldots \wedge c_n = -3$" then the answer will be no lower than the probability that a simplicity prior assigns to the shortest program that outputs $(10, 7, \ldots, -3)$.

In fact, the probability may be a fair bit higher, if the claim is part of a particularly simple sequence of sentences. In other words, logical inductors can be used to reason about *empirical* uncertainty as well as logical uncertainty, by using $\mathbb{P}_\infty$ as a full-fledged sequence predictor:

Theorem 4.6.5 (Domination of the Universal Semimeasure)**.** *Let $(b_1, b_2, \ldots)$ be a sequence of zero-arity predicate symbols in $\mathcal{L}$ not mentioned in Γ, and let $\sigma_{\leq n} = (\sigma_1, \ldots, \sigma_n)$ be any finite bitstring. Define*

$$\mathbb{P}_\infty(\sigma_{\leq n}) := \mathbb{P}_\infty(\text{``}(b_1 \leftrightarrow \underline{\sigma_1} = 1) \wedge (b_2 \leftrightarrow \underline{\sigma_2} = 1) \wedge \ldots \wedge (b_n \leftrightarrow \underline{\sigma_n} = 1)\text{''}),$$

such that, for example, $\mathbb{P}_\infty(01101) = \mathbb{P}_\infty(\text{``}\neg b_1 \wedge b_2 \wedge b_3 \wedge \neg b_4 \wedge b_5\text{''})$. Let M be a universal continuous semimeasure. Then there is some positive constant C such that for any finite bitstring $\sigma_{\leq n}$,

$$\mathbb{P}_\infty(\sigma_{\leq n}) \geq C \cdot M(\sigma_{\leq n}).$$

(Proof in: G.5.)

In other words, logical inductors can be viewed as a computable approximation to a normalized probability distribution that dominates the universal semimeasure. In fact, this dominance is strict:

Theorem 4.6.6 (Strict Domination of the Universal Semimeasure)**.** *The universal continuous semimeasure does not dominate $\mathbb{P}_\infty$; that is, for any positive constant C there is some finite bitstring $\sigma_{\leq n}$ such that*

$$\mathbb{P}_\infty(\sigma_{\leq n}) > C \cdot M(\sigma_{\leq n}).$$

(Proof in: G.6.)

In particular, by Theorem 4.6.3 (Uniform Non-Dogmatism), logical inductors assign positive probability to the set of all completions of theories like PA and ZFC, whereas universal semimeasures do not. This is why we can't construct approximately coherent beliefs about logic by fixing an enumeration of logical sentences and conditioning a universal semimeasure on more axioms of Peano arithmetic each day: the probabilities that the semimeasure assigns to those conjunctions must go to zero, so the conditional probabilities may misbehave. (If this were not the case, it would be possible to sample a complete extension of Peano arithmetic with positive probability, because universal semimeasures are approximable from below; but this is impossible. See the proof of Theorem 4.6.6 for details.) While $\mathbb{P}_\infty$ is limit-computable, it is not approximable from below, so (roughly speaking) it can and does outperform the universal semimeasure when reasoning about arithmetical claims.

4.7 Conditionals

One way to interpret Theorem 4.6.5 (Domination of the Universal Semimeasure) is that when we condition $\mathbb{P}_\infty$ on independent sentences about which it knows nothing, it performs empirical (scientific) induction. We will now show that when we condition $\overline{\mathbb{P}}$, it also performs logical induction.

In probability theory, it is common to discuss conditional probabilities such as $\Pr(A \mid B) := \Pr(A \wedge B)/\Pr(B)$ (for any B with $\Pr(B) > 0$), where $\Pr(A \mid B)$ is interpreted as the probability of A restricted to world where B is true. In the domain of logical uncertainty, we can define conditional probabilities in the analogous way:

Definition 4.7.1 (Conditional Probability)**.** *Let ϕ and ψ be sentences, and let $\mathbb{V}$ be a valuation with $\mathbb{V}(\psi) > 0$. Then we define*

$$\mathbb{V}(\phi \mid \psi) := \begin{cases} \mathbb{V}(\phi \wedge \psi)/\mathbb{V}(\psi) & \text{if } \mathbb{V}(\phi \wedge \psi) < \mathbb{V}(\psi) \\ 1 & \text{otherwise.} \end{cases}$$

Given a valuation sequence $\overline{\mathbb{V}}$, we define

$$\overline{\mathbb{V}}(- \mid \psi) := (\mathbb{V}_1(- \mid \psi), \mathbb{V}_2(- \mid \psi), \ldots).$$

Defining $\mathbb{V}(\phi \mid \psi)$ to be 1 if $\mathbb{V}(\psi) = 0$ is nonstandard, but convenient for our theorem statements and proofs. The reader is welcome to ignore the conditional probabilities in cases where $\mathbb{V}(\psi) = 0$, or to justify our definition from the principle of explosion (which says that from a contradiction, anything follows). This definition also caps $\mathbb{V}(\phi \mid \psi)$ at 1, which is necessary because there's no guarantee that $\mathbb{V}$ knows that $\phi \wedge \psi$ should have a lower probability than ψ. For example, if it takes $\overline{\mathbb{P}}$ more than 17 days to learn how "$\wedge$" interacts with ϕ and ψ, then it might be the case that $\mathbb{P}_{17}(\phi \wedge \psi) = 0.12$ and $\mathbb{P}_{17}(\psi) = 0.01$, in which case the uncapped "conditional probability" of $\phi \wedge \psi$ given ψ according to $\mathbb{P}_{17}$ would be twelve hundred percent.

This fact doesn't exactly induce confidence in $\overline{\mathbb{P}}(- \mid \psi)$. Nevertheless, we have the following theorem:

Theorem 4.7.2 (Closure Under Conditioning)**.** *The sequence $\overline{\mathbb{P}}(- \mid \psi)$ is a logical inductor over $\Gamma \cup \{\psi\}$. Furthermore, given any efficiently computable enumeration $\Psi \subseteq \mathcal{S}$ of sentences, the sequence*

$$\left(\mathbb{P}_1(- \mid \psi_1), \mathbb{P}_2(- \mid \psi_1 \wedge \psi_2), \mathbb{P}_3(- \mid \psi_1 \wedge \psi_2 \wedge \psi_3), \ldots\right),$$

where the nth pricing is conditioned on the first n sentences in Ψ, is a logical inductor over $\Gamma \cup \Psi$. (Proof in: G.8.)

In other words, if we condition logical inductors on logical sentences, the result is still a logical inductor, and so the conditional probabilities of a logical inductor continues to satisfy all the desirable properties satisfied by all logical inductors. This also means that one can obtain a logical inductor for Peano arithmetic by starting with a logical inductor over an empty theory, and conditioning it on PA.

With that idea in mind, we will now begin examining questions about logical inductors that assume Γ can represent computable functions, such as questions about $\overline{\mathbb{P}}$'s beliefs about Γ, computer programs, and itself.

4.8 Expectations

In probability theory, it is common to ask the expected (average) value of a variable that takes on different values in different possible worlds. Emboldened by our success with conditional probabilities, we will now define a notion of the expected values of *logical* variables, and show that these are also fairly well-behaved. This machinery will be useful later when we ask logical inductors for their beliefs about themselves.

We begin by defining a notion of logically uncertain variables, which play a role analogous to the role of random variables in probability theory. For the sake

of brevity, we will restrict our attention to logically uncertain variables with their value in $[0, 1]$; it is easy enough to extend this notion to a notion of arbitrary bounded real-valued logically uncertain variables. (It does, however, require carrying a variable's bounds around everywhere, which makes the notation cumbersome.)

To define logically uncertain variables, we will need to assume that Γ is capable of representing rational numbers and proving things about them. Later, we will use expected values to construct sentences that talk about things like the expected outputs of a computer program. Thus, in this section and in the remainder of Section 4, we will assume that Γ can represent computable functions.

Definition 4.8.1 (Logically Uncertain Variable). *A **logically uncertain variable**, abbreviated **LUV**, is any formula X free in one variable that defines a unique value via Γ, in the sense that*

$$\Gamma \vdash \exists x \colon \forall x' \colon X(x') \to x' = x.$$

*We refer to that value as the **value** of X. If Γ proves that the value of X is in $[0, 1]$, we call X a $[\mathbf{0}, \mathbf{1}]$-**LUV**.*

*Given a $[0, 1]$-LUV X and a consistent world $\mathbb{W} \in \mathcal{PC}(\Gamma)$, the **value of X in** $\mathbb{W}$ is defined to be*

$$\mathbb{W}(X) := \sup \left\{ x \in [0, 1] \mid \mathbb{W}(\text{``}X \geq \underline{x}\text{''}) = 1 \right\}.$$

In other words, $\mathbb{W}(X)$ is the supremum of values that do not exceed X according to $\mathbb{W}$. (This rather roundabout definition is necessary in cases where $\mathbb{W}$ assigns X a non-standard value.)

We write $\mathcal{U}$ for the set of all $[0, 1]$-LUVs. When manipulating logically uncertain variables, we use shorthand like "$X < 0.5$" for "$\forall x \colon X(x) \to x < 0.5$". See Section 2 for details.

As an example, *Half* := "$\nu = 0.5$" is a LUV, where the unique real number that makes *Half* true is rather obvious. A more complicated LUV is

TwinPrime := "1 if the twin prime conjecture is true, 0 otherwise";

this is a deterministic quantity (assuming Γ actually proves the twin prime conjecture one way or the other), but it's reasonable for a limited reasoner to be uncertain about the value of that quantity. In general, if $f : \mathbb{N}^+ \to [0, 1]$ is a computable function then "$\underline{f}(7)$" is a LUV, because "$\underline{f}(7)$" is shorthand for the formula "$\gamma_f(7, \nu)$", where γ_f is the predicate of Γ representing f.

With LUVs in hand, we can define a notion of $\overline{\mathbb{P}}$'s expected value for a LUV X on day n with precision k. The obvious idea is to take the sum

$$\lim_{k \to \infty} \sum_{i=0}^{k-1} \frac{i}{k} \mathbb{P}_n(\text{``}\underline{i/k} < \underline{X} \leq \underline{(i+1)/k}\text{''}).$$

However, if $\mathbb{P}_n$ hasn't yet figured out that X pins down a unique value, then it might put high probability on X being in multiple different intervals, and the simple integral of a $[0, 1]$-valued LUV could fall outside the $[0, 1]$ interval. This is a nuisance when we want to treat the expectations of $[0, 1]$-LUVs as other $[0, 1]$-LUVs, so instead, we will define expectations using an analog of a cumulative distribution function. In probability theory, the expectation of a $[0, 1]$-valued random variable V with density function ρ_V is given by $\mathbb{E}(V) = \int_0^1 x \cdot \rho_V(x) dx$. We can rewrite this using integration by parts as

$$\mathbb{E}(V) = \int_0^1 \Pr(V > x) dx.$$

This motivates the following definition of expectations for LUVs:

Definition 4.8.2 (Expectation)**.** *For a given valuation* $\mathbb{V}$*, we define the **approximate expectation operator** $\mathbb{E}_k^{\mathbb{V}}$ for $\mathbb{V}$ with precision k by*

$$\mathbb{E}_k^{\mathbb{V}}(X) := \sum_{i=0}^{k-1} \frac{1}{k} \mathbb{V}(\text{“}\underline{X} > \underline{i}/\underline{k}\text{”}).$$

where X is a $[0,1]$-LUV.

This has the desirable property that $\mathbb{E}_k^{\mathbb{V}}(X) \in [0,1]$, because $\mathbb{V}(-) \in [0,1]$.

We will often want to take a limit of $\mathbb{E}_k^{\mathbb{P}_n}(X)$ as both k and n approach ∞. We hereby make the fairly arbitrary choice to focus on the case $k = n$ for simplicity, adopting the shorthand

$$\mathbb{E}_n := \mathbb{E}_n^{\mathbb{P}_n}.$$

In other words, when we examine how a logical inductor's expectations change on a sequence of sentences over time, we will (arbitrarily) consider approximate expectations that gain in precision at a rate of one unit per day.

We will now show that the expectation operator $\mathbb{E}_n$ possesses properties that make it worthy of that name.

Theorem 4.8.3 (Expectations Converge)**.** *The limit $\mathbb{E}_\infty : \mathcal{S} \to [0,1]$ defined by*

$$\mathbb{E}_\infty(X) := \lim_{n\to\infty} \mathbb{E}_n(X)$$

exists for all $X \in \mathcal{U}$. (Proof in: E.4.)

Note that $\mathbb{E}_\infty(X)$ might not be rational.

Because $\mathbb{P}_\infty$ defines a probability measure over $\mathcal{PC}(\Gamma)$, $\mathbb{E}_\infty(X)$ is the average value of $\mathbb{W}(X)$ across all consistent worlds (weighted by $\mathbb{P}_\infty$). In other words, every LUV X can be seen as a random variable with respect to the measure $\mathbb{P}_\infty$, and $\mathbb{E}_\infty$ acts as the standard expectation operator on $\mathbb{P}_\infty$. Furthermore,

Theorem 4.8.4 (Linearity of Expectation)**.** *Let $\bar{a}, \bar{b}$ be bounded $\overline{\mathbb{P}}$-generable sequences of rational numbers, and let $\overline{X}, \overline{Y}$, and $\overline{Z}$ be e.c. sequences of $[0,1]$-LUVs. If we have $\Gamma \vdash Z_n = a_n X_n + b_n Y_n$ for all n, then*

$$a_n \mathbb{E}_n(X_n) + b_n \mathbb{E}_n(Y_n) \eqsim_n \mathbb{E}_n(Z_n).$$

(Proof in: E.9.)

For our next result, we want a LUV which can be proven to take value 1 if ϕ is true and 0 otherwise.

Definition 4.8.5 (Indicator LUV)**.** *For any sentence ϕ, we define its **indicator LUV** by the formula*

$$\mathbb{1}(\phi) := \text{“}(\underline{\phi} \wedge (\nu = 1)) \vee (\neg\underline{\phi} \wedge (\nu = 0))\text{”}.$$

Observe that $\mathbb{1}(\phi)(1)$ is equivalent to ϕ, and $\mathbb{1}(\phi)(0)$ is equivalent to $\neg\phi$.

Theorem 4.8.6 (Expectations of Indicators)**.** *Let $\overline{\phi}$ be an e.c. sequence of sentences. Then*

$$\mathbb{E}_n(\mathbb{1}(\phi_n)) \eqsim_n \mathbb{P}_n(\phi_n).$$

(Proof in: E.10.)

In colloquial terms, Theorem 4.8.6 says that a logical inductor learns that asking for the expected value of $\mathbb{1}(\phi)$ is the same as asking for the probability of ϕ.

To further demonstrate that expectations work as expected, we will show that they satisfy generalized versions of all theorems proven in sections 4.2-4.5. (Readers without interest in the versions of those theorems for expectations are invited to skip to Section 4.9.)

Collected Theorems for Expectations

Definition 4.8.7 (LUV Valuation). *A LUV valuation is any function $\mathbb{U} : \mathcal{U} \to [0,1]$. Note that $\mathbb{E}_n^{\mathbb{V}}$ and $\mathbb{E}_\infty^{\mathbb{V}}$ are LUV valuations for any valuation $\mathbb{V}$ and $n \in \mathbb{N}^+$, and that every world $\mathbb{W} \in \mathcal{PC}(\Gamma)$ is a LUV valuation.*

Definition 4.8.8 (LUV Combination). *An $\boldsymbol{\mathcal{F}}$-**LUV-combination** $B : \mathcal{U} \cup \{1\} \to \mathcal{F}_n$ is an affine expression of the form*

$$B := c + \xi_1 X_1 + \cdots + \xi_k X_k,$$

*where $(X_1, \ldots, X_k)$ are $[0,1]$-LUVs and $(c, \xi_1, \ldots, \xi_k)$ are in $\mathcal{F}$. An $\boldsymbol{\mathcal{EF}}$-**LUV-combination**, an $\mathbb{R}$-**LUV-combination**, and a $\mathbb{Q}$-**LUV-combination** are defined similarly.*

*The following concepts are all defined analogously to how they are defined for sentence combinations: $B[1]$, $B[X]$, $\mathrm{rank}(B)$, $\mathbb{U}(B)$ for any LUV valuation $\mathbb{U}$, $\boldsymbol{\mathcal{F}}$-**LUV-combination progressions**, $\boldsymbol{\mathcal{EF}}$-**LUV-combination progressions**, and $\overline{\mathbb{P}}$-generable LUV-combination sequences. (See definitions 3.4.6 and 4.3.5 for details.)*

Definition 4.8.9 (Bounded LUV-Combination Sequence). *By $\mathcal{BLCS}(\overline{\mathbb{P}})$ (mnemonic: **bounded LUV-combination sequences**) we denote the set of all $\overline{\mathbb{P}}$-generable $\mathbb{R}$-LUV-combination sequences $\overline{B}$ that are bounded, in the sense that there exists some bound b such that $\|B_n\|_1 \leq b$ for all n, where $\|-\|_1$ includes the trailing coefficient.*

Theorem 4.8.10 (Expectation Provability Induction). *Let $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$ and $b \in \mathbb{R}$. If, for all consistent worlds $\mathbb{W} \in \mathcal{PC}(\Gamma)$ and all $n \in \mathbb{N}^+$, it is the case that $\mathbb{W}(B_n) \geq b$, then*

$$\mathbb{E}_n(B_n) \gtrsim_n b,$$

and similarly for $=$ and $\eqsim_n$, and for $\leq$ and $\lesssim_n$. (Proof in: E.8.)

Theorem 4.8.11 (Expectation Coherence). *Let $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(B_n) \leq \liminf_{n \to \infty} \mathbb{E}_\infty(B_n) \leq \liminf_{n \to \infty} \mathbb{E}_n(B_n),$$

and

$$\limsup_{n \to \infty} \mathbb{E}_n(B_n) \leq \limsup_{n \to \infty} \mathbb{E}_\infty(B_n) \leq \limsup_{n \to \infty} \sup_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(B_n).$$

(Proof in: E.7.)

Theorem 4.8.12 (Persistence of Expectation Knowledge). *Let $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \inf_{m \geq n} \mathbb{E}_m(B_n) = \liminf_{n \to \infty} \mathbb{E}_\infty(B_n)$$

and

$$\limsup_{n \to \infty} \sup_{m \geq n} \mathbb{E}_m(B_n) = \limsup_{n \to \infty} \mathbb{E}_\infty(B_n).$$

(Proof in: E.6.)

Theorem 4.8.13 (Expectation Preemptive Learning). *Let $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \mathbb{E}_n(B_n) = \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{E}_m(B_n)$$

and

$$\limsup_{n \to \infty} \mathbb{E}_n(B_n) = \limsup_{n \to \infty} \inf_{m \geq n} \mathbb{E}_m(B_n) \ .$$

(Proof in: E.3.)

Definition 4.8.14 (Determined via Γ (for LUV-Combinations)). *We say that a $\mathbb{R}$-LUV-combination B is **determined via Γ** if, in all worlds $\mathbb{W} \in \mathcal{PC}(\Gamma)$, the value $\mathbb{W}(B)$ is equal. Let $\mathrm{Val}_\Gamma(B)$ denote this value.*

Similarly, a sequence $\overline{B}$ of $\mathbb{R}$-LUV-combinations is said to be determined via Γ if B_n is determined via Γ for all n.

Theorem 4.8.15 (Expectation Recurring Unbiasedness). *If $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$ is determined via Γ, and $\overline{w}$ is a $\overline{\mathbb{P}}$-generable divergent weighting,*

$$\frac{\sum_{i \le n} w_i \cdot (\mathbb{E}_i(B_i) - \mathrm{Val}_\Gamma(B_i))}{\sum_{i \le n} w_i}$$

has 0 as a limit point. In particular, if it converges, it converges to 0.

Theorem 4.8.16 (Expectation Unbiasedness From Feedback). *Given $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$ that is determined via Γ, a strictly increasing deferral function f such that $\mathrm{Val}_\Gamma(A_n)$ can be computed in time $\mathcal{O}(f(n+1))$, and a $\overline{\mathbb{P}}$-generable divergent weighting w,*

$$\frac{\sum_{i \le n} w_i \cdot (\mathbb{E}_i(B_i) - \mathrm{Val}_\Gamma(B_i))}{\sum_{i \le n} w_i} \eqsim_n 0.$$

In this case, we say "$\overline{w}$ allows good feedback on $\overline{B}$". (Proof in: E.12.)

Theorem 4.8.17 (Learning Pseudorandom LUV Sequences). *Given a $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$ which is determined via Γ, if there exists a deferral function f such that for any $\overline{\mathbb{P}}$-generable f-patient divergent weighting $\overline{w}$,*

$$\frac{\sum_{i \le n} w_i \cdot \mathrm{Val}_\Gamma(B_i)}{\sum_{i \le n} w_i} \gtrsim_n 0,$$

then

$$\mathbb{E}_n(B_n) \gtrsim_n 0.$$

(Proof in: E.13.)

4.9   Trust in Consistency

The theorems above all support the hypothesis that logical inductors develop reasonable beliefs about logic. One might then wonder what a logical inductor has to say about some of the classic questions in meta-mathematics. For example, what does a logical inductor over PA say about the consistency of Peano arithmetic?

Definition 4.9.1 (Consistency Statement). *Given a recursively axiomatizable theory Γ', define the **n-consistency statement** of Γ' to be the formula with one free variable ν such that*

$$\mathrm{Con}(\Gamma')(\nu) := \text{"There is no proof of } \bot \text{ from } \underline{\Gamma'} \text{ with } \nu \text{ or fewer symbols"},$$

written in $\mathcal{L}$ using a Gödel encoding. For instance, $\mathrm{Con}(\mathsf{PA})(\text{"Ack}(10, 10)\text{"})$ says that any proof of $\bot$ from PA requires at least $\mathrm{Ack}(10, 10)$ symbols.

We further define "$\underline{\Gamma'}$ is consistent" to be the universal generalization

$$\text{"}\forall n \colon \text{there is no proof of } \bot \text{ from } \underline{\Gamma'} \text{ in } n \text{ or fewer symbols"},$$

and "$\underline{\Gamma'}$ is inconsistent" for its negation.

Theorem 4.9.2 (Belief in Finitistic Consistency)**.** *Let f be any computable function. Then*

$$\mathbb{P}_n(\mathrm{Con}(\Gamma)(\text{``}\underline{f}(\underline{n})\text{''})) \eqsim_n 1.$$

(Proof in: C.4.)

In other words, if Γ is in fact consistent, then $\overline{\overline{\mathbb{P}}}$ learns to trust it for arbitrary finite amounts of time. For any fast-growing function f you can name, $\overline{\overline{\mathbb{P}}}$ eventually learns to believe Γ is consistent for proofs of length at most, by day n at the latest. In colloquial terms, if we take a logical inductor over PA and show it a computable function f that, on each input n, tries a new method for finding an inconsistency in PA, then the logical inductor will stare at the function for a while and eventually conclude that it's not going to succeed (by learning to assign low probability to $f(n)$ proving $\bot$ from PA by day n at the latest, regardless of how long f runs). That is to say, a logical inductor over PA learns to trust Peano arithmetic *inductively*.

By the same mechanism, a logical inductor over Γ can learn inductively to trust the consistency of *any* consistent theory, including consistent theories that are stronger than Γ (in the sense that they can prove Γ consistent):

Theorem 4.9.3 (Belief in the Consistency of a Stronger Theory)**.** *Let Γ' be any recursively axiomatizable consistent theory. Then*

$$\mathbb{P}_n(\mathrm{Con}(\Gamma')(\text{``}\underline{f}(\underline{n})\text{''})) \eqsim_n 1.$$

(Proof in: C.5.)

For instance, a logical inductor over PA can learn inductively to trust the consistency of ZFC for finite proofs of arbitrary length (assuming ZFC is in fact consistent).

These two theorems alone are unimpressive. Any algorithm that assumes consistency until proven otherwise can satisfy these theorems, and because every inconsistent theory admits a finite proof of inconsistency, those naïve algorithms will disbelieve any inconsistent theory eventually. But those algorithms will still believe inconsistent theories for quite a long time, whereas logical inductors learn to distrust inconsistent theories in a timely manner:

Theorem 4.9.4 (Disbelief in Inconsistent Theories)**.** *Let $\overline{\Gamma'}$ be an e.c. sequence of recursively axiomatizable inconsistent theories. Then*

$$\mathbb{P}_n(\text{``}\underline{\Gamma'_n} \text{ is inconsistent''}) \eqsim_n 1,$$

so

$$\mathbb{P}_n(\text{``}\underline{\Gamma'_n} \text{ is consistent''}) \eqsim_n 0.$$

(Proof in: C.6.)

In other words, logical inductors learn in a timely manner to distrust inconsistent theories that can be efficiently named, even if the shortest proofs of inconsistency are very long.

Note that Theorem 4.9.2 (Belief in Finitistic Consistency) *does not say*

$$\mathbb{P}_\infty(\text{``}\underline{\Gamma} \text{ is consistent''})$$

is equal to 1, nor even that it's particularly high. On the contrary, by Theorem 4.6.2 (Non-Dogmatism), the limiting probability on that sentence is bounded away from 0 and 1 (because both that sentence and its negation are consistent with Γ). Intuitively, $\overline{D}$ never reveals evidence against the existence of non-standard numbers, so $\overline{\overline{\mathbb{P}}}$ remains open to the possibility. This is important for Theorem 4.7.2 (Closure Under Conditioning), which say that logical inductors can safely be conditioned on any sequence of statements that are consistent with Γ, but it also means that $\overline{\overline{\mathbb{P}}}$ will not give an affirmative answer to the question of whether PA is consistent in full generality.

In colloquial terms, if you hand a logical inductor any *particular* computation, it will tell you that that computation isn't going to output a proof $\perp$ from the axioms of PA, but if you ask whether PA is consistent *in general*, it will start waxing philosophical about non-standard numbers and independent sentences—not unlike a human philosopher.

A reasonable objection here is that Theorem 4.9.2 (Belief in Finitistic Consistency) is not talking about the consistency of the Peano axioms, it's talking about *computations* that search for proofs of contradiction from PA. This is precisely correct, and brings us to our next topic.

4.10 Reasoning about Halting

Consider the famous halting problem of Turing (1936). Turing proved that there is no general algorithm for determining whether or not an arbitrary computation halts. Let's examine what happens when we confront logical inductors with the halting problem.

Theorem 4.10.1 (Learning of Halting Patterns). *Let $\overline{m}$ be an e.c. sequence of Turing machines, and $\overline{x}$ be an e.c. sequence of bitstrings, such that m_n halts on input x_n for all n. Then*

$$\mathbb{P}_n(\text{``}\underline{m_n}\text{ halts on input }\underline{x_n}\text{''}) \eqsim_n 1.$$

(Proof in: C.7.)

Note that the individual Turing machines *do not* need to have fast runtime. All that is required is that the *sequence* $\overline{m}$ be efficiently computable, i.e., it must be possible to write out the source code specifying m_n in time polynomial in n. The runtime of an individual m_n is immaterial for our purposes. So long as the m_n all halt on the corresponding x_n, $\overline{\mathbb{P}}$ recognizes the pattern and learns to assign high probability to "$\underline{m_n}$ halts on input $\underline{x_n}$" no later than the nth day.

Of course, this is not so hard on its own—a function that assigns probability 1 to everything also satisfies this property. The real trick is separating the halting machines from the non-halting ones. This is harder. It is easy enough to show that $\overline{\mathbb{P}}$ learns to recognize e.c. sequences of machines that *provably* fail to halt:

Theorem 4.10.2 (Learning of Provable Non-Halting Patterns). *Let $\overline{q}$ be an e.c. sequence of Turing machines, and $\overline{y}$ be an e.c. sequence of bitstrings, such that q_n provably fails to halt on input y_n for all n. Then*

$$\mathbb{P}_n(\text{``}\underline{q_n}\text{ halts on input }\underline{y_n}\text{''}) \eqsim_n 0.$$

(Proof in: C.8.)

Of course, it's not too difficult to disbelieve that the provably-halting machines will halt; what makes the above theorem non-trivial is that $\overline{\mathbb{P}}$ learns *in a timely manner* to expect that those machines won't halt. Together, the two theorems above say that if there is any efficient method for generating computer programs that definitively either halt or don't (according to Γ) then $\overline{\mathbb{P}}$ will learn the pattern.

The above two theorems only apply to cases where Γ can prove that the machine either halts or doesn't. The more interesting case is the one where a Turing machine q fails to halt on input y, but Γ is not strong enough to prove this fact. In this case, $\mathbb{P}_\infty$'s probability of q halting on input y is positive, by Theorem 4.6.2 (Non-Dogmatism). Nevertheless, $\overline{\mathbb{P}}$ still learns to stop expecting that those machines will halt after any reasonable amount of time:

Theorem 4.10.3 (Learning not to Anticipate Halting). *Let $\overline{q}$ be an e.c. sequence of Turing machines, and let $\overline{y}$ be an e.c. sequence of bitstrings, such that q_n does not halt on input y_n for any n. Let f be any computable function. Then*

$$\mathbb{P}_n(\text{``}\underline{q_n}\text{ halts on input }\underline{y_n}\text{ within }\underline{f}(\underline{n})\text{ steps''}) \eqsim_n 0.$$

(Proof in: C.9.)

For example, let $\overline{y}$ be an enumeration of all bitstrings, and let $\overline{q}$ be the constant sequence $(q, q, \ldots)$ where q is a Turing machine that does not halt on any input. If Γ cannot prove this fact, then $\overline{\mathbb{P}}$ will never be able to attain certainty about claims that say q fails to halt, but by Theorem 4.10.3, it still learns to expect that q will run longer than any computable function you can name. In colloquial terms, while $\overline{\mathbb{P}}$ won't become certain that non-halting machines don't halt (which is impossible), it *will* put them in the "don't hold your breath" category (along with some long-running machines that do halt, of course).

These theorems can be interpreted as justifying the intuitions that many computer scientists have long held towards the halting problem: It is impossible to tell whether or not a Turing machine halts in full generality, but for large classes of well-behaved computer programs (such as e.c. sequences of halting programs and provably non-halting programs) it's quite possible to develop reasonable and accurate beliefs. The boundary between machines that compute fast-growing functions and machines that never halt is difficult to distinguish, but even in those cases, it's easy to learn to stop expecting those machines to halt within any reasonable amount of time. (See also the work of Calude and Stay [2008] for other formal results backing up this intuition.)

One possible objection here is that the crux of the halting problem (and of the Γ-trust problem) are not about making good predictions, they are about handling diagonalization and paradoxes of self-reference. Gödel's incompleteness theorem constructs a sentence that says "there is no proof of this sentence from the axioms of PA", and Turing's proof of the undecidability of the halting problem constructs a machine which halts iff some other machine thinks it loops. $\overline{\mathbb{P}}$ learning to trust Γ is different altogether from $\overline{\mathbb{P}}$ learning to trust *itself*. So let us turn to the topic of $\overline{\mathbb{P}}$'s beliefs about $\overline{\mathbb{P}}$.

4.11 Introspection

Because we're assuming Γ can represent computable functions, we can write sentences describing the beliefs of $\overline{\mathbb{P}}$ at different times. What happens when we ask $\overline{\mathbb{P}}$ about sentences that refer to itself?

For instance, consider a sentence $\psi := ``\underline{\mathbb{P}}_{\underline{n}}(\underline{\phi}) > 0.7"$ for some specific n and ϕ, where $\overline{\mathbb{P}}$'s beliefs about ψ should depend on what its beliefs about ϕ are on the nth day. Will $\overline{\mathbb{P}}$ figure this out and get the probabilities right on day n? For any particular ϕ and n it's hard to say, because it depends on whether $\overline{\mathbb{P}}$ has learned how ψ relates to $\overline{\mathbb{P}}$ and ϕ yet. If however we take an e.c. *sequence* of $\overline{\psi}$ which all say "ϕ will have probability greater than 0.7 on day n" with n varying, then we can guarantee that $\overline{\mathbb{P}}$ will learn the pattern, and start having accurate beliefs about its own beliefs:

Theorem 4.11.1 (Introspection)**.** *Let $\overline{\phi}$ be an e.c. sequence of sentences, and $\overline{a}$, $\overline{b}$ be $\overline{\mathbb{P}}$-generable sequences of probabilities. Then, for any e.c. sequence of positive rationals $\overline{\delta} \to 0$, there exists a sequence of positive rationals $\overline{\varepsilon} \to 0$ such that for all n:*

1. *if $\mathbb{P}_n(\phi_n) \in (a_n + \delta_n, b_n - \delta_n)$, then*

$$\mathbb{P}_n(``\underline{a_n} < \underline{\mathbb{P}}_{\underline{n}}(\underline{\phi_n}) < \underline{b_n}") > 1 - \varepsilon_n,$$

2. *if $\mathbb{P}_n(\phi_n) \notin (a_n - \delta_n, b_n + \delta_n)$, then*

$$\mathbb{P}_n(``\underline{a_n} < \underline{\mathbb{P}}_{\underline{n}}(\underline{\phi_n}) < \underline{b_n}") < \varepsilon_n.$$

(Proof in: F.1.)

In other words, for any pattern in $\overline{\mathbb{P}}$'s beliefs that can be efficiently written down (such as "$\overline{\mathbb{P}}$'s probabilities on $\overline{\phi}$ are between a and b on these days"), $\overline{\mathbb{P}}$ learns to believe the pattern if it's true, and to disbelieve it if it's false (with vanishing error).

At a first glance, this sort of self-reflection may seem to make logical inductors vulnerable to paradox. For example, consider the sequence of sentences

$$\overline{\chi^{0.5}} := (\text{``}\underline{\mathbb{P}_n}(\underline{\chi_n^{0.5}}) < 0.5\text{''})_{n \in \mathbb{N}^+}$$

such that $\chi_n^{0.5}$ is true iff $\overline{\mathbb{P}}$ assigns it a probability less than 50% on day n. Such a sequence can be defined by Gödel's diagonal lemma. These sentences are probabilistic versions of the classic "liar sentence", which has caused quite a ruckus in the setting of formal logic (Grim 1991; McGee 1990; Glanzberg 2001; Gupta and Belnap 1993; Eklund 2002). Because our setting is probabilistic, it's perhaps most closely related to the "unexpected hanging" paradox—$\chi_n^{0.5}$ is true iff $\overline{\mathbb{P}}$ thinks it is unlikely on day n. How do logical inductors handle this sort of paradox?

Theorem 4.11.2 (Paradox Resistance)**.** *Fix a rational $p \in (0, 1)$, and define an e.c. sequence of "paradoxical sentences" $\overline{\chi^p}$ satisfying*

$$\Gamma \vdash \underline{\chi_n^p} \leftrightarrow \left(\underline{\mathbb{P}_n}(\underline{\chi_n^p}) < \underline{p} \right)$$

for all n. Then

$$\lim_{n \to \infty} \mathbb{P}_n(\chi_n^p) = p.$$

(Proof in: F.2.)

A logical inductor responds to paradoxical sentences $\overline{\chi^p}$ by assigning probabilities that converge on p. For example, if the sentences say "$\overline{\mathbb{P}}$ will assign me a probability less than 80% on day n", then $\mathbb{P}_n$ (once it has learned the pattern) starts assigning probabilities extremely close to 80%—so close that traders can't tell if it's slightly above or slightly below. By Theorem 4.3.6 (Recurring Unbiasedness), the frequency of truth in $\chi_{\leq n}^p$ will have a limit point at 0.8 as $n \to \infty$, and by the definition of logical induction, there will be no efficiently expressible method for identifying a bias in the price.

Let us spend a bit of time understanding this result. After day n, $\chi_n^{0.8}$ is "easy" to get right, at least for someone with enough computing power to compute $\mathbb{P}_n(\chi_n^{0.8})$ to the necessary precision (it will wind up *very* close to 0.8 for large n). Before day n, we can interpret the probability of $\chi_n^{0.8}$ as the price of a share that's going to pay out \$1 if the price on day n is less than 80¢, and \$0 otherwise. What's the value of this share? Insofar as the price on day n is going to be low, the value is high; insofar as the price is going to be high, the value is low. So what actually happens on the nth day? Smart traders buy $\chi_n^{0.8}$ if its price is lower than 80¢, and sell it if its price is higher than 80¢. By the continuity constraints on the traders, each one has a price at which they stop buying $\chi_n^{0.8}$, and Theorem 4.11.2 (Paradox Resistance) tells us that the stable price exists extremely close to 80¢. Intuitively, it must be so close that traders can't tell which way it's going to go, biased on the low side, so that it looks 80% likely to be below and 20% likely to be above to any efficient inspection. For if the probability seemed more than 80% likely to be below, traders would buy; and if it seemed anymore than 20% likely to be above, traders would sell.

To visualize this, imagine that your friend owns a high-precision brain-scanner and can read off your beliefs. Imagine they ask you what probability you assign to the claim "you will assign probability <80% to this claim at precisely 10am tomorrow". As 10am approaches, what happens to your belief in this claim? If you become extremely confident that it's going to be true, then your confidence should drop. But if you become fairly confident it's going to be false, then your confidence should spike. Thus, your probabilities should oscillate, pushing your belief so close to 80% that you're not quite sure which way the brain scanner will actually call it. In response to a paradoxical claim, this is exactly how $\overline{\mathbb{P}}$ behaves, once it's learned how the paradoxical sentences work.

Thus, logical inductors have reasonable beliefs about their own beliefs even in the face of paradox. We can further show that logical inductors have "introspective access" to their own beliefs and expectations, via the medium of logically uncertain variables:

Theorem 4.11.3 (Expectations of Probabilities). *Let $\overline{\phi}$ be an efficiently computable sequence of sentences. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n \mathbb{E}_n(\text{``}\underline{\mathbb{P}_n}(\underline{\phi_n})\text{''}).$$

<div align="right">(Proof in: F.3.)</div>

Theorem 4.11.4 (Iterated Expectations). *Suppose $\overline{X}$ is an efficiently computable sequence of LUVs. Then*

$$\mathbb{E}_n(X_n) \eqsim_n \mathbb{E}_n(\text{``}\underline{\mathbb{E}_n}(\underline{X_n})\text{''}).$$

<div align="right">(Proof in: F.4.)</div>

Next, we turn our attention to the question of what a logical inductor believes about its *future* beliefs.

4.12 Self-Trust

The coherence conditions of classical probability theory guarantee that a probabilistic reasoner trusts their future beliefs, whenever their beliefs change in response to new empirical observations. For example, if a reasoner $\Pr(-)$ knows that tomorrow they'll see some evidence e that will convince them that Miss Scarlet was the murderer, then they already believe that she was the murderer today:

$$\Pr(\text{Scarlet}) = \Pr(\text{Scarlet} \mid e)\Pr(e) + \Pr(\text{Scarlet} \mid \neg e)\Pr(\neg e).$$

In colloquial terms, this says "my current beliefs are *already* a mixture of my expected future beliefs, weighted by the probability of the evidence that I expect to see."

Logical inductors obey similar coherence conditions with respect to their future beliefs, with the difference being that a logical inductor updates its belief by gaining more knowledge about *logical* facts, both by observing an ongoing process of deduction and by thinking for longer periods of time. Thus, the self-trust properties of a logical inductor follow a slightly different pattern:

Theorem 4.12.1 (Expected Future Expectations). *Let f be a deferral function (as per Definition 4.3.7), and let $\overline{X}$ denote an e.c. sequence of $[0,1]$-LUVs. Then*

$$\mathbb{E}_n(X_n) \eqsim_n \mathbb{E}_n(\text{``}\underline{\mathbb{E}_{f(n)}}(\underline{X_n})\text{''}).$$

<div align="right">(Proof in: F.5.)</div>

Roughly speaking, Theorem 4.12.1 says that a logical inductor's current expectation of X on day n is *already* equal to its expected value of X in $f(n)$ days. In particular, it learns in a timely manner to set its current expectations equal to its future expectations on any LUV. In colloquial terms, once a logical inductor has figured out how expectations work, it will never say "I currently believe that the $\overline{X}$ variables have low values, but tomorrow I'm going to learn that they have high values". Logical inductors already expect today what they expect to expect tomorrow.

It follows immediately from theorems 4.12.1 (Expected Future Expectations) and 4.8.6 (Expectations of Indicators) that the current beliefs of a logical inductor are set, in a timely manner, to equal their future expected beliefs.

Theorem 4.12.2 (No Expected Net Update). *Let f be a deferral function, and let $\overline{\phi}$ be an e.c. sequence of sentences. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n \mathbb{E}_n(\text{``}\underline{\mathbb{P}_{f(n)}}(\underline{\phi_n})\text{''}).$$

<div align="right">(Proof in: F.6.)</div>

In particular, if $\overline{\mathbb{P}}$ knows that its future self is going to assign some sequence $\overline{p}$ of probabilities to $\overline{\phi}$, then it starts assigning $\overline{p}$ to $\overline{\phi}$ in a timely manner.

Theorem 4.12.1 (Expected Future Expectations) can be generalized to cases where the LUV on day n is multiplied by an expressible feature:

Theorem 4.12.3 (No Expected Net Update under Conditionals)**.** *Let f be a deferral function, and let $\overline{X}$ denote an e.c. sequence of $[0,1]$-LUVs, and let $\overline{w}$ denote a $\overline{\mathbb{P}}$-generable sequence of real numbers in $[0,1]$. Then*

$$\mathbb{E}_n(``\underline{X_n} \cdot \underline{w}_{\underline{f(n)}}") \eqsim_n \mathbb{E}_n(``\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n}) \cdot \underline{w}_{\underline{f(n)}}").$$

(Proof in: F.7.)

To see why Theorem 4.12.3 is interesting, it helps to imagine the case where $\overline{X}$ is a series of bundles of goods and services, and w_n is $\mathrm{Ind}_{\delta_n}(\mathbb{E}_{f(n)}(X_n) > 0.7)$ for some sequence of rational numbers $\overline{\delta} \to 0$, as per Definition 4.3.2. This value is 1 if $\overline{\mathbb{P}}$ will expect the nth bundle to be worth more than 70¢ on day $f(n)$, and 0 otherwise, and intermediate if the case isn't quite clear. Then

$$\mathbb{E}_n\left(``\underline{X_n} \cdot \underline{\mathrm{Ind}}_{\delta_n}\left(\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n}) > 0.7\right)"\right)$$

can be interpreted as $\overline{\mathbb{P}}$'s expected value of the bundle on day n, in cases where $\overline{\mathbb{P}}$ is going to think it's worth at least 70¢ on day $f(n)$. Now assume that $\mathrm{Ind}_{\delta_n}(\mathbb{E}_{f(n)}(X_n)) > 0$ and divide it out of both sides, in which case the theorem roughly says

$$\mathbb{E}_{\mathrm{now}}(X \mid \mathbb{E}_{\mathrm{later}}(X) > 0.7) \eqsim \mathbb{E}_{\mathrm{now}}(\mathbb{E}_{\mathrm{later}}(X) \mid \mathbb{E}_{\mathrm{later}}(X) > 0.7),$$

which says that $\overline{\mathbb{P}}$'s expected value of the bundle now, given that it's going to think the bundle has a value of at least 70¢ later, is equal to whatever it expects to think later, conditioned on thinking later that the bundle is worth at least 70¢.

Combining this idea with indicator functions, we get the following theorem:

Theorem 4.12.4 (Self-Trust)**.** *Let f be a deferral function, $\overline{\phi}$ be an e.c. sequence of sentences, $\overline{\delta}$ be an e.c. sequence of positive rational numbers, and $\overline{p}$ be a $\overline{\mathbb{P}}$-generable sequence of rational probabilities. Then*

$$\mathbb{E}_n\left(``\underline{\mathbb{1}(\phi_n)} \cdot \underline{\mathrm{Ind}}_{\delta_n}\left(\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n}) > \underline{p_n}\right)"\right) \gtrsim_n p_n \cdot \mathbb{E}_n\left(``\underline{\mathrm{Ind}}_{\delta_n}\left(\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n}) > \underline{p_n}\right)"\right).$$

(Proof in: F.8.)

Very roughly speaking, if we squint at Theorem 4.12.4, it says something like

$$\mathbb{E}_{\mathrm{now}}(\phi \mid P_{\mathrm{later}}(\phi) > p) \gtrsim p,$$

i.e., if we ask $\overline{\mathbb{P}}$ what it would believe about ϕ now if it learned that it was going to believe ϕ with probability at least p in the future, then it will answer with a probability that is at least p.

As a matter of fact, Theorem 4.12.4 actually says something slightly weaker, which is also more desirable. Let each ϕ_n be the self-referential sentence "$\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n}) < 0.5$" which says that the future $\mathbb{P}_{f(n)}$ will assign probability less than 0.5 to ϕ_n. Then, conditional on $\mathbb{P}_{f(n)}(\phi_n) \geq 0.5$, $\mathbb{P}_n$ should believe that the probability of ϕ_n is 0. And indeed, this is what a logical inductor will do:

$$\mathbb{P}_n\left(``\underline{\phi_n} \wedge (\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n}) \geq 0.5)"\right) \eqsim_n 0,$$

by Theorem 4.2.3 (Persistence of Knowledge), because each of those conjunctions is disprovable. This is why Theorem 4.12.4 uses continuous indicator functions: With discrete conjunctions, the result would be undesirable (not to mention false).

What Theorem 4.12.4 says is that $\overline{\mathbb{P}}$ attains self-trust of the "if in the future I will believe x is very likely, then it must be because x is very likely" variety, while retaining the ability to think it can outperform its future self's beliefs when its future self confronts paradoxes. In colloquial terms, if we ask "what's your probability on the paradoxical sentence ϕ_n given that your future self believes it with probability *exactly* 0.5?" then $\overline{\mathbb{P}}$ will answer "very low", but if we ask "what's your probability on the paradoxical sentence ϕ_n given that your future self believes it with probability *extremely close to* 0.5?" then $\overline{\mathbb{P}}$ will answer "roughly 0.5."

Still speaking roughly, this means that logical inductors trust their future beliefs to be accurate and only change for good reasons. Theorem 4.12.4 says that if you ask "what's the probability of ϕ, given that in the future you're going to believe it's more than 95% likely?" then you'll get an answer that's no less than 0.95, even if the logical inductor currently thinks that ϕ is unlikely.

5 Construction

In this section, we show how given any deductive process $\overline{D}$, we can construct a computable belief sequence, called $\overline{\text{LIA}}$, that satisfies the logical induction criterion relative to $\overline{D}$. Roughly speaking, $\overline{\text{LIA}}$ works by simulating an economy of traders and using Brouwer's fixed point theorem to set market prices such that no trader can exploit the market relative to $\overline{D}$.

We will build LIA from three subroutines called MarketMaker, Budgeter, and TradingFirm. Intuitively, MarketMaker will be an algorithm that sets market prices by anticipating what a single trader is about to do, Budgeter will be an algorithm for altering a trader to stay within a certain budget, and TradingFirm will be an algorithm that uses Budgeter to combine together an infinite sequence of carefully chosen e.c. traders (via a sum calculable in finite time) into a single trader that exploits a given market if any e.c. trader exploits that market. Then, LIA will work by using MarketMaker to make a market not exploitable by TradingFirm and hence not exploitable by any e.c. trader, thereby satisfying the logical induction criterion.

To begin, we will need a few basic data types for our subroutines to pass around:

Definition 5.0.1 (Belief History). *An **n-belief history** $\mathbb{P}_{\leq n} = (\mathbb{P}_1, \ldots, \mathbb{P}_n)$ is a finite list of belief states of length n.*

Definition 5.0.2 (Strategy History). *An **n-strategy history** $T_{\leq n} = (T_1, \ldots, T_n)$ is a finite list of trading strategies of length n, where T_i is an i-strategy.*

Definition 5.0.3 (Support). *For any valuation $\mathbb{V}$ we define*

$$\text{Support}(\mathbb{V}) := \{\phi \in \mathcal{S} \mid \mathbb{V}(\phi) \neq 0\},$$

and for any n-strategy T_n we define

$$\text{Support}(T_n) := \{\phi \in \mathcal{S} \mid T_n[\phi] \not\equiv 0\}.$$

Observe that for any belief state $\mathbb{P}$ and any n-strategy T_n, $\text{Support}(\mathbb{P})$ and $\text{Support}(T_n)$ are computable from the finite lists representing $\mathbb{P}$ and T_n.

5.1 Constructing MarketMaker

Here we define the MarketMaker subroutine and establish its key properties. Intuitively, given any trader $\overline{T}$ as input, on each day n, MarketMaker looks at the trading strategy T_n and the valuations $\mathbb{P}_{\leq n-1}$ output by MarketMaker on previous days. It then uses an approximate fixed point (guaranteed to exist by Brouwer's fixed point theorem) that sets prices $\mathbb{P}_n$ for that day such that when the trader's strategy T_n reacts to the prices, the resulting trade $T_n(\mathbb{P}_{\leq n})$ earns at most a very small positive amount of value in any world. Intuitively, the fixed point finds the trader's "fair prices", such that they abstain from betting, except possibly to buy sentences at a price very close to \$1 or sell them at a price very close to \$0, thereby guaranteeing that very little value can be gained from the trade.

Lemma 5.1.1 (Fixed Point Lemma). *Let T_n be any n-strategy, and let $\mathbb{P}_{\leq n-1}$ be any $(n-1)$-belief history. There exists a valuation $\mathbb{V}$ with $\text{Support}(\mathbb{V}) \subseteq \text{Support}(T_n)$ such that*

$$\text{for all worlds } \mathbb{W} \in \mathcal{W}: \quad \mathbb{W}\left(T_n(\mathbb{P}_{\leq n-1}, \mathbb{V})\right) \leq 0. \tag{5.1.1}$$

Proof. We will use Brouwer's fixed point theorem to find "prices" $\mathbb{V}$ such that T_n only ever buys shares for \$1 or sells them for \$0, so it cannot make a profit in any world. Intuitively, we do this by making a "price adjustment" mapping called fix that moves prices toward 1 or 0 (respectively) as long as T_n would buy or sell (respectively) any shares at those prices, and finding a fixed point of that mapping.

First, we let $\mathcal{S}' = \text{Support}(T_n)$ and focus on the set

$$\mathcal{V}' := \{\mathbb{V} \mid \text{Support}(\mathbb{V}) \subseteq \mathcal{S}'\}.$$

Observe that $\mathcal{V}'$ is equal to the natural inclusion of the finite-dimensional cube $[0,1]^{\mathcal{S}'}$ in the space of all valuations $\mathcal{V} = [0,1]^{\mathcal{S}}$. We now define our "price adjustment" function $\text{fix} : \mathcal{V}' \to \mathcal{V}'$ as follows:

$$\text{fix}(\mathbb{V})(\phi) := \max(0, \ \min(1, \ \mathbb{V}(\phi) + T(\mathbb{P}_{\leq n-1}, \mathbb{V})[\phi])).$$

This map has the odd property that it adds prices and trade volumes, but it does the trick. Notice that fix is a function from the compact, convex space $\mathcal{V}'$ to itself, so if it is continuous, it satisfies the antecedent of Brouwer's fixed point theorem. Observe that fix is in fact continuous, because trade strategies are continuous. Indeed, we required that trade strategies be continuous for precisely this purpose. Thus, by Brouwer's fixed point theorem, fix has at least one fixed point $\mathbb{V}^{\text{fix}}$ that satisfies, for all sentences $\phi \in \mathcal{S}'$,

$$\mathbb{V}^{\text{fix}}(\phi) = \max(0, \ \min(1, \ \mathbb{V}^{\text{fix}}(\phi) + T_n(\mathbb{P}_{\leq n-1}, \mathbb{V}^{\text{fix}})[\phi])).$$

Fix a world $\mathbb{W}$ and observe from this equation that if T_n buys some shares of $\phi \in \mathcal{S}'$ at these prices, i.e. if $T_n(\mathbb{P}_{\leq n-1}, \mathbb{V}^{\text{fix}})[\phi] > 0$, then $\mathbb{V}^{\text{fix}}(\phi) = 1$, and in particular, $\mathbb{W}(\phi) - \mathbb{V}^{\text{fix}}(\phi) \leq 0$. Similarly, if T_n sells some shares of ϕ, i.e. if $T_n(\mathbb{P}_{\leq n}, \mathbb{V}^{\text{fix}})[\phi] < 0$, then $\mathbb{V}^{\text{fix}}(\phi) = 0$, so $\mathbb{W}(\phi) - \mathbb{V}^{\text{fix}}(\phi) \geq 0$. In either case, we have

$$0 \geq (\mathbb{W}(\phi) - \mathbb{V}^{\text{fix}}(\phi)) \cdot T_n(\mathbb{P}_{\leq n-1}, \mathbb{V}^{\text{fix}})[\phi]$$

since the two factors always have opposite sign (or at least one factor is 0). Summing over all ϕ, remembering that $T_n(\mathbb{V}_{\leq n})[\phi] = 0$ for $\phi \notin \mathcal{S}'$, gives

$$0 \geq \sum_{\phi \in \mathcal{S}} (\mathbb{W}(\phi) - \mathbb{V}^{\text{fix}}(\phi)) \cdot T_n(\mathbb{P}_{\leq n-1}, \mathbb{V}^{\text{fix}})[\phi]$$

$$= \mathbb{W}(T_n(\mathbb{P}_{\leq n}, \mathbb{V}^{\text{fix}})) - \mathbb{V}^{\text{fix}}(T_n(\mathbb{P}_{\leq n-1}, \mathbb{V}^{\text{fix}}))$$

since the values of the "cash" terms $\mathbb{W}(T_n(\mathbb{P}_{\leq n}, \mathbb{V}^{\text{fix}})[1])$ and $\mathbb{V}^{\text{fix}}(T_n(\mathbb{P}_{\leq n}, \mathbb{V}^{\text{fix}})[1])$ are by definition both equal to $T_n(\mathbb{P}_{\leq n}, \mathbb{V}^{\text{fix}})[1]$ and therefore cancel. But

$$\mathbb{V}^{\text{fix}}(T_n(\mathbb{P}_{\leq n-1}, \mathbb{V}^{\text{fix}})) = 0$$

by definition of a trading strategy, so for any world $\mathbb{W}$, we have

$$0 \geq \mathbb{W}(T_n(\mathbb{P}_{\leq n-1}, \mathbb{V}^{\text{fix}})).$$

$\square$

Definition/Proposition 5.1.2 (`MarketMaker`). *There exists a computable function, henceforth named* `MarketMaker`, *satisfying the following definition. Given as input any $n \in \mathbb{N}^+$, any n-strategy T_n, and any $(n-1)$-belief history $\mathbb{P}_{\leq n-1}$,* `MarketMaker`$_n(T_n, \mathbb{P}_{\leq n-1})$ *returns a belief state $\mathbb{P}$ with $\text{Support}(\mathbb{P}) \subseteq \text{Support}(T_n)$ such that*

$$\text{for all worlds } \mathbb{W} \in \mathcal{W}: \quad \mathbb{W}(T_n(\mathbb{P}_{\leq n-1}, \mathbb{P})) \leq 2^{-n}. \tag{5.1.2}$$

Proof. Essentially, we will find a rational approximation $\mathbb{P}$ to the fixed point $\mathbb{V}^{\text{fix}}$ in the previous lemma, by brute force search. This requires some care, because the set of all worlds is uncountably infinite.

First, given T_n and $\mathbb{P}_{\leq n-1}$, let $\mathcal{S}' := \text{Support}(T_n)$, $\mathcal{V}' = \{\mathcal{V} \mid \text{Support}(\mathbb{V}) \subseteq \mathcal{S}'\}$, and take $\mathbb{V}^{\text{fix}} \in \mathcal{V}'$ satisfying (5.1.1). Let $\mathcal{W}' := \{\mathbb{W}' \mid \text{Support}(\mathbb{W}) \in \mathcal{S}'\}$, and for any $\mathbb{W}$, define $\mathbb{W}' \in \mathcal{W}$ by

$$\mathbb{W}'(\phi) := \begin{cases} \mathbb{W}(\phi) & \text{if } \phi \in \mathcal{S}', \\ 0 & \text{otherwise.} \end{cases}$$

Observe that for any $\mathbb{W} \in \mathcal{W}$, the function $\mathcal{V}' \to \mathbb{R}$ given by

$$\mathbb{V} \mapsto \mathbb{W}(T_n(\mathbb{P}_{\leq n-1}, \mathbb{V})) = \mathbb{W}'(T_n(\mathbb{P}_{\leq n-1}, \mathbb{V}))$$

is a continuous function of $\mathbb{V}$ that depends only on $\mathbb{W}'$. Since the set $\mathcal{W}'$ is finite, the function

$$\mathbb{V} \mapsto \sup_{\mathbb{W} \in \mathcal{W}} \mathbb{W}(T_n(\mathbb{P}_{\leq n-1}, \mathbb{V})) = \max_{\mathbb{W}' \in \mathcal{W}'} \mathbb{W}'(T_n(\mathbb{P}_{\leq n-1}, \mathbb{V}))$$

is the maximum of a finite number of continuous non-positive functions, and is therefore continuous and non-positive. Hence there is some neighborhood in $\mathcal{V}'$ around $\mathbb{V}^{\text{fix}}$ with image in $(-\infty, 2^{-n}) \subset \mathbb{R}$. By the density of rational points in $\mathcal{V}'$, there is therefore some belief state $\mathbb{P} \in \mathcal{V}' \cap \mathbb{Q}^{\mathcal{S}}$ satisfying (5.1.2), as needed.

It remains to show that such a $\mathbb{P}$ can in fact be found by brute force search. First, recall that a belief state $\mathbb{P}$ is a rational-valued finite-support map from $\mathcal{S}$ to $[0,1]$, and so can be represented by a finite list of pairs (ϕ, q) with $\phi \in \mathcal{S}$ and $q \in \mathbb{Q} \cap [0,1]$. Since $\mathcal{S}$ and $[0,1] \cap \mathbb{Q}$ are computably enumerable, so is the set of all belief states.

Thus, we can computably "search" though all possible $\mathbb{P}$s, so we need only establish that given n, T_n, and $\mathbb{P}_{\leq n-1}$ we can computably decide whether each $\mathbb{P}$ in our search satisfies (5.1.2) until we find one. First note that the finite set $\text{Support}(T_n)$ can be computed by searching the expression specifying T_n for all the sentences ϕ that occur within it. Moreover, equation (5.1.2) need only be be checked for worlds $\mathbb{W}' \in \mathcal{W}'$, since any other $\mathbb{W}$ returns the same value as its corresponding $\mathbb{W}'$. Now, for any fixed world $\mathbb{W}' \in \mathcal{W}'$ and candidate $\mathbb{P}$, we can compute each value in the language of expressible features

$$\mathbb{W}'(T_n(\mathbb{P}_{\leq n-1}, \mathbb{P})) = T_n(\mathbb{P}_{\leq n-1}, \mathbb{P})[\phi] + \sum_{\phi \in \mathcal{S}'} \mathbb{W}'(\phi) \cdot T_n(\mathbb{P}_{\leq n-1}, \mathbb{P})[\phi]$$

directly by evaluating the expressible features $T_n[\phi]$ on the given belief history $(\mathbb{P}_{\leq n-1}, \mathbb{P})$, as $\phi \in \mathcal{S}'$ varies. Since $\mathcal{W}'$ is a finite set, we can do this for all $\mathbb{W}'$ with a finite computation. Thus, checking whether a belief state $\mathbb{P}$ satisfies condition (5.1.2) is computably decidable, and a solution to (5.1.2) can therefore be found by enumerating all belief states $\mathbb{P}$ and searching through them for the first one that works. $\square$

Lemma 5.1.3 (`MarketMaker` Inexploitability). *Let $\overline{T}$ be any trader. The sequence of belief states $\overline{\mathbb{P}}$ defined recursively by*

$$\mathbb{P}_n := \texttt{MarketMaker}_n(T_n, \mathbb{P}_{\leq n-1}),$$

with base case $\mathbb{P}_1 = \texttt{MarketMaker}(T_1, ())$, is not exploited by $\overline{T}$ relative to any deductive process $\overline{D}$.

Proof. By the definition of `MarketMaker`, we have that for every n, the belief state $\mathbb{P} = \mathbb{P}_n$ satisfies equation (5.1.2), i.e.,

$$\text{for all worlds } \mathbb{W} \in \mathcal{W} \text{ and all } n \in \mathbb{N}^+: \quad \mathbb{W}(T_n(\overline{\mathbb{P}})) \leq 2^{-n}.$$

Hence by linearity of $\mathbb{W}$, for all $n \in \mathbb{N}^+$ we have:

$$\mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}})\right) = \sum_{i \leq n} \mathbb{W}(T_i(\overline{\mathbb{P}})) \leq \sum_{i \leq n} 2^{-i} < 1.$$

Therefore, given any deductive process $\overline{D}$,

$$\sup\left\{\mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}})\right) \,\Big|\, n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n)\right\} \leq 1 < \infty,$$

so $\overline{T}$ does not exploit $\overline{\mathbb{P}}$ relative to $\overline{D}$. $\qquad\square$

5.2   Constructing `Budgeter`

Here we introduce a subroutine for turning a trader with potentially infinite losses into a trader that will never have less than $-\$b$ in any world $\mathbb{W} \in \mathcal{PC}(D_n)$ on any day n, for some bound b, in such a way that does not affect the trader if it wouldn't have fallen below $-\$b$ to begin with.

Definition/Proposition 5.2.1 (`Budgeter`). *Given any deductive process $\overline{D}$, there exists a computable function, henceforth called $\mathtt{Budgeter}^{\overline{D}}$, satisfying the following definition. Given inputs n and $b \in \mathbb{N}^+$, an n-strategy history $T_{\leq n}$, and an $(n-1)$-belief history $\mathbb{P}_{\leq n-1}$, $\mathtt{Budgeter}^{\overline{D}}$ returns an n-strategy $\mathtt{Budgeter}_n^{\overline{D}}(b, T_{\leq n}, \mathbb{P}_{\leq n-1})$, such that*

$$\textit{if:} \quad \mathbb{W}\left(\sum_{i \leq m} T_i(\mathbb{P}_{\leq i})\right) \leq -b \textit{ for some } m < n \textit{ and } \mathbb{W} \in \mathcal{PC}(D_m),$$

$$\textit{then:} \quad \mathtt{Budgeter}_n^{\overline{D}}(b, T_{\leq n}, \mathbb{P}_{\leq n-1}) = 0,$$

$$\textit{else:} \quad \mathtt{Budgeter}_n^{\overline{D}}(b, T_{\leq n}, \mathbb{P}_{\leq n-1}) = \qquad\qquad (5.2.1)$$

$$T_n \cdot \inf_{\mathbb{W} \in \mathcal{PC}(D_n)} \left[\max\left(1, \frac{-\mathbb{W}(T_n)}{b + \mathbb{W}\left(\sum_{i \leq n-1} T_i(\mathbb{P}_{\leq i})\right)}\right)\right]^{-1}.$$

Proof. Let $\mathcal{S}' = \bigcup_{i \leq n} \mathrm{Support}(T_i)$, $\mathcal{W}' = \{\mathbb{W} \mid \mathrm{Support}(\mathbb{W}) \subseteq \mathcal{S}'\}$, and for any world $\mathbb{W}$, write

$$\mathbb{W}'(\phi) := \begin{cases} \mathbb{W}(\phi) & \text{if } \phi \in \mathcal{S}', \\ 0 & \text{otherwise.} \end{cases}$$

Now, observe that we can computably check the "if" statement in the function definition. This is because $\mathbb{W}(\sum_{i \leq m} T_i(\mathbb{P}_{\leq i}))$ depends only on $\mathbb{W}' \in \mathcal{W}'$, a finite set. We can check whether $\mathbb{W}' \in \mathcal{PC}(D_m)$ in finite time by checking whether any assignment of truth values to the finite set of prime sentences occurring in sentences of D_n yields the assignment $\mathbb{W}'$ on $\mathrm{Support}(\mathbb{W}')$. The set of sentences D_n is computable given n, because $\overline{D}$ is computable by definition.

It remains to show that the "else" expression can be computed and returns an n-trading strategy. First, the infimum can be computed over $\mathbb{W}' \in \mathcal{W}' \cap \mathcal{PC}(D_n)$, a finite set, since the values in the inf depend only on $\mathbb{W}'$, and the inf operator itself can be re-expressed in the language of expressible features using max and multiplication by (-1). The values $\mathbb{W}'(T_n)$ and $\mathbb{W}'(\sum_{i \leq n-1} T_i(\mathbb{P}_{\leq i}))$ are finite sums, and the denominator $b + \mathbb{W}(\sum_{i \leq n-1} T_i(\mathbb{P}_{\leq i}))$ is a fixed positive rational (so we can safely multiply by its reciprocal). The remaining operations are all single-step evaluations in the language of expressible valuation features, completing the proof. $\qquad\square$

Let us reflect on the meaning of these operations. The quantity $b + \mathbb{W}(\sum_{i < n} T_i(\mathbb{P}_{\leq i}))$ is the amount of money the trader has available on day n according to $\mathbb{W}$ (assuming

they started with a budget of b), and $-\mathbb{W}(T_n)$ is the amount they're going to lose on day n according to $\mathbb{W}$ as a function of the upcoming prices, and so the infimum above is the trader's trade on day n scaled down such that they can't overspend their budget according to any world propositionally consistent with D_n.

Lemma 5.2.2 (Properties of `Budgeter`). *Let $\overline{T}$ be any trader, and $\overline{\mathbb{P}}$ be any sequence of belief states. Given n and b, let B_n^b denote* `Budgeter`$_n^{\overline{D}}(b, T_{\leq n}, \mathbb{P}_{\leq n-1})$*. Then:*

1. *for all $b, n \in \mathbb{N}^+$, if for all $m \leq n$ and $\mathbb{W} \in \mathcal{PC}(D_m)$ we have $\mathbb{W}\left(\sum_{i \leq m} T_i(\overline{\mathbb{P}})\right) > -b$, then*

$$B_n^b(\overline{\mathbb{P}}) = T_n(\overline{\mathbb{P}});$$

2. *for all $b, n \in \mathbb{N}^+$ and all $\mathbb{W} \in \mathcal{PC}(D_n)$, we have*

$$\mathbb{W}\left(\sum_{i \leq n} B_i^b(\overline{\mathbb{P}})\right) \geq -b;$$

3. *If $\overline{T}$ exploits $\overline{\mathbb{P}}$ relative to $\overline{D}$, then so does $\overline{B}^b$ for some $b \in \mathbb{N}^+$.*

Part 1.

Proof. Suppose that for some time step n, for all $m \leq n$ and all worlds $\mathbb{W} \in \mathcal{PC}(D_m)$ plausible at time m we have

$$\mathbb{W}\left(\sum_{i \leq m} T_i(\overline{\mathbb{P}})\right) > -b,$$

so by linearity of $\mathbb{W}(-)$, we have in particular that

$$b + \mathbb{W}\left(\sum_{i \leq n-1} T_i(\overline{\mathbb{P}})\right) > -\mathbb{W}\left(T_n(\overline{\mathbb{P}})\right).$$

Since $n - 1 \leq n$, the LHS is positive, so we have

$$1 > \frac{-\mathbb{W}\left(T_n(\overline{\mathbb{P}})\right)}{b + \mathbb{W}\left(\sum_{i \leq n-1} T_i(\overline{\mathbb{P}})\right)}.$$

Therefore, by the definition of `Budgeter`$^{\overline{D}}$ (and $T_i(\overline{\mathbb{P}}) = T_i(\mathbb{P}_{\leq i})$), since the "if" clause doesn't trigger by the assumption on the $\mathbb{W}\left(\sum_{i \leq m} T_i(\overline{\mathbb{P}})\right)$ for $m < n$,

$$B_n^b(\overline{\mathbb{P}}) \equiv T_n(\overline{\mathbb{P}}) \cdot \inf_{\mathbb{W} \in \mathcal{PC}(D_n)} 1 \Big/ \max\left(1, \frac{-\mathbb{W}(T_n(\overline{\mathbb{P}}))}{b + \mathbb{W}\left(\sum_{i \leq n-1} T_i(\overline{\mathbb{P}})\right)}\right)$$

$$= T_n(\mathbb{P}_{\leq n}) \cdot \inf_{\mathbb{W} \in \mathcal{PC}(D_n)} 1/1$$

$$= T_n(\overline{\mathbb{P}})$$

as needed. $\qquad\square$

Part 2.

Proof. Suppose for a contradiction that for some n and some $\mathbb{W} \in \mathcal{PC}(D_n)$,

$$\mathbb{W}\left(\sum_{i \leq n} B_i^b(\overline{\mathbb{P}})\right) < -b.$$

Assume that n is the least such day, and fix some such $\mathbb{W} \in \mathcal{PC}(D_n)$. By the minimality of n it must be that $\mathbb{W}(B_n^b(\overline{\mathbb{P}})) < 0$, or else we would have

$\mathbb{W}\left(\sum_{i \leq n-1} B_i^b(\overline{\mathbb{P}})\right) < -b$. Since $B_n^b(\overline{\mathbb{P}})$ is a non-negative multiple of $T_n(\overline{\mathbb{P}})$, we also have $\mathbb{W}(T_n(\overline{\mathbb{P}})) < 0$. However, since $B_n^b \not\equiv 0$, from the definition of $\texttt{Budgeter}^{\overline{D}}$ we have

$$\mathbb{W}\left(B_n^b\right) = \mathbb{W}\left(T_n(\overline{\mathbb{P}})\right) \cdot \left(\inf_{\mathbb{W}' \in \mathcal{PC}(D_n)} 1 \Big/ \max\left(1, \frac{-\mathbb{W}'(T_n(\overline{\mathbb{P}}))}{b + \mathbb{W}'(\sum_{i \leq n-1} T_i(\overline{\mathbb{P}}))}\right)\right)$$

$$\geq \mathbb{W}\left(T_n(\overline{\mathbb{P}})\right) \cdot 1 \Big/ \max\left(1, \frac{-\mathbb{W}(T_n(\overline{\mathbb{P}}))}{b + \mathbb{W}(\sum_{i \leq n-1} T_i(\overline{\mathbb{P}}))}\right) \quad \text{(since } \mathbb{W}\left(T_n(\overline{\mathbb{P}})\right) < 0)$$

$$\geq \mathbb{W}\left(T_n(\overline{\mathbb{P}})\right) \cdot \frac{b + \mathbb{W}(\sum_{i \leq n-1} T_i(\overline{\mathbb{P}}))}{-\mathbb{W}(T_n(\overline{\mathbb{P}}))}$$

since $-\mathbb{W}\left(T_n(\overline{\mathbb{P}})\right) > 0$ and $B_n^b \not\equiv 0$ implies $b + \mathbb{W}(\sum_{i \leq n-1} T_i(\overline{\mathbb{P}})) > 0$. Hence, this

$$= -b - \mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}})\right).$$

Further, since $B_n^b \not\equiv 0$, we have

for all $j \leq n-1$: $\quad \mathbb{W}\left(\sum_{i \leq j} T_i(\overline{\mathbb{P}})\right) > -b$, which by Part 1 implies that

for all $j \leq n-1$: $\quad B_j^b(\overline{\mathbb{P}}) = T_j(\overline{\mathbb{P}})$, therefore

$$\mathbb{W}(B_n^b) \geq -b - \mathbb{W}\left(\sum_{i \leq n-1} B_i^b(\overline{\mathbb{P}})\right), \text{ hence}$$

$$\mathbb{W}\left(\sum_{i \leq n} B_i^b(\overline{\mathbb{P}})\right) \geq -b. \qquad \square$$

Part 3.

Proof. By definition of exploitation, the set

$$\left\{\mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}})\right) \,\Big|\, n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n)\right\}$$

is unbounded above, and is strictly bounded below by some integer b. Then by Part 1, for all n we have $T_n(\overline{\mathbb{P}}) = B_n^b(\overline{\mathbb{P}})$. Thus,

$$\left\{\mathbb{W}\left(\sum_{i \leq n} B_i^b(\overline{\mathbb{P}})\right) \,\Big|\, n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n)\right\}$$

is unbounded above and bounded below, i.e., $\overline{B}^b$ exploits $\overline{\mathbb{P}}$ relative to $\overline{D}$. $\qquad \square$

5.3 Constructing `TradingFirm`

Next we define `TradingFirm`, which combines an (enumerable) infinite sequence of e.c. traders into a single "supertrader" that exploits a given belief sequence $\overline{\mathbb{P}}$ relative to $\overline{D}$ if any e.c. trader does. It does this by taking each e.c. trader, budgeting it, and scaling its trades down so that traders later in the sequence carry less weight to begin with.

To begin, we will need a computable sequence that includes every e.c. trader at least once. The following trick is standard, but we include it here for completeness:

Proposition 5.3.1 (Redundant Enumeration of e.c. Traders)**.** *There exists a computable sequence* $(\overline{T}^k)_{k \in \mathbb{N}^+}$ *of e.c. traders such that every e.c. trader occurs at least once in the sequence.*

Proof. Fix a computable enumeration of all ordered pairs (M_k, f_k) where M_k is a Turing machine and f_k is a polynomial with coefficients in $\mathbb{Z}$. We define a computable function

$$\text{ECT} : \{\text{Turing machines}\} \times \{\text{Integer polynomials}\} \times (n \in \mathbb{N}^+) \to \{n\text{-strategies}\}$$

that runs as follows: $\text{ECT}(M, f, n)$ first runs $M(n)$ for up to $f(n)$ time steps, and if in that time $M(n)$ halts and returns a valid n-strategy T_n, then $\text{ECT}(M, f, n)$ returns that strategy, otherwise it returns 0 (as an n-strategy). Observe that $\text{ECT}(M_k, f_k, -)$ is always an e.c. trader, and that every e.c. trader occurs as $\text{ECT}(M_k, f_k, -)$ for some k. $\qquad\square$

Definition/Proposition 5.3.2 (`TradingFirm`). *Given any deductive process $\overline{D}$, there exists a computable function, henceforth called $\texttt{TradingFirm}^{\overline{D}}$, satisfying the following definition. By Proposition 5.3.1, we fix a computable enumeration $\overline{T}^k$ including every e.c. trader at least once, and let*

$$S_n^k = \begin{cases} T_n^k & \text{if } n \geq k \\ 0 & \text{otherwise.} \end{cases}$$

Given input $n \in \mathbb{N}^+$ and an $(n-1)$-belief history $\mathbb{P}_{\leq n-1}$, $\texttt{TradingFirm}^{\overline{D}}$ returns an n-strategy given by

$$\texttt{TradingFirm}_n^{\overline{D}}(\mathbb{P}_{\leq n-1}) = \sum_{k \in \mathbb{N}^+} \sum_{b \in \mathbb{N}^+} 2^{-k-b} \cdot \texttt{Budgeter}_n^{\overline{D}}(b, S_{\leq n}^k, \mathbb{P}_{\leq n-1}). \quad (5.3.2)$$

Proof. We need only show that the infinite sum in equation (5.3.2) is equivalent to a computable finite sum. Writing

$$B_n^{b,k} = \texttt{Budgeter}_n^{\overline{D}}(b, S_{\leq n}^k, \mathbb{P}_{\leq n-1}),$$

(an n-strategy), the sum on the RHS of (5.3.2) is equivalent to

$$\sum_{k \in \mathbb{N}^+} \sum_{b \in \mathbb{N}^+} 2^{-k-b} \cdot B_n^{b,k}.$$

Since $S_n^k = 0$ for $k > n$, we also have $B_n^{b,k} = 0$ for $k > n$, so the sum is equivalent to

$$= \sum_{k \leq n} \sum_{b \in \mathbb{N}^+} 2^{-k-b} \cdot B_n^{b,k}.$$

Now, assume C_n is a positive integer such that $\sum_{i \leq n} \|S_i^k(\overline{\mathbb{V}})\|_1 < C_n$ for all $k \leq n$ and any valuation sequence $\overline{\mathbb{V}}$ (we will show below that such a C_n can be computed from $\mathbb{P}_{\leq n-1}$). Since the valuations $\mathbb{W}$ and $\overline{\mathbb{P}}$ are always $[0,1]$-valued, for any $m \leq n$ the values $\mathbb{W}\left(\sum_{i \leq m} S_i^k(\mathbb{P}_{\leq m})\right)$ are bounded below by $-\sum_{i \leq m} \|S_i^k(\mathbb{P}_{\leq m})\|_1 > -C_n$. By property 1 of $\texttt{Budgeter}^{\overline{D}}$ (Lemma 5.2.2.1), $B_n^{b,k} = S_n^k$ when $b > C_n$, so the sum is equivalent to

$$= \left(\sum_{k \leq n} \sum_{b \leq C_n} 2^{-k-b} \cdot B_n^{b,k}\right) + \left(\sum_{k \leq n} \sum_{b > C_n} 2^{-k-b} \cdot S_n^k\right)$$

$$= \left(\sum_{k \leq n} \sum_{b \leq C_n} 2^{-k-b} \cdot B_n^{b,k}\right) + \left(\sum_{k \leq n} 2^{-k-C_n} \cdot S_n^k\right)$$

which is a finite sum of trading strategies, and hence is itself a trading strategy. Since the $B_n^{b,k}$ and the S_n^k are computable from $\mathbb{P}_{\leq n-1}$, this finite sum is computable.

It remains to justify our assumption that integers C_n can be computed from $\mathbb{P}_{\leq n-1}$ with $C_n > \sum_{i \leq n} \|S_i^k(\overline{\mathbb{V}})\|_1$ for all $k \leq n$ and $\overline{\mathbb{V}}$. To see this, first consider how to bound a single expressible feature ξ. We can show by induction on the structure of ξ (see A.2) that, given constant bounds on the absolute value $|\zeta(\overline{\mathbb{V}})|$ of each subexpression ζ of ξ, we can compute a constant bound on $|\xi(\overline{\mathbb{V}})|$; for example,

the bound on $\zeta \cdot \eta$ is the product of the bound on ζ and the bound on η. Thus, given a single trading strategy S_i^k and any ϕ, we can compute a constant upper bound on $|S_i^k[\phi](\overline{\mathbb{V}})|$ for all $\overline{\mathbb{V}}$. Since $\|S_i^k(\overline{\mathbb{V}})\|_1 \leq \sum_{\phi \in \mathrm{Support}(S_i^k)} 2|S_i^k[\phi](\overline{\mathbb{V}})|$ and $\mathrm{Support}(S_i^k)$ is computable, we can bound each $\|S_i^k(\overline{\mathbb{V}})\|_1$, and hence also $\sum_{i \leq n} \|S_i^k(\overline{\mathbb{V}})\|_1$, as needed.

$\square$

Lemma 5.3.3 (Trading Firm Dominance)**.** *Let $\overline{\mathbb{P}}$ be any sequence of belief states, and $\overline{D}$ be a deductive process. If there exists any e.c. trader $\overline{T}$ that exploits $\overline{\mathbb{P}}$ relative to $\overline{D}$, then the sequence*

$$\left(\mathtt{TradingFirm}_n^{\overline{D}}(\mathbb{P}_{\leq n-1}) \right)_{n \in \mathbb{N}^+}$$

also exploits $\overline{\mathbb{P}}$ (relative to $\overline{D}$).

Proof. Suppose that some e.c. trader exploits $\overline{\mathbb{P}}$. That trader occurs as $\overline{T}^k$ for some k in the enumeration used by $\mathtt{TradingFirm}^{\overline{D}}$. First, we show that $\overline{S^k}$ (from the definition of $\mathtt{TradingFirm}^{\overline{D}}$) also exploits $\overline{\mathbb{P}}$. It suffices to show that there exist constants $c_1 \in \mathbb{R}^+$ and $c_2 \in \mathbb{R}$ such that for all $n \in \mathbb{N}^+$ and $\mathbb{W} \in \mathcal{PC}(D_n)$,

$$\mathbb{W}\left(\sum_{i \leq n} S_n^k(\mathbb{P}) \right) \geq c_1 \cdot \mathbb{W}\left(\sum_{i \leq n} T_n^k(\mathbb{P}) \right) + c_2.$$

Taking $c_1 = 1$ and $c_2 = -\sum_{i < k} \|T_i^k(\overline{\mathbb{P}}))\|_1$, where $\|\cdot\|_1$ denotes the ℓ_1 norm on $\mathbb{R}$-combinations of sentences, we have

$$\mathbb{W}\left(\sum_{i \leq n} S_n^k(\mathbb{P}) \right) \geq 1 \cdot \mathbb{W}\left(\sum_{i \leq n} T_n^k(\mathbb{P}) \right) - \left(\sum_{i < k} \|T_i^k(\overline{\mathbb{P}})\|_1 \right),$$

so $\overline{S^k}$ exploits $\overline{\mathbb{P}}$. By Lemma 5.2.2.3, we thus have that for some $b \in \mathbb{N}^+$, the trader $\overline{B}^{b,k}$ given by

$$B_n^{b,k} := \mathtt{Budgeter}_n^{\overline{D}}(b, S_{\leq n}^k, \mathbb{P}_{\leq n-1})$$

also exploits $\overline{\mathbb{P}}$.

Next, we show that the trader $\overline{F}$ given by

$$F_n := \mathtt{TradingFirm}_n^{\overline{D}}(\mathbb{P}_{\leq n-1})$$

exploits $\overline{\mathbb{P}}$. Again, it suffices to show that there exist constants $c_1 \in \mathbb{R}^+$ and $c_2 \in \mathbb{R}$ such that for all $n \in \mathbb{N}^+$ and $\mathbb{W} \in \mathcal{PC}(D_n)$,

$$\mathbb{W}\left(\sum_{i \leq n} F_i \right) \geq c_1 \cdot \mathbb{W}\left(\sum_{i \leq n} B_i^{b,k} \right) + c_2.$$

It will suffice to take $c_1 = 2^{-k-b}$ and $c_2 = -2$, because we have

$$\mathbb{W}\left(\sum_{i \leq n} F_i \right) - 2^{-k-b} \cdot \mathbb{W}\left(\sum_{i \leq n} B_i^{b,k} \right)$$

$$= \sum_{(k',b') \neq (k,b)} 2^{-k'-b'} \cdot \mathbb{W}\left(\sum_{i \leq n} B_i^{b',k'} \right)$$

$$\geq \sum_{(k',b') \neq (k,b)} 2^{-k'-b'} \cdot (-b') \geq -2$$

56

by Lemma 5.2.2.2, hence

$$\mathbb{W}\left(\sum_{i\leq n} F_i\right) \geq 2^{-k-b} \cdot \mathbb{W}\left(\sum_{i\leq n} B_i^{b,k}\right) - 2.$$

Thus, $\overline{F}$ exploits $\overline{\mathbb{P}}$. □

5.4   Constructing $\overline{\texttt{LIA}}$

We are finally ready to build $\texttt{LIA}$. With the subroutines above, the idea is now fairly simple: we pit $\texttt{MarketMaker}$ and $\texttt{TradingFirm}$ against each other in a recursion, and $\texttt{MarketMaker}$ wins. Imagine that on each day, $\texttt{TradingFirm}$ outputs an ever-larger mixture of traders, then $\texttt{MarketMaker}$ carefully examines that mixture and outputs a belief state on which that mixture makes at most a tiny amount of money on net.

Definition/Algorithm 5.4.1 (A Logical Induction Algorithm)**.** *Given a deductive process $\overline{D}$, define the computable belief sequence $\overline{\texttt{LIA}} = (\texttt{LIA}_1, \texttt{LIA}_2, \ldots)$ recursively by*

$$\texttt{LIA}_n := \texttt{MarketMaker}_n(\texttt{TradingFirm}_n^{\overline{D}}(\texttt{LIA}_{\leq n-1}), \texttt{LIA}_{\leq n-1}),$$

beginning from the base case $\texttt{LIA}_{\leq 0} := ()$.

Theorem 5.4.2 ($\texttt{LIA}$ is a Logical Inductor)**.** *$\overline{\texttt{LIA}}$ satisfies the logical induction criterion relative to $\overline{D}$, i.e., LIA is not exploitable by any e.c. trader relative to the deductive process $\overline{D}$.*

Proof. By Lemma 5.3.3, if any e.c. trader exploits $\overline{\texttt{LIA}}$ (relative to $\overline{D}$), then so does the trader $\overline{F} := (\texttt{TradingFirm}_n^{\overline{D}}(\texttt{LIA}_{\leq n-1}))_{n\in\mathbb{N}^+}$. By Lemma 5.1.3, $\overline{F}$ does not exploit $\overline{\texttt{LIA}}$. Therefore no e.c. trader exploits $\overline{\texttt{LIA}}$. □

5.5   Questions of Runtime and Convergence Rates

In this paper, we have optimized our definitions for the theoretical clarity of results rather than for the efficiency of our algorithms. This leaves open many interesting questions about the relationship between runtime and convergence rates of logical inductors that have not been addressed here. Indeed, the runtime of $\texttt{LIA}$ is under-specified because it depends heavily on the particular enumerations of traders and rational numbers used in the definitions of $\texttt{TradingFirm}$ and $\texttt{MarketMaker}$.

For logical inductors in general, there will be some tradeoff between the runtime of $\mathbb{P}_n$ as a function of n and how quickly the values $\mathbb{P}_n(\phi)$ converge to $\mathbb{P}_\infty(\phi)$ as n grows. Quantifying this tradeoff may be a fruitful source of interesting open problems. Note, however, the following important constraint on the convergence rate of any logical inductor, regardless of its implementation, which arises from the halting problem:

Proposition 5.5.1 (Uncomputable Convergence Rates)**.** *Let $\overline{\mathbb{P}}$ be a logical inductor over a theory Γ that can represent computable functions, and suppose $f : \mathcal{S} \times \mathbb{Q}^+ \to \mathbb{N}$ is a function such that for every sentence ϕ, if $\Gamma \vdash \phi$ then $\mathbb{P}_n(\phi) > 1 - \varepsilon$ for all $n > f(\phi, \varepsilon)$. Then f must be uncomputable.*

Proof. Suppose for contradiction that such a computable f were given. We will show that f could be used to computably determine whether $\Gamma \vdash \phi$ for an arbitrary sentence ϕ, a task which is known to be impossible for a first-order theory that can represent computable functions. (If we assumed further that Γ were sound as a theory of the natural numbers, this would allow us to solve the halting problem by letting ϕ be a sentence of the form "M halts".)

Given a sentence ϕ, we run two searches in parallel. If we find that $\Gamma \vdash \phi$, then we return True. If we find that for some $b, n \in \mathbb{N}^+$ we have

$$n > f\left(\phi, \frac{1}{b}\right) \text{ and } \mathbb{P}_n(\phi) \leq 1 - \frac{1}{b}, \tag{5.5.1}$$

57

then we return False. Both of these conditions are computably enumerable since f, $\mathbb{P}_n$, and verifying witnesses to $\Gamma \vdash \phi$ are computable functions.

Suppose first that $\Gamma \vdash \phi$. Then by definition of f we have $\mathbb{P}_n(\phi) > 1 - \frac{1}{b}$ for all $n > f\left(\phi, \frac{1}{b}\right)$, and hence we find a witness for $\Gamma \vdash \phi$ and return True. Now suppose that $\Gamma \nvdash \phi$. Then by Theorem 4.6.2 (Non-Dogmatism) we have that $\mathbb{P}_\infty(\phi) < 1 - \varepsilon$ for some $\varepsilon > 0$, and hence for some b and all sufficiently large n we have $\mathbb{P}_n(\phi) < 1 - 1/b$. Therefore 5.5.1 holds and we return False. Thus our search always halts and returns a Boolean value that correctly indicates whether $\Gamma \vdash \phi$. $\qquad \square$

6 Selected Proofs

In this section, we exhibit a few selected stand-alone proofs of certain key theorems. These theorems hold for any $\overline{\mathbb{P}}$ satisfying the logical induction criterion, which we recall here:

Definition 3.0.1 (The Logical Induction Criterion). *A market $\overline{\mathbb{P}}$ is said to satisfy the **logical induction criterion** relative to a deductive process $\overline{D}$ if there is no efficiently computable trader $\overline{T}$ that exploits $\overline{\mathbb{P}}$ relative to $\overline{D}$. A market $\overline{\mathbb{P}}$ meeting this criterion is called a **logical inductor over $\overline{D}$**.*

Only our notation (Section 2), framework (Section 3), and continuous threshold indicator (Definition 4.3.2) are needed to understand the results and proofs in this section. Shorter proofs of these theorems can be found in the appendix, but those rely on significantly more machinery.

6.1 Convergence

Recall Theorem 4.1.1 and the proof sketch given:

Theorem 4.1.1 (Convergence). *The limit $\mathbb{P}_\infty : \mathcal{S} \to [0,1]$ defined by*

$$\mathbb{P}_\infty(\phi) := \lim_{n \to \infty} \mathbb{P}_n(\phi)$$

exists for all ϕ.

Proof sketch.

> Roughly speaking, if $\overline{\mathbb{P}}$ never makes up its mind about ϕ, then it can be exploited by a trader arbitraging shares of ϕ across different days. More precisely, suppose by way of contradiction that the limit $\mathbb{P}_\infty(\phi)$ does not exist. Then for some $p \in [0,1]$ and $\varepsilon > 0$, we have $\mathbb{P}_n(\phi) < p - \varepsilon$ infinitely often and also $\mathbb{P}_n(\phi) > p + \varepsilon$ infinitely often. A trader can wait until $\mathbb{P}_n(\phi) < p - \varepsilon$ and then buy a share in ϕ at the low market price of $\mathbb{P}_n(\phi)$. Then the trader waits until some later m such that $\mathbb{P}_m(\phi) > p + \varepsilon$, and sells back the share in ϕ at the higher price. This trader makes a total profit of 2ε every time $\mathbb{P}_n(\phi)$ oscillates in this way, at no risk, and therefore exploits $\overline{\mathbb{P}}$. Since $\overline{\mathbb{P}}$ implements a logical inductor, this is not possible; therefore the limit $\mathbb{P}_\infty(\phi)$ must in fact exist.

We will define a trader $\overline{T}$ that executes a strategy similar to this one, and hence exploits the market $\overline{\mathbb{P}}$ if $\lim_{n \to \infty} \mathbb{P}_n(\phi)$ diverges. To do this, there are two technicalities we must deal with. First, the strategy outlined above uses a discontinuous function of the market prices $\mathbb{P}_n(\phi)$, and therefore is not permitted. This is relatively easy to fix using the continuous indicator functions of Definition 4.3.2.

The second technicality is more subtle. Suppose we define our trader to buy ϕ-shares whenever their price $\mathbb{P}_n(\phi)$ is low, and sell them back whenever their price is high. Then it is possible that the trader makes the following trades in sequence against the market $\overline{\mathbb{P}}$: buy 10 ϕ-shares on consecutive days, then sell 10 ϕ-shares; then buy 100 ϕ-shares consecutively, and then sell them off; then buy 1000 ϕ-shares, then sell them off; and so on. Although this trader makes profit on each batch, it

always spends more on the next batch, taking larger and larger risks (relative to the remaining plausible worlds). Then the plausible value of this trader's holdings will be unbounded below, and so it does not exploit $\overline{\mathbb{P}}$. In short, this trader is not tracking its budget, and so may have unboundedly negative plausible net worth. We will fix this problem by having our trader $\overline{T}$ track how many net ϕ-shares it has bought, and not buying too many, thereby maintaining bounded risk. This will be sufficient to prove the theorem.

Proof of Theorem 4.1.1. Suppose by way of contradiction that the limit $\mathbb{P}_\infty$ does not exist. Then, for some sentence ϕ and some rational numbers $p \in [0,1]$ and $\varepsilon > 0$, we have that $\mathbb{P}_n(\phi) < p - \varepsilon$ infinitely often and $\mathbb{P}_n(\phi) > p + \varepsilon$ infinitely often. We will show that $\overline{\mathbb{P}}$ can be exploited by a trader $\overline{T}$ who buys below and sells above these prices infinitely often, contrary to the logical induction criterion.

Definition of the trader $\overline{T}$. We will define $\overline{T}$ recursively along with another sequence of $\mathcal{EF}$-combinations $\overline{H}$ (mnemonic: "holdings") which tracks the sum of the trader's previous trades. Our base cases are

$$T_1 := \overline{0}$$

$$H_1 := \overline{0}.$$

For $n > 1$, we define a recurrence whereby $\overline{T}$ will buy some ϕ-shares whenever $\phi^{*n} < p - \varepsilon/2$, up to $(1 - H_{n-1}[\phi])$ shares when $\phi^{*n} < p - \varepsilon$, and sells some ϕ-shares whenever $\phi^{*n} > p + \varepsilon/2$, up to H_{n-1} shares when $\phi^{*n} > p + \varepsilon$:

$$
\begin{aligned}
T_n[\phi] &:= (1 - H_{n-1}[\phi]) \cdot \mathrm{Ind}_{\varepsilon/2}(\phi^{*n} < p - \varepsilon/2) \\
&\quad - H_{n-1}[\phi] \cdot \mathrm{Ind}_{\varepsilon/2}(\phi^{*n} > p + \varepsilon/2), \\
T_n &:= T_n[\phi] \cdot (\phi - \phi^{*n}) \\
H_n &:= H_{n-1} + T_n.
\end{aligned}
\tag{6.1.1}
$$

The trade coefficients $T[\phi]$ are chosen so that the number of ϕ-shares $H_n[\phi]$ that it owns is always in $[0,1]$ (it never buys more than $1 - H_{n-1}[\phi]$ and never sells more than $H_{n-1}[\phi]$). Observe that each T_n is a valid trading strategy for day n (see Definition 3.4.4) because it is of the form $\xi \cdot (\phi - \phi^{*n})$.

To complete the definition, we must ague that $\overline{T}$ is efficiently computable. For this, observe that the $3n + 2$ definition ($:=$) equations defining $T_1, \ldots, T_n$ above can be written down in time polynomial in n. Thus, a combination of feature expressions defining T_n from scratch can be written down in $\mathrm{poly}(n)$ time (indeed, the expression is just a concatenation of n copies of the three "$:=$" equations written above, along with the base cases), so $\overline{T}$ is efficiently computable.

Proof of exploitation. To show $\overline{T}$ exploits $\overline{\mathbb{P}}$ over $\overline{D}$, we must compute upper and lower bounds on the set of plausible values $\mathbb{W}(H_n(\overline{\mathbb{P}}))$ (since $H_n = \sum_{i \leq n} T_i$) for worlds $\mathbb{W} \in \mathcal{PC}(D_n)$.

While proving exploitation, we leave the constant argument $\overline{\mathbb{P}}$ implicit to reduce clutter, writing, e.g., ϕ^{*i} for $\phi^{*i}(\overline{\mathbb{P}}) = \mathbb{P}_i(\phi)$, $T_n[\phi]$ for $T_n[\phi](\overline{\mathbb{P}})$, and so on.

First, since each $T_i[1] = -T[\phi] \cdot \phi^{*i}$, the trader's "cash" held on day n is

$$H_n[1] = \sum_{i \leq n} T_i[1] = -\sum_{i \leq n} T_i[\phi] \cdot \phi^{*i}$$

which we can regroup, to compare the prices ϕ^{*i} to p, as

$$
\begin{aligned}
H_n[1] &= \sum_{i \leq n} \big(T_i[\phi] \cdot (p - \phi^{*i})\big) - p \cdot \sum_{i \leq n} T_i[\phi] \\
&= \sum_{i \leq n} \big(T_i[\phi] \cdot (p - \phi^{*i})\big) - p \cdot H_n[\phi].
\end{aligned}
$$

Now, $T_i[\phi] > 0$ iff $\phi^{*i} < p - \varepsilon/2$, and $T_i[\phi] < 0$ iff $\phi^{*i} > p + \varepsilon/2$, so for all i the product $T_i[\phi] \cdot (p - \phi^{*i})$ is greater than $|T_i[\phi]| \cdot \varepsilon/2$:

$$H_n[1] \geq -p \cdot H_n[\phi] + \sum_{i \leq n} |T_i[\phi]| \cdot \varepsilon/2.$$

Moreover, by design, $H_n[\phi] \in [0, 1]$ for all n, so

$$H_n[1] \geq -p + \sum_{i \leq n} |T_i[\phi]| \cdot \varepsilon/2.$$

Now, by assumption, ϕ^{*i} lies above and below $(p - \varepsilon, p + \varepsilon)$ infinitely often, so from equation (6.1.1), $H_i[\phi] = 0$ and $H_i[\phi] = 1$ infinitely often. Since the sum $\sum_{i \leq n} |T_i[\phi]|$ is the total variation in the sequence $H_i[\phi]$, it must diverge (by the triangle inequality) as $n \to \infty$, so

$$\lim_{n \to \infty} H_n[1] = \infty.$$

Moreover, in any world $\mathbb{W}$, the trader's non-cash holdings $H_n[\phi] \cdot \phi$ have value $\mathbb{W}(H_n[\phi] \cdot \phi) = H_n[\phi] \cdot \mathbb{W}(\phi) \geq 0$ (since $H_n[\phi] > 0$), so its combined holdings $H_n = H_n[1] + H_n[\phi] \cdot \phi$ have value

$$\mathbb{W}(H_n) = \mathbb{W}(H_n[1] + H_n[\phi] \cdot \phi) = H_n[1] + H_n[\phi] \cdot \mathbb{W}(\phi) \geq H_n[1]$$

so in *every* world $\mathbb{W}$ we have

$$\lim_{n \to \infty} \mathbb{W}(H_n) = \infty.$$

This contradicts that $\overline{\mathbb{P}}$ is a logical inductor; therefore, the limit $\mathbb{P}_\infty(\phi)$ must exist. $\qquad\square$

6.2 Limit Coherence

Recall Theorem 4.1.2:

Theorem 4.1.2 (Limit Coherence)**.** $\mathbb{P}_\infty$ *is coherent, i.e., it gives rise to an internally consistent probability measure* Pr *on the set* $\mathcal{PC}(\Gamma)$ *of all worlds consistent with* Γ, *defined by the formula*

$$\mathrm{Pr}(\mathbb{W}(\phi) = 1) := \mathbb{P}_\infty(\phi).$$

In particular, if Γ *contains the axioms of first-order logic, then* $\mathbb{P}_\infty$ *defines a probability measure on the set of first-order completions of* Γ.

Proof of Theorem 4.1.2. By Theorem 4.1.1 (Convergence), the limit $\mathbb{P}_\infty(\phi)$ exists for all sentences $\phi \in \mathcal{S}$. Therefore, $\mathrm{Pr}(\mathbb{W}(\phi) = 1) := \mathbb{P}_\infty(\phi)$ is well-defined as a function of basic subsets of the set of all consistent worlds $\mathcal{PC}(D_\infty) = \mathcal{PC}(\Gamma)$.

Gaifman (1964) shows that Pr extends to a probability measure over $\mathcal{PC}(\Gamma)$ so long as the following three implications hold for all sentences ϕ and ψ:

- If $\Gamma \vdash \phi$, then $\mathbb{P}_\infty(\phi) = 1$.

- If $\Gamma \vdash \neg\phi$, then $\mathbb{P}_\infty(\phi) = 0$.

- If $\Gamma \vdash \neg(\phi \wedge \psi)$, then $\mathbb{P}_\infty(\phi \vee \psi) = \mathbb{P}_\infty(\phi) + \mathbb{P}_\infty(\psi)$.

Since the three conditions are quite similar in form, we will prove them simultaneously using four exemplar traders and parallel arguments.

Definition of the traders. Suppose that one of the three conditions is violated by a margin of ε, i.e., one of the following four cases holds:

(L^1) $\Gamma \vdash \phi$, but $\qquad$ (I^1) $\mathbb{P}_\infty(\phi) < 1 - \varepsilon$;

(L^2) $\Gamma \vdash \neg\phi$, but $\qquad$ (I^2) $\mathbb{P}_\infty(\phi) > \varepsilon$;

(L^3) $\Gamma \vdash \neg(\phi \wedge \psi)$, but $\quad$ (I^3) $\mathbb{P}_\infty(\phi \vee \psi) < \mathbb{P}_\infty(\phi) + \mathbb{P}_\infty(\psi) - \varepsilon$; or

(L^4) $\Gamma \vdash \neg(\phi \wedge \psi)$, but $\quad$ (I^4) $\mathbb{P}_\infty(\phi \vee \psi) > \mathbb{P}_\infty(\phi) + \mathbb{P}_\infty(\psi) + \varepsilon$.

Let $i \in \{1,2,3,4\}$ be the case that holds. Since the limit $\mathbb{P}_\infty$ exists, there is some sufficiently large time s_ε such that for all $n > s_\varepsilon$, the inequality I^i holds with n in place of ∞. Furthermore, since $\overline{D}$ is a Γ-complete deductive process, for some sufficiently large s_Γ and all $n > s_\Gamma$, the logical condition L^i holds with D_n in place of Γ. Thus, letting $s := \max(s_\varepsilon, s_\Gamma)$, for $n > s$ one of the following cases holds:

(L_n^1) $D_n \vdash \phi$, but $\qquad$ (I_1^n) $\mathbb{P}_n(\phi) < 1 - \varepsilon$;

(L_n^2) $D_n \vdash \neg\phi$, but $\qquad$ (I_2^n) $\mathbb{P}_n(\phi) > \varepsilon$;

(L_n^3) $D_n \vdash \neg(\phi \wedge \psi)$, but $\quad$ (I_3^n) $\mathbb{P}_n(\phi \vee \psi) < \mathbb{P}_n(\phi) + \mathbb{P}_n(\psi) - \varepsilon$; or

(L_n^4) $D_n \vdash \neg(\phi \wedge \psi)$, but $\quad$ (I_4^n) $\mathbb{P}_n(\phi \vee \psi) > \mathbb{P}_n(\phi) + \mathbb{P}_n(\psi) + \varepsilon$.

(When interpreting these, be sure to remember that each D_n is finite, and $D \vdash$ indicates using provability using only propositional calculus, i.e., modus ponens. In particular, the axioms of first order logic are not assumed to be in D_n.)

We now define, for each of the above four cases, a trader that will exploit the market $\overline{\mathbb{P}}$. For $n > s$, let

$$T_n^1 := \phi - \phi^{*n}$$
$$T_n^2 := -(\phi - \phi^{*n})$$
$$T_n^3 := ((\phi \vee \psi) - (\phi \vee \psi)^{*n}) - (\phi - \phi^{*n}) - (\psi - \psi^{*n})$$
$$T_n^4 := (\phi - \phi^{*n}) + (\psi - \psi^{*n}) - ((\phi \vee \psi) - (\phi \vee \psi)^{*n})$$

and for $n \leq s$ let $T_n^i = 0$. Each T_n^i can be written down in $\mathcal{O}(\log(n))$ time (the constant s can be hard-coded at a fixed cost), so these $\overline{T}^i$ are all e.c. traders.

Proof of exploitation. We leave the constant argument $\overline{\mathbb{P}}$ implicit to reduce clutter, writing, e.g., ϕ^{*i} for $\phi^{*i}(\overline{\mathbb{P}}) = \mathbb{P}_i(\phi)$, $T_n[\phi]$ for $T_n[\phi](\overline{\mathbb{P}})$, and so on.

Consider case 1, where L_n^1 and I_n^1 hold for $n > s$, and look at the trader $\overline{T}^1$. For any $n > s$ and any world $\mathbb{W} \in \mathcal{PC}(D_n)$, by linearity of $\mathbb{W}$ we have

$$\mathbb{W}\left(\sum_{i \leq n} T_i^1\right) = \sum_{i \leq n} T_i^1[\phi] \cdot \left(\mathbb{W}(\phi) - \phi^{*i}\right)$$

but $T_i^1[\phi] \equiv 1$ iff $i > s$, so this sum is

$$= \sum_{s < i \leq n} 1 \cdot \left(\mathbb{W}(\phi) - \phi^{*i}\right).$$

Now, by our choice of s, $\mathbb{W}(\phi) = 1$, and $i > s$ implies $\phi^{*i} < 1 - \varepsilon$, so this is

$$\geq \sum_{s < i \leq n} (1 - (1 - \varepsilon))$$
$$= \varepsilon \cdot (n - s)$$
$$\to \infty \text{ as } n \to \infty.$$

In particular, $\overline{T}^1$ exploits $\overline{\mathbb{P}}$, i.e., the set of values

$$\left\{ \mathbb{W}\left(\sum_{i \leq n} T_i\right)(\overline{\mathbb{P}}) \;\middle|\; n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n) \right\}$$

is bounded below but not bounded above. The analysis for case 2 is identical: if L_n^2 and I_n^2 hold for $n > s$, then $\overline{T}^2$ exploits $\overline{\mathbb{P}}$.

Now consider case 3, where L_n^3 and I_n^3 hold for $n > s$. Then for any time step $n > s$ and any world $\mathbb{W} \in \mathcal{PC}(D_n)$,

$$\mathbb{W}\left(\sum_{i \leq n} T_i^3\right) = \sum_{i \leq n} \left((\mathbb{W}(\neg(\phi \wedge \psi)) - (\phi \vee \psi)^{*i}) - (\mathbb{W}(\phi) - \phi^{*i}) - (\mathbb{W}(\psi) - \psi^{*i})\right)$$

$$= \sum_{s < i \leq n} (\mathbb{W}(\phi \vee \psi) - \mathbb{W}(\phi) - \mathbb{W}(\psi)) - \left((\phi \vee \psi)^{*i} - \phi^{*i} - \psi^{*i}\right)$$

but by our choice of s, $\mathbb{W}(\phi \vee \psi) - \mathbb{W}(\phi) - \mathbb{W}(\psi) = 0$, and $i > s$ implies the inequality $(\phi \vee \psi)^{*i} - \phi^{*i} - \psi^{*i} < -\varepsilon$, so the above sum is

$$\geq \sum_{s < i \leq n} \varepsilon$$

$$= \varepsilon \cdot (n - s) \to \infty \text{ as } n \to \infty.$$

So $\overline{T}^3$ exploits $\overline{\mathbb{P}}$, contradicting the logical induction criterion. The analysis for case 4 is identical. Hence, all four implications must hold for $\overline{\mathbb{P}}$ to satisfy the logical induction criterion. $\square$

6.3 Non-dogmatism

Recall Theorem 4.6.2:

Theorem 4.6.2 (Non-Dogmatism)**.** *If $\Gamma \nvdash \phi$ then $\mathbb{P}_\infty(\phi) < 1$, and if $\Gamma \nvdash \neg\phi$ then $\mathbb{P}_\infty(\phi) > 0$.*

Proof of Theorem 4.6.2. We prove the second implication, since the first implication is similar, with selling in place of buying. Suppose for a contradiction that $\Gamma \nvdash \neg\phi$ but that $\mathbb{P}_\infty(\phi) = 0$.

Definition of the trader $\overline{T}$. We define $\overline{T}$ recursively, along with helper functions $\overline{\beta}^k$ that will ensure that for every k, our trader will by one share of ϕ for a price of at most 2^{-k}:

for $k = 1, \ldots, n$:
$$\beta_k^k := 0$$
for $i = k + 1, \ldots, n$:
$$\beta_i^k := \text{Ind}_{2^{-k-1}}(\phi^{*i} < 2^{-k}) \cdot \left(1 - \sum_{j=k}^{i-1} \beta_j^k\right)$$
$$T_i[\phi] := \sum_{j \leq i} \beta_j^k$$
$$T_i := T_i[\phi] \cdot (\phi - \phi^{*i})$$

Note that all the equations defining T_n can be written down (from scratch) in $\mathcal{O}(n^3 \log(n))$ time, so $\overline{T}$ is an e.c. trader.

Proof of exploitation. We leave the constant argument $\overline{\mathbb{P}}$ implicit to reduce clutter, writing, e.g., ϕ^{*i} for $\phi^{*i}(\overline{\mathbb{P}}) = \mathbb{P}_i(\phi)$, $T_n[\phi]$ for $T_n[\phi](\overline{\mathbb{P}})$, and so on.

Observe from the recursion above for $\overline{T}$ that for all $i > 0$ and $k > 0$,

$$0 \leq \sum_{j=k}^{i} \beta_j^k \leq 1$$

62

and for any i and any $k \leq i$,
$$\beta_i^k \geq 0.$$

Next, observe that for any $k > 0$, for $i \geq$ some threshold $f(k)$, we will have $\phi^{*i} < 2^{-k-1}$, in which case the indicator in the definition of β_i^k will equal 1, at which point $\sum_{j=k}^{i} \beta_j^k = 1$. Thus, for all $n \geq f(k)$,

$$\sum_{i=k}^{n} \beta_i^k = 1.$$

Letting $H_n = \sum_{i \leq n} T_i$, the following shows that our trader will eventually own an arbitrarily large number of ϕ-shares:

$$
\begin{aligned}
H_n[\phi] &= \sum_{i \leq n} \sum_{k \leq i} \beta_i^k \\
&= \sum_{k \leq n} \sum_{k \leq i \leq n} \beta_i^k \\
&\geq \sum_{\substack{k \leq n \\ f(k) \leq n}} \sum_{k \leq i \leq n} \beta_i^k \\
&= \sum_{\substack{k \leq n \\ f(k) \leq n}} 1 \quad \to \infty \quad \text{as} \quad n \to \infty
\end{aligned}
\tag{6.3.1}
$$

Next we show that our trader never spends more than a total of \$1.

$$H_n[1] = -\sum_{i \leq n} \sum_{k \leq i} \beta_i^k \cdot \phi^{*i},$$

but the indicator function defining β_i^k ensures that $\phi^{*i} \leq 2^{-k}$ whenever β_i^k is non-zero, so this is

$$
\begin{aligned}
&\geq -\sum_{i \leq n} \sum_{k \leq i} \beta_i^k \cdot 2^{-k} \\
&= -\sum_{k \leq n} 2^{-k} \cdot \sum_{k \leq i \leq n} \beta_i^k \\
&\geq -\sum_{k \leq n} 2^{-k} \cdot 1
\end{aligned}
$$

Now, for any world $\mathbb{W}$, since $H_n[\phi] \geq 0$ for all n and $\mathbb{W}(\phi) \geq 0$, we have

$$
\begin{aligned}
\mathbb{W}(H_n) &= H_n[1] + H_n[\phi]\mathbb{W}(\phi) \\
&\geq -1 + 0 \cdot 0 \geq -1
\end{aligned}
$$

so the values $\mathbb{W}(H_n)$ are bounded below as n varies. Moreover, since $\Gamma \nvdash \neg\phi$, for every n there is always some $\mathbb{W} \in \mathcal{PC}(D_n)$ where $\mathbb{W}(\phi) = 1$ (since any consistent truth assignment can be extended to a truth assignment on all sentences), in which case

$$\mathbb{W}(H_n) \geq -1 + H_n[\phi] \cdot 1$$

But by equation 6.3.1, this $\lim_{n \to \infty} H_n[\phi] = \infty$, so $\lim_{n \to \infty} \mathbb{W}(H_n) = \infty$ as well. Hence, our e.c. trader exploits the market, contradicting the logical induction criterion. Therefore, if $\mathbb{P}_\infty(\phi) = 0$, we must have $\Gamma \vdash \neg\phi$. $\qquad \square$

6.4 Provability Induction

Recall:

Theorem 4.2.1 (Provability Induction)**.** *Let $\overline{\phi}$ be an e.c. sequence of theorems. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n 1.$$

Furthermore, let $\overline{\psi}$ be an e.c. sequence of disprovable sentences. Then

$$\mathbb{P}_n(\psi_n) \eqsim_n 0.$$

Proof of Theorem 4.2.1. This result is actually special case of Theorem 4.4.2, which will be proven in the next section. For, suppose $\overline{\phi}$ is an e.c. sequence of sentences with $\Gamma \vdash \phi_n$ for all n. Notice that for every i, the indicator $\mathrm{Thm}_\Gamma(\phi_i)$, specifying whether ϕ_i is a theorem of Γ, evaluates to 1. Therefore we immediately have that for any divergent weighting w at all,

$$\lim_{n\to\infty} \frac{\sum_{i<n} w(i) \cdot \mathrm{Thm}_\Gamma \phi_i}{\sum_{i<n} w(i)} = 1.$$

That is, the sequence $\overline{\phi}$ is pseudorandom (over any class of weightings) with frequency 1. Hence, by Learning Pseudorandom Frequencies (Theorem 4.4.2),

$$\mathbb{P}_n(\phi_n) \eqsim_n 1,$$

as desired. The proof that $\mathbb{P}_n(\psi_n) \eqsim_n 0$ proceeds analogously. $\square$

Examining the proof of Theorem 4.4.2 in the special case of provability induction yields some intuition. In this case, the trader defined in that proof essentially buys ϕ_n-shares every round that $\mathbb{P}_n(\phi_n) < 1 - \varepsilon$. To avoid overspending, it tracks which ϕ_n have been proven so far, and never has more than 1 total share outstanding. Since eventually each ϕ_n is guaranteed to be valued at 1 in every plausible world, the value of the trader is increased by at least ε (times the number of ϕ_n-shares it purchased) infinitely often. In this way, the trader makes profits for so long as $\overline{\mathbb{P}}$ fails to recognize the pattern $\overline{\phi}$ of provable sentences.

6.5 Learning Pseudorandom Frequencies

Recall Theorem 4.4.2:

Theorem 4.4.2 (Learning Pseudorandom Frequencies)**.** *Let $\overline{\phi}$ be an e.c. sequence of decidable sentences. If $\overline{\phi}$ is pseudorandom with frequency p over the set of all $\mathbb{P}$-generable divergent weightings, then*

$$\mathbb{P}_n(\phi_n) \eqsim_n p.$$

Before beginning the proof, the following intuition may be helpful. If the theorem does not hold, assume without loss of generality that $\overline{\mathbb{P}}$ repeatedly underprices the ϕ_n. Then a trader can buy ϕ_n-shares whenever their price goes below $p - \varepsilon$. By the assumption that the truth values of the ϕ_n are pseudorandom, roughly p proportion of the shares will pay out. Since the trader only pays at most $p - \varepsilon$ per share, on average they make ε on each trade, so over time they exploit the market. All we need to do is make the trades continuous, and ensure that the trader does not go below a fixed budget (as in the proof of Theorem 4.1.1).

Proof of Theorem 4.4.2. Suppose for a contradiction that $\overline{\phi}$ is an e.c. sequence of Γ-decidable sentences such that for every $\overline{\mathbb{P}}$-generable divergent weighting w,

$$\lim_{n\to\infty} \frac{\sum_{i<n} w(i) \cdot \mathrm{Thm}_\Gamma \phi_i}{\sum_{i<n} w(i)} = p,$$

but nevertheless, for some $\varepsilon > 0$ and infinitely many n, $|\mathbb{P}_n(\phi_n) - p| > \varepsilon$. Without loss of generality, assume that for infinite many n,

$$\mathbb{P}_n(\phi_n) < p - \varepsilon.$$

(The argument for the case where $\mathbb{P}_n(\phi_n) > p + \varepsilon$ infinitely often will be the same, and one of these two cases must obtain.)

Definition of the trader $\overline{T}$. We define $\mathrm{Open} : (\mathcal{S} \times \mathbb{N}) \to \mathbb{B}$ to be the following (potentially very slow) computable function:

$$\mathrm{Open}(\phi, n) = \begin{cases} 0 & \text{if } D_n \vdash \phi \text{ or } D_n \vdash \neg\phi; \\ 1 & \text{otherwise.} \end{cases}$$

Open is computable because (remembering that $\vdash$ stands for propositional provability) we can just search through all truth assignments to the prime sentences of all sentences in D_n that make the sentences in D_n true, and see if they all yield the same truth value to ϕ. We now define a much faster function $\mathrm{MO} : (\mathbb{N} \times \mathbb{N}) \to \mathbb{B}$ (mnemonic: "maybe open") by

$$\mathrm{MO}(\phi, n) = \begin{cases} 0 & \begin{array}{l} \text{if for some } m \leq n, \mathrm{Open}(\phi, m) \\ \text{returns } 0 \text{ in } \leq n \text{ steps} \end{array} \\ 1 & \text{otherwise.} \end{cases}$$

Observe that $\mathrm{MO}(\phi, n)$ runs in $\mathcal{O}(n^2)$ time, and that for any decidable ϕ,

- $\mathrm{MO}(\phi, n) = 0$ for some sufficiently large n;
- if $\mathrm{MO}(\phi, n) = 0$ then $\mathrm{Open}(\phi, n) = 0$;
- if $\mathrm{MO}(\phi, m) = 0$ and $n > m$ then $\mathrm{MO}(\phi, n) = 0$.

(Note that MO may assign a value of 1 when Open does not, hence the mnemonic "maybe open".)

We will now use MO to define a trader $\overline{T}$ recursively, along with a helper function β to ensure that it never holds a total of more than 1 unit of open (fractional) shares. We let $\beta_1 = 0$ and for $n \geq 1$,

$$\beta_n := 1 - \sum_{i < n} \mathrm{MO}(\phi_i, n) T_i[\phi_i];$$
$$T_n[\phi_n] := \beta_n \cdot \mathrm{Ind}_{\varepsilon/2}(\phi_n^{*n} < p - \varepsilon/2);$$
$$T_n := T_n[\phi] \cdot (\phi_n - \phi_n^{*n}).$$

Observe that the expressible feature T_n can be computed (from scratch) in $\mathrm{poly}(n)$ time using MO, so $\overline{T}$ is an e.c. trader. Notice also that β_n and all the $T_n(\phi)$ are always in $[0, 1]$.

A divergent weighting. For the rest of the proof, we leave the constant argument $\overline{\mathbb{P}}$ implicit to reduce clutter, writing, e.g., ϕ_i^{*i} for $\phi_i^{*i}(\overline{\mathbb{P}}) = \mathbb{P}_i(\phi)$, $T_n[\phi]$ for $T_n[\phi](\overline{\mathbb{P}})$, and so on.

We will show that the sequence of trade coefficients $w_n = T_n[\phi_n]$ made by $\overline{T}$ against the market $\overline{\mathbb{P}}$ form a $\overline{\mathbb{P}}$-generable divergent weighting. Our trader $\overline{T}$ is efficiently computable and $T_n[\phi_n] \in [0, 1]$ for all n, so it remains to show that, on input $\mathbb{P}_{\leq n}$,

$$\sum_{n \in \mathbb{N}^+} T_n[\phi_n] = \infty.$$

Suppose this were not the case, so that for some sufficiently large m,

$$\sum_{m < j} T_j[\phi_j] < 1/2. \tag{6.5.1}$$

65

By the definition of MO, there exists some large m' such that for all $i < m$, $\text{MO}(\phi_i, m') = 0$. At that point, for any $n > m'$, we have

$$\begin{aligned}
\beta_n := 1 - \sum_{i<n} T_i[\phi_i] \cdot \text{MO}(\phi_i, n) \\
= 1 - \sum_{m<i<n} T_i[\phi_i] \cdot \text{MO}(\phi_i, n) \\
\geq 1 - \sum_{m<i} T_i[\phi_i]
\end{aligned}$$

which, by equation (6.5.1), means that

$$\beta_n \geq 1/2.$$

Then, by the earlier supposition on $\overline{\mathbb{P}}$, for some $n > m'$ we have $\mathbb{P}_n(\phi_n) < p - \varepsilon$, at which point

$$T_n[\phi_n] = \beta_n \cdot \text{Ind}_{\varepsilon/2}(\phi_n^{*n} < p - \varepsilon/2) \geq \beta_n \cdot 1 \geq 1/2$$

which contradicts the $1/2$ bound in equation (6.5.1). Hence, the sum $\sum_i T_i[\phi_n]$ must instead be bounded. This means $(T_n[\phi_n])_{n \in \mathbb{N}^+}$ is a $\overline{\mathbb{P}}$-generable divergent weighting.

Proof of exploitation. Now, by definition of $\overline{\phi}$ being pseudorandom with frequency p over the class of $\overline{\mathbb{P}}$-generable divergent weightings, we have that

$$\lim_{n \to \infty} \frac{\sum_{i \leq n} T_i[\phi_i] \cdot \text{Thm}_\Gamma(\phi_i)}{\sum_{i \leq n} T_i[\phi_i]} = p.$$

Thus, for all sufficiently large n,

$$\sum_{i \leq n} T_i[\phi_i] \cdot \text{Thm}_\Gamma(\phi_i) \geq (p - \varepsilon/4) \cdot \sum_{i \leq n} T_i[\phi_i].$$

Now, since our construction makes $\beta_n \in [0, 1]$ for all n, we have

$$\sum_{i \leq n} T_i[\phi_i] \cdot \text{MO}(\phi_i) \leq 1.$$

Also,

$$\mathbb{W}(\phi_i) \geq \text{Thm}_\Gamma(\phi_i) - \text{MO}(\phi_i).$$

Multiplying this by $T_i[\phi_i]$ and summing over i gives

$$\begin{aligned}
\sum_{i \leq n} T_i[\phi_i] \cdot \mathbb{W}(\phi_i) \geq \left(\sum_{i \leq n} T_i[\phi_i] \cdot \text{Thm}_\Gamma(\phi_i) \right) - \left(\sum_{i \leq n} T_i[\phi_i] \cdot \text{MO}(\phi_i) \right) \\
\geq \left(\sum_{i \leq n} T_i[\phi_i] \cdot \text{Thm}_\Gamma(\phi_i) \right) - 1 \\
\geq -1 + (p - \varepsilon/4) \sum_{i \leq n} T_i[\phi_i].
\end{aligned}$$

By the definition of $\overline{T}$, and since $\phi_i^{*i} \leq (p - \varepsilon/2)$ whenever $T_i[\phi_i] \neq 0$,

$$-\sum_{i \leq n} T_i[\phi_i] \cdot \phi_i^{*i} \geq -(p - \varepsilon/2) \sum_{i \leq n} T_i[\phi_i].$$

66

Adding the above two inequalities gives

$$\mathbb{W}\left(\sum_{i\le n} T_i\right) \ge -1 + (\varepsilon/4)\sum_{i\le n} T_i[\phi_i]$$
$$\to \infty \text{ as } n \to \infty$$

because $T_i[\phi_i]$ is a divergent weighting (as shown above). Hence, $\overline{T}$ exploits the market $\overline{\mathbb{P}}$, contradicting the logical induction criterion. Therefore, for $\overline{\mathbb{P}}$ to satisfy the logical induction criterion, we must have

$$\lim_{n\to\infty} \mathbb{P}_n(\phi_n) = p.$$

$\square$

6.6 Provability Induction

Recall Theorem 4.2.1:

Theorem 4.2.1 (Provability Induction)**.** *Let $\overline{\phi}$ be an e.c. sequence of theorems. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n 1.$$

Furthermore, let $\overline{\psi}$ be an e.c. sequence of disprovable sentences. Then

$$\mathbb{P}_n(\psi_n) \eqsim_n 0.$$

Proof of Theorem 4.2.1. Suppose $\overline{\phi}$ is an e.c. sequence of sentences with $\Gamma \vdash \phi_n$ for all n. Notice that for every i, the indicator $\mathrm{Thm}_\Gamma(\phi_i)$ evaluates to 1. Therefore we immediately have that for any divergent weighting w at all,

$$\lim_{n\to\infty} \frac{\sum_{i<n} w(i) \cdot \mathrm{Thm}_\Gamma \phi_i}{\sum_{i<n} w(i)} = 1.$$

That is, the sequence $\overline{\phi}$ is pseudorandom (over any class of weightings) with frequency 1. Hence, by Learning Pseudorandom Frequencies (Theorem 4.4.2),

$$\mathbb{P}_n(\phi_n) \eqsim_n 1,$$

as desired. The proof that $\mathbb{P}_n(\psi_n) \eqsim_n 0$ proceeds analogously. $\square$

Examining the proof of Theorem 4.4.2 (Learning Pseudorandom Frequencies) in the special case of provability induction yields some intuition. In this case, the trader defined in that proof essentially buys ϕ_n-shares every round that $\mathbb{P}_n(\phi_n) < 1 - \varepsilon$. To avoid overspending, it tracks which ϕ_n have been proven so far, and never has more than 1 total share outstanding. Since eventually each ϕ_n is guaranteed to be valued at 1 in every plausible world, the value of the trader is increased by at least ε (times the number of ϕ_n-shares it purchased) infinitely often. In this way, the trader makes profits for so long as $\overline{\mathbb{P}}$ fails to recognize the pattern $\overline{\phi}$ of provable sentences.

7 Discussion

We have proposed the *logical induction criterion* as a criterion on the beliefs of deductively limited reasoners, and we have shown that reasoners who satisfy this criterion (*logical inductors*) possess many desirable properties when it comes to developing beliefs about logical statements (including statements about mathematical facts, long-running computations, and the reasoner themself). We have also given a computable algorithm `LIA` for constructing a logical inductor. We will now discuss applications of logical induction (Section 7.1) and speculate about how and why we think this framework works (Section 7.2). We then discuss a few variations on our framework (Section 7.3) before concluding with a discussion of a few open questions (Section 7.4).

7.1 Applications

Logical inductors are not intended for practical use. The algorithm to compare with logical induction is not Belief Propagation (an efficient method for approximate inference in Bayesian networks [Pearl 1988]) but Solomonoff's theory of inductive inference (an uncomputable method for making ideal predictions about empirical facts [Solomonoff 1964a]). Just as Solomnoff's sequence predictor assigns probabilities to all possible observations and learns to predict any computable environment, logical inductors assign probabilities to all possible sentences of logic and learns to recognize any efficiently computable pattern between logical claims.

Solomonoff's theory involves a predictor that considers all computable hypotheses about their observations, weighted by simplicity, and uses Bayesian inference to zero in on the best computable hypothesis. This (uncomputable) algorithm is impractical, but has nevertheless been of theoretical use: its basic idiom—consult a series of experts, reward accurate predictions, and penalize complexity—is commonplace in statistics, predictive analytics, and machine learning. These "ensemble methods" often perform quite well in practice. Refer to Opitz and Maclin (1999) and Dietterich (2000) for reviews of popular and successful ensemble methods.

One of the key applications of logical induction, we believe, is the development of an analogous idiom for scenarios where reasoners are uncertain about logical facts. Logical inductors use a framework similar to standard ensemble methods, with a few crucial differences that help them manipulate logical uncertainty. The experts consulted by logical inductors don't make predictions about what is going to happen next; instead, they observe the aggregated advice of all the experts (including themselves) and attempt to exploit inefficiencies in that aggregate model. A trader doesn't need to have an opinion about whether or not ϕ is true; they can exploit the fact that ϕ and $\neg\neg\phi$ have different probabilities without having any idea what ϕ says or what that's supposed to mean. This idea and others yield an idiom for building models that integrate logical patterns and obey logical constraints.

In a different vein, we expect that logical inductors can already serve as a drop-in replacement for formal models of reasoners that assume logical omniscience and/or perfect Bayesianism, such as in game theory, economics, or theoretical models of artificial reasoners.

The authors are particularly interested in tools that help AI scientists attain novel statistical guarantees in settings where robustness and reliability guarantees are currently difficult to come by. For example, consider the task of designing an AI system that reasons about the behavior of computer programs, or that reasons about its own beliefs and its own effects on the world. While practical algorithms for achieving these feats are sure to make use of heuristics and approximations, we believe scientists will have an easier time designing robust and reliable systems if they have some way to relate those approximations to theoretical algorithms that are known to behave well in principle (in the same way that Auto-Encoding Variational Bayes can be related to Bayesian inference [Kingma and Welling 2013]). Modern models of rational behavior are not up to this task: formal logic is inadequate when it comes to modeling self-reference, and probability theory is inadequate when it comes to modeling logical uncertainty. We see logical induction as a first step towards models of rational behavior that work in settings where agents must reason about themselves, while deductively limited.

When it comes to the field of meta-mathematics, we expect logical inductors to open new avenues of research on questions about what sorts of reasoning systems can achieve which forms of self-trust. The specific type of self-trust that logical inductors achieve (via, e.g., Theorem 4.12.4) is a subtle subject, and worthy of a full paper in its own right. As such, we will not go into depth here.

7.2 Analysis

Mathematicians, scientists, and philosophers have taken many different approaches towards the problem of unifying logic with probability theory. (For a sample, refer to Section 1.2.) In this subsection, we will speculate about what makes the logical

induction framework tick, and why it is that logical inductors achieve a variety of desiderata. The authors currently believe that the following three points are some of the interesting takeaways from the logical induction framework:

Following Solomonoff and Gaifman. One key idea behind our framework is our paradigm of making predictions by combining advice from an ensemble of experts in order to assign probabilities to all possible logical claims. This merges the framework of Solomonoff (1964a) with that of Gaifman (1964), and it is perhaps remarkable that this can be made to work. Say we fix an enumeration of all prime sentences of first-order logic, and then hook `LIA` (Algorithm 5.4.1) up to a theorem prover that enumerates theorems of `PA` (written using that enumeration). Then all `LIA` ever "sees" (from the deductive process) is a sequence of sets like

$$\{\#92305 \text{ or } \#19666 \text{ is true}; \#50105 \text{ and } \#68386 \text{ are true}; \#8517 \text{ is false}\}.$$

From this and this alone, `LIA` develops accurate beliefs about all possible arithmetical claims. `LIA` does this in a manner that outpaces the underlying deductive process and satisfies the desiderata listed above. If instead we hook `LIA` up to a `ZFC`-prover, it develops accurate beliefs about all possible set-theoretic claims. This is very reminiscent of Solomonoff's framework, where all the predictor sees is a sequence of 1s and 0s, and they start figuring out precisely which environment they're interacting with.

This is only one of many possible approaches to the problem of logical uncertainty. For example, Adams' probability logic (1996) works in the other direction, using logical axioms to put constraints on an unknown probability distribution and then using deduction to infer properties of that distribution. Markov logic networks (Richardson and Domingos 2006) construct a belief network that contains a variable for every possible way of grounding out each logical formula, which makes them quite ill-suited to the problem of reasoning about the behavior of complex Turing machines.[8] In fact, there is no consensus about what form an algorithm for "good reasoning" under logical uncertainty should take. Empiricists such as Hintikka (1962) and Fagin et al. (1995) speak of a set of modal operators that help differentiate between different types of knowledge; AI scientists such as Russell and Wefald (1991b), Hay et al. (2012), and Lin et al. (2015) speak of algorithms that are reasoning about complicated facts while also making decisions about what to reason about next; mathematicians such as (Briol, Oates, Girolami, Osborne, and Sejdinovic 2015; Briol, Oates, Girolami, and Osborne 2015; Hennig, Osborne, and Girolami 2015) speak of numerical algorithms that give probabilistic answers to particular questions where precise answers are difficult to generate.

Our approach achieves some success by building an approximately-coherent distribution over all logical claims. Of course, logical induction does not solve all the problems of reasoning under deductive limitation—far from it! They do not engage in meta-cognition (in the sense of Russell and Wefald [1991b]) to decide which facts to reason about next, and they do not give an immediate practical tool (as in the case of probabilistic integration [Briol, Oates, Girolami, Osborne, and Sejdinovic 2015]), and they have abysmal runtime and uncomputable convergence bounds. It is our hope that the methods logical inductors use to aggregate expert advice will eventually yield algorithms that are useful for various applications, in the same way that useful ensemble methods can be derived from Solomonoff's theory of inductive inference.

Keep the experts small. One of the key differences between our framework and Solomonoff-inspired ensemble methods is that our "experts" are not themselves predicting the world. In standard ensemble methods, the prediction algorithm weighs

---

8. Reasoning about the behavior of a Turing machine using a Markov logic network would require having one node in the graph for every intermediate state of the Turing machine for every input, so doing inference using that graph is not much easier than simply running the Turing machine. Thus, Markov logic networks are ill-suited for answering questions about how a reasoner should predict the behavior of computations that they cannot run.

advice from a number of experts, where the experts themselves are also making predictions. The "master algorithm" rewards the experts for accuracy and penalizes them for complexity, and uses a weighted mixture of the experts to make their own prediction. In our framework, the master algorithm is still making predictions (about logical facts), but the experts themselves are not necessarily predictors. Instead, the experts are "traders", who get to see the current model (constructed by aggregating information from a broad class of traders) and attempt to exploit inefficiencies in that aggregate model. This allows traders to identify (and eliminate) inconsistencies in the model even if they don't know what's actually happening in the world. For example, if a trader sees that $\mathbb{P}(\phi) + \mathbb{P}(\neg\phi) \ll 1$, they can buy shares of both ϕ and $\neg\phi$ and make a profit, even if they have no idea whether ϕ is true or what ϕ is about. In other words, letting the experts buy and sell shares (instead of just making predictions), and letting them see the aggregate model, allows them to contribute knowledge to the model, even if they have no idea what's going on in the real world.

We can imagine each trader as contributing a small piece of logical knowledge to a model—each trader gets to say "look, I don't know what you're trying to predict over there, but I do know that this piece of your model is inconsistent". By aggregating all these pieces of knowledge, our algorithm builds a model that can satisfy many different complicated relationships, even if every individual expert is only tracking a single simple pattern.

Make the trading functions continuous. As stated above, our framework gets significant mileage from showing each trader the aggregate model created by input from all traders, and letting them profit from identifying inconsistencies in that model. Showing traders the current market prices is not trivial, because the market prices on day n depend on which trades are made on day n, creating a circular dependency. Our framework breaks this cycle by requiring that the traders use continuous betting strategies, guaranteeing that stable beliefs can be found.

In fact, it's fairly easy to show that something like continuity is strictly necessary, if the market is to have accurate beliefs about itself. Consider again the paradoxical sentence $\chi := \text{``}\underline{\mathbb{P}_n}(\underline{\chi}) < 0.5\text{''}$ which is true iff its price in $\overline{\mathbb{P}}$ is less than 50¢ on day n. If, on day n, traders were allowed to buy when $\chi < 0.5$ and sell otherwise, then there is no equilibrium price. Continuity guarantees that the equilibrium price will always exist.

This guarantee protects logical inductors from the classic paradoxes of self-reference—as we have seen, it allows $\overline{\mathbb{P}}$ to develop accurate beliefs about its current beliefs, and to trust its future beliefs in most cases. We attribute the success of logical inductors in the face of paradox to the continuity conditions, and we suspect that it is a general-purpose method that deductively limited reasoners can use to avoid the classic paradoxes.

7.3 Variations

One notable feature of the logical induction framework is its generality. The framework is not tied to a polynomial-time notion of efficiency, nor to any specific model of computation. All the framework requires is a method of enumerating possible patterns of logic (the "traders") on the one hand, and a method of enumerating provable sentences of logic (the "deductive process") on the other. Our algorithm then gives a method for aggregating those patterns into a combined model that respects the logical patterns that actually hold.

The framework would work just as well if we used the set of linear-time traders in place of the set of poly-time traders. Of course, the market built out of linear-time traders would not satisfy all the same desirable properties—but the *method of induction*, which consists of aggregating knowledge from a collection of traders and letting them all see the combined model and attempt to exploit it, would remain unchanged.

There is also quite a bit of flexibility in the definition of a trader. Above, traders are defined to output continuous piecewise-rational functions of the market prices. We could restrict this definition (e.g., by having traders output continuous piecewise-

linear functions of the market prices), or broaden it (by replacing piecewise-rational with a larger class), or change the encoding scheme entirely. For instance, we could have the traders output not functions but upper-hemicontinuous relations specifying which trades they are willing to purchase; or we could give them oracle access to the market prices and have them output trades (instead of trading strategies). Alternatively, we could refrain from giving traders access to the market prices altogether, and instead let them sample truth values for sentences according to that sentence's probability, and then consider markets that are almost surely not exploited by any of these traders.

In fact, our framework is not even specific to the domain of logic. Strictly speaking, all that is necessary is a set of atomic events that can be "true" or "false", a language for talking about Boolean combinations of those atoms, and a deductive process that asserts things about those atoms (such as "$a \wedge \neg b$") over time. We have mainly explored the case where the atoms are prime sentences of first order logic, but the atoms could just as easily be bits in a webcam image, in which case the inductor would learn to predict patterns in the webcam feed. In fact, some atoms could be reserved for the webcam and others for prime sentences, yielding an inductor that does empirical and logical induction simultaneously.

For the sake of brevity, we leave the development of this idea to future works.

7.4   Open Questions

With Definition 3.0.1, we have presented a simple criterion on deductively limited reasoners, such that any reasoner who meets the criterion satisfies a large number of desiderata, and any reasoner that fails to meet the criterion can have their beliefs exploited by an efficient trader. With $\overline{\text{LIA}}$ we have shown that this criterion can be met in practice by computable reasoners.

The logical induction criterion bears a strong resemblance to the "no Dutch book" criteria used by Ramsey (1931), de Finetti (1937), Teller (1973), and Lewis (1999) to support Bayesian probability theory. This fact, and the fact that a wide variety of desirable properties follow directly from a single simple criterion, imply that logical induction captures a portion of what it means to do good reasoning under deductive limitations. That said, logical induction leaves a number of problems wide open. Here we discuss four, recalling desiderata from Section 1.1:

Desideratum 15 (Decision Rationality)**.** *The algorithm for assigning probabilities to logical claims should be able to target specific, decision-relevant claims, and it should reason about those claims as efficiently as possible given the computing resources available.*

In the case of logical inductors, we can interpret this desideratum as saying that it should be possible to tell a logical inductor to reason about one sentence in particular, and have it efficiently allocate resources towards that task. For example, we might be curious about Goldbach's conjecture, and wish to tell a logical inductor to develop its beliefs about that particular question, i.e. by devoting its computing resources in particular to sentences that relate to Goldbach's conjecture (such as sentences that might imply or falsify it).

Our algorithm for logical induction does not do anything of this sort, and there is no obvious mechanism for steering its deliberations. In the terminology of Hay et al. (2012), $\overline{\text{LIA}}$ does not do metalevel reasoning, i.e., it does nothing akin to "thinking about what to think about". That said, it is plausible that logical induction could play a role in models of bounded decision-making agents. For example, when designing an artificial intelligence (AI) algorithm that *does* try to reason about Goldbach's conjecture, it would be quite useful for that algorithm to have access to a logical inductor that tells it which other mathematical facts are likely related (and how). We can imagine a resource-constrained algorithm directing computing resources while consulting a partially-trained logical inductor, occasionally deciding that the best use of resources is to train the logical inductor further. At the moment, these ideas are purely speculative; significant work remains to be done to see how

logical induction bears on the problem of allocation of scarce computing resources when reasoning about mathematical facts.

Desideratum 16 (Answers Counterpossible Questions). *When asked questions about contradictory states of affairs, a good reasoner should give reasonable answers.*

In the year 1993, if you asked a mathematician about what we would know about mathematics if Fermat's last theorem was false, they would talk about how that would imply the existence of non-modular elliptic curves. In the year 1994, Fermat's last theorem was proven true, so by the principle of explosion, we now know that if Fermat's last theorem were false, then 1=2 and $\sqrt{2}$ is rational, because from a contradiction, anything follows. The first sort of answer seems more reasonable, and indeed, reasoning about counterpossibilities (i.e., proving a conjecture false by thinking about what would follow if it were true) is a practice that mathematicians engage in regularly. A satisfactory treatment of counterpossibilities has proven elusive; see (Cohen 1990; Vander Laan 2004; Brogaard and Salerno 2007; Krakauer 2012; Bjerring 2014) for some discussion and ideas. One might hope that a good treatment of logical uncertainty would naturally result in a good treatment of counterpossibilities.

There are intuitive reasons to expect that a logical inductor has reasonable beliefs about counterpossibilities. In the days before $\overline{D}$ has (propositionally) ruled out worlds inconsistent with Fermat's last theorem, $\overline{\mathbb{P}}$ has to have beliefs that allow for Fermat's last theorem to be false, and if the proof is a long time in coming, those beliefs are likely reasonable. However, we do not currently have any guarantees of this form—$\mathbb{P}_\infty$ still assigns probability 0 to Fermat's last theorem being false, and so the conditional probabilities are not guaranteed to be reasonable, so we haven't yet found anything satisfactory to say with confidence about $\overline{\mathbb{P}}$'s counterpossible beliefs.

While the discussion of counterpossibilities may seem mainly academic, Soares and Fallenstein (2015) have argued that counterpossibilities are central to the problem of designing robust decision-making algorithms. Imagine a deterministic agent `agent` evaluating three different "possible scenarios" corresponding to three different actions the agent could take. Intuitively, we want the nth scenario (modeled inside the agent) to represent what would happen if the agent took the nth action, and this requires reasoning about what would happen if `agent(observation)` had the output `a` vs `b` vs `c`. Thus, a better understanding of counterpossible reasoning could yield better decision algorithms. Significant work remains to be done to understand and improve the way that logical inductors answer counterpossible questions.

Desideratum 17 (Use of Old Evidence). *When a bounded reasoner comes up with a new theory that neatly describes anomalies in the old theory, that old evidence should count as evidence in favor of the new theory.*

The canonical example of the problem of old evidence is Einstein's development of the theory of general relativity and its retrodiction of the precession in Mercury's orbit. For hundreds of years before Einstein, astronomers knew that Newton's equations failed to model this precession, and Einstein's retrodiction counted as a large boost for his theory. This runs contrary to Bayes' theorem, which says that a reasoner should wring every drip of information out of every observation the moment that the evidence appears. A Bayesian reasoner keeps tabs on all possible hypotheses at all times, and so they never find a new hypothesis in a burst of insight, and reward it for retrodictions. Humans work differently—scientists spent centuries without having even one good theory for the precession of Mercury, and the difficult scientific labor of Einstein went into *inventing the theory.*

There is a weak sense in which logical inductors solve the problem of old evidence— as time goes on, they get better and better at recognizing patterns in the data that they have already seen, and integrating those old patterns into their new models. That said, a strong solution to the problem of old evidence isn't just about finding new ways to use old data every so often; it's about giving a satisfactory account of how to algorithmically *generate new scientific theories.* In that domain, logical induction has much less to say: they "invent" their "theories" by sheer brute force, iterating over all possible polynomial-time methods for detecting patterns in data.

There is some hope that logical inductors will shed light on the question of how to build accurate models of the world in practice, just as ensemble methods yield models that are better than any individual expert in practice. However, the task of using logical inductors to build practical models in some limited domain is wide open.

Desideratum 14 (Efficiency)**.** *The algorithm for assigning probabilities to logical claims should run efficiently, and be usable in practice.*

Logical inductors are far from efficient, but they do raise an interesting empirical question. While the theoretically ideal ensemble method (the universal semimeasure [Li and Vitányi 1993]) is uncomputable, practical ensemble methods often make very good predictions about their environments. It is therefore plausible that practical logical induction-inspired approaches could manage logical uncertainty well in practice. Imagine we pick some limited domain of reasoning, and a collection of constant- and linear-time traders. Imagine we use standard approximation methods (such as gradient descent) to find approximately-stable market prices that aggregate knowledge from those traders. Given sufficient insight and tweaking, would the resulting algorithm be good at learning to respect logical patterns in practice? This is an empirical question, and it remains to be tested.

7.5   Acknowledgements

References

Aaronson, Scott. 2013. "Why philosophers should care about computational complexity." In *Computability: Turing, Gödel, Church, and Beyond,* edited by B. Jack Copeland, Carl J. Posy, and Oron Shagrir. MIT Press.

Adams, Ernest W. 1996. *A Primer of Probability Logic.* University of Chicago Press.

Akama, Seiki, and Newton C. A. da Costa. 2016. "Why Paraconsistent Logics?" In *Towards Paraconsistent Engineering,* edited by Seiki Akama, 7–24. Springer.

Bacharach, Michael. 1994. "The epistemic structure of a theory of a game." *Theory and Decision* 37 (1): 7–48.

Battigalli, Pierpaolo, and Giacomo Bonanno. 1999. "Recent results on belief, knowledge and the epistemic foundations of game theory." *Research in Economics* 53 (2): 149–225.

Beygelzimer, Alina, John Langford, and David M. Pennock. 2012. "Learning Performance of Prediction Markets with Kelly Bettors." In *11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012),* edited by Vincent Conitzer, Michael Winikoff, Wiebe van der Hoek, and Lin Padgham, 1317–1318. International Foundation for Autonomous Agents / Multiagent Systems.

Binmore, Kenneth. 1992. "Foundations of game theory." In *Advances in Economic Theory: Sixth World Congress,* edited by Jean-Jacques Laffont, 1:1–31.

Bjerring, Jens Christian. 2014. "On Counterpossibles." *Philosophical Studies* 168 (2): 327–353.

Blair, Howard A., and V.S. Subrahmanian. 1989. "Paraconsistent Logic Programming." *Theoretical Computer Science* 68 (2): 135–154.

Boole, George. 1854. *An investigation of the laws of thought: on which are founded the mathematical theories of logic and probabilities.* Dover Publications.

Briol, François-Xavier, Chris Oates, Mark Girolami, and Michael A. Osborne. 2015. "Frank-Wolfe Bayesian Quadrature: Probabilistic Integration with Theoretical Guarantees." In *Advances in Neural Information Processing Systems 28 (NIPS 2015),* edited by C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, 1162–1170. Curran Associates, Inc.

Briol, François-Xavier, Chris Oates, Mark Girolami, Michael A. Osborne, and Dino Sejdinovic. 2015. "Probabilistic Integration." arXiv: `1512.00933 [stat.ML]`.

Brogaard, Berit, and Joe Salerno. 2007. "Why Counterpossibles are Non-Trivial." *The Reasoner* 1 (1): 5–6.

Calude, Cristian S, and Michael A Stay. 2008. "Most programs stop quickly or never halt." *Advances in Applied Mathematics* 40 (3): 295–308.

Campbell-Moore, Catrin. 2015. "How to Express Self-Referential Probability. A Kripkean Proposal." *The Review of Symbolic Logic* 8 (04): 680–704.

Carnap, Rudolf. 1962. *Logical Foundations of Probability.* University of Chicago Press.

Christiano, Paul. 2014. *Non-Omniscience, Probabilistic Inference, and Metamathematics.* Technical report 2014–3. Berkeley, CA: Machine Intelligence Research Institute. `http://intelligence.org/files/Non-Omniscience.pdf`.

Christiano, Paul F., Eliezer Yudkowsky, Marcello Herreshoff, and Mihály Bárász. 2013. *Definability of Truth in Probabilistic Logic.* Early Draft. Berkeley, CA: Machine Intelligence Research Institute, April 2. `https://intelligence.org/files/DefinabilityTruthDraft.pdf`.

Cohen, Daniel. 1990. "On What Cannot Be." In *Truth or Consequences: Essays in Honor of Nuel Belnap,* 123–132. Kluwer Academic Publishers.

de Finetti, Bruno. 1937. "Foresight: Its Logical Laws, Its Subjective Sources." In *Studies in Subjective Probability,* edited by Henry E. Kyburg and Howard E.K. Smokler. Huntington, New York: Roger E. Kreiger Publishing Co.

De Raedt, Luc. 2008. *Logical and Relational Learning.* Cognitive Technologies. Springer.

De Raedt, Luc, and Kristian Kersting. 2008. "Probabilistic Inductive Logic Programming." In *Probabilistic Inductive Logic Programming,* 1–27. Springer.

De Raedt, Luc, and Angelika Kimmig. 2015. "Probabilistic (logic) programming concepts." *Machine Learning* 100 (1): 5–47.

Demski, Abram. 2012. "Logical Prior Probability." Edited by Joscha Bach, Ben Goertzel, and Matthew Iklé. *Artificial General Intelligence. 5th International Conference, AGI 2012* (New York), no. 7716: 50–59.

Dietterich, Thomas G. 2000. "Ensemble Methods in Machine Learning." In *1st International workshop on Multiple Classifier Systems (MCS 2000),* 1–15. Springer.

Eells, Ellery. 1990. "Bayesian problems of old evidence." *Scientific Theories* 14:205–223.

Eklund, Matti. 2002. "Inconsistent Languages." *Philosophy and Phenomenological Research* 64 (2): 251–275.

Enderton, Herbert B. 2001. *A Mathematical Introduction to Logic.* Academic Press.

Fagin, Ronald, and Joseph Y. Halpern. 1987. "Belief, awareness, and limited reasoning." *Artificial intelligence* 34 (1): 39–76.

Fagin, Ronald, Joseph Y. Halpern, Yoram Moses, and Moshe Vardi. 1995. *Reasoning about knowledge, vol. 4.* MIT Press Cambridge.

Fuhrmann, André. 2013. "Relevant logics, modal logics and theory change." PhD diss., Australian National University - Research School of Social Sciences.

Gaifman, Haim. 1964. "Concerning Measures in First Order Calculi." *Israel Journal of Mathematics* 2 (1): 1–18.

Gaifman, Haim, and Marc Snir. 1982. "Probabilities over rich languages, testing and randomness." *Journal of Symbolic Logic* 47 (03): 495–548.

Garber, Daniel. 1983. "Old evidence and logical omniscience in Bayesian confirmation theory." *Testing scientific theories* 10:99–131.

Gärdenfors, Peter. 1988. *Knowledge in Flux: Modeling the Dynamics of Epistemic States.* Bradford Books, MIT Press.

Garrabrant, Scott, Tsvi Benson-Tilsen, Siddharth Bhaskar, Abram Demski, Joanna Garrabrant, George Koleszarik, and Evan Lloyd. 2016. "Asymptotic Logical Uncertainty and the Benford Test." In *9th Conference on Artificial General Intelligence (AGI-16),* edited by Bas Steunebrink, Pei Wang, and Ben Goertzel, 9782:202–211. Lecture Notes in Artificial Intelligence. Springer International Publishing.

Garrabrant, Scott, Benya Fallenstein, Abram Demski, and Nate Soares. 2016. "Inductive Coherence." arXiv: `1604.05288 [cs.AI]`.

Garrabrant, Scott, Nate Soares, and Jessica Taylor. 2016. "Asymptotic Convergence in Online Learning with Unbounded Delays." arXiv: `1604.05280`.

Gerla, Giangiacomo. 2013. *Fuzzy logic: mathematical tools for approximate reasoning.* Vol. 11. Trends in Logic. Springer.

Glanzberg, Michael. 2001. "The Liar in Context." *Philosophical Studies* 103 (3): 217–251.

Glymour, Clark. 1980. *Theory and Evidence.* Princeton University Press.

Gödel, Kurt, Stephen Cole Kleene, and John Barkley Rosser. 1934. *On Undecidable Propositions of Formal Mathematical Systems.* Princeton, NJ: Institute for Advanced Study.

Good, Irving J. 1950. *Probability and the Weighing of Evidence.* Charles Griffin, London.

Grim, Patrick. 1991. *The Incomplete Universe: Totality, Knowledge, and Truth.* Mit Press.

Guarino, Nicola. 1998. "Formal Ontology and Information Systems." In *Formal Ontology in Information Systems: Proceedings of FOIS'98,* 46:3–15. Frontiers in Artificial Intelligence and Applications. Amsterdam: IOS Press.

Gupta, Anil, and Nuel D. Belnap. 1993. *The Revision Theory of Truth.* MIT Press.

Hacking, Ian. 1967. "Slightly More Realistic Personal Probability." *Philosophy of Science* 34 (4): 311–325.

Hailperin, Theodore. 1996. "Sentential probability logic."

Halpern, Joseph Y. 2003. *Reasoning about Uncertainty.* Cambridge, MA: MIT Press.

Hay, Nicholas, Stuart J. Russell, Solomon Eyal Shimony, and David Tolpin. 2012. "Selecting Computations: Theory and Applications." In *Uncertainty in Artificial Intelligence (UAI-'12),* edited by Nando de Freitas and Kevin Murphy, 346–355.

Hennig, Philipp, Michael A. Osborne, and Mark Girolami. 2015. "Probabilistic numerics and uncertainty in computations." *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 471 (2179).

Hilbert, David. 1902. "Mathematical Problems." *Bulletin of the American Mathematical Society* 8 (10): 437–480.

Hintikka, Jaakko. 1962. *Knowledge and belief: An introduction to the logic of the two notions.* Cornell University Press.

———. 1979. "Impossible Possible Worlds Vindicated." In *Game-Theoretical Semantics: Essays on Semantics by Hintikka, Carlson, Peacocke, Rantala, and Saarinen,* edited by Esa Saarinen, 367–379. Springer.

Hutter, Marcus, John W. Lloyd, Kee Siong Ng, and William T. B. Uther. 2013. "Probabilities on Sentences in an Expressive Logic." *Journal of Applied Logic* 11 (4): 386–420.

Jaynes, E. T. 2003. *Probability Theory: The Logic of Science.* Edited by G. Larry Bretthorst. New York: Cambridge University Press.

Jeffrey, Richard. 1983. "Bayesianism with a human face." *Testing scientific theories, minnesota studies in the philosophy of science* 10:133–156.

Joyce, James M. 1999. *The Foundations of Causal Decision Theory.* Cambridge Studies in Probability, Induction and Decision Theory. New York: Cambridge University Press.

Kersting, Kristian, and Luc De Raedt. 2007. "Bayesian Logic Programming: Theory and Tool." Chap. 10 in *Introduction to Statistical Relational Learning,* edited by Lise Getoor and Ben Taskar, 291–322. MIT Press.

Khot, Tushar, Niranjan Balasubramanian, Eric Gribkoff, Ashish Sabharwal, Peter Clark, and Oren Etzioni. 2015. "Markov Logic Networks for Natural Language Question Answering." arXiv: `1507.03045 [stat.ML]`.

Kingma, Diederik P, and Max Welling. 2013. "Auto-Encoding Variational Bayes." arXiv: `1312.6114 [stat.ML]`.

Klir, George, and Bo Yuan. 1995. *Fuzzy Sets and Fuzzy Logic.* New Jersey: Prentice Hall.

Kok, Stanley, and Pedro Domingos. 2005. "Learning the structure of Markov logic networks." In *22nd International Conference on Machine Learning (ICML '05),* 441–448. ACM.

Kolmogorov, A.N. 1950. "Foundations of the Theory of Probability."

Krakauer, Barak. 2012. "Counterpossibles." PhD diss., University of Massachusetts - Amherst.

Kucera, Antonin, and Andre Nies. 2011. "Demuth randomness and computational complexity." *Annals of Pure and Applied Logic* 162 (7): 504–513.

Lewis, David. 1999. *Papers in metaphysics and epistemology.* Vol. 2. Cambridge University Press.

Li, Ming, and Paul M. B. Vitányi. 1993. *An Introduction to Kolmogorov Complexity and its Applications.* 1st ed. New York: Springer.

Lin, Christopher H, Andrey Kolobov, Ece Kamar, and Eric Horvitz. 2015. "Metareasoning for Planning Under Uncertainty." arXiv: `1505.00399 [cs.AI]`.

Lindley, Dennis V. 1991. "Making Decisions."

Lipman, Barton L. 1991. "How to decide how to decide how to...: Modeling limited rationality." *Econometrica: Journal of the Econometric Society* 59 (4): 1105–1125.

Łoś, Jerzy. 1955. "On the Axiomatic Treatment of Probability." *Colloquium Mathematicae* 3 (2): 125–137.

Lowd, Daniel, and Pedro Domingos. 2007. "Efficient Weight Learning for Markov Logic Networks." In *11th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2007),* edited by Joost N. Kok, Jacek Koronacki, Ramon Lopez de Mantaras, Stan Matwin, Dunja Mladenič, and Andrzej Skowron, 200–211. Springer.

McCallum, Andrew, Karl Schultz, and Sameer Singh. 2009. "Factorie: Probabilistic programming via imperatively defined factor graphs." In *Advances in Neural Information Processing Systems 22 (NIPS 2009),* edited by Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta, 1249–1257. Curran Associates, Inc.

McGee, Vann. 1990. *Truth, Vagueness, and Paradox: An Essay on the Logic of Truth.* Hackett Publishing.

Meyer, J-J Ch., and Wiebe Van Der Hoek. 1995. *Epistemic logic for AI and computer science.* Cambridge Tracts in Theoretical Computer Science 41. Cambridge University Press.

Mihalkova, Lilyana, Tuyen Huynh, and Raymond J Mooney. 2007. "Mapping and Revising Markov Logic Networks for Transfer Learning." In *22nd Conference on Artificial Intelligence (AAAI-07),* 1:608–614. AAAI Press.

Mortensen, Chris. 2013. *Inconsistent Mathematics.* Vol. 312. Mathematics and Its Applications. Springer.

Muggleton, Stephen H., and Hiroaki Watanabe. 2014. *Latest Advances in Inductive Logic Programming.* World Scientific.

Muiño, David Picado. 2011. "Measuring and Repairing Inconsistency in Probabilistic Knowledge Bases." *International Journal of Approximate Reasoning* 52 (6): 828–840.

Opitz, David, and Richard Maclin. 1999. "Popular Ensemble Methods: An Empirical Study." *Journal of Artificial Intelligence Research* 11:169–198.

Pearl, Judea. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* San Mateo, CA: Morgan Kaufmann.

Polya, George. 1990. *Mathematics and Plausible Reasoning: Patterns of plausible inference.* Vol. 2. Princeton University Press.

Potyka, Nico. 2015. "Solving Reasoning Problems for Probabilistic Conditional Logics with Consistent and Inconsistent Information." PhD diss., Fernuniversität.

Potyka, Nico, and Matthias Thimm. 2015. "Probabilistic Reasoning with Inconsistent Beliefs Using Inconsistency Measures." In *24th International Joint Conference on Artificial Intelligence (IJCAI-15),* 3156–3163. Buenos Aires, Argentina: AAAI Press.

Priest, Graham. 2002. "Paraconsistent Logic." In *Handbook of Philosophical Logic,* 6:287–393. Kluwer Academic Publishers.

Ramsey, Frank Plumpton. 1931. "Truth and Probability." In *The Foundations of Mathematics and other Logical Essays,* edited by Richard Bevan Braithwaite, 156–198. New York: Harcourt, Brace.

Rantala, Veikko. 1979. "Urn Models: A New Kind of Non-Standard Model for First-Order Logic." In *Game-Theoretical Semantics: Essays on Semantics by Hintikka, Carlson, Peacocke, Rantala, and Saarinen,* edited by Esa Saarinen, 347–366. Springer.

Richardson, Matthew, and Pedro Domingos. 2006. "Markov Logic Networks." *Machine Learning* 62 (1-2): 107–136.

Rubinstein, Ariel. 1998. *Modeling Bounded Rationality.* MIT Press.

Russell, Stuart J. 2016. "Rationality and Intelligence: A Brief Update." In *Fundamental Issues of Artificial Intelligence,* 376:7–28. Synthese Library. Springer.

Russell, Stuart J., and Eric H. Wefald. 1991a. *Do the Right Thing: Studies in Limited Rationality.* MIT Press.

———. 1991b. "Principles of Metareasoning." *Artificial intelligence* 49 (1-3): 361–395.

Savage, Leonard J. 1954. "The foundations of statistics."

———. 1967. "Difficulties in the theory of personal probability." *Philosophy of Science* 34 (4): 305–310.

Sawin, Will, and Abram Demski. 2013. "Computable probability distributions which converge on believing true $\Pi 1$ sentences will disbelieve true $\Pi 2$ sentences."

Schlesinger, George N. 1985. *Range of Epistemic Logic.* Scots Philosophical Monograph. Elsevier Science Ltd.

Simon, Herbert Alexander. 1982. *Models of bounded rationality: Empirically grounded economic reason.* Vol. 3. MIT Press.

Singla, Parag, and Pedro Domingos. 2005. "Discriminative Training of Markov Logic Networks." In *20th National Conference on Artificial Intelligence (AAAI-05),* 2:868–873. AAAI Press.

Soares, Nate, and Benja Fallenstein. 2015. "Toward Idealized Decision Theory." arXiv: `1507.01986 [cs.AI]`.

Solomonoff, Ray J. 1964a. "A Formal Theory of Inductive Inference. Part I." *Information and Control* 7 (1): 1–22.

———. 1964b. "A Formal Theory of Inductive Inference. Part II." *Information and Control* 7 (2): 224–254.

Sowa, John F. 1999. *Knowledge Representation: Logical, Philosophical, and Computational Foundations.* Brooks-Cole.

Sprenger, Jan. 2015. "A Novel Solution to the Problem of Old Evidence." *Philosophy of Science* 82 (3): 383–401.

Teller, Paul. 1973. "Conditionalization and observation." *Synthese* 26 (2): 218–258.

Thimm, Matthias. 2013a. "Inconsistency Measures for Probabilistic Logics." *Artificial Intelligence* 197:1–24.

———. 2013b. "Inconsistency measures for probabilistic logics." *Artificial Intelligence* 197:1–24.

Tran, Son D, and Larry S Davis. 2008. "Event modeling and recognition using markov logic networks." In *European Conference on Computer Vision,* 610–623. Springer.

Turing, Alan M. 1936. "On Computable Numbers, with an Application to the Entscheidungsproblem." *Proceedings of the London Mathematical Society,* 2nd ser., 42 (230–265).

Vajda, Steven. 1972. *Probabilistic Programming.* Probability and Mathematical Statistics: A Series of Monographs and Textbooks. Academic Press.

Vander Laan, David. 2004. "Counterpossibles and similarity." Chap. 20 in *Lewisian Themes: The Philosophy of David K. Lewis,* edited by Frank Jackson and Graham Priest, 258–275. Oxford University Press.

von Neumann, John, and Oskar Morgenstern. 1944. *Theory of Games and Economic Behavior.* 1st ed. Princeton, NJ: Princeton University Press.

Wang, Jue, and Pedro M. Domingos. 2008. "Hybrid Markov Logic Networks." In *23rd Conference on Artificial Intelligence (AAAI-08),* 2:1106–1111. AAAI Press.

Wood, Frank, Jan-Willem van de Meent, and Vikash Mansinghka. 2014. "A New Approach to Probabilistic Programming Inference." In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS),* 33:1024–1032. JMLR Workshop and Conference Proceedings.

Yen, John, and Reza Langari. 1999. *Fuzzy Logic: Intelligence, Control, and Information.* Vol. 1. Pearson.

Zhang, Yitang. 2014. "Bounded gaps between primes." *Annals of Mathematics* 179 (3): 1121–1174.

Zilberstein, Shlomo. 2008. "Metareasoning and bounded rationality." *Metareasoning: Thinking about Thinking, MIT Press, forthcoming.*

Zvonkin, Alexander K., and Leonid A. Levin. 1970. "The Complexity of Finite Objects and the Development of the Concepts of Information and Randomness by Means of the Theory of Algorithms." *Russian Mathematical Surveys* 25 (6): 83–124.

Zynda, Lyle. 1995. "Old evidence and new theories." *Philosophical Studies* 77 (1): 67–95.

A   Preliminaries

A.1   Organization of the Appendix

The appendix is organized differently from the paper. Here we describe the broad dependency structure of the proofs and mention the theorems that are proven by constructing explicit traders (rather than as corollaries). Note that theorems that were proven in Section 6 are also proven here, but differently (generally much more concisely, as a corollary of some other theorem).

A. Preliminaries. Appendix A.2 describes expressible features in full detail. Appendix A.3 defines some notions for combinations, and defines when a sequence of traders can be "efficiently emulated", which will be useful in B, D.1, and G.

B. Convergence. Appendix B.1 introduces a tool for constructing traders (Lemma B.1.3, Return on Investment) that is used in B and D.1. Appendices B.2 (Affine Preemptive Learning) and B.5 (Persistence of Affine Knowledge) prove those theorems using Lemma B.1.3, and the remainder of B derives some corollaries (convergence and non-affine special cases).

C. Coherence. Appendix C.1 proves Affine Coherence, giving (Affine) Provability Induction as corollaries. The remainder of C derives corollaries of Provability Induction (consistency and halting) and of Affine Provability Induction (coherence and exclusive-exhaustive relationships).

D. Statistics. Appendix D.1 proves Affine Recurring Unbiasedness using Lemma B.1.3, giving Simple Calibration (D.3) as a corollary. Appendices D.4 (Affine Unbiasedness From Feedback) and D.6 (Learning Pseudorandom Affine Sequences) prove those theorems by constructing traders, and the remainder of Appendix D derives corollaries (varied and non-affine cases).

E. Expectations. Appendix E.2 proves the Mesh Independence Lemma by constructing a trader, and E.1 and E.5 prove two other lemmas on expectations; basic properties of expectations such as convergence and linearity are also proved. These proofs rely on theorems proven in B and C. The remainder of E proves analogs for expectations of the convergence, coherence, and statistical theorems by applying their affine versions to $\mathcal{F}$-combinations expressing expectations.

F. Introspection and Self-Trust. The first part of Appendix F proves introspection properties using Affine Provability Induction and Expectation Provability Induction. The remainder derives the self-trust properties as applications of theorems proven in Appendix E.

G. Non-Dogmatism and Closure. Appendix G is mostly self-contained. Appendix G.1 proves a simple analog of the return on investment lemma with stronger hypotheses; this is applied to constructing traders in G.2 (Uniform Non-Dogmatism), G.3 (Occam Bounds), and G.5 (Domination of the Universal Semimeasure), with non-dogmatism and strict domination as corollaries. Appendix G.8 (Conditionals on Theories) uses uniform non-dogmatism, preemptive learning, and G.7 (Closure under Finite Perturbations).

A.2   Expressible Features

This section can be safely skipped and referred back to as desired.

Recall that a trading strategy for day n is given by an affine combination of sentences with expressible feature coefficients. As such, a machine that implements a trader must use some notation for writing down those features. Here, to be fully rigorous, we will make an explicit choice of notation for expressible features. Recall their definition:

Definition 3.4.3 (Expressible Feature)**.** *An **expressible feature** $\xi \in \mathcal{F}$ is a valuation feature expressible by an algebraic expression built from price features ϕ^{*n} for each $n \in \mathbb{N}^+$ and $\phi \in \mathcal{S}$, rational numbers, addition, multiplication, $\max(-, -)$, and a "safe reciprocation" function $\max(1, -)^{-1}$.*

*We write $\mathcal{EF}$ for the set of all expressible features, $\mathcal{EF}_n$ for the set of expressible features of rank $\leq n$, and define an **$\mathcal{EF}$-progression** to be a sequence $\overline{\xi}$ such that $\xi_n \in \mathcal{EF}_n$.*

A (multi-line) string representing an expressible feature will be called a *well-formed feature expression*, and will be built from smaller expressions involving variables (mainly to save space when a particular expression would otherwise need to be repeated many times).

We define the set of *variable feature expressions* Ξ inductively to include:

- Past and present market prices: for all $i \leq n$ and for all $\psi \in \mathcal{S}$, there is a symbol $\psi^{*i} \in \Xi$.

- Rationals: $\mathbb{Q} \subset \Xi$.

- Variables: $V \subset \Xi$.

Further, if $\xi \in \Xi$ and $\zeta \in \Xi$, then the following operations on them are as well:

- Addition: $\xi + \zeta \in \Xi$.

- Multiplication: $\xi \cdot \zeta \in \Xi$.

- Maximum: $\max(\xi, \zeta) \in \Xi$.

- Safe reciprocation: $1/\max(1, \xi) \in \Xi$.

These operations are sufficient to generate all the expressible features we will need. For example,

$$-\xi := (-1) \cdot \xi;$$
$$\min(\xi, \zeta) := -\max(-\xi, -\zeta);$$
$$|\xi| := \max(\xi, -\xi);$$

and when $\zeta \geq \varepsilon$ for some constant $\varepsilon > 0$, we can define

$$\xi/\zeta := (1/\varepsilon) \cdot \xi/\max(1, (1/\varepsilon) \cdot \zeta).$$

We now define a *well-formed feature expression* to be a (multi-line) string of the following form:

$$v_1 := \text{(feature expression with no variables)};$$
$$v_2 := \text{(feature expression involving } v_1);$$
$$\cdots$$
$$v_k := \text{(feature expression involving } v_1, \ldots, v_{k-1});$$
$$\text{return (feature expression involving } v_1, \ldots, v_k),$$

where the final expression after "return " is the expression evaluated to actually compute the expressible feature defined by this code block.

Examples

The following well-formed feature expression defines a rank 7 expressible feature:

$$v_1 := \phi_1^{*7} + \phi_2^{*4}$$
$$v_2 := v_1 - 1$$
$$\text{return } 3 \cdot \max(v_1, v_2).$$

If the market at time 7 has $\mathbb{P}_7(\phi_1) = 0.8$ and the market at time 4 had $\mathbb{P}_4(\phi_2) = 0$, then this expressible feature evaluates to

$$3 \cdot \max(v_1, v_2) = 3 \cdot \max(0.8, -0.2) = 2.4.$$

An n-strategy can now be written down in a very similar format, sharing variable definitions used in the various coefficients to save space. For example, the following code defines a 7-strategy:

$$v_1 := \phi_1^{*7} + \phi_2^{*4}$$
$$v_2 := v_1 \cdot v_2$$
$$T[\phi_1] := 3 \cdot \max(v_1, v_2)$$
$$T[\phi_2] := 6 \cdot \max(v_1, v_2).$$
$$T := \sum_{i=1}^{2} T[\phi_i] \cdot (\phi_i - \phi_i^{*n})$$
$$\text{return } T$$

Notice that the function ϕ_1^{*7} returning the current market price of ϕ_1 affects (via v_1) how many shares of ϕ_1 this trader buys. This is permitted, and indeed is crucial for allowing traders to base their trades on the current market prices.

Dynamic programming for traders

We will often define traders that make use of indexed variables that are defined recursively in terms of previous indices, as in e.g. the proof of Theorem 4.1.1 (Convergence) in Section 6.1. In particular, we often have traders refer to their own past trades, e.g. using expressible features of the form $T_i[\phi]$ for $i < n$ to define their trade at time n. This can be written down in polynomial time using the expression language for features, via dynamic programming. For example, to use previous trades, a trader can recapitulate all the variables used in all its previous trading strategies. As long as the trading strategies are efficiently computable given previous trades as variables, they are still efficiently computable without them (possibly with a higher-degree polynomial).

A.3   Definitions

Price of a Combination

Definition A.3.1 (Price of a Combination)**.** *Given any affine combination*

$$A = c + \xi_1 \phi_1 + \cdots + \xi_k \phi_k$$

of rank $\leq n$, observe that the map $\overline{\mathbb{V}} \mapsto \mathbb{V}_n(A)$ is an expressible feature, called the **price** *of A on day n, and is given by the expression*

$$A^{*n} := c + \xi_1 {\phi_1}^{*n} + \cdots + \xi_k {\phi_k}^{*n}.$$

For any valuation sequence $\overline{\mathbb{U}}$, observe by linearity and associativity that

$$(\mathbb{V}(A))(\overline{\mathbb{U}}) = \mathbb{V}(A(\overline{\mathbb{U}})) = c(\overline{\mathbb{U}}) + \sum_{\phi} \xi_{\phi}(\overline{\mathbb{U}}) \mathbb{V}(\phi).$$

Buying a Combination

Definition A.3.2 (Buying a Combination)**.** *Given any $\mathcal{EF}$-combination $A^{\dagger}$ of rank $\leq n$, we define a corresponding n-strategy called* **buying $A^{\dagger}$ on day n** *to equal*

$$A^{\dagger} - A^{\dagger *n}.$$

Observe that buying A on day n is indeed an n-strategy.

$\mathcal{F}$-Combinations Corresponding to $\mathcal{F}$-LUV Combinations

Definition A.3.3 (Ex)**.** *Let* $B := c + \xi_1 X_1 + \cdots + \xi_k X_k$ *be an* $\mathcal{F}$-*LUV combination. Define*

$$\mathrm{Ex}_m(A) = c + \xi_1 \sum_{i=0}^{m-1} \frac{1}{m} (\text{``}\underline{X_1} > \underline{i}/\underline{m}\text{''}) + \cdots + \xi_k \sum_{i=0}^{m-1} \frac{1}{m} (\text{``}\underline{X_k} > \underline{i}/\underline{m}\text{''})$$

to be a $\mathcal{F}$-*affine combination corresponding to* B. *Note that* $\mathbb{V}(\mathrm{Ex}_m(B)) = \mathbb{E}_m^{\mathbb{V}}(B)$. *Also note that if* $(B_n)_n$ *is bounded, then* $(\mathrm{Ex}_n(B_n))_n$ *is bounded; we will use this fact freely in what follows.*

Efficiently Emulatable Sequence of Traders

In Appendices B, D.1, and G, we will construct traders that allocate their money across multiple strategies for exploiting the market. In order to speak unambiguously about multiple overlapping long-term strategies for making trades, we define the notion of a sequence of traders that can be efficiently emulated by one trader.

Definition A.3.4 (Efficiently Emulatable Sequence of Traders)**.** *We say that a sequence of traders* $(\overline{T}^k)_k$ *is* efficiently emulatable *if*

- *the sequence of programs that compute the* $\overline{T}^k$ *can be efficiently generated;*

- *those programs for* $\overline{T}^k$ *have uniformly bounded runtime, i.e., there exists a constant* c *such that for all* k *and all times* n, *the program that computes* $\overline{T}^k$ *runs in time* $\mathcal{O}(n^c)$; *and*

- *for all* k *and all* $n < k$, *we have that* $\overline{T}_n^k$ *is the zero trade.*

Efficiently emulatable sequences are so named because a single trader $\overline{T}$ can emulate the entire sequence of traders $(\overline{T}^k)_k$. That is, on time n, $\overline{T}$ can directly compute all the trading strategies T_n^k for $k \leq n$ by listing the appropriate programs and running them on input n. This can be done in polynomial time by definition of an efficiently emulatable sequence. We require that $\overline{T}^k$ does not make non-zero trades before time k so that the emulator $\overline{T}$ need not truncate any trades made by the $\overline{T}^k$.

B   Convergence Proofs

B.1   Return on Investment

This section provides a useful tool for constructing traders, which will be applied in Appendix B and Appendix D.1. The reader may wish to first begin with the proof in Appendix B.2 of Theorem 4.5.7 as motivation of the return on investment lemma.

Statement of the ε-ROI lemma.   If we have a logical inductor $\overline{\mathbb{P}}$, we know that $\overline{\mathbb{P}}$ cannot be exploited by any trader. It will often be easy to show that if $\overline{\mathbb{P}}$ fails to satisfy some property, then there is a trader $\overline{T}$ that takes advantage of a specific, one-shot opportunity to trade against the market in a way that is guaranteed to eventually be significantly higher value than the size of the original investment; and that such opportunities arise infinitely often. In order to use such a situation to ensure that the market $\overline{\mathbb{P}}$ satisfies the property, we will now show that logical inductors are not susceptible to repeatable methods for making a guaranteed, substantial profit.

To define a notion of return on investment, we first define the "magnitude" of a trade made by a trader, so that we can talk about traders that are profitable in

proportion to the size of their trades:

$$\|T(\overline{\mathbb{P}})\|_{\mathrm{mg}} := \sum_{\phi \in \mathcal{S}} |T[\phi](\overline{\mathbb{P}})|.$$

This number will be called the **magnitude** of the trade. It is just the total number of shares traded by T against the market $\overline{\mathbb{P}}$, whether the shares are bought or sold. Note that the magnitude is *not* the same as the $\| - \|_1$-norm of $T(\overline{\mathbb{P}})$; the magnitude omits the constant term $T[1](\overline{\mathbb{P}})$.

The magnitude is a simple bound on the value of the holdings $T_n(\overline{\mathbb{P}})$: for any world $\mathbb{W}$ (plausible or not),

$$\left| \sum_{\phi \in \mathcal{S}} T_n[\phi](\overline{\mathbb{P}}) \cdot (\mathbb{W}(\phi) - \mathbb{P}_n(\phi)) \right| \leq \sum_{\phi \in \mathcal{S}} \left| T_n[\phi](\overline{\mathbb{P}}) \right| \cdot 1 = \|T_n(\overline{\mathbb{P}})\|_{\mathrm{mg}},$$

since $\mathbb{W}(\phi) \in \{0,1\}$ and $\mathbb{P}_n(\phi) \in [0,1]$. Now we define the total magnitude of a trader over time.

Definition B.1.1 (Magnitude of a Trader)**.** *The **magnitude** $\|\overline{T}(\overline{\mathbb{P}})\|_{\mathrm{mg}}$ of a trader $\overline{T}$ against the market $\overline{\mathbb{P}}$ is*

$$\|\overline{T}(\overline{\mathbb{P}})\|_{\mathrm{mg}} := \sum_{n \in \mathbb{N}^+} \|T_n(\overline{\mathbb{P}})\|_{\mathrm{mg}} \equiv \sum_{n \in \mathbb{N}^+} \sum_{\phi \in \mathcal{S}} |T_n[\phi](\overline{\mathbb{P}})|.$$

The magnitude of $\overline{T}$ is the total number of shares it trades (buys or sells) over all time.

Now we define what it means for a trader to increase its net value by a substantial fraction of its investment, i.e., its magnitude.

Definition B.1.2 (ε Return on Investment)**.** *For $\varepsilon > 0$, we say that a trader $\overline{T}$ trading against $\overline{\mathbb{P}}$ has ε return on investment or ε-ROI if, for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,*

$$\lim_{n \to \infty} \mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}}) \right) \geq \varepsilon \|\overline{T}(\overline{\mathbb{P}})\|_{\mathrm{mg}}.$$

In words, a trader $\overline{T}$ has ε-ROI if, in the limit of time and deduction, the value of its holdings is in every world at least ε times its total investment $\|\overline{T}(\overline{\mathbb{P}})\|_{\mathrm{mg}}$. Note that this does not merely say that $\overline{T}$ recoups at least an ε fraction of its original cost; rather, the net value is guaranteed in all worlds consistent with Γ to have increased by an ε fraction of the magnitude $\|\overline{T}(\overline{\mathbb{P}})\|_{\mathrm{mg}}$ of $\overline{T}$'s trades.

Recall from Definition 4.3.5 that a sequence $\overline{\alpha}$ of rationals is $\overline{\mathbb{P}}$-generable if there is some e.c. $\mathcal{EF}$-progression $\overline{\alpha^\dagger}$ such that $\alpha_n^\dagger(\overline{\mathbb{P}}) = \alpha_n$ for all n.

Lemma B.1.3 (No Repeatable ε-ROI)**.** *Let $\overline{\mathbb{P}}$ be a logical inductor with respect to some deductive process $\overline{D}$, and let $(\overline{T}^k)_{k \in \mathbb{N}^+}$ be an efficiently emulatable sequence of traders (Definition A.3.4). Suppose that for some fixed $\varepsilon > 0$, each trader $\overline{T}^k$ has ε-ROI. Suppose further that there is some $\overline{\mathbb{P}}$-generable sequence $\overline{\alpha}$ such that for all k,*

$$\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = \alpha_k.$$

Then

$$\lim_{k \to \infty} \alpha_k = 0.$$

In words, this says roughly that there is no efficient, repeatable method for producing a substantial guaranteed return on an investment. The condition that $\overline{\alpha}$ is $\overline{\mathbb{P}}$-generable will help with the budgeting done by the trader that emulates the sequence $(\overline{T}^k)_k$.

Proof strategy.   We will construct a trader $\overline{T}$ that emulates the sequence $(\overline{T}^k)$ in a manner such that if the traders $\overline{T}^k$ did not make trades of vanishing limiting value, then our trader $\overline{T}$ would accrue unbounded profit by repeatedly making investments that are guaranteed to pay out by a substantial amount. Very roughly, on time n, $\overline{T}$ will sum together the trades T_n^k made by all the $\overline{T}^k$ with $k \leq n$. In this way, $\overline{T}$ will accrue all the profits made by each of the $\overline{T}^k$.

The main problem we have to deal with is that $\overline{T}$ risks going deeper and deeper into debt to finance its investments, as discussed before the proof of Theorem 4.1.1 (Convergence) in Section 6.1. That is, it may be that each $\overline{T}^k$ makes an investment that takes a very long time for all the worlds $\mathbb{W} \in \mathcal{PC}(D_n)$ plausible at time n to value highly. In the meanwhile, $\overline{T}$ continues to spend money buying shares and taking on risk from selling shares that might plausibly demand a payout. In this way, despite the fact that each of its investments will eventually become profitable, $\overline{T}$ may have holdings with unboundedly negative plausible value.

To remedy this, we will have our trader $\overline{T}$ keep track of its "debt" and of which investments have already paid off, and then scale down new traders $\overline{T}^k$ so that $\overline{T}$ maintains a lower bound on the plausible value of its holdings. Roughly speaking, $\overline{T}$ at time n checks whether the current holdings of $\overline{T}^k$ are guaranteed to have a positive value in all plausible worlds, for each $k \leq n$. Then $\overline{T}$ sums up the total magnitudes α_k of all the trades ever made by those $\overline{T}^k$ whose trades are not yet guaranteed to be profitable. This sum is used to scale down all trades made by $\overline{T}^n$, so that the total magnitude of the unsettled investments made by $\overline{T}$ will remain bounded.

Proof of Lemma B.1.3.

Proof. We now prove Lemma B.1.3.

We can assume without loss of generality that each $\alpha_n \leq 1$ by dividing $\overline{T}^k$'s trades by $\max(1, \alpha_k)$.

Checking profitability of investments.   At time n, our trader $\overline{T}$ runs a (possibly very slow) search process to enumerate traders $\overline{T}^k$ from the sequence $(\overline{T}^k)_k$ that have made trades that are already guaranteed to be profitable, as judged by what is plausible according to the deductive process $\overline{D}$ with respect to which $\overline{\mathbb{P}}$ is a logical inductor. That is, $\overline{T}$ runs a search for pairs of numbers $k, m \in \mathbb{N}^+$ such that:

$$\sum_{i \leq m} \|T_i^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} \geq (1 - \varepsilon/3)\alpha_k \qquad \text{(few future trades), and}$$

$$\inf_{\mathbb{W} \in \mathcal{PC}(D_m)} \mathbb{W}\left(\sum_{i \leq m} T_i^k(\overline{\mathbb{P}})\right) \geq (2\varepsilon/3)\alpha_k \qquad \text{(guaranteed profit).}$$

If the trader $\overline{T}^k$ has few future trades and guaranteed profit at time m then we say that the trader's holdings have matured. We denote the least such m by $m(k)$.

The first condition (few future trades) says that $\overline{T}^k$ has made trades of total magnitude at least $(1 - \varepsilon/3)\alpha_k$ after time k up until time m. By the assumption that $\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = \alpha_k$, for each k there is some time step m such that this condition holds. By that same assumption, $\overline{T}^k$ will make trades of total magnitude at most $(\varepsilon/3)\alpha_k$ in all time steps after m.

The second condition (guaranteed profit) says that the minimum value plausible at time m of all trades made by $\overline{T}^k$ up until time m is at least $(2\varepsilon/3)\alpha_k$. By the assumption that $\overline{T}^k$ has ε-ROI, i.e., that the minimum value of $\overline{T}^k$ is eventually at least $\varepsilon\|T_k^k\|_{\mathrm{mg}}$, the condition of guaranteed profit will hold at some m.

The idea is that, since $\overline{T}^k$ will trade at most $(\varepsilon/3)\alpha_k$ shares after m and the holdings of $\overline{T}^k$ from trades up until the current time have minimum plausible value at least $(2\varepsilon/3)\alpha_k$, it is guaranteed that the holdings of $\overline{T}^k$ at any time after m will have minimum plausible value at least $(\varepsilon/3)\alpha_k$. This will allow our trader $\overline{T}$ to "free up" funds allocated to emulating $\overline{T}^k$, in the sense that there is no longer ever any plausible net downside to the holdings from trades made by $\overline{T}^k$.

Definition of the trader $\overline{T}$. For fixed k and m, these two conditions refer to specific individual computations (namely D_m, the $T_i^k(\overline{\mathbb{P}})$ for $i \leq m$, and α_k). On time step n, for all $k, j \leq n$, our trader $\overline{T}$ sets Boolean variables $\mathrm{open}(k, j) := 0$ if it is verified in j steps of computation that the holdings of $\overline{T}^k$ have matured; and $\mathrm{open}(k, j) := 1$ if $\overline{T}^k$ has open investments. Since the holdings of each $\overline{T}^k$ will eventually mature, for all k there is some n such that $\mathrm{open}(k, n) = 0$.

Let $\overline{\alpha^\dagger}$ be an e.c. $\mathcal{EF}$ progression such that for each n we have $\alpha_n^\dagger(\overline{\mathbb{P}}) = \alpha_n$. Then $\overline{T}$ outputs the trading strategy

$$T_n := \sum_{k \leq n} \beta_k^\dagger \cdot T_n^k,$$

where the $\beta_k^\dagger$ are defined recursively by

$$\beta_k^\dagger := 1 - \sum_{i < k} \mathrm{open}(i, k) \beta_i^\dagger \alpha_i^\dagger.$$

That is, the machine computing $\overline{T}$ outputs the definitions of the budget variables $\beta_k^\dagger$ for each $k \leq n$, and then lists the trades

$$\mathrm{return}\ \phi := \sum_{k \leq n} \beta_k^\dagger T_n^k[\phi]$$

for each ϕ listed by any of the trades T_n^k for $k \leq n$. As shorthand, we write $\beta_k := \beta_k^\dagger(\overline{\mathbb{P}})$. Notice that since $(\overline{T}^k)_k$ is efficiently emulatable, we have $\forall k : \forall i < k : T_i^k \equiv 0$, and therefore

$$\forall n : T_n(\overline{\mathbb{P}}) = \sum_{k \in \mathbb{N}^+} \beta_k T_n^k(\overline{\mathbb{P}}).$$

Note that each $\mathrm{open}(i, k)$ is pre-computed by the machine that outputs our trader $\overline{T}$ and then is encoded as a constant in the expressible feature $\beta_k^\dagger$. The trade coordinate $T_n[\phi]$ is an expressible feature because the $\beta_k^\dagger$ and $T_n^k[\phi]$ are expressible features.

Budgeting the traders $\overline{T}^k$. Since we assumed each $\alpha_k \leq 1$, it follows from the definition of the budget variable $\beta_k^\dagger$ that

$$\beta_k \alpha_k \leq 1 - \sum_{i < k} \mathrm{open}(i, k) \beta_i \alpha_i,$$

and then β_k is used as the constant scaling factor for $\overline{T}^k$ in the sum defining $\overline{T}$'s trades. In this way, we maintain the invariant that for any n,

$$\sum_{k \leq n} \mathrm{open}(k, n) \beta_k \alpha_k \leq 1.$$

Indeed, by induction on k, using the fact that $\mathrm{open}(i, n)$ implies $\mathrm{open}(i, m)$ for $m \geq n$, we have $\beta_k \geq 0$ and the above invariant holds.

In words, this says that out of all the traders $\overline{T}^k$ with investments still open at time n, the sum of the magnitudes $\beta_k \alpha_k$ of their total investments (as budgeted by the β_k) is bounded by 1.

Analyzing the value of $\overline{T}$'s holdings. Now we lower bound the value of the holdings of $\overline{T}$ from trades against the market $\overline{\mathbb{P}}$. Fix any time step n and world $\mathbb{W} \in \mathcal{PC}(D_n)$ plausible at time n. Then we have that the value of $\overline{T}$'s holdings at time n is

$$\mathbb{W}(\textstyle\sum_{i \leq n} T_i(\overline{\mathbb{P}})) = \sum_{k \leq n} \mathbb{W}(\textstyle\sum_{i \leq n} \beta_k T_i^k(\overline{\mathbb{P}}))$$

by linearity and by definition of our trader $\overline{T}$;

$$= \sum_{\substack{k \leq n \\ \text{open}(k,n)}} \mathbb{W}(\textstyle\sum_{i \leq n} \beta_k T_i^k(\overline{\mathbb{P}})) + \sum_{\substack{k \leq n \\ \neg\text{open}(k,n)}} \mathbb{W}(\textstyle\sum_{i \leq n} \beta_k T_i^k(\overline{\mathbb{P}}))$$

again by linearity. We analyze the first term, the value of the holdings that have not yet matured, as follows:

$$\sum_{\substack{k \leq n \\ \text{open}(k,n)}} \mathbb{W}(\textstyle\sum_{i \leq n} \beta_k T_i^k(\overline{\mathbb{P}})) \geq - \sum_{\substack{k \leq n \\ \text{open}(k,n)}} \beta_k \sum_{i \leq n} \|T_i^k(\overline{\mathbb{P}})\|_{\text{mg}}$$

$$\geq - \sum_{\substack{k \leq n \\ \text{open}(k,n)}} \beta_k \sum_{i \in \mathbb{N}^+} \|T_i^k(\overline{\mathbb{P}})\|_{\text{mg}}$$

$$= - \sum_{k \leq n} \text{open}(k,n)\beta_k \alpha_k$$

$$\geq -1,$$

by the previous discussion of the β_k. In short, the β_k were chosen so that the total magnitude of all of $\overline{T}$'s holdings from trades made by any $\overline{T}^k$ that haven't yet matured stays at most 1, so that its plausible value stays at least -1.

Now we analyze the second term in the value of $\overline{T}$'s holdings, representing the value of the holdings that have already matured, as follows:

$$\sum_{\substack{k \leq n \\ \neg\text{open}(k,n)}} \mathbb{W}(\textstyle\sum_{i \leq n} \beta_k T_i^k(\overline{\mathbb{P}}))$$

$$= \sum_{\substack{k \leq n \\ \neg\text{open}(k,n)}} \left(\mathbb{W}(\textstyle\sum_{i \leq m(k)} \beta_k T_i^k(\overline{\mathbb{P}})) + \mathbb{W}(\textstyle\sum_{m(k) < i \leq n} \beta_k T_i^k(\overline{\mathbb{P}})) \right)$$

where $m(k)$ is minimal such that $\overline{T}^k$ has guaranteed profit and makes few future trades at time $m(k)$, as defined above;

$$\geq \sum_{\substack{k \leq n \\ \neg\text{open}(k,n)}} \left(\beta_k(2\varepsilon/3)\alpha_k - \sum_{i > m(k)} \beta_k \|T_i^k(\overline{\mathbb{P}})\|_{\text{mg}} \right)$$

since by definition of $m(k)$ and the guaranteed profit condition, the value of the holdings of $\overline{T}^k$ from its trades up until time $m(k)$ is at least $(2\varepsilon/3)\alpha_k$ in any world in D_n;

$$\geq \sum_{\substack{k \leq n \\ \neg\text{open}(k,n)}} (\beta_k(2\varepsilon/3)\alpha_k - \beta_k(\varepsilon/3)\alpha_k)$$

since $\overline{T}^k$ is guaranteed to make trades of magnitude at most $(\varepsilon/3)\alpha_k$ after time $m(k)$;

$$= \sum_{\substack{k \leq n \\ \neg\text{open}(k,n)}} \beta_k(\varepsilon/3)\alpha_k.$$

86

Completing our analysis, we have a lower bound on the value in $\mathbb{W}$ of the holdings of $\overline{T}$ at time n:

$$\mathbb{W}(\textstyle\sum_{i \leq n} T_i(\overline{\mathbb{P}})) \geq -1 + \sum_{\substack{k \leq n \\ \neg\mathrm{open}(k,n)}} \beta_k(\varepsilon/3)\alpha_k.$$

$\overline{T}$ exploits $\overline{\mathbb{P}}$ unless $\overline{\alpha}$ vanishes. Since $\overline{T}$ is an efficient trader and $\overline{\mathbb{P}}$ is a logical inductor, $\overline{T}$ does not exploit $\overline{\mathbb{P}}$. That is, the set

$$\left\{ \mathbb{W}\left(\textstyle\sum_{i \leq n} T_i(\overline{\mathbb{P}}) \right) \;\middle|\; n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n) \right\}$$

is bounded above, since it is bounded below by -1 by the above analysis. In words, the plausible value of $\overline{T}$'s holdings is always at least -1, so by the logical induction criterion it cannot go to infinity. Therefore, again by the above analysis, we must have

$$\lim_{n \to \infty} \sum_{\substack{k \leq n \\ \neg\mathrm{open}(k,n)}} \beta_k(\varepsilon/3)\alpha_k < \infty.$$

As shown above, for any k the conditions for $\neg\mathrm{open}(k,n)$ will eventually be met by all sufficiently large n. Thus

$$\lim_{n \to \infty} \sum_{\substack{k \leq n \\ \neg\mathrm{open}(k,n)}} \beta_k(\varepsilon/3)\alpha_k = \sum_k (\varepsilon/3)\beta_k\alpha_k < \infty.$$

Now we show that $\lim_{k \to \infty} \alpha_k = 0$. Suppose by way of contradiction that for some $\delta \in (0,1)$, $\alpha_k > \delta$ for infinitely many k, but nevertheless for some sufficiently large time step n, we have

$$\sum_{i > n} \beta_i \alpha_i < 1/2.$$

Recall that for each $i \leq n$, at some time $n(i)$, $\mathrm{open}(i, n(i)) = 0$ verifies that the holdings of $\overline{T}^i$ have matured. Let N be any number greater than $n(i)$ for all $i \leq n$. Then

$$\sum_{i < N} \mathrm{open}(i, N)\beta_i\alpha_i = \sum_{i \leq n} 0 \cdot \beta_i\alpha_i + \sum_{n < i < N} \mathrm{open}(i, N)\beta_i\alpha_i$$

$$\leq 0 + \sum_{n < i < N} \beta_i\alpha_i$$

$$\leq 1/2.$$

So for infinitely many sufficiently large k we have

$$\alpha_k\beta_k = \alpha_k \left(1 - \sum_{i < k} \mathrm{open}(i, k)\beta_i\alpha_i \right)$$

$$\geq \alpha_k(1 - 1/2)$$

$$\geq \delta/2.$$

Thus

$$\sum_k (\varepsilon/3)\beta_k\alpha_k = \infty,$$

contradicting that this sum is bounded. Therefore in fact $\alpha_k \eqsim_k 0$, as desired. $\quad\square$

B.2 Affine Preemptive Learning

Theorem 4.5.7 (Affine Preemptive Learning). *Let $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \mathbb{P}_n(A_n) = \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{P}_m(A_n)$$

and

$$\limsup_{n \to \infty} \mathbb{P}_n(A_n) = \limsup_{n \to \infty} \inf_{m \geq n} \mathbb{P}_m(A_n) \ .$$

Proof strategy: buying combinations that will appreciate, and ROI. The inequality

$$\liminf_{n \to \infty} \mathbb{P}_n(A_n) \geq \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{P}_m(A_n)$$

states roughly that $\mathbb{P}_n$ cannot infinitely often underprice the $\mathbb{R}$-combination A_n by a substantial amount in comparison to any price $\mathbb{P}_m(A_n)$ assigned to A_n by $\mathbb{P}_m$ at any future time $m \geq n$.

Intuitively, if the market $\overline{\mathbb{P}}$ did not satisfy this inequality, then $\overline{\mathbb{P}}$ would be exploitable by a trader that buys the $\mathbb{R}$-combination A_n when its price is low, and then sells it back when, inevitably, the price is substantially higher. If we have sold back all our shares in some sentence ϕ, then there is no contribution, positive or negative, to our net value from our ϕ-shares (as opposed to their prices); for every share we owe, there is a matching share that we hold. So if we buy low and sell high, we have made a profit off of the price differential, and once the inter-temporal arbitrage is complete we have not taken on any net risk from our stock holdings.

The fact that we can accrue stock holdings that we are guaranteed to eventually sell back for more than their purchase price is not sufficient to exploit the market. It may be the case that at every time n we spend \$1 on some $\mathbb{R}$-combination that we eventually sell back at \$2, but not until time $4n$. (That is, until time $4n$, the price remains low.) Then at every time n we owe $-\$n$ in cash, but only have around \$2$(n/4)$ worth of cash from shares we have sold off, for a net value of around $-n/2$. Thus we have net value unbounded below and hence do not exploit the market, despite the fact that each individual investment we make is eventually guaranteed to be profitable.

To avoid this obstacle, we will apply the ε-return on investment lemma (Lemma B.1.3) to the sequence of traders $(\overline{T}^k)_k$ that enforce the inequality at time k as described above. That is, $\overline{T}^k$ myopically "keeps $\overline{\mathbb{P}}$ sensible about $\overline{A}$" at time k by buying the $\mathbb{R}$-combination A_k described above if that $\mathbb{R}$-combination is under-priced at time k, and otherwise $\overline{T}^k$ does nothing. The ROI Lemma guarantees that the inequality cannot infinitely often fail substantially, or else this sequence would have δ-ROI for some δ.

The main technical difficulty is that we have to buy the $\mathbb{R}$-combination A_n at time n (if it is underpriced), wait for the price of the combination to increase substantially, and then sell it off, possibly over multiple time steps. The traders $\overline{T}^k$ will therefore have to track what fraction of their initial investment they have sold off at any given time.

Proof.

Proof. We show the first equality; the second equality follows from the first by considering the negated sequence $(-A_n)_n$.

Since for all n we have $\sup_{m \geq n} \mathbb{P}_m(A_n) \geq \mathbb{P}_n(A_n)$, the corresponding inequality in the limit infimum is immediate.

Suppose for contradiction that the other inequality doesn't hold, so that

$$\liminf_{n \to \infty} \mathbb{P}_n(A_n) < \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{P}_m(A_n) \ .$$

Then there are rational numbers $\varepsilon > 0$ and b such that we have

$$\liminf_{n \to \infty} \mathbb{P}_n(A_n) < b - \varepsilon < b + \varepsilon < \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{P}_m(A_n) \ .$$

Therefore we can fix some sufficiently large s_ε such that:

- for all $n > s_\varepsilon$, we have $\sup_{m \geq n} \mathbb{P}_m(A_n) > b + \varepsilon$, and

- for infinitely many $n > s_\varepsilon$, we have $\mathbb{P}_n(A_n) < b - \varepsilon$.

We will assume without loss of generality that each $\|A_n\|_{\mathrm{mg}} \leq 1$; they are assumed to be bounded, so they can be scaled down appropriately.

An efficiently emulatable sequence of traders. Let $\overline{A^\dagger}$ be an $\mathcal{EF}$-combination progression such that $A_n^\dagger(\overline{\mathbb{P}}) = A_n$ for all n. We now define our sequence of traders $(\overline{T}^k)_k$. For $k \leq s_\varepsilon$, define $\overline{T}^k$ to be the zero trading strategy at all times n.

For $k > s_\varepsilon$, define T_n^k to be the zero trading strategy for $n < k$, and define T_k^k to be the trading strategy

$$T_k^k := \mathrm{Under}_k \cdot \left(A_k^\dagger - A_k^{\dagger * k} \right) \ ,$$

where

$$\mathrm{Under}_k := \mathrm{Ind}_{\varepsilon/2} \left(A_k^{\dagger * k} < b - \varepsilon/2 \right) \ .$$

This is a buy order for the $\mathbb{R}$-combination A_k, scaled down by the continuous indicator function Under_k for the event that $\mathbb{P}_k$ has underpriced that $\mathbb{R}$-combination at time k. Then, for times $n > k$, we define $\overline{T}^k$ to submit the trading strategy

$$T_n^k := -F_n \cdot \left(\mathrm{Under}_k \cdot \left(A_k^\dagger - A_k^{\dagger * k} \right) \right) \ ,$$

where we define $F_n \geq 0$ recursively in the previous fractions F_i:

$$F_n := \mathrm{Over}_n^k \cdot \left(1 - \sum_{k < i < n} F_i \right) ,$$

using the continuous indicator $\mathrm{Over}_n^k := \mathrm{Ind}_{\varepsilon/2} \left(A_k^{\dagger * n} > b + \varepsilon/2 \right)$ of the $\mathbb{R}$-combination being overpriced at time n.

In words, T_n^k is a sell order for the $\mathbb{R}$-combination A_k, scaled down by the fraction Under_k of this $\mathbb{R}$-combination that $\overline{T}^k$ purchased at time k, and also scaled down by the fraction F_n of the original purchase T_k^k that will be sold on this time step. That is, $\sum_{k < i < n} F_i$ the total fraction of the original purchase $\mathrm{Under}_k \cdot A_k$ that has already been sold off on all previous rounds since time k. Then T_n^k sells off the remaining fraction $1 - \sum_{k < i < n} F_i$ of the $\mathbb{R}$-combination $\mathrm{Under}_k \cdot A_k$, scaled down by the extent Over_n^k to which A_k is overpriced at time n.

Notice that since $\mathrm{Over}_i^k \in [0, 1]$ for all i, by induction on n we have that $\sum_{k < i \leq n} F_i \leq 1$ and $F_n \geq 0$. This justifies thinking of the F_i as portions of the original purchase being sold off.

By assumption, the $\mathcal{EF}$-combination progression $\overline{A^\dagger}$ is e.c. Also, each trader $\overline{T}^k$ does not trade before time k. Therefore the sequence of traders $(\overline{T}^k)_k$ is efficiently emulatable (see A.2 on dynamic programming). (The constant s_ε before which the $\overline{T}^{k \leq s_\varepsilon}$ make no trades can be hard-coded in the efficient enumeration.)

($\varepsilon/2$) return on investment for $\overline{T}^k$. Now we show that each $\overline{T}^k$ has ($\varepsilon/2$)-ROI; i.e., for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\lim_{n \to \infty} \mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}})\right) \geq (\varepsilon/2)\|\overline{T}^k(\overline{\mathbb{P}})\|_{\text{mg}}.$$

In words, this says that the trades made by $\overline{T}^k$ across all time are valued positively in any $\mathbb{W} \in \mathcal{PC}(\Gamma)$, by a fixed fraction ($\varepsilon/2$) of the magnitude of $\overline{T}^k$. For $k \leq s_\varepsilon$, this is immediate since $\|\overline{T}^k(\overline{\mathbb{P}})\|_{\text{mg}} = 0$ by definition.

For each $k > s_\varepsilon$, by definition $\overline{T}^k$ makes a trade of magnitude $\|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\text{mg}} = \text{Under}_k(\overline{\mathbb{P}}) \cdot \|A_k\|_{\text{mg}}$, followed by trades of magnitude

$$\sum_{n>k} F_n \|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\text{mg}} \leq \|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\text{mg}} \ ,$$

by the earlier comment that the F_n are non-negative and sum to at most 1. Furthermore, by assumption, there is some $m > k$ such that $\mathbb{P}_m(A_k) > b + \varepsilon$. At that point, $\text{Over}_m^k(\overline{\mathbb{P}}) = 1$, so that $F_m(\overline{\mathbb{P}}) = \left(1 - \sum_{k<i<m} F_i(\overline{\mathbb{P}})\right)$; intuitively this implies that at time m, $\overline{T}^k$ will sell off the last of its stock holdings from trades in A_k. Formally we have

$$\sum_{k<i\leq m} \|\overline{T}_i^k(\overline{\mathbb{P}})\|_{\text{mg}} = \left(F_m(\overline{\mathbb{P}}) + \sum_{k<i<m} F_i(\overline{\mathbb{P}})\right) \cdot \text{Under}_k(\overline{\mathbb{P}}) \cdot \|A_k\|_{\text{mg}}$$

$$= \|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\text{mg}} \ .$$

Furthermore, for all times $M > m$ we have $F_M(\overline{\mathbb{P}}) = 0$, so that $T_M^k(\overline{\mathbb{P}}) \equiv 0$. Therefore $\|\overline{T}^k(\overline{\mathbb{P}})\|_{\text{mg}} = \sum_{k \leq i \leq n} \|\overline{T}_i^k(\overline{\mathbb{P}})\|_{\text{mg}} = 2\|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\text{mg}}$.

Now fix any world $\mathbb{W} \in \mathcal{PC}(\Gamma)$. Then the limiting value of $\overline{T}^k$ in $\mathbb{W}$ is:

$$\lim_{n \to \infty} \mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}})\right) = \mathbb{W}\left(\sum_{k \leq i \leq m} T_i^k(\overline{\mathbb{P}})\right)$$

since by the above analysis, T_i^k is the zero trade for $i < k$ and for $i > m$;

$$= \mathbb{W}\left(T_k^k(\overline{\mathbb{P}}) + \sum_{k<i\leq m} T_i^k(\overline{\mathbb{P}})\right)$$

$$= \text{Under}_k(\overline{\mathbb{P}}) \cdot \mathbb{W}(A_k - \mathbb{P}_k(A_k))$$

$$+ \text{Under}_k(\overline{\mathbb{P}}) \cdot \mathbb{W}\left(\sum_{k<i\leq m}(-F_i(\overline{\mathbb{P}})) \cdot (A_k - \mathbb{P}_i(A_k))\right)$$

by linearity, by the definition of the trader $\overline{T}^k$, and since by definition $A_k^{\dagger *k}(\overline{\mathbb{P}}) = \mathbb{P}_k(A_k^\dagger(\overline{\mathbb{P}})) = \mathbb{P}_k(A_k)$. Note that the prices $\mathbb{P}_i(A_k)$ of A_k in the summation change with the time step i. Then

$$= \text{Under}_k(\overline{\mathbb{P}}) \cdot \mathbb{W}(A_k - A_k)$$

$$+ \text{Under}_k(\overline{\mathbb{P}}) \cdot \mathbb{W}\left(-\mathbb{P}_k(A_k) + \sum_{k<i\leq m} F_i \cdot \mathbb{P}_i(A_k)\right)$$

$$\lim_{n \to \infty} \mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}})\right) = \text{Under}_k(\overline{\mathbb{P}}) \cdot \left(-\mathbb{P}_k(A_k) + \sum_{k<i\leq m} F_i \cdot \mathbb{P}_i(A_k)\right) \ ,$$

using linearity to rearrange the terms in the first and second lines, and using that $\sum_{k<i\leq m} F_i = 1$ as shown above. Note that this last quantity does not contain any stock holdings whose value depends on the world; intuitively this is because $\overline{T}^k$ sold off exactly all of its initial purchase. The remaining quantity is the difference between the price at which the $\mathbb{R}$-combination A_k was bought and the prices at which it was sold over time, scaled down by the fraction $\text{Under}_k(\overline{\mathbb{P}})$ of A_k that $\overline{T}^k$ purchased at time k.

If at time step k the $\mathbb{R}$-combination A_k was not underpriced, i.e., $\mathrm{Under}_k(\overline{\mathbb{P}}) = 0$, then
$$\lim_{n\to\infty} \mathbb{W}\left(\sum_{i\le n} T_i^k(\overline{\mathbb{P}})\right) = 0 = (\varepsilon/2)\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} \ ,$$

as desired. On the other hand, suppose that $\mathrm{Under}_k(\overline{\mathbb{P}}) > 0$. That is,
$$\mathrm{Ind}_{\varepsilon/2}\left(\mathbb{P}_k(A_k) < b - \varepsilon/2\right) > 0 \ ,$$

i.e., A_k was actually underpriced at time k. Therefore

$$\lim_{n\to\infty} \mathbb{W}\left(\sum_{i\le n} T_i^k(\overline{\mathbb{P}})\right) \ge \mathrm{Under}_k(\overline{\mathbb{P}}) \cdot \left(-(b-\varepsilon/2) + \sum_{k<i\le m} F_i(\overline{\mathbb{P}}) \cdot (b+\varepsilon/2)\right)$$

since when $F_i(\overline{\mathbb{P}}) > 0$ we have $\mathrm{Over}_n^k(\overline{\mathbb{P}}) > 0$ and hence $\mathbb{P}_i(A_k) \ge b + \varepsilon/2$, and using the fact that $\mathrm{Under}_k(\overline{\mathbb{P}})$ is nonnegative;
$$= \mathrm{Under}_k(\overline{\mathbb{P}}) \cdot \varepsilon$$
$$\ge (\varepsilon/2)\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} \ ,$$

since

$$\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = 2\|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = 2 \cdot \mathrm{Under}_k(\overline{\mathbb{P}}) \cdot \|A_k\|_{\mathrm{mg}} \le 2 \cdot \mathrm{Under}_k(\overline{\mathbb{P}})$$

Thus $\overline{T}^k$ has $(\varepsilon/2)$-ROI.

Deriving a contradiction. We have shown that the sequence of traders $(\overline{T}^k)_k$ is bounded, efficiently emulatable, and has $(\varepsilon/2)$-return on investment. The remaining condition to Lemma B.1.3 states that for all k, the magnitude $\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}}$ of all trades made by $\overline{T}^k$ must equal α_k for some $\overline{\mathbb{P}}$-generable $\overline{\alpha_k}$. This condition is satisfied for $\alpha_k := 2\|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}}$, since as shown above, $\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = 2\|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}}$.

Therefore we can apply Lemma B.1.3 (the ROI lemma) to the sequence of traders $(\overline{T}^k)_k$. We conclude that $\alpha_k \eqsim_k 0$. Recall that we supposed by way of contradiction that the $\mathbb{R}$-combinations in $\overline{A_k}$ are underpriced infinitely often. That is, for infinitely many days k, $\mathbb{P}_k(A_k) < b - \varepsilon$. But for any such $k > s_\varepsilon$, $\overline{T}_k^k$ purchases a full $\mathbb{R}$-combination A_k, and then sells off the resulting stock holdings for at least $b + \varepsilon/2$, at which point $\overline{T}^k$ has profited by at least ε. More precisely, for these k we have

$$\mathrm{Under}_k(\overline{\mathbb{P}}) = \mathrm{Ind}_{\varepsilon/2}\left(\mathbb{P}_k(A_k) < b - \varepsilon/2\right) = 1.$$

Since
$$\alpha_k = 2\|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = 2 \cdot \mathrm{Under}_k(\overline{\mathbb{P}}) \cdot \|A_k\|_{\mathrm{mg}} = 2\|A_k\|_{\mathrm{mg}}$$

and $\|A_k\|_{\mathrm{mg}} \ge \varepsilon/2$ (since $\mathbb{P}_m(A_k) - \mathbb{P}_k(A_k) \ge \varepsilon$ for some m), we have $\alpha_k \ge \varepsilon$ for infinitely many k, which contradicts $\alpha_k \eqsim_k 0$. $\qquad\square$

B.3 Preemptive Learning

Theorem 4.2.4 (Preemptive Learning). *Let $\overline{\phi}$ be an e.c. sequence of sentences. Then*
$$\liminf_{n\to\infty} \mathbb{P}_n(\phi_n) = \liminf_{n\to\infty} \sup_{m\ge n} \mathbb{P}_m(\phi_n).$$

Furthermore,
$$\limsup_{n\to\infty} \mathbb{P}_n(\phi_n) = \limsup_{n\to\infty} \inf_{m\ge n} \mathbb{P}_m(\phi_n).$$

Proof. This is a special case of Theorem 4.5.7 (Affine Preemptive Learning), using the combination $A_n := \phi_n$. $\qquad\square$

B.4 Convergence

Theorem 4.1.1 (Convergence). *The limit $\mathbb{P}_\infty : \mathcal{S} \to [0, 1]$ defined by*

$$\mathbb{P}_\infty(\phi) := \lim_{n \to \infty} \mathbb{P}_n(\phi)$$

exists for all ϕ.

Proof. By Theorem 4.2.4 (Preemptive Learning),

$$\liminf_{n \to \infty} \mathbb{P}_n(\phi) = \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{P}_m(\phi)$$
$$= \lim_{k \to \infty} \inf_{n \geq k} \sup_{m \geq n} \mathbb{P}_m(\phi)$$
$$= \limsup_{n \to \infty} \mathbb{P}_n(\phi).$$

Since the $\liminf$ and $\limsup$ of $\mathbb{P}_n(\phi)$ are equal, the limit exists. $\qquad\square$

B.5 Persistence of Affine Knowledge

Let $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$. Then

$$\liminf_{n \to \infty} \inf_{m \geq n} \mathbb{P}_m(A_n) = \liminf_{n \to \infty} \mathbb{P}_\infty(A_n)$$

and

$$\limsup_{n \to \infty} \sup_{m \geq n} \mathbb{P}_m(A_n) = \limsup_{n \to \infty} \mathbb{P}_\infty(A_n).$$

Proof strategy: keeping $\mathbb{P}_m$ reasonable on all $\mathcal{AF}_{n \leq m}$. In the same vein as the proof in Appendix B.2 of Theorem 4.5.7, the inequality

$$\liminf_{n \to \infty} \inf_{m \geq n} \mathbb{P}_m(A_n) \geq \liminf_{n \to \infty} \mathbb{P}_\infty(A_n)$$

says roughly that $\mathbb{P}_m$ cannot underprice the $\mathbb{R}$-combination A_n by a substantial amount infinitely often, where the $\mathbb{R}$-combination is "underpriced" in comparison to the value of the $\mathbb{R}$-combination as judged by the limiting belief state $\mathbb{P}_\infty$.

As the proof of the present theorem is quite similar to the proof of Theorem 4.5.7, we will highlight the differences in this proof, and otherwise give a relatively terse proof.

Intuitively, if the market $\overline{\mathbb{P}}$ did not satisfy the present inequality then $\overline{\mathbb{P}}$ would be exploitable by a trader that buys A_n at any time m such that its price $\mathbb{P}_m(A_n)$ is lower than its eventual price, and then sells back the $\mathbb{R}$-combination when the price rises.

We would like to apply the return on investment lemma (Lemma B.1.3) as in Theorem 4.5.7. One natural attempt is to have, for each n, a trader for that watches the price of A_n at all times $m \geq n$, buying low and selling high. This proof strategy may be feasible, but does not follow straightforwardly from the ROI lemma: those traders may be required to make multiple purchases of A_n in order to guard against their prices *ever* dipping too low. This pattern of trading may violate the condition for applying the ROI lemma that requires traders to have a total trading volume that is predictable by a $\overline{\mathbb{P}}$-generable $\mathcal{EF}$-progression (in order to enable verifiable budgeting).

Thus we find it easier to index our traders by the time rather than by A_n. That is, we will define a sequence of traders $(\overline{T}^k)_k$, where the trader $\overline{T}^k$ ensures that $\mathbb{P}_k$ does not assign too low a price to any A_n for $n \leq k$. Specifically, $\overline{T}^k$ at time k buys any $\mathbb{R}$-combination A_n for $n \leq k$ with $\mathbb{P}_k(A_n)$ sufficiently low, and then sells back each such purchase as the price $\mathbb{P}_m(A_n)$ rises. In this way, if $\mathbb{R}$-combinations are ever underpriced at any time above the main diagonal, there is a trader ready to buy that $\mathbb{R}$-combination in full.

Proof.

Proof. We show the first equality; the second equality follows from the first by considering the negated progression $(-A_n)_n$.

For every n, since $\mathbb{P}_\infty$ is the limit of the $\mathbb{P}_m$ and since $\mathbb{V}(A_n)$ is continuous as a function of the valuation $\mathbb{V}$, we have that $\inf_{m\geq n}\mathbb{P}_m(A_n) \leq \mathbb{P}_\infty(A_n)$. Therefore the corresponding inequality in the limit infimum is immediate.

Suppose for contradiction that the other inequality doesn't hold, so that

$$\liminf_{n\to\infty}\inf_{m\geq n}\mathbb{P}_m(A_n) < \liminf_{n\to\infty}\mathbb{P}_\infty(A_n).$$

Then there are rational numbers $\varepsilon > 0$ and b such that we have

$$\liminf_{n\to\infty}\inf_{m\geq n}\mathbb{P}_m(A_n) < b - \varepsilon < b + \varepsilon < \liminf_{n\to\infty}\mathbb{P}_\infty(A_n) \ ,$$

and therefore we can fix some sufficiently large s_ε such that

- for all $n > s_\varepsilon$, we have $\mathbb{P}_\infty(A_n) > b + \varepsilon$, and

- for infinitely many $n > s_\varepsilon$, we have $\inf_{m\geq n}\mathbb{P}_m(A_n) < b - \varepsilon$.

We will assume without loss of generality that each $\|A_n\|_{\mathrm{mg}} \leq 1$; they are assumed to be bounded, so they can be scaled down appropriately.

An efficiently emulatable sequence of traders.   Now we define our sequence of traders $(\overline{T}^k)_k$. Let $\overline{A^\dagger}$ be an $\mathcal{EF}$-combination progression such that $A^\dagger{}_n(\overline{\mathbb{P}}) = A_n$ for all n. For $n < k$, define T_n^k to be the zero trading strategy. Define T_k^k to be the trading strategy

$$T_k^k := \sum_{s_\varepsilon < n \leq k} \left(\mathrm{Under}_k^n \cdot \left(A_n^\dagger - A_n^{\dagger * k}\right)\right) \ ,$$

where

$$\mathrm{Under}_k^n := \mathrm{Ind}_{\varepsilon/2}\left(A_n^{\dagger * k} < b - \varepsilon/2\right) \ .$$

This is a buy order for each $\mathbb{R}$-combination A_n for $s_\varepsilon < n \leq k$, scaled down by the continuous indicator function Under_k^n for the event that A_n is underpriced at time k by $\overline{\mathbb{P}}$. Then, for time steps $m > k$, we define T_m^k to be the trading strategy

$$T_m^k := F_m \cdot \sum_{s_\varepsilon < n \leq k} \left(-\mathrm{Under}_k^n \cdot \left(A_n^\dagger - A_n^{\dagger * m}\right)\right) \ ,$$

where

$$F_m := \left(1 - \sum_{k < i < m} F_i\right) \cdot \prod_{s_\varepsilon < n \leq k} \mathrm{Over}_m^n$$

and

$$\mathrm{Over}_m^n := \mathrm{Ind}_{\varepsilon/2}\left(A_n^{\dagger * m} > b + \varepsilon/2\right) \ .$$

This trade is a sell order for the entire $\mathbb{R}$-combination comprising the sum of the scaled $\mathbb{R}$-combinations $\mathrm{Under}_k^n(\overline{\mathbb{P}}) \cdot A_n$ for $s_\varepsilon < n \leq k$ purchased at time k by T_k^k, scaled down by the fraction $F_m(\overline{\mathbb{P}})$. We define F_m so that it represents the fraction of the original purchase made by the trader $\overline{T}^k$ that has not yet been sold off by time m, scaled down by the continuous indicator $\prod_{s_\varepsilon < n \leq k}\mathrm{Over}_m^n$ for the event that all of those $\mathbb{R}$-combinations A_n for $s_\varepsilon < n \leq k$ are overpriced at time m.

By assumption, the $\mathcal{EF}$-combination progression $\overline{A^\dagger}$ is e.c., and each trader $\overline{T}^k$ does not trade before time k. Therefore the sequence of traders $(\overline{T}^k)_k$ is efficiently emulatable (see A.2 on dynamic programming).

$(\varepsilon/2)$ return on investment for $\overline{T}^k$. Now we show that each $\overline{T}^k$ has $(\varepsilon/2)$-ROI, i.e., for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$:

$$\lim_{n \to \infty} \mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}})\right) \geq (\varepsilon/2)\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} \ .$$

Roughly speaking, $\overline{T}^k$ gets $(\varepsilon/2)$-ROI for the same reason as the traders in the proof of Theorem 4.5.7: the stock holdings from each A_n that $\overline{T}^k$ purchased will be sold off for at least $(\varepsilon/2)$-ROI, so the sum of the $\mathbb{R}$-combinations is sold off for $(\varepsilon/2)$-ROI.

Since $\mathrm{Over}_i^n(\overline{\mathbb{P}}) \in [0,1]$ for all n and i, by induction on m we have that $\sum_{k < i \leq m} F_i(\overline{\mathbb{P}}) \leq 1$ and $F_m(\overline{\mathbb{P}}) \geq 0$. Therefore for each $k > s_\varepsilon$, by definition $\overline{T}^k$ makes a trade of magnitude $\|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = \sum_{s_\varepsilon < n \leq k} \mathrm{Under}_k^n(\overline{\mathbb{P}}) \cdot \|A_n\|_{\mathrm{mg}}$, followed by trades of magnitude

$$\sum_{n > k} F_n(\overline{\mathbb{P}})\|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} \leq \|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} \ .$$

By assumption, there is some time m such that $\mathbb{P}_m(A_n) > b + \varepsilon/2$ for all $s_\varepsilon < n \leq k$. At that point, $\mathrm{Over}_m^n(\overline{\mathbb{P}}) = 1$ for each such n, so that $F_n(\overline{\mathbb{P}}) = \left(1 - \sum_{k < i < n} F_i(\overline{\mathbb{P}})\right)$. Then at time m, $\overline{T}^k$ will sell of the last of its stock holdings from trades in A_n, so that $\sum_{k < i \leq m} \|\overline{T}_i^k(\overline{\mathbb{P}})\|_{\mathrm{mg}}$ is equal to

$$\left(F_m(\overline{\mathbb{P}}) + \sum_{k < i < m} F_i(\overline{\mathbb{P}})\right) \cdot \sum_{s_\varepsilon < n \leq k} \mathrm{Under}_k^n(\overline{\mathbb{P}}) \cdot \|A_n\|_{\mathrm{mg}} = \|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} \ .$$

Furthermore, for all times $M > m$ we have $F_M(\overline{\mathbb{P}}) = 0$, so that $T_M^k(\overline{\mathbb{P}}) \equiv 0$. Therefore $\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = \sum_{k \leq i \leq n} \|\overline{T}_i^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = 2\|\overline{T}_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}}$.

From this point, the proof of return on investment is essentially identical to the analogous proof of Theorem 4.5.7. The only difference is that here the trader $\overline{T}^k$ holds a combination of $\mathbb{R}$-combinations. Therefore we will not belabor the details; inserting a summation $\sum_{s_\varepsilon < n \leq k}$ in front of the trades made by the traders in the proof of Theorem 4.5.7 will produce the precise derivation.

In short, since $\overline{T}^k$ will eventually hold no net shares, the value of its holdings is determined by the prices of the shares it trades, regardless of plausible worlds. By definition, $\overline{T}^k$ purchases a mixture of $\mathbb{R}$-combinations

$$\sum_{s_\varepsilon < n \leq k} \left(\mathrm{Under}_k^n(\overline{\mathbb{P}}) \cdot A_n\right),$$

where each A_n with $\mathrm{Under}_k^n(\overline{\mathbb{P}}) > 0$ has price $\mathbb{P}_k(A_n)$ at most $b - \varepsilon/2$ at time k. Then $\overline{T}^k$ sells off that mixture, at times for which each $\mathbb{R}$-combination has price at least $b + \varepsilon/2$. Thus $\overline{T}^k$ eventually has holdings with value at least

$$\sum_{s_\varepsilon < n \leq k} \left(\mathrm{Under}_k^n(\overline{\mathbb{P}}) \cdot (b + \varepsilon/2 - (b - \varepsilon/2))\right)$$
$$= \sum_{s_\varepsilon < n \leq k} \left(\mathrm{Under}_k^n(\overline{\mathbb{P}}) \cdot \varepsilon\right)$$
$$\geq \varepsilon \sum_{s_\varepsilon < n \leq k} \left(\mathrm{Under}_k^n(\overline{\mathbb{P}}) \cdot |A_n|\right)$$
$$\geq \varepsilon\|T_k^k(\overline{\mathbb{P}})\|_{\mathrm{mg}}$$
$$= (\varepsilon/2)\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} \ .$$

Thus, $\overline{T}^k$ has $(\varepsilon/2)$-ROI.

Deriving a contradiction. We have shown that the sequence of traders $(\overline{T}^k)_k$ is bounded, efficiently emulatable, and has $(\varepsilon/2)$-return on investment. The remaining condition to Lemma B.1.3 states that for all k, the magnitude $\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}}$ of all trades made by $\overline{T}^k$ must equal α_k for some $\overline{\mathbb{P}}$-generable $\overline{\alpha_k}$. This condition is satisfied for $\alpha_k := 2\|\overline{T}^k_k(\overline{\mathbb{P}})\|_{\mathrm{mg}}$, since as shown above, $\|\overline{T}^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = 2\|\overline{T}^k_k(\overline{\mathbb{P}})\|_{\mathrm{mg}}$.

Therefore we can apply Lemma B.1.3 (the ROI lemma) to the sequence of traders $(\overline{T}^k)_k$. We conclude that $\alpha_k \eqsim_k 0$. Recall that we supposed by way of contradiction that infinitely often, some A_n is underpriced. That is, for infinitely many times k and indices $s_\varepsilon < n \le k$, $\mathbb{P}_k(A_n) < b - \varepsilon$.

But for any such k and n, T^k_k will purchase the full $\mathbb{R}$-combination A_n, as

$$\mathrm{Under}^n_k(\overline{\mathbb{P}}) = \mathrm{Ind}_{\varepsilon/2}\left(\mathbb{P}_k(A_n) < b - \varepsilon/2\right) = 1 \ .$$

Now $\alpha_k = 2\|\overline{T}^k_k\|_{\mathrm{mg}} \ge 2 \cdot \mathrm{Under}^n_k\|A_n\|_{\mathrm{mg}} = 2\|A_n\|_{\mathrm{mg}}$, and $\|A_n\|_{\mathrm{mg}} \ge \varepsilon/2$ (since $\mathbb{P}_m(A_n) - \mathbb{P}_k(A_n) \ge \varepsilon$ for some m). So $\alpha_k \ge \varepsilon$ infinitely often, contradicting $\alpha_k \eqsim_k 0$. $\qquad\square$

B.6 Persistence of Knowledge

Theorem 4.2.3 (Persistence of Knowledge)**.** *Let $\overline{\phi}$ be an e.c. sequence of sentences, and $\overline{p}$ be an e.c. sequence of rational-number probabilities. If $\mathbb{P}_\infty(\phi_n) \eqsim_n p_n$, then*

$$\sup_{m \ge n} |\mathbb{P}_m(\phi_n) - p_n| \eqsim_n 0.$$

Furthermore, if $\mathbb{P}_\infty(\phi_n) \lesssim_n p_n$, then

$$\sup_{m \ge n} \mathbb{P}_m(\phi_n) \lesssim_n p_n,$$

and if $\mathbb{P}_\infty(\phi_n) \gtrsim_n p_n$, then

$$\inf_{m \ge n} \mathbb{P}_m(\phi_n) \gtrsim_n p_n.$$

Proof. The second and third statements are a special case of Theorem 4.5.6 (Persistence of Affine Knowledge), using the combination $A_n := \phi_n$; the first statement follows from the second and third. $\qquad\square$

C Coherence Proofs

C.1 Affine Coherence

Theorem 4.5.5 (Affine Coherence)**.** *Let $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(A_n) \le \liminf_{n \to \infty} \mathbb{P}_\infty(A_n) \le \liminf_{n \to \infty} \mathbb{P}_n(A_n),$$

and

$$\limsup_{n \to \infty} \mathbb{P}_n(A_n) \le \limsup_{n \to \infty} \mathbb{P}_\infty(A_n) \le \limsup_{n \to \infty} \sup_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(A_n).$$

Proof. We show the first series of inequalities; the second series follows from the first by considering the negated progression $(-A_n)_n$. Let $\overline{A^\dagger}$ be an $\mathcal{EF}$-combination progression such that $A^\dagger{}_n(\overline{\mathbb{P}}) = A_n$ for all n.

Connecting $\mathcal{PC}(\Gamma)$ to $\mathbb{P}_\infty$. First we show that

$$\liminf_{n\to\infty} \inf_{\mathbb{W}\in\mathcal{PC}(\Gamma)} \mathbb{W}(A_n) \le \liminf_{n\to\infty} \mathbb{P}_\infty(A_n).$$

It suffices to show the stronger statement that for any $n \in \mathbb{N}^+$,

$$\inf_{\mathbb{W}\in\mathcal{PC}(\Gamma)} \mathbb{W}(A_n) \le \mathbb{P}_\infty(A_n).$$

This is a generalization of coherence in the limit to affine relationships; its proof will follow a strategy essentially identical to the one used in the proof of Theorem 4.1 (coherence) to show the particular coherence relationships that are sufficient to imply ordinary probabilistic coherence. That is, we will construct a trader that waits for the coherence relationship to approximately hold (so to speak) and for the price of the corresponding $\mathbb{R}$-combination to approximately converge, and then buys the combination repeatedly if it is underpriced.

Suppose by way of contradiction that the inequality does not hold, so for some fixed n there are rational numbers $\varepsilon > 0$ and b and a time step s_ε such that for all $m > s_\varepsilon$ we have

$$\mathbb{P}_m(A_n) < b - \varepsilon < b + \varepsilon < \inf_{\mathbb{W}\in\mathcal{PC}(D_m)} \mathbb{W}(A_n) \ .$$

Therefore we can define a trader $\overline{T}$ that waits until time s_ε, and thereafter buys a full $\mathbb{R}$-combination A_n on every time step. That is, we take T_m to be the zero trading strategy for $m \le s_\varepsilon$, and we define T_m for $m > s_\varepsilon$ to be

$$T_m := A_n^\dagger - A_n^{\dagger * m}.$$

Intuitively, since the infimum over plausible worlds of the value of the stocks in this $\mathbb{R}$-combination is already substantially higher than its price, the value of the total holdings of our trader $\overline{T}$ immediately increases by at least 2ε. More formally, we have that for any time m and any $\mathbb{W} \in \mathcal{PC}(D_m)$,

$$\mathbb{W}\left(\sum_{i\le m} T_i(\overline{\mathbb{P}})\right) = \mathbb{W}\left(\sum_{s_\varepsilon < i \le m} T_i(\overline{\mathbb{P}})\right)$$

since $T_m \equiv 0$ for $m \le s_\varepsilon$;

$$= \sum_{s_\varepsilon < i \le m} \mathbb{W}(A_n) - \mathbb{P}_i(A_n)$$

by linearity, by definition of T_i, and since $A_n^\dagger(\overline{\mathbb{P}}) \equiv A_n$ and $A_n^{\dagger * m}(\overline{\mathbb{P}}) \equiv \mathbb{P}_m(A_n)$;

$$\ge \sum_{s_\varepsilon < i \le m} b + \varepsilon - (b - \varepsilon)$$

$$= 2\varepsilon(m - s_\varepsilon).$$

This is bounded below by 0 and unbounded above as m goes to ∞. Thus $\overline{T}$ exploits the market $\overline{\mathbb{P}}$, contradicting that $\overline{\mathbb{P}}$ is a logical inductor. Therefore in fact we must have

$$\liminf_{n\to\infty} \inf_{\mathbb{W}\in\mathcal{PC}(\Gamma)} \mathbb{W}(A_n(\overline{\mathbb{P}})) \le \liminf_{n\to\infty} \mathbb{P}_\infty(A_n(\overline{\mathbb{P}})),$$

as desired.

Connecting $\mathbb{P}_n$ to $\mathbb{P}_\infty$ and to fast diagonals. Now we show that

$$\liminf_{n\to\infty} \mathbb{P}_\infty(A_n) \le \liminf_{n\to\infty} \mathbb{P}_n(A_n).$$

This says, roughly speaking, that affine relationships that hold in the limiting belief state $\mathbb{P}_\infty$ also hold along the main diagonal. We show this inequality in two steps. First, by Theorem 4.5.6 (Persistence of affine knowledege), we have

$$\liminf_{n\to\infty} \mathbb{P}_\infty(A_n) \leq \liminf_{n\to\infty} \inf_{m\geq n} \mathbb{P}_m(A_n).$$

This says roughly that if the limiting beliefs end up satisfying some sequence of affine relationships, then eventually all belief states above the main diagonal satisfy that relationship to at least the same extent. Second, it is immediate that

$$\liminf_{n\to\infty} \inf_{m\geq n} \mathbb{P}_m(A_n) \leq \liminf_{n\to\infty} \mathbb{P}_n(A_n),$$

since for all n, $\inf_{m\geq n} \mathbb{P}_m(A_m) \leq \mathbb{P}_n(A_n)$. Thus we have the desired inequality. $\square$

C.2 Affine Provability Induction

Theorem 4.5.4 (Affine Provability Induction)**.** *Let $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$ and $b \in \mathbb{R}$. If, for all consistent worlds $\mathbb{W} \in \mathcal{PC}(\Gamma)$ and all $n \in \mathbb{N}^+$, it is the case that $\mathbb{W}(A_n) \geq b$, then*

$$\mathbb{P}_n(A_n) \gtrsim_n b,$$

and similarly for $=$ and $\eqsim_n$, and for $\leq$ and $\lesssim_n$.

Proof. We prove the statement in the case of $\geq$; the case of $\leq$ is analogous, and the case of $=$ follows from the other two cases. By Theorem 4.5.5 (Affine Coherence),

$$\liminf_{n\to\infty} \mathbb{P}_n(A_n) \geq \liminf_{n\to\infty} \inf_{\mathbb{W}\in\mathcal{PC}(\Gamma)} \mathbb{W}(A_n) \geq b.$$

$\square$

C.3 Provability Induction

Theorem 4.2.1 (Provability Induction)**.** *Let $\overline{\phi}$ be an e.c. sequence of theorems. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n 1.$$

Furthermore, let $\overline{\psi}$ be an e.c. sequence of disprovable sentences. Then

$$\mathbb{P}_n(\psi_n) \eqsim_n 0.$$

Proof. Since $\overline{\phi}$ is a sequence of theorems, for all n and $\mathbb{W} \in \mathcal{PC}(\Gamma)$, $\mathbb{W}(\phi_n) = 1$. So by Theorem 4.5.4 (Affine Provability Induction),

$$\mathbb{P}_n(\phi_n) \eqsim_n 1.$$

Similarly, since $\overline{\psi}$ is a sequence of disprovable sentences, for all n and $\mathbb{W} \in \mathcal{PC}(\Gamma)$, $\mathbb{W}(\psi_n) = 0$. So by Theorem 4.5.4 (Affine Provability Induction),

$$\mathbb{P}_n(\psi_n) \eqsim_n 0.$$

$\square$

C.4 Belief in Finitistic Consistency

Theorem 4.9.2 (Belief in Finitistic Consistency)**.** *Let f be any computable function. Then*

$$\mathbb{P}_n(\mathrm{Con}(\Gamma)(\text{``}\underline{f(\underline{n})}\text{''})) \eqsim_n 1.$$

Proof. Since each statement $\mathrm{Con}(\Gamma)(\text{``}\underline{f(\underline{n})}\text{''})$ is computable and true, and Γ can represent computable functions, each of these statements is provable in Γ. Now apply Theorem 4.2.1 (Provability Induction) to get the desired property. $\square$

C.5   Belief in the Consistency of a Stronger Theory

Theorem 4.9.3 (Belief in the Consistency of a Stronger Theory)**.** *Let Γ' be any recursively axiomatizable consistent theory. Then*

$$\mathbb{P}_n(\mathrm{Con}(\Gamma')(\text{``}\underline{f(\underline{n})}\text{''})) \backsimeq_n 1.$$

Proof. Since each statement $\mathrm{Con}(\Gamma')(\text{``}\underline{f(\underline{n})}\text{''})$ is computable and true, and Γ can represent computable functions, each of these statements is provable in Γ. Now apply Theorem 4.2.1 (Provability Induction) to get the desired property. $\square$

C.6   Disbelief in Inconsistent Theories

Theorem 4.9.4 (Disbelief in Inconsistent Theories)**.** *Let $\overline{\Gamma'}$ be an e.c. sequence of recursively axiomatizable inconsistent theories. Then*

$$\mathbb{P}_n(\text{``}\underline{\Gamma'_{\underline{n}}}\text{ is inconsistent''}) \backsimeq_n 1,$$

so

$$\mathbb{P}_n(\text{``}\underline{\Gamma'_{\underline{n}}}\text{ is consistent''}) \backsimeq_n 0.$$

Proof. Since each statement "$\underline{\Gamma'_{\underline{n}}}$ is inconsistent" is provable in PA, and Γ can represent computable functions, each of these statements is provable in Γ. Now apply Theorem 4.2.1 (Provability Induction) to get the first desired property.

   Similarly, since each statement "$\underline{\Gamma'_{\underline{n}}}$ is consistent" is disprovable in PA, and Γ can represent computable functions, each of these statements is disprovable in Γ. Now apply Theorem 4.2.1 (Provability Induction) to get the second desired property. $\square$

C.7   Learning of Halting Patterns

Theorem 4.10.1 (Learning of Halting Patterns)**.** *Let $\overline{m}$ be an e.c. sequence of Turing machines, and $\overline{x}$ be an e.c. sequence of bitstrings, such that m_n halts on input x_n for all n. Then*
$$\mathbb{P}_n(\text{``}\underline{m_{\underline{n}}}\text{ halts on input }\underline{x_{\underline{n}}}\text{''}) \backsimeq_n 1.$$

Proof. Since each statement "$\underline{m_{\underline{n}}}$ halts on input $\underline{x_{\underline{n}}}$" is computable and true, and Γ can represent computable functions, each of these statements is provable in Γ. Now apply Theorem 4.2.1 (Provability Induction) to get the desired property. $\square$

C.8   Learning of Provable Non-Halting Patterns

Theorem 4.10.2 (Learning of Provable Non-Halting Patterns)**.** *Let $\overline{q}$ be an e.c. sequence of Turing machines, and $\overline{y}$ be an e.c. sequence of bitstrings, such that q_n provably fails to halt on input y_n for all n. Then*

$$\mathbb{P}_n(\text{``}\underline{q_{\underline{n}}}\text{ halts on input }\underline{y_{\underline{n}}}\text{''}) \backsimeq_n 0.$$

Proof. Each statement "$\underline{q_{\underline{n}}}$ halts on input $\underline{y_{\underline{n}}}$" is disprovable in Γ. Now apply Theorem 4.2.1 (Provability Induction) to get the desired property. $\square$

C.9   Learning not to Anticipate Halting

Theorem 4.10.3 (Learning not to Anticipate Halting)**.** *Let $\overline{q}$ be an e.c. sequence of Turing machines, and let $\overline{y}$ be an e.c. sequence of bitstrings, such that q_n does not halt on input y_n for any n. Let f be any computable function. Then*

$$\mathbb{P}_n(\text{``}\underline{q_{\underline{n}}}\text{ halts on input }\underline{y_{\underline{n}}}\text{ within }\underline{f(\underline{n})}\text{ steps''}) \backsimeq_n 0.$$

Proof. Since each statement "$\underline{q_{\underline{n}}}$ halts on input $\underline{y_{\underline{n}}}$ within $\underline{f(\underline{n})}$ steps" is computable and false, and Γ can represent computable functions, each of these statements is disprovable in Γ. Now apply Theorem 4.2.1 (Provability Induction) to get the desired property. $\square$

C.10   Limit Coherence

Theorem 4.1.2 (Limit Coherence). *$\mathbb{P}_\infty$ is coherent, i.e., it gives rise to an internally consistent probability measure* $\Pr$ *on the set $\mathcal{PC}(\Gamma)$ of all worlds consistent with Γ, defined by the formula*

$$\Pr(\mathbb{W}(\phi) = 1) := \mathbb{P}_\infty(\phi).$$

In particular, if Γ contains the axioms of first-order logic, then $\mathbb{P}_\infty$ defines a probability measure on the set of first-order completions of Γ.

Proof. The limit $\mathbb{P}_\infty(\phi)$ exists by Theorem 4.1.1 (Convergence), so $\mathbb{P}_\infty$ is well-defined. Gaifman (1964) shows that $\mathbb{P}_\infty$ defines a probability measure over $\mathcal{PC}(\Gamma)$ so long as the following three implications hold for all sentences ϕ an ψ:

- If $\Gamma \vdash \phi$, then $\mathbb{P}_\infty(\phi) = 1$,

- If $\Gamma \vdash \neg\phi$, then $\mathbb{P}_\infty(\phi) = 0$,

- If $\Gamma \vdash \neg(\phi \wedge \psi)$, then $\mathbb{P}_\infty(\phi \vee \psi) = \mathbb{P}_\infty(\phi) + \mathbb{P}_\infty(\psi)$.

Let us demonstrate each of these three properties.

- Assume that $\Gamma \vdash \phi$. By Theorem 4.2.1 (Provability Induction), $\mathbb{P}_\infty(\phi) = 1$.

- Assume that $\Gamma \vdash \neg\phi$. By Theorem 4.2.1 (Provability Induction), $\mathbb{P}_\infty(\phi) = 0$.

- Assume that $\Gamma \vdash \neg(\phi \wedge \psi)$. For all $\mathbb{W} \in \mathcal{PC}(\Gamma)$, $\mathbb{W}(\phi \vee \psi) = \mathbb{W}(\phi) + \mathbb{W}(\psi)$. So by Theorem 4.5.4 (Affine Provability Induction), $\mathbb{P}_\infty(\phi \vee \psi) = \mathbb{P}_\infty(\phi) + \mathbb{P}_\infty(\psi)$.

□

C.11   Learning Exclusive-Exhaustive Relationships

Theorem 4.5.1 (Learning Exclusive-Exhaustive Relationships). *Let $\overline{\phi^1}, \dots, \overline{\phi^k}$ be k e.c. sequences of sentences, such that for all n, Γ proves that $\phi_n^1, \dots, \phi_n^k$ are exclusive and exhaustive (i.e. exactly one of them is true). Then*

$$\mathbb{P}_n(\phi_n^1) + \cdots + \mathbb{P}_n(\phi_n^k) \eqsim_n 1.$$

Proof. Define $A_n := \phi_n^1 + \cdots + \phi_n^k$. Note that for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$, $\mathbb{W}(A_n) = 1$.
So by Theorem 4.5.4 (Affine Provability Induction) and linearity,

$$\mathbb{P}_n(\phi_n^1) + \cdots + \mathbb{P}_n(\phi_n^k) = \mathbb{P}_n(\phi_n^1 + \cdots + \phi_n^k) = \mathbb{P}_n(A_n) \eqsim_n 1.$$

□

D   Statistical Proofs

D.1   Affine Recurring Unbiasedness

Theorem 4.5.9 (Affine Recurring Unbiasedness). *If $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$ is determined via Γ, and $\overline{w}$ is a $\overline{\mathbb{P}}$-generable divergent weighting,*

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(A_i) - \mathrm{Val}_\Gamma(A_i))}{\sum_{i \leq n} w_i}$$

has 0 as a limit point. In particular, if it converges, it converges to 0.

Proof. Define

$$\mathrm{Bias}_n := \frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(A_i) - \mathrm{Val}_\Gamma(A_i))}{\sum_{i \leq n} w_i}.$$

Our proof consists of three steps:

1. Proving $\limsup_{n\to\infty} \mathrm{Bias}_n \geq 0$.

2. Noting that the first argument can be applied to the sequence $(-A_n)_n$ to prove $\liminf_{n\to\infty} \mathrm{Bias}_n \leq 0$.

3. Proving that, given these facts, $(\mathrm{Bias}_n)_n$ has 0 as a limit point.

The first step will be deferred. The second step is trivial. We will now show that the third step works given that the previous two do:

Let $a := \liminf_{n\to\infty} \mathrm{Bias}_n \leq 0$ and $b := \limsup_n \mathrm{Bias}_n \geq 0$. If $a = 0$ or $b = 0$ then of course 0 is a limit point. Otherwise, let $a < 0 < b$. If 0 is not a limit point of $(\mathrm{Bias}_n)_n$, then there are $\varepsilon > 0$ and $N \in \mathbb{N}$ such that $\forall n > N : \mathrm{Bias}_n \notin (-\varepsilon, \varepsilon) \subseteq (a, b)$. Choose $M > N$ such that $\mathrm{Bias}_M \in (\varepsilon, b]$ and for all $n > M$, $\mathrm{Bias}_n - \mathrm{Bias}_{n+1} < \varepsilon$; sufficiently late adjacent terms are close because $\sum_{i \leq n} w_i$ goes to ∞ and the absolute difference between successive numerators is at most 1. Then $(\mathrm{Bias}_n)_{n>M}$ must remain positive (it cannot cross the 2ε-wide gap), contradicting that a is also a limit point and $a < 0$.

At this point we have shown that the second and third steps follow from the first step, so we need only show the first step: $\limsup_{n\to\infty} \mathrm{Bias}_n \geq 0$. Suppose this is not the case. Then there is some natural N and rational $\varepsilon \in (0, 1)$ such that for all $n \geq N$,

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(A_i) - \mathrm{Val}_\Gamma(A_i))}{\sum_{i \leq n} w_i} < -2\varepsilon$$

or equivalently,

$$\sum_{i \leq n} w_i \cdot (\mathrm{Val}_\Gamma(A_i) - \mathbb{P}_i(A_i)) > 2\varepsilon \sum_{i \leq n} w_i.$$

An efficiently emulatable sequence of traders. We will consider an infinite sequence of traders, each of which will buy a "run" of $\mathbb{R}$-combinations, and which will have ε-ROI. Then we will apply Lemma B.1.3 to derive a contradiction.

Without loss of generality, assume each $\|A_n\|_{\mathrm{mg}} \leq 1$; they are uniformly bounded and can be scaled down without changing the theorem statement. Let $\overline{w^\dagger}$ be an e.c. $\mathcal{EF}$ progression such that $w_n^\dagger(\overline{\mathbb{P}}) = w_n$. Let $\overline{A^\dagger}$ be an e.c. $\mathcal{EF}$-combination progression such that $A_n^\dagger(\overline{\mathbb{P}}) = A_n$. Let $\overline{A^\dagger}$ be equal to

$$A_n^\dagger = c + \xi_1 \phi_1 + \cdots + \xi_{l(n)} \phi_{l(n)},$$

and define the expressible feature $\|A_n^\dagger\|_{\mathrm{mg}} := \sum_{i=1}^{l(n)} |\xi_i|$.

For $k < N$, trader $\overline{T}^k$ will be identically zero. For $k \geq N$, trader $\overline{T}^k$ will buy some number of copies of $A_n^\dagger$ on day n; formally,

$$T_n^k := \gamma_{k,n}^\dagger \cdot (A_n^\dagger - A_n^{\dagger * n}),$$

with $\gamma_{k,n}^\dagger$ to be defined later. To define $\gamma_{k,n}^\dagger$, we will first define a scaling factor on the trader's purchases:

$$\delta_k := \frac{\varepsilon}{1 + k}.$$

Now we recursively define

$$\gamma_{k,n}^\dagger := [n \geq k] \min\left(\delta_k \cdot w_n^\dagger, 1 - \sum_{i \leq n-1} \gamma_{k,i}^\dagger \|A_n^\dagger\|_{\mathrm{mg}}\right),$$

where $[n \geq k]$ is Iverson bracket applied to $n \geq k$, i.e. the 0-1 indicator of that condition. This sequence of traders is efficiently emulatable, because $\overline{A^\dagger}$ and $\overline{w^\dagger}$ are e.c., and $\overline{T}^k$ makes no trades before day k.

Analyzing the trader. On day n, trader $\overline{T}^k$ attempts to buy $\delta_k w_n$ copies of A_n, but caps its total budget at 1 dollar; the min in the definition of $\gamma_{k,n}^\dagger$ ensures that $\|\overline{T}^k(\mathbb{P})\|_{\mathrm{mg}} \le 1$.

Observe that $\sum_{i \le n} \|\overline{T}_i^k(\mathbb{P})\|_{\mathrm{mg}} = \max\{1, \sum_{i=k}^n \delta_k w_n \|A_n\|_{\mathrm{mg}}\}$. We can use this to show that $\|\overline{T}^k(\mathbb{P})\|_{\mathrm{mg}} = 1$. For all $n \ge N$,

$$2\varepsilon \sum_{i \le n} w_i < \sum_{i \le n} w_i \cdot (\mathrm{Val}_\Gamma(A_i) - \mathbb{P}_i(A_i)) \le 2\sum_{i \le n} w_i \|A_i\|_{\mathrm{mg}}.$$

Since the left hand side goes to ∞ as $n \to \infty$, so does the right hand side. So indeed, $\sum_{n=k}^\infty \delta_k w_n \|A_n\|_{\mathrm{mg}} = \infty$, and $\|\overline{T}^k(\mathbb{P})\|_{\mathrm{mg}} = 1$.

$\overline{T}^k$ has ε return on investment. We will now show that each trader $\overline{T}^k$ has ε return on investment. Trivially, for $k < N$, $\overline{T}^k$ has ε return on investment, because $\|\overline{T}^k(\mathbb{P})\|_{\mathrm{mg}} = 0$. So consider $k \ge N$.

As shorthand, let $\gamma_{k,n} := \gamma_{k,n}^\dagger(\overline{\mathbb{P}})$. Let m_k be the first $m \ge k$ for which $\gamma_{k,m} < \delta_k w_n$. We have $\gamma_{k,n} = \delta_k w_n$ for $k \le n < m_k$, and $\gamma_{k,n} = 0$ for $n > m_k$. So for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$
\begin{aligned}
&\mathbb{W}\left(\sum_{n=1}^\infty T_n^k(\overline{\mathbb{P}})\right) \\
&= \sum_{n=1}^\infty \mathbb{W}(T_n^k(\overline{\mathbb{P}})) \\
&= \sum_{k \le n \le m_k} \mathbb{W}(T_n^k(\overline{\mathbb{P}})) \\
&= \sum_{k \le n \le m_k} \delta_k w_n (\mathbb{W}(A_n) - \mathbb{P}_n(A_n)) - (\delta_k w_{m_k} - \gamma_{k,m_k})(\mathbb{W}(A_{m_k}) - \mathbb{P}_n(A_{m_k})) \\
&\ge \sum_{k \le n \le m_k} \delta_k w_n (\mathbb{W}(A_n) - \mathbb{P}_n(A_n)) - \delta_k \\
&= \sum_{n \le m_k} \delta_k w_n (\mathbb{W}(A_n) - \mathbb{P}_n(A_n)) - \sum_{n < k} \delta_k w_n (\mathbb{W}(A_n) - \mathbb{P}_n(A_n)) - \delta_k \\
&\ge \sum_{n \le m_k} \delta_k w_n (\mathbb{W}(A_n) - \mathbb{P}_n(A_n)) - (k\delta_k + \delta_k) \\
&= \sum_{n \le m_k} \delta_k w_n (\mathrm{Val}_\Gamma A_n - \mathbb{P}_n(A_n)) - \varepsilon \\
&\ge 2\varepsilon - \varepsilon \\
&= \varepsilon.
\end{aligned}
$$

So each trader $\overline{T}^k$ with $k \ge N$ makes at least ε profit with trades of total magnitude 1, ensuring that it has ε return on investment.

Deriving a contradiction. Note that the magnitudes of the traders are $\overline{\mathbb{P}}$-generable (the first $N-1$ have magnitude 0 and the rest have magnitude 1). By Lemma B.1.3, $\|\overline{T}^k\|_{\mathrm{mg}} \eqsim_k 0$. $\|\overline{T}^k\|_{\mathrm{mg}} = 1$ for all $k \ge N$ (by the above analysis), so this is a contradiction.

$\square$

D.2 Recurring Unbiasedness

Theorem 4.3.6 (Recurring Unbiasedness). *Given an e.c. sequence of decidable sentences $\overline{\phi}$ and a $\overline{\mathbb{P}}$-generable divergent weighting $\overline{w}$, the sequence*

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(\phi_i) - \mathrm{Thm}_\Gamma(\phi_i))}{\sum_{i \leq n} w_i}$$

has 0 as a limit point. In particular, if it converges, it converges to 0.

Proof. This is a special case of Theorem 4.5.9 (Affine Recurring Unbiasedness). $\square$

D.3 Simple Calibration

Theorem 4.3.3 (Recurring Calibration). *Let $\overline{\phi}$ be an e.c. sequence of decidable sentences, a and b be rational numbers, $\overline{\delta}$ be an e.c. sequence of positive rational numbers, and suppose that $\sum_n \left(\mathrm{Ind}_{\delta_i}(a < \mathbb{P}_i(\phi_i) < b) \right)_{i \in \mathbb{N}^+} = \infty$. Then, if the sequence*

$$\left(\frac{\sum_{i \leq n} \mathrm{Ind}_{\delta_i}(a < \mathbb{P}_i(\phi_i) < b) \cdot \mathrm{Thm}_\Gamma(\phi_i)}{\sum_{i \leq n} \mathrm{Ind}_{\delta_i}(a < \mathbb{P}_i(\phi_i) < b)} \right)_{n \in \mathbb{N}^+}$$

converges, it converges to a point in $[a, b]$. Furthermore, if it diverges, it has a limit point in $[a, b]$.

Proof. Define $w_i := \mathrm{Ind}_{\delta_i}(a < \phi_i^{*i} < b)$. By Theorem 4.3.6 (Recurring Unbiasedness), the sequence

$$\left(\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(\phi_i) - \mathrm{Thm}_\Gamma(\phi_i))}{\sum_{i \leq n} w_i} \right)_{n \in \mathbb{N}^+}$$

has 0 as a limit point. Let $n_1, n_2, \ldots$ be a the indices of a subsequence of this sequence that converges to zero. We also know that for all n high enough,

$$a \leq \frac{\sum_{i \leq n} w_i \mathbb{P}_i(\phi_i)}{\sum_{i \leq n} w_i} \leq b$$

because $w_i = 0$ whenever $\mathbb{P}_i(\phi_i) \notin [a, b]$. Now consider the sequence

$$\left(\frac{\sum_{i \leq n_k} w_i \cdot \mathrm{Thm}_\Gamma(\phi_i)}{\sum_{i \leq n_k} w_i} \right)_{k \in \mathbb{N}^+}$$

$$= \left(\frac{\sum_{i \leq n_k} w_i \mathbb{P}_i(\phi_i)}{\sum_{i \leq n} w_i} - \frac{\sum_{i \leq n_k} w_i \cdot (\mathbb{P}_i(\phi_i) - \mathrm{Thm}_\Gamma(\phi_i))}{\sum_{i \leq n_k} w_i} \right)_{k \in \mathbb{N}^+}$$

The first term is bounded between a and b, and the second term goes to zero, so the sequence has a $\liminf$ at least a and a $\limsup$ no more than b. By the Bolzano-Weierstrass theorem, this sequence has a convergent subsequence, whose limit must be between a and b. This subsequence is also a subsequence of

$$\left(\frac{\sum_{i \leq n} w_i \cdot \mathrm{Thm}_\Gamma(\phi_i)}{\sum_{i \leq n} w_i} \right)_{n \in \mathbb{N}^+}$$

which proves that this sequence has a limit point in $[a, b]$, as desired. $\square$

D.4 Affine Unbiasedness From Feedback

Theorem 4.5.10 (Affine Unbiasedness from Feedback). *Given $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$ that is determined via Γ, a strictly increasing deferral function f such that $\mathrm{Val}_\Gamma(A_n)$ can be computed in time $\mathcal{O}(f(n+1))$, and a $\overline{\mathbb{P}}$-generable divergent weighting w,*

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(A_i) - \mathrm{Val}_\Gamma(A_i))}{\sum_{i \leq n} w_i} \eqsim_n 0.$$

In this case, we say "$\overline{w}$ allows good feedback on $\overline{A}$".

Proof. Without loss of generality, assume that each $\|A_n\|_1 \leq 1$. Define

$$\mathrm{Bias}_k := \frac{\sum_{i \leq k} w_{f(i)} \cdot (\mathbb{P}_{f(i)}(\phi_{f(i)}) - \mathrm{Thm}_\Gamma(\phi_{f(i)}))}{\sum_{i \leq k} w_{f(i)}}$$

and observe that $\mathrm{Bias}_k \eqsim_k 0$ implies the theorem statement, since we need only consider the sum over n in the support of f. We wish to show that $\mathrm{Bias}_k \eqsim_k 0$. We will show a trader that exploits $\overline{\mathbb{P}}$ under the assumption that $\liminf_{k \to \infty} \mathrm{Bias}_k < 0$, proving $\mathrm{Bias}_k \gtrsim_k 0$. We can apply the same argument to the sequence $(-A_n)_n$ to get $\mathrm{Bias}_k \lesssim_k 0$.

Suppose $\liminf_{k \to \infty} \mathrm{Bias}_k < 0$. Under this supposition, infinitely often, $\mathrm{Bias}_k < -3\varepsilon$ for some rational $0 < \varepsilon < 1/6$.

Defining the trader. Let $\overline{w^\dagger}$ be an e.c. $\mathcal{EF}$ progression such that $w_n^\dagger(\overline{\mathbb{P}}) = w_n$. Let $\overline{A^\dagger}$ be an e.c. $\mathcal{EF}$-combination progression such that $A_n^\dagger(\overline{\mathbb{P}}) = A_n$. Recursively define

$$\beta_i^\dagger := \varepsilon \cdot \mathrm{Wealth}_i^\dagger \cdot w_i^\dagger$$
$$\mathrm{Wealth}_i^\dagger := 1 + \sum_{j \leq i-1} \beta_j^\dagger \cdot \left(A_{f(j)}^{\dagger * f(j+1)} - A_{f(j)}^{\dagger * f(j)} \right)$$

in order to define the trader

$$T_n := \begin{cases} \beta_i^\dagger \cdot \left(A_{f(i)}^\dagger - A_{f(i)}^{\dagger * n} \right) - [i > 1] \cdot \beta_{i-1}^\dagger \cdot \left(A_{f(i-1)}^\dagger - A_{f(i-1)}^{\dagger * n} \right) & \text{if } \exists i : n = f(i) \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\beta_i^\dagger$ and $\mathrm{Wealth}_i^\dagger$ have rank at most $f(i-1)$, and so T_n has rank at most n.

Analyzing the trader. As shorthand, let $\beta_i := \beta_i^\dagger(\overline{\mathbb{P}})$, and $\mathrm{Wealth}_i := \mathrm{Wealth}_i^\dagger(\overline{\mathbb{P}})$.

Intuitively, on day $f(i)$, $\overline{T}$ buys $A_{f(i)}$ according to a fraction of its "wealth" Wealth_i (how much money $\overline{T}$ would have if it started with one dollar), and then sells $A_{f(i)}$ at a later time $f(i+1)$. Thus, $\overline{T}$ makes money if the price of $A_{f(i)}$ is greater at time $f(i+1)$ than at time $f(i)$.

Betting according to a fraction of wealth resembles the Kelley betting strategy and ensures that the trader never loses more than \$1. $\mathrm{Wealth}_i - 1$ is a lower bound on $\overline{T}$'s worth in any world $\mathbb{W} \in \mathcal{PC}(D_{f(i)})$, based on trades $\overline{T}$ makes on $\mathbb{R}$-combinations $A_{f(1)}$ through $A_{f(i-1)}$. Thus, since the number of copies of A_n that $\overline{T}$ buys is no more than ε times its current wealth, and $\|A_n\| \leq 1$, T's minimum worth is bounded below by -1.

Now it will be useful to write Wealth_i in log space. Intuitively, this should be enlightening because T always bets a fraction of its wealth (similar to a Kelley bettor), so its winnings multiply over time rather than adding. By induction,

$$\log \mathrm{Wealth}_i = \sum_{j \leq i-1} \log \left(1 + \varepsilon w_j (\mathbb{P}_{f(j+1)}(A_{f(j)}) - \mathbb{P}_{f(j)}(A_{f(j)})) \right)$$

This statement is trivial when $i = 1$. For the inductive step, we have

$$
\begin{aligned}
\log \mathrm{Wealth}_{i+1} &= \log\left(\mathrm{Wealth}_i + \beta_i(\mathbb{P}_{f(i+1)}(A_{f(i)}) - \mathbb{P}_{f(i)}(A_{f(i)}))\right) \\
&= \log\left(\mathrm{Wealth}_i + \varepsilon \cdot \mathrm{Wealth}_i \cdot w_i(\mathbb{P}_{f(i+1)}(A_{f(i)}) - \mathbb{P}_{f(i)}(A_{f(i)}))\right) \\
&= \log \mathrm{Wealth}_i + \log\left(1 + \varepsilon w_j(\mathbb{P}_{f(j+1)}(A_{f(j)}) - \mathbb{P}_{f(j)}(A_{f(j)}))\right) \\
&= \sum_{j \leq i} \log\left(1 + \varepsilon w_j(\mathbb{P}_{f(j+1)}(A_{f(j)}) - \mathbb{P}_{f(j)}(A_{f(j)}))\right)
\end{aligned}
$$

For $|x| \leq 1/2$, we have $\log(1 + x) \geq x - x^2$. Therefore, since $\varepsilon < 1/4$, $w_j \leq 1$, and $|\mathbb{P}_{f(j+1)}(A_{f(j)}) - \mathbb{P}_{f(i)}(A_{f(j)})| \leq 1$,

$$
\begin{aligned}
\log \mathrm{Wealth}_i &\geq \sum_{j \leq i-1} \Big(\varepsilon w_j(\mathbb{P}_{f(j+1)}(A_{f(j)}) - \mathbb{P}_{f(j)}(A_{f(j)})) \\
&\qquad\qquad - \varepsilon^2 w_j^2(\mathbb{P}_{f(j+1)}(A_{f(j)}) - \mathbb{P}_{f(j)}(A_{f(j)}))^2\Big) \\
&\geq \sum_{j \leq i-1} \left(\varepsilon w_j(\mathbb{P}_{f(j+1)}(A_{f(j)}) - \mathbb{P}_{f(j)}(A_{f(j)})) - \varepsilon^2\right) \\
&= \sum_{j \leq i-1} \varepsilon\left(w_j(\mathbb{P}_{f(j+1)}(A_{f(j)}) - \mathbb{P}_{f(j)}(A_{f(j)})) - \varepsilon\right)
\end{aligned}
$$

At this point, it will be useful to show a relation between $\mathbb{P}_{f(j+1)}(A_{f(j)})$ and $\mathrm{Val}_\Gamma(A_{f(j)})$. Consider the sequence

$$
A'_n := \begin{cases} A_{f(i)} - \mathrm{Val}_\Gamma(A_{f(i)}) & \text{if } \exists i : f(i+1) = n \\ 0 & \text{otherwise} \end{cases}
$$

which is in $\mathcal{BCS}(\overline{\mathbb{P}})$ because $\mathrm{Val}_\Gamma(A_{f(j)})$ is computable in time polynomial in $f(j+1)$. Of course, each A'_n has value 0 in any world $\mathbb{W} \in \mathcal{PC}(\Gamma)$. So by Theorem 4.5.4 (Affine Provability Induction),

$$
\mathbb{P}_n(A'_n) \eqsim_n 0
$$

so for all sufficiently high j, $\mathbb{P}_{f(j+1)}(A_{f(j)}) \geq \mathrm{Val}_\Gamma(A_{f(j)}) - \varepsilon$. Thus, for some constant C,

$$
\begin{aligned}
\log \mathrm{Wealth}_i &\geq \sum_{j \leq i-1} \varepsilon\left(w_j(\mathrm{Val}_\Gamma(A_{f(j)}) - \mathbb{P}_{f(j)}(A_{f(j)})) - 2\varepsilon\right) - C \\
&= \varepsilon\left(\sum_{j \leq i-1} w_j\right)(-\mathrm{Bias}_{i-1} - 2\varepsilon) - C'
\end{aligned}
$$

Now, for infinitely many i it is the case that $\mathrm{Bias}_{i-1} < -3\varepsilon$. So it is infinitely often the case that $\log \mathrm{Wealth}_i \geq \varepsilon^2 \sum_{j \leq i-1} w_i - C'$. Since $\overline{w}$ is divergent, T's eventual wealth (and therefore max profit) can be arbitrarily high. Thus, T exploits $\overline{\mathbb{P}}$. $\qquad\square$

D.5   Unbiasedness From Feedback

Theorem 4.3.8 (Unbiasedness From Feedback). *Let $\overline{\phi}$ be any e.c. sequence of decidable sentences, and $\overline{w}$ be any $\overline{\mathbb{P}}$-generable divergent weighting. If there exists a strictly increasing deferral function f such that the support of $\overline{w}$ is contained in the image of f and $\mathrm{Thm}_\Gamma(\phi_{f(n)})$ is computable in $\mathcal{O}(f(n+1))$ time, then*

$$
\frac{\sum_{i \leq n} w_i \cdot (\mathbb{P}_i(\phi_i) - \mathrm{Thm}_\Gamma(\phi_i))}{\sum_{i \leq n} w_i} \eqsim_n 0.
$$

In this case, we say "$\overline{w}$ allows good feedback on $\overline{\phi}$".

Proof. This is a special case of Theorem 4.5.10 (Affine Unbiasedness from Feedback). $\qquad\square$

D.6 Learning Pseudorandom Affine Sequences

Theorem 4.5.11 (Learning Pseudorandom Affine Sequences). *Given a $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$ which is determined via Γ, if there exists deferral function f such that for any $\overline{\mathbb{P}}$-generable f-patient divergent weighting $\overline{w}$,*

$$\frac{\sum_{i \leq n} w_i \cdot \mathrm{Val}_\Gamma(A_i)}{\sum_{i \leq n} w_i} \gtrsim_n 0,$$

then

$$\mathbb{P}_n(A_n) \gtrsim_n 0,$$

and similarly for $\eqsim_n$, and $\lesssim_n$.

Proof. We will prove the statement in the case of $\gtrsim_n$; the case of $\lesssim_n$ follows by negating the $\mathbb{R}$-combination sequence $\overline{A}$, and the case of $\eqsim_n$ follows from the other two. Suppose it is not the case that $\mathbb{P}_n(A_n) \gtrsim_n 0$. Then there is a rational $\varepsilon > 0$ such that $\mathbb{P}_n(A_n) < -2\varepsilon$ infinitely often.

Defining the trader. Let $\overline{A^\dagger}$ be an e.c. $\mathcal{EF}$-combination progression such that $A_n^\dagger(\overline{\mathbb{P}}) = A_n$. Let an affine combination A be considered *settled* by day m if $\mathbb{W}(A) = \mathrm{Val}_\Gamma(A)$ for each $\mathbb{W} \in \mathcal{PC}(D_m)$. We may write $\mathrm{Settled}(n, m)$ to be the proposition that A_n is settled by day m. $\mathrm{Settled}(n, m)$ is decidable; let `settled` be a Turing machine deciding $\mathrm{Settled}(n, m)$ given (n, m). Now we define a lower-approximation to Settled:

$\mathrm{DefinitelySettled}(n, m) :\leftrightarrow \exists i \leq m : \mathtt{settled}(n, i)$ returns true within m steps.

Note that

- $\mathrm{DefinitelySettled}(n, m)$ can be decided in time polynomial in m when $n \leq m$,

- $\mathrm{DefinitelySettled}(n, m) \rightarrow \mathrm{Settled}(n, m)$, and

- If $\mathrm{Settled}(n, m)$, then $\mathrm{DefinitelySettled}(n, M)$ for some $M \geq m$.

To define the trader, first we will define $\overline{\alpha^\dagger}$ recursively by

$$\alpha_n^\dagger := (1 - C_n^\dagger) \, \mathrm{Ind}_\varepsilon(A_n^{\dagger *n} < -\varepsilon)$$
$$C_n^\dagger := \sum_{i < n} [\neg \mathrm{DefinitelySettled}(i, n) \vee f(i) > n] \cdot \alpha_i^\dagger.$$

The trader itself buys $\alpha_n^\dagger(\overline{\mathbb{P}})$ copies of the combination $A^\dagger{}_n(\overline{\mathbb{P}})$ on day n:

$$T_n := \alpha_n^\dagger \cdot (A_n^\dagger - A_n^{\dagger *n}).$$

Intuitively, $C_n^\dagger$ is the total number of copies of $\mathbb{R}$-combinations that the trader has bought that are either possibly-unsettled (according to DefinitelySettled), or whose deferral time $f(i)$ is past the current time n.

Analyzing the trader. As shorthand, define $\alpha_n := \alpha_n^\dagger(\overline{\mathbb{P}})$ and $C_n := C_n^\dagger(\overline{\mathbb{P}})$. Some important properties of $\overline{T}$ are:

- Each $C_n \leq 1$.

- $\alpha_n = 1 - C_n$ when $\mathbb{P}_n(A_n) < -2\varepsilon$.

- Whenever $\alpha_n > 0$, $\mathbb{P}_n(A_n) < -\varepsilon$.

- $\sum_{n \in \mathbb{N}^+} \alpha_n = \infty$. Suppose this sum were finite. Then there is some time N for which $\sum_{n \geq N} \alpha_n < 1/2$. For some future time N', and each $n < N$, we have DefinitelySettled$(n, N') \wedge f(n) \leq N'$. This implies that $C_n < 1/2$ for each $n \geq N'$. However, consider the first $n \geq N'$ for which $\mathbb{P}_n(A_n) < -2\varepsilon$. Since $C_n < 1/2$, $\alpha_n \geq 1/2$. But this contradicts $\sum_{n \geq N} \alpha_n < 1/2$.

Let $b \in \mathbb{Q}$ be such that each $\|A_n\|_{\mathrm{mg}} < b$. Consider this trader's profit at time m in world $\mathbb{W} \in \mathcal{PC}(D_m)$:

$$\sum_{n \leq m} \alpha_n(\mathbb{W}(A_n) - \mathbb{P}_n(A_n))$$

$$\geq \sum_{n \leq m} \alpha_n(\mathbb{W}(A_n) + \varepsilon)$$

$$\geq \sum_{n \leq m} \alpha_n(\mathrm{Val}_\Gamma(A_n) + \varepsilon) - 2b$$

where the last inequality follows because $\sum_{n \leq m}[\neg\mathrm{Settled}(n, m)]\alpha_n \leq 1$, and an unsettled copy of A_n can only differ by $2b$ between worlds, while settled copies of A_n must have the same value in all worlds in $\mathcal{PC}(D_m)$.

T holds no more than 1 copy of $\mathbb{R}$-combinations unsettled by day M, and an unsettled combination's value can only differ by $2b$ between different worlds while settled affine combinations must have the same value in different worlds in $\mathcal{PC}(D_m)$. We now show that this quantity goes to ∞, using the fact that $\overline{A}$ is pseudorandomly positive.

Observe that $\overline{\alpha}$ is a divergent weighting. It is also f-patient, since $\sum_{n \leq m}[f(n) \geq m]\alpha_n \leq 1$. So by assumption,

$$\liminf_{m \to \infty} \frac{\sum_{n \leq m} \alpha_n \cdot \mathrm{Val}_\Gamma(A_n)}{\sum_{n \leq m} \alpha_n} \geq 0.$$

At this point, note that

$$\sum_{n \leq m} \alpha_n(\mathrm{Val}_\Gamma(A_n) + \varepsilon) = \left(\sum_{n \leq m} \alpha_n\right)\left(\frac{\sum_{n \leq m} \alpha_n \cdot \mathrm{Val}_\Gamma(A_n)}{\sum_{n \leq m} \alpha_n} + \varepsilon\right).$$

For all sufficiently high m, $\frac{\sum_{n \leq m} \alpha_n \cdot \mathrm{Val}_\Gamma(A_n)}{\sum_{n \leq m} \alpha_n} \geq -\varepsilon/2$, and $\sum_{n \in \mathbb{N}^+} \alpha_n = \infty$, so

$$\liminf_{m \to \infty} \sum_{n \leq m} \alpha_n(\mathrm{Val}_\Gamma(A_n) + \varepsilon) = \infty.$$

If we define $g(m)$ to be the minimum plausible worth at time m over plausible worlds $\mathbb{W} \in \mathcal{PC}(D_m)$, we see that $g(m)$ limits to infinity, implying that the trader's maximum worth goes to infinity. The fact that $g(m)$ limits to infinity also implies that $g(m)$ is bounded from below, so the trader's minimum worth is bounded from below. Thus, this trader exploits the market $\overline{\mathbb{P}}$.

$\square$

D.7 Learning Varied Pseudorandom Frequencies

Definition 4.4.4 (Varied Pseudorandom Sequence)**.** *Given a deferral function f, a set S of f-patient divergent weightings, an e.c. sequence $\overline{p}$ of Γ-decidable sentences, and a $\mathbb{P}$-generable sequence $\overline{p}$ of rational probabilities, $\overline{\phi}$ is called a $\overline{\mathbf{p}}$-varied pseudorandom sequence (relative to S) if, for all $\overline{w} \in S$,*

$$\frac{\sum_{i \leq n} w_i \cdot (p_i - \mathrm{Thm}_\Gamma(\phi_i))}{\sum_{i \leq n} w_i} \overline{\approx}_n 0.$$

*Furthermore, we can replace $\eqsim_n$ with $\gtrsim_n$ or $\lesssim_n$, in which case we say $\overline{\phi}$ is **varied pseudorandom above $\overline{p}$** or **varied pseudorandom below $\overline{p}$**, respectively.*

Theorem 4.4.5 (Learning Varied Pseudorandom Frequencies)**.** *Given an e.c. sequence $\overline{\phi}$ of Γ-decidable sentences and a $\overline{\mathbb{P}}$-generable sequence $\overline{p}$ of rational probabilities, if there exists some f such that $\overline{\phi}$ is $\overline{p}$-varied pseudorandom (relative to all f-patient $\overline{\mathbb{P}}$-generable divergent weightings), then*

$$\mathbb{P}_n(\phi_n) \eqsim_n p_n.$$

Furthermore, if $\overline{\phi}$ is varied pseudorandom above or below $\overline{p}$, then the $\eqsim_n$ may be replaced with $\gtrsim_n$ or $\lesssim_n$ (respectively).

Proof. We will prove the statement in the case of pseudorandom above; the case of pseudorandom below is analogous, and the case of pseudorandom follows from the other cases.

Define $A_n := \phi_n - p_n$ and note that $\overline{A} \in \mathcal{BCS}(\overline{\mathbb{P}})$. Observe that, because $\overline{A}$ is varied pseudorandomness above $\overline{p}$, for any f-patient divergent weighting w,

$$\frac{\sum_{i \leq n} w_i \operatorname{Val}_\Gamma A_n}{\sum_{i \leq n} w_i} \gtrsim_n 0.$$

Now apply Theorem 4.5.11 (Learning Pseudorandom Affine Sequences) to get

$$\mathbb{P}_n(A_n) = \mathbb{P}_n(\phi_n) - p_n \gtrsim_n 0.$$

$\square$

D.8 Learning Pseudorandom Frequencies

Definition 4.4.1 (Pseudorandom Sequence)**.** *Given a set S of divergent weightings (Definition 4.3.4), a sequence $\overline{\phi}$ of decidable sentences is called **pseudorandom with frequency p** over S if, for all weightings $\overline{w} \in S$,*

$$\lim_{n \to \infty} \frac{\sum_{i \leq n} w_i \cdot \operatorname{Thm}_\Gamma(\phi_i)}{\sum_{i \leq n} w_i}$$

exists and is equal to p.

Theorem 4.4.2 (Learning Pseudorandom Frequencies)**.** *Let $\overline{\phi}$ be an e.c. sequence of decidable sentences. If $\overline{\phi}$ is pseudorandom with frequency p over the set of all $\overline{\mathbb{P}}$-generable divergent weightings, then*

$$\mathbb{P}_n(\phi_n) \eqsim_n p.$$

Proof. Let q be any rational number less than p. Note that $\overline{\phi}$ is varied pseudorandom above q, so by Theorem 4.4.5 (Learning Varied Pseudorandom Frequencies),

$$\mathbb{P}_n(\phi_n) \gtrsim_n q.$$

But we could have chosen any rational $q < p$, so $\mathbb{P}_n(\phi_n) \gtrsim_n p$. An analogous argument shows $\mathbb{P}_n(\phi_n) \lesssim_n p$. $\square$

E Expectations Proofs

E.1 Consistent World LUV Approximation Lemma

Lemma E.1.1. *Let $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$ be a $\mathbb{R}$-LUV combination bounded by some rational number b. For all natural numbers n and all $\mathbb{W} \in \mathcal{PC}(\Gamma)$, we have*

$$|\mathbb{E}_n^{\mathbb{W}}(B) - \mathbb{W}(B)| \leq b/n.$$

Proof. Let $\mathbb{W} \in \mathcal{PC}(\Gamma)$. For any $[0,1]$-LUV X, by Definition 4.8.2,

$$|\mathbb{E}_n^{\mathbb{W}}(X) - \mathbb{W}(X)| = \left| \sum_{i=0}^{n-1} \frac{1}{n} \mathbb{W}(\text{"}\underline{X} > \underline{i}/\underline{n}\text{"}) - \mathbb{W}(X) \right|$$

Since Γ can represent computable functions, the number of i values in $\{0, \ldots, n-1\}$ for which $\mathbb{W}(\text{"}\underline{X} > \underline{i}/\underline{n}\text{"}) = 1$ is at least $\lfloor n\mathbb{W}(X) \rfloor \geq n\mathbb{W}(X) - 1$, so

$$\sum_{i=0}^{n-1} \frac{1}{n} \mathbb{W}(\text{"}\underline{X} > \underline{i}/\underline{n}\text{"}) \geq \mathbb{W}(X) - 1/n.$$

Similarly, the number of i values in $\{0, \ldots, n-1\}$ for which $\mathbb{W}(\text{"}\underline{X} > \underline{i}/\underline{n}\text{"})$ is no more than $\lceil n\mathbb{W}(X) \rceil \leq n\mathbb{W}(X) + 1$, so

$$\sum_{i=0}^{n-1} \frac{1}{n} \mathbb{W}(\text{"}\underline{X} > \underline{i}/\underline{n}\text{"}) \leq \mathbb{W}(X) + 1/n.$$

We now have

$$\begin{aligned}
|\mathbb{E}_n^{\mathbb{W}}(B) - \mathbb{W}(B)| &= |c_1(\mathbb{E}_n^{\mathbb{W}}(X_1) - \mathbb{W}(X_1)) + \cdots + c_k(\mathbb{E}_n^{\mathbb{W}}(X_k) - \mathbb{W}(X_k))| \\
&\leq c_1|\mathbb{E}_n^{\mathbb{W}}(X_1) - \mathbb{W}(X_1)| + \cdots + c_k|\mathbb{E}_n^{\mathbb{W}}(X_k) - \mathbb{W}(X_k)| \\
&\leq c_1/n + \cdots + c_k/n \\
&\leq b/n.
\end{aligned}$$

$\square$

E.2 Mesh Independence Lemma

Lemma E.2.1. *Let $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$. Then*

$$\lim_{n\to\infty} \sup_{m \geq n} |\mathbb{E}_n^{\mathbb{P}^m}(B_n) - \mathbb{E}_m(B_n)| = 0.$$

Proof. We will prove the claim that

$$\limsup_{m\to\infty} \max_{n \leq m} \left(|\mathbb{E}_n^{\mathbb{P}^m}(B_n) - \mathbb{E}_m(B_n)| - (2/n) \right) \leq 0.$$

This claim implies that, for any $\varepsilon > 0$, there are only finitely many (n, m) with $n \leq m$ such that $|\mathbb{E}_n^{\mathbb{P}^m}(B_n) - \mathbb{E}_m(B_n)| > 2/n + \varepsilon$, which in turn implies that, for any $\varepsilon' > 0$, there are only finitely many (n, m) with $n \leq m$ such that $|\mathbb{E}_n^{\mathbb{P}^m}(B_n) - \mathbb{E}_m(B_n)| > \varepsilon'$. This is sufficient to show the statement of the theorem.

We will now prove

$$\limsup_{m\to\infty} \max_{n \leq m} \left(\mathbb{E}_n^{\mathbb{P}^m}(B_n) - \mathbb{E}_m(B_n) - (2/n) \right) \leq 0.$$

The proof with $\mathbb{E}_m(B_n) - \mathbb{E}_n^{\mathbb{P}^m}(B_n)$ instead is analogous, and together these inequalities prove the claim.

Suppose this inequality does not hold. Then there is some rational $\varepsilon > 0$ such that for infinitely many m,

$$\max_{n \leq m}(\mathbb{E}_n^{\mathbb{P}^m}(B_n) - \mathbb{E}_m(B_n) - (2/n)) > \varepsilon.$$

Let $\overline{B^\dagger}$ be an e.c. $\mathcal{EF}$-combination progression such that $B_n^\dagger(\overline{\mathbb{P}}) = B_n$. Assume without loss of generality that each $\|B_n\|_1 \leq 1$ (they are assumed to be bounded and can be scaled down appropriately). Define $\mathcal{EF}$-combinations

$$A_{n,m}^\dagger := \mathrm{Ex}_n(B_n^\dagger) - \mathrm{Ex}_m(B_n^\dagger) - 2/n,$$

using the $\mathcal{F}$-combinations Ex defined in A.3. As shorthand, we write $A_{n,m} :=$ $A_{n,m}^\dagger(\overline{\mathbb{P}})$. By Lemma E.1.1, for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$, $\mathbb{W}(A_{n,m}) \leq 0$. We aim to show $\mathbb{P}_m(A_{n,m}) < \varepsilon$ for all sufficiently high m and $n \leq m$, but we cannot immediately derive this using Theorem 4.5.4 (Affine Provability Induction), since A has two indices. We get around this difficulty by taking a "softmax" over possible values of n given a fixed value of m. Specifically, for $n \leq m$, define expressible features (of rank m)

$$\alpha_{n,m}^\dagger := \mathrm{Ind}_{\varepsilon/2}\left(A_{n,m}^{\dagger *m} > \varepsilon/2\right) \cdot \left(1 - \sum_{i<n} \alpha_{i,m}^\dagger\right).$$

As shorthand, we write $\alpha_{n,m} := \alpha_{n,m}^\dagger(\overline{\mathbb{P}})$. Intuitively, $\alpha_{n,m}$ will distribute weight among n values for which $A_{m,n}$ is overpriced at time m. Now we define the $\mathcal{EF}$-combination progression

$$G_m^\dagger := \sum_{n \leq m} \alpha_{n,m} \cdot A_{n,m}^\dagger.$$

As shorthand, we write $G_m := G_m^\dagger(\overline{\mathbb{P}})$. Fix m and suppose that $\mathbb{P}_m(A_{n,m}) \geq \varepsilon$ for some $n \leq m$. Then $\sum_{n \leq m} \alpha_{n,m} = 1$. Therefore,

$$\mathbb{P}_m(G_m) = \sum_{n \leq m} \alpha_{n,m} \mathbb{P}_m(A_{n,m}) \geq \sum_{n \leq m} \alpha_{n,m} \cdot \varepsilon/2 = \varepsilon/2.$$

So if we can show $\mathbb{P}_m(G_m) \lesssim_m 0$, that will be sufficient to show that $\max_{n \leq m} \mathbb{P}_m(A_{n,m}) < \varepsilon$ for all sufficiently high m. We now show this. Let $\mathbb{W} \in \mathcal{PC}(\Gamma)$. Since each $\alpha_{n,m} \geq 0$, and $\mathbb{W}(A_{n,m}) \leq 0$, we have $\mathbb{W}(G_m) \leq 0$. So by Theorem 4.5.4 (Affine Provability Induction), $\mathbb{P}_m(G_m) \lesssim_m 0$; here we use that $(G_m)_m$ is bounded, since the $A_{n,m}$ are bounded and since for each m, $\sum_{n \leq m} \alpha_{n,m} \leq 1$ by construction.

So for all sufficiently high m we have $\max_{n \leq m} \mathbb{P}_m(A_{n,m}) < \varepsilon$ (or equivalently, $\max_{n \leq m}(\mathbb{E}_n^{\mathbb{P}_m}(B_n) - \mathbb{E}_m(B_n)) < 2/n + \varepsilon$). But this contradicts our assumption that for infinitely many m,

$$\max_{n \leq m}(\mathbb{E}_n^{\mathbb{P}_m}(B_n) - \mathbb{E}_m(B_n) - (2/n)) > \varepsilon.$$

$\square$

E.3 Expectation Preemptive Learning

Theorem 4.8.13 (Expectation Preemptive Learning)**.** *Let $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \mathbb{E}_n(B_n) = \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{E}_m(B_n)$$

and

$$\limsup_{n \to \infty} \mathbb{E}_n(B_n) = \limsup_{n \to \infty} \inf_{m \geq n} \mathbb{E}_m(B_n) \ .$$

Proof. We prove only the first statement; the proof of the second statement is analogous. Apply Theorem 4.5.7 (Affine Preemptive Learning) to the bounded sequence $(\mathrm{Ex}_n(B_n))_n$ to get

$$\liminf_{n \to \infty} \mathbb{E}_n^{\mathbb{P}_n}(B_n) = \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{E}_n^{\mathbb{P}_m}(B_n),$$

using that by definition $\mathbb{P}_m(\mathrm{Ex}_n(B_n)) = \mathbb{E}_n^{\mathbb{P}_m}(B_n)$. By Lemma E.2.1,

$$\lim_{n \to \infty} \sup_{m \geq n} |\mathbb{E}_n^{\mathbb{P}_m}(B_n) - \mathbb{E}_m(B_n)| = 0$$

so

$$\liminf_{n \to \infty} \sup_{m \geq n} \mathbb{E}_n^{\mathbb{P}_m}(B_n) = \liminf_{n \to \infty} \sup_{m \geq n} \mathbb{E}_m(B_n).$$

$\square$

E.4 Expectations Converge

Theorem 4.8.3 (Expectations Converge). *The limit* $\mathbb{E}_\infty : \mathcal{S} \to [0,1]$ *defined by*

$$\mathbb{E}_\infty(X) := \lim_{n\to\infty} \mathbb{E}_n(X)$$

exists for all $X \in \mathcal{U}$.

Proof. By applying Theorem 4.8.13 (Expectation Preemptive Learning) to the constant sequence $X, X, \ldots$, we have

$$\liminf_{n\to\infty} \mathbb{E}_n(X) = \liminf_{n\to\infty} \sup_{m\geq n} \mathbb{E}_m(X) = \limsup_{n\to\infty} \mathbb{E}_n(X).$$

$\square$

E.5 Limiting Expectation Approximation Lemma

Lemma E.5.1. *For any* $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$,

$$|\mathbb{E}_n^{\mathbb{P}_\infty}(B_n) - \mathbb{E}_\infty(B_n)| \approx_n 0.$$

Proof. By Lemma E.2.1 and by continuity of $\mathbb{V} \mapsto \mathbb{E}_n^{\mathbb{V}}(B_n)$,

$$\lim_{n\to\infty} |\mathbb{E}_n^{\mathbb{P}_\infty}(B_n) - \mathbb{E}_\infty(B_n)| = \lim_{n\to\infty} \lim_{m\to\infty} |\mathbb{E}_n^{\mathbb{P}_m}(B_n) - \mathbb{E}_m(B_n)|$$
$$\leq \lim_{n\to\infty} \sup_{m\geq n} |\mathbb{E}_n^{\mathbb{P}_m}(B_n) - \mathbb{E}_m(B_n)|$$
$$= 0.$$

$\square$

E.6 Persistence of Expectation Knowledge

Theorem 4.8.12 (Persistence of Expectation Knowledge). *Let* $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$. *Then*

$$\liminf_{n\to\infty} \inf_{m\geq n} \mathbb{E}_m(B_n) = \liminf_{n\to\infty} \mathbb{E}_\infty(B_n)$$

and

$$\limsup_{n\to\infty} \sup_{m\geq n} \mathbb{E}_m(B_n) = \limsup_{n\to\infty} \mathbb{E}_\infty(B_n).$$

Proof. We prove only the first statement; the proof of the second statement is analogous. Apply Theorem 4.5.6 (Persistence of Affine Knowledge) to $(\mathrm{Ex}_n(B_n))_n$ to get

$$\liminf_{n\to\infty} \inf_{m\geq n} \mathbb{E}_n^{\mathbb{P}_m}(B_n) = \liminf_{n\to\infty} \mathbb{E}_n^{\mathbb{P}_\infty}(B_n).$$

We now show equalities on these two terms:

1. By Lemma E.2.1,

$$\lim_{n\to\infty} \sup_{m\geq n} |\mathbb{E}_n^{\mathbb{P}_m}(B_n) - \mathbb{E}_m(B_n)| = 0$$

   so

$$\liminf_{n\to\infty} \inf_{m\geq n} \mathbb{E}_m(B_n) = \liminf_{n\to\infty} \inf_{m\geq n} \mathbb{E}_n^{\mathbb{P}_m}(B_n).$$

2. By Lemma E.5.1,

$$\liminf_{n\to\infty} \mathbb{E}_n^{\mathbb{P}_\infty}(B_n) = \liminf_{n\to\infty} \mathbb{E}_\infty(B_n).$$

Together, these three equalities prove the theorem statement. $\square$

E.7 Expectation Coherence

Theorem 4.8.11 (Expectation Coherence)**.** *Let $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$. Then*

$$\liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(B_n) \leq \liminf_{n \to \infty} \mathbb{E}_\infty(B_n) \leq \liminf_{n \to \infty} \mathbb{E}_n(B_n),$$

and

$$\limsup_{n \to \infty} \mathbb{E}_n(B_n) \leq \limsup_{n \to \infty} \mathbb{E}_\infty(B_n) \leq \limsup_{n \to \infty} \sup_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(B_n).$$

Proof. We prove only the first statement; the proof of the second statement is analogous. Apply Theorem 4.5.5 (Affine Coherence) to $(\mathrm{Ex}_n(B_n))_n$ to get

$$\liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{E}_n^{\mathbb{W}}(B_n) \leq \liminf_{n \to \infty} \mathbb{E}_n^{\mathbb{P}_\infty}(B_n) \leq \liminf_{n \to \infty} \mathbb{E}_n(B_n),$$

We now show equalities on the first two terms:

1. Let b be the bound on $\overline{B}$. By Lemma E.1.1,

$$\liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} |\mathbb{E}_n^{\mathbb{W}}(B_n) - \mathbb{W}(B_n)| \leq \liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} b/n = 0$$

   so

$$\liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{E}_n^{\mathbb{W}}(B_n) = \liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(B_n).$$

2. By Lemma E.5.1,

$$\liminf_{n \to \infty} |\mathbb{E}_n^{\mathbb{P}_\infty}(B_n) - \mathbb{E}_\infty(B_n)| = 0$$

   so

$$\liminf_{n \to \infty} \mathbb{E}_n^{\mathbb{P}_\infty}(B_n) = \liminf_{n \to \infty} \mathbb{E}_\infty(B_n).$$

Together, these three equalities prove the theorem statement.

$\square$

E.8 Expectation Provability Induction

Theorem 4.8.10 (Expectation Provability Induction)**.** *Let $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$ and $b \in \mathbb{R}$. If, for all consistent worlds $\mathbb{W} \in \mathcal{PC}(\Gamma)$ and all $n \in \mathbb{N}^+$, it is the case that $\mathbb{W}(B_n) \geq b$, then*

$$\mathbb{E}_n(B_n) \gtrsim_n b,$$

and similarly for $=$ and $\eqsim_n$, and for $\leq$ and $\lesssim_n$.

Proof. We prove the statement in the case of $\geq$; the case of $\leq$ is analogous, and the case of $=$ follows from the other two cases. By Theorem 4.8.11 (Expectation Coherence),

$$\liminf_{n \to \infty} \mathbb{E}_n(B_n) \geq \liminf_{n \to \infty} \inf_{\mathbb{W} \in \mathcal{PC}(\Gamma)} \mathbb{W}(B_n) \geq b.$$

$\square$

E.9 Linearity of Expectation

Theorem 4.8.4 (Linearity of Expectation)**.** *Let $\overline{a}, \overline{b}$ be bounded $\overline{\mathbb{P}}$-generable sequences of rational numbers, and let $\overline{X}, \overline{Y}$, and $\overline{Z}$ be e.c. sequences of $[0,1]$-LUVs. If we have $\Gamma \vdash Z_n = a_n X_n + b_n Y_n$ for all n, then*

$$a_n \mathbb{E}_n(X_n) + b_n \mathbb{E}_n(Y_n) \eqsim_n \mathbb{E}_n(Z_n).$$

Proof. Observe that $\mathbb{W}(a_n X_n + b_n Y_n - Z_n) = 0$ for all n and $\mathbb{W} \in \mathcal{PC}(\Gamma)$. So by Theorem 4.8.10 (Expectation Provability Induction), $\mathbb{E}_n(a_n X_n + b_n Y_n - Z_n) \eqsim_n 0$; the theorem statement immediately follows from the definition of $\mathbb{E}_n$ applied to a LUV-combination (where $a_n X_n + b_n Y_n - Z_n$ is interpreted as a LUV-combination, not another LUV). $\square$

E.10   Expectations of Indicators

Theorem 4.8.6 (Expectations of Indicators)**.** *Let $\overline{\phi}$ be an e.c. sequence of sentences. Then*

$$\mathbb{E}_n(\mathbb{1}(\phi_n)) \eqsim_n \mathbb{P}_n(\phi_n).$$

Proof. Observe that $\mathbb{W}(\mathrm{Ex}_n(\mathbb{1}(\phi_n))) = \mathbb{W}(\phi_n)$ for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$; either

- $\mathbb{W}(\phi) = 0$ and $\mathbb{W}(\mathrm{Ex}_n(\mathbb{1}(\phi_n))) = \sum_{i=0}^{n-1} \frac{1}{n}\mathbb{W}(\text{``}\underline{\mathbb{1}(\phi_n)} > \underline{i}/\underline{n}\text{''}) = \sum_{i=0}^{n} 0 = 0$, or

- $\mathbb{W}(\phi) = 1$ and $\mathbb{W}(\mathrm{Ex}_n(\mathbb{1}(\phi_n))) = \sum_{i=0}^{n-1} \frac{1}{n}\mathbb{W}(\text{``}\underline{\mathbb{1}(\phi_n)} > \underline{i}/\underline{n}\text{''}) = \sum_{i=0}^{n} \frac{1}{n} = 1$.

So by Theorem 4.5.4 (Affine Provability Induction),

$$\mathbb{E}_n(\mathbb{1}(\phi_n)) \eqsim_n \mathbb{P}_n(\phi_n).$$

$\square$

E.11   Expectation Recurring Unbiasedness

Theorem 4.8.15 (Expectation Recurring Unbiasedness)**.** *If $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$ is determined via Γ, and $\overline{w}$ is a $\overline{\mathbb{P}}$-generable divergent weighting,*

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{E}_i(B_i) - \mathrm{Val}_\Gamma(B_i))}{\sum_{i \leq n} w_i}$$

has 0 as a limit point. In particular, if it converges, it converges to 0.

Proof. Let $\mathbb{W} \in \mathcal{PC}(\Gamma)$. Apply Theorem 4.5.9 (Affine Recurring Unbiasedness) to $(\mathrm{Ex}_n(B_n))_n$ and $\overline{w}$ to get that

$$\left(\frac{\sum_{i \leq n} w_i(\mathbb{E}_i(B_i) - \mathbb{E}_i^{\mathbb{W}}(B_n))}{\sum_{i \leq n}} \right)_{n \in \mathbb{N}^+}$$

has 0 as a limit point. Furthermore, by Lemma E.1.1, $|\mathbb{E}_i^{\mathbb{W}}(B_n) - \mathrm{Val}_\Gamma(B_n)| \leq b/n$ where b is a bound on $\overline{B}$. As a result, for any subsequence of

$$\left(\frac{\sum_{i \leq n} w_i(\mathbb{E}_i(B_i) - \mathbb{E}_i^{\mathbb{W}}(B_n))}{\sum_{i \leq n}} \right)_{n \in \mathbb{N}^+}$$

that limits to zero, the corresponding subsequence of

$$\left(\frac{\sum_{i \leq n} w_i(\mathbb{E}_i(B_i) - \mathrm{Val}_\Gamma(B_n))}{\sum_{i \leq n}} \right)_{n \in \mathbb{N}^+}$$

also limits to zero, as desired.

$\square$

E.12   Expectation Unbiasedness From Feedback

Theorem 4.8.16 (Expectation Unbiasedness From Feedback)**.** *Given $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$ that is determined via Γ, a strictly increasing deferral function f such that $\mathrm{Val}_\Gamma(A_n)$ can be computed in time $\mathcal{O}(f(n+1))$, and a $\overline{\mathbb{P}}$-generable divergent weighting w,*

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{E}_i(B_i) - \mathrm{Val}_\Gamma(B_i))}{\sum_{i \leq n} w_i} \eqsim_n 0.$$

In this case, we say "$\overline{w}$ allows good feedback on $\overline{B}$".

Proof. Let $\mathbb{W} \in \mathcal{PC}(\Gamma)$. Note that if $\mathrm{Val}_\Gamma B_n$ can be computed in time polynomial in $g(n+1)$, then so can $\mathrm{Val}_\Gamma \mathrm{Ex}_k(B_n)$. Apply Theorem 4.5.10 (Affine Unbiasedness from Feedback) to $(\mathrm{Ex}_n(B_n))_n$ to get

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{E}_i(B_i) - \mathbb{E}_i^{\mathbb{W}}(B_i))}{\sum_{i \leq n} w_i} \varpropto_n 0.$$

Furthermore, by Lemma E.1.1, $|\mathbb{E}_i^{\mathbb{W}}(B_n) - \mathrm{Val}_\Gamma(B_n)| \leq b/n$ where b is a bound on $\overline{B}$. As a result,

$$\frac{\sum_{i \leq n} w_i \cdot (\mathbb{E}_i(B_i) - \mathrm{Val}_\Gamma(B_i))}{\sum_{i \leq n} w_i} \varpropto_n 0.$$

as desired. $\qquad\square$

E.13 Learning Pseudorandom LUV Sequences

Theorem 4.8.17 (Learning Pseudorandom LUV Sequences). *Given a $\overline{B} \in \mathcal{BLCS}(\overline{\mathbb{P}})$ which is determined via Γ, if there exists a deferral function f such that for any $\overline{\mathbb{P}}$-generable f-patient divergent weighting $\overline{w}$,*

$$\frac{\sum_{i \leq n} w_i \cdot \mathrm{Val}_\Gamma(B_i)}{\sum_{i \leq n} w_i} \gtrsim_n 0,$$

then

$$\mathbb{E}_n(B_n) \gtrsim_n 0.$$

Proof. We will prove the statement in the case of $\gtrsim$; the case of $\lesssim$ is analogous, and the case of $\varpropto$ follows from the other cases.

Let b be the bound of $\overline{B}$. Let $\mathbb{W} \in \mathcal{PC}(\Gamma)$. First, note that by Lemma E.1.1, $|\mathbb{E}_i^{\mathbb{W}}(B_i) - \mathrm{Val}_\Gamma(B_i)| \leq b/i$. Therefore,

$$\frac{\sum_{i \leq n} w_i \cdot \mathbb{E}_i^{\mathbb{W}}(B_i)}{\sum_{i \leq n} w_i} \varpropto_n \frac{\sum_{i \leq n} w_i \cdot \mathrm{Val}_\Gamma(B_i)}{\sum_{i \leq n} w_i} \gtrsim_n 0.$$

So we may apply Theorem 4.5.11 (Learning Pseudorandom Affine Sequences) to $(\mathrm{Ex}_n(B_n))_n$ to get

$$\mathbb{E}_n(B_n) \gtrsim_n 0.$$

$\qquad\square$

F Introspection and Self-Trust Proofs

F.1 Introspection

Theorem 4.11.1 (Introspection). *Let $\overline{\phi}$ be an e.c. sequence of sentences, and $\overline{a}$, $\overline{b}$ be $\overline{\mathbb{P}}$-generable sequences of probabilities. Then, for any e.c. sequence of positive rationals $\overline{\delta} \to 0$, there exists a sequence of positive rationals $\overline{\varepsilon} \to 0$ such that for all n:*

1. *if $\mathbb{P}_n(\phi_n) \in (a_n + \delta_n, b_n - \delta_n)$, then*

$$\mathbb{P}_n(\text{``}\underline{a_n} < \underline{\mathbb{P}_n}(\underline{\phi_n}) < \underline{b_n}\text{''}) > 1 - \varepsilon_n,$$

2. *if $\mathbb{P}_n(\phi_n) \notin (a_n - \delta_n, b_n + \delta_n)$, then*

$$\mathbb{P}_n(\text{``}\underline{a_n} < \underline{\mathbb{P}_n}(\underline{\phi_n}) < \underline{b_n}\text{''}) < \varepsilon_n.$$

Proof. Define $\psi_n := \text{``}\underline{a_n} < \underline{\mathbb{P}_n}(\underline{\phi_n}) < \underline{b_n}\text{''}$.

Proof of the first statement.   Observe that for all n, and all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\mathrm{Ind}_{\delta_n}(a_n < \mathbb{P}_n(\phi_n) < b_n) \cdot (1 - \mathbb{W}(\psi_n)) = 0,$$

since regardless of $\mathbb{P}_n(\phi_n)$, one of the two factors is 0. Thus, applying Theorem 4.5.4 (Affine Provability Induction) gives

$$\mathrm{Ind}_{\delta_n}(a_n < \mathbb{P}_n(\phi_n) < b_n) \cdot (1 - \mathbb{P}_n(\psi_n)) \eqsim_n 0.$$

Define

$$\varepsilon_n := \mathrm{Ind}_{\delta_n}(a_n < \mathbb{P}_n(\phi_n) < b_n) \cdot (1 - \mathbb{P}_n(\psi_n)) + 1/n$$

and note that $\varepsilon_n > 0$ and $\varepsilon_n \eqsim_n 0$. For any n for which $\mathbb{P}_n(\phi_n) \in (a_n + \delta_n, b_n - \delta_n)$, the first factor is 1, so $\mathbb{P}_n(\psi_n) = 1 - \varepsilon_n + 1/n > 1 - \varepsilon_n$.

Proof of the second statement.   Observe that for all n, and all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\big(\mathrm{Ind}_{\delta_n}(\mathbb{P}_n(\phi_n) < a_n) + \mathrm{Ind}_{\delta_n}(\mathbb{P}_n(\phi_n) > b_n)\big) \cdot \mathbb{W}(\psi_n) = 0,$$

since regardless of $\mathbb{P}_n(\phi_n)$, one of the factors is 0. Thus, applying Theorem 4.5.4 (Affine Provability Induction) gives

$$\big(\mathrm{Ind}_{\delta_n}(\mathbb{P}_n(\phi_n) < a_n) + \mathrm{Ind}_{\delta_n}(\mathbb{P}_n(\phi_n) > b_n)\big) \cdot \mathbb{P}_n(\psi_n) \eqsim_n 0.$$

Define

$$\varepsilon_n := \big(\mathrm{Ind}_{\delta_n}(\mathbb{P}_n(\phi_n) < a_n) + \mathrm{Ind}_{\delta_n}(\mathbb{P}_n(\phi_n) > b_n)\big) \cdot \mathbb{P}_n(\psi_n) + 1/n$$

and note that $\varepsilon_n > 0$ and $\varepsilon_n \eqsim_n 0$. For any n for which $\mathbb{P}_n(\phi_n) \notin (a_n - \delta_n, b_n + \delta_n)$, the first factor is 1, so $\mathbb{P}_n(\psi_n) < \varepsilon_n$. $\qquad\square$

F.2   Paradox Resistance

Theorem 4.11.2 (Paradox Resistance)**.** *Fix a rational $p \in (0, 1)$, and define an e.c. sequence of "paradoxical sentences" $\overline{\chi^p}$ satisfying*

$$\Gamma \vdash \underline{\chi_n^p} \leftrightarrow \left(\underline{\mathbb{P}_n}(\underline{\chi_n^p}) < \underline{p}\right)$$

for all n. Then

$$\lim_{n \to \infty} \mathbb{P}_n(\chi_n^p) = p.$$

Proof. We prove $\mathbb{P}_n(\phi_n) \gtrsim_n p$ and $\mathbb{P}_n(\phi_n) \lesssim_n p$ individually.

1. Suppose it is not the case that $\mathbb{P}_n(\phi_n) \gtrsim_n p$, so $\mathbb{P}_n(\phi_n) < p - \varepsilon$ infinitely often for some $\varepsilon > 0$. Observe that for all n, and all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\mathrm{Ind}_{1/n}(\mathbb{P}_n(\phi_n) < p) \cdot (1 - \mathbb{W}(\phi_n)) = 0,$$

since regardless of $\mathbb{P}_n(\phi_n)$, one of the factors is 0. Thus, applying Theorem 4.5.4 (Affine Provability Induction) yields

$$\mathrm{Ind}_{1/n}(\mathbb{P}_n(\phi_n) < p) \cdot (1 - \mathbb{P}_n(\phi_n)) \eqsim_n 0. \qquad (\text{F.2.1})$$

But infinitely often,

$$\mathrm{Ind}_{1/n}(\mathbb{P}_n(\phi_n) < p) \cdot (1 - \mathbb{P}_n(\phi_n)) \geq 1 \cdot (1 - (p - \varepsilon)) \geq \varepsilon$$

which contradicts equation (F.2.1).

114

2. Suppose it is not the case that $\mathbb{P}_n(\phi_n) \lesssim_n p$, so $\mathbb{P}_n(\phi_n) > p + \varepsilon$ infinitely often for some $\varepsilon > 0$. Observe that for all n, and all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\mathrm{Ind}_{1/n}(\mathbb{P}_n(\phi_n) > p) \cdot \mathbb{W}(\phi_n) = 0,$$

since regardless of $\mathbb{P}_n(\phi_n)$, one of the factors is 0. Thus, applying Theorem 4.5.4 (Affine Provability Induction) yields

$$\mathrm{Ind}_{1/n}(\mathbb{P}_n(\phi_n) > p) \cdot \mathbb{P}_n(\phi_n) \eqsim_n 0. \tag{F.2.2}$$

But infinitely often,

$$\mathrm{Ind}_{1/n}(\mathbb{P}_n(\phi_n) > p) \cdot \mathbb{P}_n(\phi_n) \geq 1 \cdot (p + \varepsilon) \geq \varepsilon$$

which contradicts equation (F.2.2).

$\square$

F.3 Expectations of Probabilities

Theorem 4.11.3 (Expectations of Probabilities)**.** *Let $\overline{\phi}$ be an efficiently computable sequence of sentences. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n \mathbb{E}_n(\text{``}\underline{\mathbb{P}_n}(\underline{\phi_n})\text{''}).$$

Proof. Observe that for all n, and for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$, $\mathbb{W}(\mathbb{P}_n(\phi_n) - \text{``}\underline{\mathbb{P}_n}(\underline{\phi_n})\text{''}) = 0$ (where $\mathbb{P}_n(\phi_n)$ is a number and $\text{``}\underline{\mathbb{P}_n}(\underline{\phi_n})\text{''}$) is a LUV). Thus, by Theorem 4.8.10 (Expectation Provability Induction),

$$\mathbb{P}_n(\phi_n) - \mathbb{E}_n(\text{``}\underline{\mathbb{P}_n}(\underline{\phi_n})\text{''}) \eqsim_n 0.$$

$\square$

F.4 Iterated Expectations

Theorem 4.11.4 (Iterated Expectations)**.** *Suppose $\overline{X}$ is an efficiently computable sequence of LUVs. Then*

$$\mathbb{E}_n(X_n) \eqsim_n \mathbb{E}_n(\text{``}\underline{\mathbb{E}_n}(\underline{X_n})\text{''}).$$

Proof. Observe that for all n, and for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$, $\mathbb{W}(\mathbb{E}_n(X_n) - \text{``}\underline{\mathbb{E}_n}(\underline{X_n})\text{''}) = 0$ (where $\mathbb{E}_n(X_n)$ is a number and $\text{``}\underline{\mathbb{E}_n}(\underline{X_n})\text{''}$) is a LUV). Thus, by Theorem 4.8.10 (Expectation Provability Induction),

$$\mathbb{E}_n(X_n) - \mathbb{E}_n(\text{``}\underline{\mathbb{E}_n}(\underline{X_n})\text{''}) \eqsim_n 0.$$

$\square$

F.5 Expected Future Expectations

Theorem 4.12.1 (Expected Future Expectations)**.** *Let f be a deferral function (as per Definition 4.3.7), and let $\overline{X}$ denote an e.c. sequence of $[0,1]$-LUVs. Then*

$$\mathbb{E}_n(X_n) \eqsim_n \mathbb{E}_n(\text{``}\underline{\mathbb{E}_{f(n)}}(\underline{X_n})\text{''}).$$

Proof. Let $Y_m := X_n$ if $m = f(n)$ for some n, and $Y_m := \text{``}0\text{''}$ otherwise. Observe that $(Y_m)_m$ is e.c.. By Theorem 4.11.4 (Iterated Expectations),

$$\mathbb{E}_{f(n)}(X_n) \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{E}_{f(n)}}(\underline{X_n})\text{''}).$$

We now manipulate the encodings $\underline{f(n)}$ (for the number $f(n)$) and $\underline{f}(\underline{n})$ (for the program computing f and its input n). Observe than for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\mathbb{W}(``\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n})") = \mathbb{W}(``\underline{\mathbb{E}}_{\underline{f}(\underline{n})}(\underline{X_n})").$$

So by Theorem 4.8.10 (Expectation Provability Induction),

$$\mathbb{E}_{f(n)}(X_n) \eqsim_n \mathbb{E}_{f(n)}(``\underline{\mathbb{E}}_{\underline{f}(\underline{n})}(\underline{X_n})").$$

By Theorem 4.8.13 (Expectation Preemptive Learning),

$$\mathbb{E}_n(X_n) \eqsim_n \mathbb{E}_n(``\underline{\mathbb{E}}_{\underline{f}(\underline{n})}(\underline{X_n})").$$

$\square$

F.6 No Expected Net Update

Theorem 4.12.2 (No Expected Net Update)**.** *Let f be a deferral function, and let $\overline{\phi}$ be an e.c. sequence of sentences. Then*

$$\mathbb{P}_n(\phi_n) \eqsim_n \mathbb{E}_n(``\underline{\mathbb{P}}_{\underline{f}(\underline{n})}(\underline{\phi_n})").$$

Proof. Let $\psi_m := \phi_n$ if $m = f(n)$ for some n, and $\psi_m := \bot$ otherwise. Observe that $(\psi_m)_m$ is e.c.. By Theorem 4.11.3 (Expectations of Probabilities),

$$\mathbb{P}_{f(n)}(\phi_n) \eqsim_n \mathbb{E}_{f(n)}(``\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n})").$$

We now manipulate the encodings $\underline{f(n)}$ and $\underline{f}(\underline{n})$. Observe that for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\mathbb{W}(``\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n})") = \mathbb{W}(``\underline{\mathbb{P}}_{\underline{f}(\underline{n})}(\underline{\phi_n})").$$

So by Theorem 4.8.10 (Expectation Provability Induction),

$$\mathbb{P}_{f(n)}(\phi_n) \eqsim_n \mathbb{E}_{f(n)}(``\underline{\mathbb{P}}_{\underline{f}(\underline{n})}(\underline{\phi_n})").$$

By Theorem 4.8.6 (Expectations of Indicators),

$$\mathbb{E}_{f(n)}(\mathbb{1}(\phi_n)) \eqsim_n \mathbb{E}_{f(n)}(``\underline{\mathbb{P}}_{\underline{f}(\underline{n})}(\underline{\phi_n})").$$

By Theorem 4.8.13 (Expectation Preemptive Learning),

$$\mathbb{E}_n(\mathbb{1}(\phi_n)) \eqsim_n \mathbb{E}_n(``\underline{\mathbb{P}}_{\underline{f}(\underline{n})}(\underline{\phi_n})").$$

By Theorem 4.8.6 (Expectations of Indicators),

$$\mathbb{P}_n(\phi_n) \eqsim_n \mathbb{E}_n(``\underline{\mathbb{P}}_{\underline{f}(\underline{n})}(\underline{\phi_n})").$$

$\square$

F.7 No Expected Net Update under Conditionals

Theorem 4.12.3 (No Expected Net Update under Conditionals)**.** *Let f be a deferral function, and let $\overline{X}$ denote an e.c. sequence of $[0,1]$-LUVs, and let $\overline{w}$ denote a $\overline{\mathbb{P}}$-generable sequence of real numbers in $[0,1]$. Then*

$$\mathbb{E}_n(``\underline{X_n} \cdot \underline{w}_{\underline{f}(\underline{n})}") \eqsim_n \mathbb{E}_n(``\underline{\mathbb{E}}_{\underline{f}(\underline{n})}(\underline{X_n}) \cdot \underline{w}_{\underline{f}(\underline{n})}").$$

Proof. By Theorem 4.11.4 (Iterated Expectations) and Theorem 4.8.10 (Expectation Provability Induction),

$$\mathbb{E}_{f(n)}(X_n) \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n})\text{''}) \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n})\text{''})$$

and thus

$$\mathbb{E}_{f(n)}(X_n) \cdot w_{f(n)} \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n})\text{''}) \cdot w_{f(n)}.$$

Observe that for all n, and for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\mathbb{W}(X_n) \cdot w_{f(n)} = \mathbb{W}(\text{``}\underline{X_n} \cdot \underline{w}_{\underline{f(n)}}\text{''}).$$

So by Theorem 4.8.10 (Expectation Provability Induction),

$$\mathbb{E}_{f(n)}(X_n) \cdot w_{f(n)} \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{X_n} \cdot \underline{w}_{\underline{f(n)}}\text{''}).$$

Similarly, for all n and all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\mathbb{W}(\text{``}\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n})\text{''}) \cdot w_{f(n)} \eqsim_n \mathbb{W}(\text{``}\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n}) \cdot \underline{w}_{\underline{f(n)}}\text{''}).$$

So by Theorem 4.8.10 (Expectation Provability Induction),

$$\mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n})\text{''}) \cdot w_{f(n)} \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n}) \cdot \underline{w}_{\underline{f(n)}}\text{''}).$$

Combining these,

$$\mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n}) \cdot \underline{w}_{\underline{f(n)}}\text{''}) \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{X_n} \cdot \underline{w}_{\underline{f(n)}}\text{''}).$$

So by Theorem 4.8.13 (Expectation Preemptive Learning),

$$\mathbb{E}_n(\text{``}\underline{\mathbb{E}}_{\underline{f(n)}}(\underline{X_n}) \cdot \underline{w}_{\underline{f(n)}}\text{''}) \eqsim_n \mathbb{E}_n(\text{``}\underline{X_n} \cdot \underline{w}_{\underline{f(n)}}\text{''}).$$

$\square$

F.8 Self-Trust

Theorem 4.12.4 (Self-Trust)**.** *Let f be a deferral function, $\overline{\phi}$ be an e.c. sequence of sentences, $\overline{\delta}$ be an e.c. sequence of positive rational numbers, and $\overline{p}$ be a $\overline{\mathbb{P}}$-generable sequence of rational probabilities. Then*

$$\mathbb{E}_n\left(\text{``}\underline{\mathbb{1}(\phi_n)} \cdot \underline{\text{Ind}}_{\underline{\delta_n}}\left(\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n}) > \underline{p_n}\right)\text{''}\right) \gtrsim_n p_n \cdot \mathbb{E}_n\left(\text{``}\underline{\text{Ind}}_{\underline{\delta_n}}\left(\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n}) > \underline{p_n}\right)\text{''}\right).$$

Proof. Define $\alpha_n := \text{Ind}_{\delta_n}(P_{f(n)}(\phi_n) > p_n)$. By Theorem 4.11.3 (Expectations of Probabilities),

$$\mathbb{P}_{f(n)}(\phi_n) \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n})\text{''})$$

and so

$$\mathbb{P}_{f(n)}(\phi_n) \cdot \alpha_n \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{P}}_{\underline{f(n)}}(\underline{\phi_n})\text{''}) \cdot \alpha_n.$$

Observe that for all $\mathbb{W} \in \mathcal{PC}(\Gamma)$,

$$\mathbb{W}(\mathbb{1}(\phi_n)) \cdot \alpha_n = \mathbb{W}(\text{``}\underline{\mathbb{1}(\phi_n)} \cdot \underline{\alpha}_{\underline{n}}\text{''}).$$

So by Theorem 4.8.6 (Expectations of Indicators) and Theorem 4.8.10 (Expectation Provability Induction),

$$\mathbb{P}_{f(n)}(\phi_n) \cdot \alpha_n \eqsim_n \mathbb{E}_{f(n)}(\mathbb{1}(\phi_n)) \cdot \alpha_n \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{1}(\phi_n)} \cdot \underline{\alpha}_{\underline{n}}\text{''}).$$

117

By two more similar applications of Theorem 4.8.10 (Expectation Provability Induction),

$$\mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{P}_{f(n)}(\phi_n)}\text{''}) \cdot \alpha_n \eqsim_n \mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{P}_{f(n)}(\phi_n)} \cdot \underline{\alpha_n}\text{''}) \gtrsim_n p_n \cdot \mathbb{E}_{f(n)}(\text{``}\underline{\alpha_n}\text{''}).$$

Combining these,

$$\mathbb{E}_{f(n)}(\text{``}\underline{\mathbb{1}(\phi_n)} \cdot \underline{\alpha_n}\text{''}) \gtrsim_n p_n \cdot \mathbb{E}_{f(n)}(\text{``}\underline{\alpha_n}\text{''}).$$

By Theorem 4.8.13 (Expectation Preemptive Learning),

$$\mathbb{E}_n(\text{``}\underline{\mathbb{1}(\phi_n)} \cdot \underline{\alpha_n}\text{''}) \gtrsim_n p_n \cdot \mathbb{E}_n(\text{``}\underline{\alpha_n}\text{''}).$$

$\square$

G Non-Dogmatism and Closure Proofs

G.1 Parametric Traders

Now we show that there is no uniform strategy (i.e., efficiently emulatable sequence of traders) for exposing the logical inductor $\overline{\mathbb{P}}$ to increasing finite amounts of risk, without also taking on increasing finite amounts of risk.

Lemma G.1.1 (Parametric Traders)**.** *Let $\overline{\mathbb{P}}$ be a logical inductor over $\overline{D}$. Then there does not exist an efficiently emulatable sequence of traders $(\overline{T}^k)_k$ such that for all k, the set*

$$\left\{ \mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}}) \right) \,\Big|\, n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n) \right\}$$

of plausible values of $\overline{T}^k$'s holdings is bounded below by -1 and has supremum at least k.

In words, this lemma states that if $\overline{\mathbb{P}}$ is a logical inductor then there is no efficiently emulatable sequence of traders $(\overline{T}^k)_k$ such that each $\overline{T}^k$ never risks more than \$1, and exposes $\overline{\mathbb{P}}$ to at least \$$k$ of plausible risk. To show this lemma, roughly speaking we sum together scaled versions of some of the $\overline{T}^k$ so that the sum of their risks converges but the set of their plausible profits diverges. In this proof only we will use the abbreviation $h(j) := j2^j$ for $j \in \mathbb{N}^+$.

Proof. Suppose for contradiction that such a sequence $(\overline{T}^k)_k$ exists. Define a trader $\overline{T}$ by the formula

$$T_n := \sum_{j:h(j) \leq n} \frac{T_n^{h(j)}}{2^j}.$$

This is well-defined as it is a finite sum of trading strategies, and it is efficiently computable in n because $(\overline{T}^k)_k$ is efficiently emulatable. Then for any time n and any world $\mathbb{W} \in \mathcal{PC}(D_n)$,

$$\mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}}) \right) = \mathbb{W}\left(\sum_{i \leq n} \sum_{j:h(j) \leq n} \frac{T_i^{h(j)}(\overline{\mathbb{P}})}{2^j} \right)$$

by definition of $\overline{T}$ and since $T_n^{h(j)} \equiv 0$ if $h(j) > n$;

$$= \sum_{j:h(j) \leq n} \frac{1}{2^j} \mathbb{W}\left(\sum_{i \leq n} T_i^{h(j)}(\overline{\mathbb{P}}) \right)$$

118

by linearity;

$$\geq \sum_{j \in \mathbb{N}^+} \frac{1}{2^j} \cdot (-1)$$

$$\geq -1,$$

by the assumption that the plausible values $\mathbb{W}\left(\sum_{i \leq n} T_i^{h(j)}(\overline{\mathbb{P}})\right)$ are bounded below by -1. Furthermore, for any $k \in \mathbb{N}^+$, consider the trader $\overline{T}^{h(k)}$. By assumption, for some time n and some world $\mathbb{W} \in \mathcal{PC}(D_n)$, we have

$$\mathbb{W}\left(\sum_{i \leq n} T_i^{h(k)}(\overline{\mathbb{P}})\right) \geq h(k) \equiv k2^k.$$

Then, by the above analysis, we have

$$\mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}})\right) \geq \frac{1}{2^k} \cdot \mathbb{W}\left(\sum_{i \leq n} T_i^{h(k)}(\overline{\mathbb{P}})\right) + \sum_{j \in \mathbb{N}^+} \frac{1}{2^j} \cdot (-1)$$

$$\geq \frac{k2^k}{2^k} - 1$$

$$= k - 1.$$

Thus we have shown that the plausible values $\mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}})\right)$ of our trader $\overline{T}$ are bounded below by -1 but unbounded above, i.e. $\overline{T}$ exploits the market $\overline{\mathbb{P}}$. This contradicts that $\overline{\mathbb{P}}$ is a logical inductor, showing that this sequence $(\overline{T}^k)_k$ cannot exist.

$\square$

G.2   Uniform Non-Dogmatism

Recall Theorem 4.6.3:

Theorem 4.6.3 (Uniform Non-Dogmatism)**.** *For any computably enumerable sequence of sentences $\overline{\phi}$ such that $\Gamma \cup \overline{\phi}$ is consistent, there is a constant $\varepsilon > 0$ such that for all n,*

$$\mathbb{P}_\infty(\phi_n) \geq \varepsilon.$$

Roughly speaking, to show this, we will construct a parametric trader using Lemma G.1.1 by defining an efficiently emulatable sequence $(\overline{T}^k)_k$ of traders. Each trader $\overline{T}^k$ will attempt to "defend" the probabilities of the ϕ_i from dropping too far by buying no more than $k+1$ total shares in various ϕ_i when they are priced below $1/(k+1)$. If the property doesn't hold of $\overline{\mathbb{P}}$, then each $\overline{T}^k$ will buy a full $(k+1)$-many shares, at a total price of at most -1. But since the ϕ_i are all collectively consistent, there is always a plausible world that values the holdings of $\overline{T}^k$ at no less than $k + 1 - 1 = k$. Then the parametric trader that emulates $(\overline{T}^k)_k$ will exploit $\overline{\mathbb{P}}$, contradicting that $\overline{\mathbb{P}}$ is a logical inductor.

Proof. We can assume without loss of generality that for each ϕ_i that appears in the sequence of sentences $\overline{\phi}$, that same sentence ϕ_i appears in $\overline{\phi}$ infinitely often, by transforming the machine that enumerates $\overline{\phi}$ into a machine that enumerates $(\phi_1, \phi_1, \phi_2, \phi_1, \phi_2, \phi_3, \phi_1, \cdots)$. Futhermore, we can assume that $\overline{\phi}$ is efficiently computable by, if necessary, padding $\overline{\phi}$ with copies of $\top$ while waiting for the enumeration to list the next ϕ_i.

Constructing the traders. We now define our sequence $(\overline{T}^k)_k$ of traders. For $n < k$, let T_n^k be the zero trading strategy. For $n \geq k$, define T_n^k to be the trading strategy

$$T_n^k := (k + 1 - \text{Bought}_n^k) \cdot \text{Low}_n^k \cdot (\phi_n - \phi_n^{*n}),$$

where

$$\text{Low}_n^k := \text{Ind}_{1/(2k+2)} \left(\phi_n^{*n} < \frac{1}{k+1} \right)$$

and

$$\text{Bought}_n^k := \sum_{i \leq n-1} \|T_i^k\|_{\text{mg}}.$$

We will make use of the convention for writing the coefficients of a trade:

$$T_n^k[\phi_n] \equiv (k + 1 - \text{Bought}_n^k) \cdot \text{Low}_n^k.$$

In words, T_n^k is a buy order for $(k + 1 - \text{Bought}_n^k)$-many shares of ϕ_n, scaled down by the extent Low_n^k to which ϕ_n is priced below $1/(k+1)$ at time n. The quantity Bought_n^k measures the total number of shares that $\overline{T}^k$ has purchased before the current time step n.

For some fixed polynomial independent of k, the $\overline{T}^k$ are uniformly computable with runtime bounded by that polynomial as a function of n, using the discussion in A.2 on dynamic programming. Furthermore, $T_n^k \equiv 0$ for $n < k$ by definition. Hence $(\overline{T}^k)_k$ is an efficiently emulatable sequence of traders as defined in A.3.4.

Note that by the definition of T_n^k, the magnitude $\|T_n^k(\overline{\mathbb{P}})\|_{\text{mg}}$ of the trade is bounded by $k + 1 - \text{Bought}_n^k(\overline{\mathbb{P}})$. By definition of Bought_n^k and by induction on n, we have that

$$\text{Bought}_1^k(\overline{\mathbb{P}}) = 0 \leq k + 1$$

as Bought_1^k is an empty sum, and

$$\begin{aligned}
\text{Bought}_{n+1}^k(\overline{\mathbb{P}}) &= \sum_{i \leq n+1} \|T_i^k(\overline{\mathbb{P}})\|_{\text{mg}} \\
&= \text{Bought}_n^k(\overline{\mathbb{P}}) + \|T_n^k(\overline{\mathbb{P}})\|_{\text{mg}} \\
&\leq \text{Bought}_n^k(\overline{\mathbb{P}}) + k + 1 - \text{Bought}_n^k(\overline{\mathbb{P}}) \\
&= k + 1.
\end{aligned}$$

In words, $\overline{T}^k$ never trades more than $k + 1$ shares in total. Furthermore, since by definition Low is always non-negative, we have that

$$\|T_n^k(\overline{\mathbb{P}})\|_{\text{mg}} = |T_n^k[\phi_n](\overline{\mathbb{P}})| = |(k + 1 - \text{Bought}_n^k(\overline{\mathbb{P}})) \cdot \text{Low}_n^k(\overline{\mathbb{P}})| \geq 0.$$

Analyzing the value of $\overline{T}^k$. Fix $\overline{T}^k$ and a time step n. For any plausible world $\mathbb{W} \in \mathcal{PC}(D_n)$, the value in $\mathbb{W}$ of holdings from trades made by $\overline{T}^k$ up to time n is

$$\begin{aligned}
\mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}}) \right) &= \mathbb{W}\left(\sum_{i \leq n} T_i^k[\phi_i](\overline{\mathbb{P}}) \cdot (\phi_i - \phi_i^{*i}(\overline{\mathbb{P}})) \right) \\
&= \sum_{i \leq n} T_i^k[\phi_i](\overline{\mathbb{P}}) \cdot \mathbb{W}(\phi_i) \\
&\quad + \sum_{i \leq n} T_i^k[\phi_i](\overline{\mathbb{P}}) \cdot (-\mathbb{P}_i(\phi_i)),
\end{aligned}$$

by linearity and by the definition $\phi_i^{*i}(\overline{\mathbb{P}}) \equiv \mathbb{P}_i(\phi_i)$. We analyze the second term first, which represents the contribution to the value of $\overline{T}^k$ from the prices of the ϕ_i-shares

that it has purchased up to time n. We have that

$$\sum_{i \leq n} T_i^k[\phi_i](\overline{\mathbb{P}}) \cdot (-\mathbb{P}_i(\phi_i))$$

$$\geq \sum_{i \leq n} \|T_i^k\|_{\mathrm{mg}} \cdot \left(-\frac{1}{k+1}\right)$$

since $\|T_i^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = T_i^k[\phi_i](\overline{\mathbb{P}}) \geq 0$ and since $\mathbb{P}_i(\phi_i) \leq 1/(k+1)$ whenever $\mathrm{Low}_i^k(\overline{\mathbb{P}})$ is non-zero;

$$\geq -\frac{k+1}{k+1}$$
$$= -1,$$

since $\overline{T}^k$ never purchases more than $k+1$ shares. Now consider the value

$$\sum_{i \leq n} T_i^k[\phi_i](\overline{\mathbb{P}}) \cdot \mathbb{W}(\phi_i)$$

in $\mathbb{W}$ of the stock holdings from trades made by $\overline{T}^k$ up to time n. Since both $\mathbb{W}(\phi_i)$ and $T_i^k[\phi_i](\overline{\mathbb{P}})$ are non-negative, this value is non-negative. Hence we have shown that

$$\mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}})\right) \geq -1 + 0 = -1,$$

i.e. the total value of $\overline{T}^k$ is bounded below by -1.

Furthermore, since $\Gamma \cup \overline{\phi}$ is consistent, there is always a plausible world $\mathbb{W} \in \mathcal{PC}(D_n)$ such that $\forall i \leq n : \mathbb{W}(\phi_i) = 1$, and therefore

$$\mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}})\right) \geq -1 + \sum_{i \leq n} T_n^k[\phi](\overline{\mathbb{P}}).$$

Exploitation by the parametric trader. Now suppose by way of contradiction that the market $\overline{\mathbb{P}}$ does not satisfy the uniform non-dogmatism property. Then for every k, in particular the property does not hold for $\varepsilon = 1/(2k+2)$, so there is some ϕ_i in the sequence $\overline{\phi}$ such that $\mathbb{P}_\infty(\phi_i) < 1/(2k+2)$. Since by assumption ϕ_i appears infinitely often in $\overline{\phi}$, for some sufficiently large n we have $\mathbb{P}_n(\phi_n) \equiv \mathbb{P}_n(\phi_i) < 1/(2k+2)$, at which point

$$\mathrm{Low}_n^k(\overline{\mathbb{P}}) = \mathrm{Ind}_{1/(2k+2)}\left(\mathbb{P}_n(\phi_n) < \frac{1}{k+1}\right) = 1.$$

Therefore

$$T_n^k[\phi_n] = (k+1 - \mathrm{Bought}_n^k),$$

so that

$$\sum_{i \leq n} \|T_i^k(\overline{\mathbb{P}})\|_{\mathrm{mg}} = \mathrm{Bought}_n^k(\overline{\mathbb{P}}) + k + 1 - \mathrm{Bought}_n^k(\overline{\mathbb{P}}) = k+1.$$

Thus

$$\mathbb{W}\left(\sum_{i \leq n} T_i^k(\overline{\mathbb{P}})\right) \geq -1 + k + 1 = k.$$

In words, once the price of some ϕ_i dips below $1/(2k+2)$, the trader $\overline{T}^k$ will purchase the remaining $k+1 - \mathrm{Bought}_n^k(\overline{\mathbb{P}})$ shares it will ever buy. Then in a world $\mathbb{W}$ that witnenesses that $\overline{\phi}$ is consistent with Γ, all the shares held by $\overline{T}^k$ are valued at \$1 each, so $\overline{T}^k$ has stock holdings valued at $k+1$, and cash holdings valued at no less than -1.

121

Therefore each $\overline{T}^k$ has plausible value bounded below by -1 and at least k in some plausible world at some time step, and therefore Lemma G.1.1 applies, contradicting that $\overline{\mathbb{P}}$ is a logical inductor. Therefore in fact $\overline{\mathbb{P}}$ does satisfy the uniform non-dogmatism property.

$\square$

G.3   Occam Bounds

Recall Theorem 4.6.4:

Theorem 4.6.4 (Occam Bounds)**.** *There exists a fixed positive constant C such that for any sentence ϕ with prefix complexity $\kappa(\phi)$, if $\Gamma \nvdash \neg\phi$, then*

$$\mathbb{P}_\infty(\phi) \geq C2^{-\kappa(\phi)},$$

and if $\Gamma \nvdash \phi$, then

$$\mathbb{P}_\infty(\phi) \leq 1 - C2^{-\kappa(\phi)}.$$

We show the result for ϕ such that $\Gamma \nvdash \neg\phi$; the result for $\Gamma \nvdash \phi$ follows by considering $\neg\neg\phi$ and using the coherence of $\mathbb{P}_\infty$.

Roughly speaking, we will construct an efficiently emulatable sequence of traders $(\overline{T}^k)_k$ where $\overline{T}^k$ attempts to ensure that $\mathbb{P}_n(\phi)$ does not drop below $2^{-\kappa(\phi)}/(k+1)$ for any ϕ. We do this by having $\overline{T}^k$ purchase shares in any ϕ that are underpriced in this way, as judged by a computable approximation from below of $2^{-\kappa(\phi)}$. The trader $\overline{T}^k$ will purchase at most $k+1$ shares in each ϕ, and hence spend at most \$$2^{-\kappa(\phi)}$ for each ϕ and at most \$1 in total. On the other hand, if the market $\overline{\mathbb{P}}$ does not satisfy the Occam property with constant $C = 1/(k+1)$, then for some ϕ with $\Gamma \nvdash \neg\phi$, we will have that $\overline{T}^k$ purchases a full $k+1$ shares in ϕ. Since there is always a plausible world that values ϕ at \$1, $\overline{T}^k$ will have a plausible value of at least \$$k$, taking into account the \$1 maximum total prices paid. This contradicts Lemma G.1.1, so in fact $\overline{\mathbb{P}}$ satisfies the Occam property.

To implement this strategy, we will use tools similar to those used in the proof of Theorem 4.6.3, and the proof is similar in spirit, so we will elide some details.

Proof. Observe that $2^{-\kappa(\phi)}$ is approximable from below uniformly in ϕ, since we can (slowly) enumerate all prefixes on which our fixed UTM halts and outputs ϕ. Let $\overline{\phi}$ be an efficiently computable enumeration of all sentences. Let M be a Turing machine that takes an index i into our enumeration and takes a time n, and outputs a non-negative rational number. We further specify that M runs in time polynomial in $i+n$, satisfies $\forall n, i : M(n, i) \leq 2^{-\kappa(\phi_i)}$, and satisfies $\lim_{t\to\infty} M(n, i) = 2^{-\kappa(\phi_i)}$. Note that since we are using prefix complexity, we have $\sum_{\phi\in\mathcal{S}} 2^{-\kappa(\phi)} \leq 1$. (We can assume without loss of generality that $M(n, i) > 0$ for all $i \leq n$, by padding the sequence $\overline{\phi}$ with ϕ_1 while waiting until $M(n, i) > 0$ to enumerate ϕ_i, using the fact that our UTM outputs each ϕ for some prefix.)

We define a sequence of traders $(\overline{T}^k)_k$. For $n < k$, define T_n^k to be the zero trading strategy. For $n \geq k$, define T_n^k to be the trading strategy given by

$$T_n^k := \sum_{i\leq n}(k + 1 - \mathrm{Bought}_n^k(i)) \cdot \mathrm{Low}_n^k(i) \cdot (\phi_i - \phi_i^{*n}),$$

where

$$\mathrm{Low}_n^k(i) := \mathrm{Ind}_{M(n,i)/(2k+2)}\left(\phi^{*n} < \frac{M(n,i)}{k+1}\right)$$

and

$$\mathrm{Bought}_n^k(i) := \sum_{j\leq n-1} T_j^k[\phi_i].$$

This is similar to the parametric trader used in the proof of Theorem 4.6.3, except that here on time n, $\overline{T}^k$ buys any ϕ_i when $\mathbb{P}_n(\phi_i)$ is too low, and $\text{Bought}_n^k(i)$ tracks the number of shares bought by $\overline{T}^k$ in each ϕ_i individually up to time n. By the discussion of dynamic programming in A.2, $(\overline{T}^k)_k$ is an efficiently emulatable sequence of traders. (We use that $M(n, i)$ is pre-computed by the machine that computes $\overline{T}^k$, and hence appears in the feature expression for T_n^k as a constant which is strictly positive by assumption.)

Observe that for each i, by induction on n we have $\text{Bought}_n^k(i) \leq k + 1$, so that $\overline{T}^k$ only buys positive shares in the various ϕ_i, and $\overline{T}^k$ only buys up to $k + 1$ shares of ϕ_i. Further, $\overline{T}^k$ only buys ϕ_i-shares at time n if the price $\mathbb{P}_n(\phi_i)$ is below $1/(k+1)$-th of the approximation $M(n, i)$ to $2^{-\kappa(\phi)}$, i.e.

$$\mathbb{P}_n(\phi_i) < \frac{M(n, i)}{k + 1} \leq \frac{2^{-\kappa(\phi_i)}}{k + 1}.$$

Therefore $\overline{T}^k$ spends at most $\$2^{-\kappa(\phi_i)}$ on ϕ_i-shares, and hence spends at most $\$1$ in total.

Suppose for contradiction that $\overline{\mathbb{P}}$ does not satisfy the Occam property. Then for every k there exists some ϕ_i such that the limiting price of ϕ_i is

$$\mathbb{P}_\infty(\phi_i) < \frac{2^{-\kappa(\phi_i)}}{(2k + 2)},$$

but nevertheless $\Gamma \nvdash \neg\phi_i$. Then for some time step n we will have that $\mathbb{P}_n(\phi_i) < M(n, i)/(2k + 2)$, and hence $\text{Low}_n^k(i) = 1$. At that point $\overline{T}^k$ will purchase $k + 1 - \text{Bought}_n^k(i)$ shares in ϕ_i, bringing $\text{Bought}_{n+1}^k(i)$ to $k + 1$; that is, $\overline{T}^k$ will have bought $k + 1$ shares of ϕ_i. Since ϕ is consistent with Γ, there is some plausible world $\mathbb{W} \in \mathcal{PC}(D_n)$ that values those shares at $\$1$ each, so that the total value of all of the holdings from trades made by $\overline{T}^k$ is at least k. By Lemma G.1.1 this contradicts that $\overline{\mathbb{P}}$ is a logical inductor, so in fact $\overline{\mathbb{P}}$ must satisfy the Occam property. $\qquad\square$

G.4   Non-Dogmatism

Theorem 4.6.2 (Non-Dogmatism)**.** *If* $\Gamma \nvdash \phi$ *then* $\mathbb{P}_\infty(\phi) < 1$, *and if* $\Gamma \nvdash \neg\phi$ *then* $\mathbb{P}_\infty(\phi) > 0$.

Proof. This is a special case of 4.6.4, since $\kappa(\phi) > 0$ for any ϕ. $\qquad\square$

G.5   Domination of the Universal Semimeasure

Theorem 4.6.5 (Domination of the Universal Semimeasure)**.** *Let* $(b_1, b_2, \ldots)$ *be a sequence of zero-arity predicate symbols in* $\mathcal{L}$ *not mentioned in* Γ*, and let* $\sigma_{\leq n} = (\sigma_1, \ldots, \sigma_n)$ *be any finite bitstring. Define*

$$\mathbb{P}_\infty(\sigma_{\leq n}) := \mathbb{P}_\infty(\text{``}(b_1 \leftrightarrow \underline{\sigma_1} = 1) \wedge (b_2 \leftrightarrow \underline{\sigma_2} = 1) \wedge \ldots \wedge (b_n \leftrightarrow \underline{\sigma_n} = 1)\text{''}),$$

such that, for example, $\mathbb{P}_\infty(01101) = \mathbb{P}_\infty(\text{``}\neg b_1 \wedge b_2 \wedge b_3 \wedge \neg b_4 \wedge b_5\text{''})$*. Let* M *be a universal continuous semimeasure. Then there is some positive constant* C *such that for any finite bitstring* $\sigma_{\leq n}$*,*

$$\mathbb{P}_\infty(\sigma_{\leq n}) \geq C \cdot M(\sigma_{\leq n}).$$

Proof. Let $(\overline{\sigma}^i)_i$ be an e.c. enumeration of all finite strings. Let $l(\overline{\sigma})$ be the length of $\overline{\sigma}$. Define

$$\phi_i := \text{``}(b_1 \leftrightarrow \underline{\sigma_1^i} = 1) \wedge (b_2 \leftrightarrow \underline{\sigma_2^i} = 1) \wedge \ldots \wedge (b_{l(\overline{\sigma}^i)} \leftrightarrow \underline{\sigma_{l(\overline{\sigma}^i)}^i} = 1)\text{''}$$

to be the sentence saying that the string $(b_1, b_2, \ldots)$ starts with σ^i. Suppose the theorem is not true; we will construct a sequence of parametric traders to derive a contradiction through Lemma G.1.1.

Defining a sequence of parametric traders. To begin, let $A(\overline{\sigma}, n)$ be a lower-approximation of $M(\overline{\sigma})$ that can be computed in time polynomial in n and the length of $\overline{\sigma}$. Specifically, A must satisfy

- $A(\overline{\sigma}, n) \leq M(\overline{\sigma})$, and

- $\lim_{n \to \infty} A(\overline{\sigma}, n) = M(\overline{\sigma})$.

Now, recursively define

$$
\alpha^{\dagger}_{k,n,i} := \begin{cases} \mathrm{Ind}_{\frac{1}{4(k+1)}} \left(\frac{\phi^{*n}_i}{A(\overline{\sigma}^i, n)} < \frac{1}{2(k+1)} \right) & \text{if } A(\overline{\sigma}^i, n) > 0 \wedge n \geq k \wedge i \leq n \\ 0 & \text{otherwise} \end{cases}
$$

$$
\beta^{\dagger}_{k,n,i} := \min(\alpha^{\dagger}_{k,n,i}, 1 - \gamma^{\dagger}_{k,n,i})
$$

$$
\gamma^{\dagger}_{k,n,i} := \sum_{m<n, j \leq m} \beta^{\dagger}_{k,m,j} \phi^{*m}_j + \sum_{j<i} \beta^{\dagger}_{k,n,j} \phi^{*n}_j
$$

in order to define a parametric trader

$$
T^k_n := \sum_{i \leq n} \beta_{k,n,i} \cdot (\phi_i - \phi^{*n}_i)
$$

As shorthand, we write $\alpha_{k,n,i} := \alpha^{\dagger}_{k,n,i}(\overline{\mathbb{P}})$, $\beta_{k,n,i} := \beta^{\dagger}_{k,n,i}(\overline{\mathbb{P}})$, $\gamma_{k,n,i} := \gamma^{\dagger}_{k,n,i}(\overline{\mathbb{P}})$. Intuitively,

- $\alpha_{k,n,i}$ is the number of copies ϕ_i that $\overline{T}^k$ would buy on day n if it were not for budgetary constraints. It is high if $\mathbb{P}_n$ obviously underprices ϕ_i relative to M (which can be checked by using the lower-approximation A).

- $\beta_{k,n,i}$ is the actual number of copies of ϕ_i that $\overline{T}^k$ buys, which is capped by budgetary constraints.

- $\gamma_{k,n,i}$ is the amount of money T^k has spent on propositions $\phi_1, \ldots, \phi_{n-1}$ "before considering" buying ϕ_i on day n. We imagine that, on each day n, the trader goes through propositions in the order $\phi_1, \ldots, \phi_n$.

Analyzing the sequence of parametric traders. Observe that T^k spends at most \$1 in total, since $\beta_{k,n,i} \leq 1 - \gamma_{k,n,i}$. Now we will analyze the trader's maximum payout. Assume by contradiction that $\mathbb{P}_\infty$ does not dominate M. Define

$$
\mathrm{Purchased}_{k,n,i} := \sum_{m \leq n} \beta_{k,n,i}
$$

to be the number of shares of $\overline{\sigma}^i$ that T has bought by time n, and

$$
\mathrm{MeanPayout}_{k,n} := \sum_{i \in \mathbb{N}^+} M(\overline{\sigma}^i) \mathrm{Purchased}_{k,n,i}.
$$

to be the "mean" value of stocks purchased by time n according to the semimeasure M. Both of these quantities are nondecreasing in n. Now we show that there is some N such that $\mathrm{MeanPayout}_{k,n} \geq k + 1$ for all $n \geq N$:

- Every purchase costing c corresponds to $\mathrm{MeanPayout}_{k,n}$ increasing by at least $c \cdot 2(k+1)$. This is because the trader only buys ϕ_i when $\frac{\mathbb{P}_n(\phi_i)}{A(\overline{\sigma}_i, n)} < \frac{1}{2(k+1)}$, and $A(\overline{\sigma}_i, n) \leq M(\overline{\sigma}_i)$.

- For some N, $\text{MeanPayout}_{k,N} \geq k + 1$. Suppose this is not the case. Since we are supposing that $\mathbb{P}_\infty$ does not dominate the universal semimeasure, there is some i such that $\mathbb{P}_\infty(\phi_i) < \frac{M(\overline{\sigma}^i)}{8(k+1)}$. So we will have $\mathbb{P}_n(\phi_i) < \frac{M(\overline{\sigma}^i)}{8(k+1)}$ for infinitely many n; let $\mathcal{N}$ be the set of such n.

  For all sufficiently high n we have $A(\overline{\sigma}^i, n) \geq M(\overline{\sigma}^i)/2$, so for all sufficiently high $n \in \mathcal{N}$,

  $$\frac{\mathbb{P}_n(\phi_i)}{A(\overline{\sigma}^i, n)} \leq \frac{\mathbb{P}_n(\phi_i)}{M(\overline{\sigma}^i)/2} \leq \frac{1}{4(k+1)}$$

  and so there is some infinite subset $\mathcal{N}' \subseteq \mathcal{N}$ for which $\alpha_{k,n,i} = 1$. By assumption, $\forall n : \text{MeanPayout}_{k,n} < k + 1$, so the trader never has spent more than \$1/2 (using the previous step), so $\gamma_{k,n,i} \leq 1/2$. This means $\beta_{k,n,i} \geq 1/2$, which implies an increase in mean payout $\text{MeanPayout}_{k,n} - \text{MeanPayout}_{k,n-1} \geq M(\overline{\sigma}^i) > 0$. But this increase happens for infinitely many n, so $\lim_{n \to \infty} \text{MeanPayout}_{k,n} = \infty$. This contradicts the assumption that $\text{MeanPayout}_{k,N} < k + 1$ for all N.

- $\text{MeanPayout}_{k,N}$ is nondecreasing in N, so $\text{MeanPayout}_{k,n} \geq k + 1$ for all $n \geq N$.

Using this lower bound on $\text{MeanPayout}_{k,n}$, we would now like to show that T^k's purchases pay out at least $k + 1$ in some $W \in \mathcal{PC}(D_\infty)$. To do this, define

$$\text{MaxPayout}_{k,n} := \sup_{\overline{\sigma} \in \mathbb{B}^{\leq \mathbb{N}+}} \sum_{\sigma'^i \text{ prefix of } \overline{\sigma}} \text{Purchased}(\overline{\sigma}^i, n)$$

to be the maximum amount that T^k's purchases pay out over all possible strings (finite and infinite). Since M is a semimeasure over finite and infinite bitstrings, we have $\text{MeanPayout}(n) \leq \text{MaxPayout}(n)$. Since each ϕ_i is independent of Γ, $\overline{T}^k$'s maximum worth is at least

$$\limsup_{n \to \infty} \text{MaxPayout}(\overline{\varepsilon}, n) - 1 \geq \limsup_{n \to \infty} \text{MeanPayout}(\overline{\varepsilon}, n) - 1 \geq k + 1 - 1 = k.$$

This is sufficient to show a contradiction using Lemma G.1.1. $\qquad\square$

G.6   Strict Domination of the Universal Semimeasure

Recall Theorem 4.6.6 (Strict Domination of the Universal Semimeasure):

Theorem 4.6.6 (Strict Domination of the Universal Semimeasure). *The universal continuous semimeasure does not dominate $\mathbb{P}_\infty$; that is, for any positive constant C there is some finite bitstring $\sigma_{\leq n}$ such that*

$$\mathbb{P}_\infty(\sigma_{\leq n}) > C \cdot M(\sigma_{\leq n}).$$

Proof. Consider the sets of codes for Turing machines

$$A_0 := \{M \mid M \text{ halts on input 0 and outputs 0}\}$$

and

$$A_1 := \{M \mid M \text{ halts on input 0 and outputs 1}\}.$$

Both of these sets are computably enumerable and disjoint, so by Theorem 4.6.3 (Uniform Non-Dogmatism), $\mathbb{P}_\infty$ assigns positive measure to the set $[A]$ of infinite bitstrings that encode a separating set for A_0 and A_1, i.e., a set A such that $A \cap A_0 = \emptyset$ and $A \supseteq A_1$.

Thus it suffices to show that the universal semimeasure assigns 0 measure to $[A]$. This is a known result from computability theory, using the fact that A_0 and A_1

are recursively inseparable; see for example Kucera and Nies 2011. Here we give an elementary proof sketch.

Suppose for contradiction that m computes a universal semimeasure and $m([A]) = r > 0$; we argue that we can compute some separating set A. Let $q \in [4r/5, r] \cap \mathbb{Q}$. There is some fixed k such that the finite binary subtree A_k consisting of finite prefixes of length k of strings in $[A]$ is assigned $m(A_k) \in [r, 6r/5]$.

On input n, we can run m on the set of strings of length up to n until the set of extensions of strings in A_k has measure at least q; this will happen because $m([A]) > q$. Then we output 0 if the majority of the measure is on strings with nth bit equal to 0, and we output 1 otherwise. If we output 0 but in fact $n \in A_1$, then there is measure at most $6r/5 - 2r/5 = 4r/5$ on extensions of strings in A_k that are consistent with separating A_0 and A_1; but this is impossible, as $[A]$ has measure r. Likewise if we output 1 then it can't be that $n \in A_0$. Thus we have recursively separated A_0 and A_1, contradicting that A_0 and A_1 are recursively inseparable. $\qquad\square$

G.7   Closure under Finite Perturbations

Recall Theorem 4.6.1 (Closure under Finite Perturbations):

Theorem 4.6.1 (Closure under Finite Perturbations)**.** *Let $\overline{\mathbb{P}}$ and $\overline{\mathbb{P}'}$ be markets with $\mathbb{P}_n = \mathbb{P}'_n$ for all but finitely many n. Then $\overline{\mathbb{P}}$ is a logical inductor if and only if $\overline{\mathbb{P}'}$ is a logical inductor.*

In short, a trader that exploits $\overline{\mathbb{P}}$ also exploits $\overline{\mathbb{P}'}$ since all but finitely many of its trades are identically valued. The proof mainly concerns a minor technical issue; we have to make a small adjustment to the trader to ensure that it makes exactly the same trades against $\overline{\mathbb{P}'}$ as it does against $\overline{\mathbb{P}}$.

Proof. Assume there is a trader $\overline{T}$ which exploits $\overline{\mathbb{P}}$. We will construct a new trader $\overline{T}'$ that exploits $\overline{\mathbb{P}'}$. Fix N large enough that $\mathbb{P}_n = \mathbb{P}'_n$ for all $n \geq N$.

We will define $\overline{T}'$ so that it makes the same trades against the market $\overline{\mathbb{P}'}$ as the trader $\overline{T}$ makes against $\overline{\mathbb{P}}$. That is, we want that for all n,

$$T'_n(\overline{\mathbb{P}'}) = T_n(\overline{\mathbb{P}}).$$

It is insufficient to set the trading strategy T'_n equal to T_n for all n. This is because T_n may infinitely often make different trades given the history $\mathbb{P}'_{\leq n}$ instead of the history $\mathbb{P}_{\leq n}$. For example, it may be that every day $\overline{T}$ buys $\mathbb{V}_1(\phi)$-many shares in ϕ against $\mathbb{V}$; in this case if $\mathbb{P}'_1(\phi) \neq \mathbb{P}_1(\phi)$, then at each time n, $T_n(\overline{\mathbb{P}'})$ will buy a different number of shares from $T_n(\overline{\mathbb{P}})$. Roughly speaking, we will patch this problem by copying $\overline{T}$, but feeding it "false reports" about the market prices so that it appears to the T_n that they are reacting to $\overline{\mathbb{P}}$ rather than $\overline{\mathbb{P}'}$.

More precisely, let F be a computable function from feature expressions to feature expressions, in the expression language discussed in A.2. For a feature expression α, we define $F(\alpha)$ to be identical to α but with all occurrences of an expression ϕ^{*i} for $i < N$ replaced by a constant $\mathbb{P}_i(\phi)$.

Note that F is efficiently computable: by the assumption that $\mathbb{P}_n = \mathbb{P}'_n$ for all $n \geq N$, only finitely many constants $\mathbb{P}_i(\phi)$ are needed, and can be hard-coded into F. Furthermore, F behaves as intended: for any α, we have $F(\alpha)(\overline{\mathbb{P}'}) = \alpha(\overline{\mathbb{P}})$ (using a slight abuse of notation, treating α as both an expression and as the feature thus expressed). This follows by structural induction the expression α, where every step is trivial except the base cases for symbols ϕ^{*i} with $i < N$, which follow from the definition of F. Now we define

$$T'_n := \sum_{\phi \in \mathcal{S}} F(T_n[\phi])(\phi - \phi^{*n})$$

for any n. This is efficiently computable because T_n and F are both e.c. Furthermore, for all $n \geq N$, we have that $T'_n(\overline{\mathbb{P}'}) = T_n(\overline{\mathbb{P}})$. Therefore for any n we have that

$$\left| \mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}}) \right) - \mathbb{W}\left(\sum_{i \leq n} T'_i(\overline{\mathbb{P}'}) \right) \right|$$
$$\leq \left| \mathbb{W}\left(\sum_{i < N} T_i(\overline{\mathbb{P}}) \right) - \mathbb{W}\left(\sum_{i < N} T'_i(\overline{\mathbb{P}'}) \right) \right|,$$

which is a fixed constant, where we use that all terms for $i \geq N$ cancel with each other. This says that at all times and all plausible worlds, there is a fixed upper bound on the difference between the values of $\overline{T}$ against $\overline{\mathbb{P}}$ and of $\overline{T}'$ against $\overline{\mathbb{P}'}$. Thus if

$$\left\{ \mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}}) \right) \,\middle|\, n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n) \right\}$$

is bounded below but unbounded above, then so is

$$\left\{ \mathbb{W}\left(\sum_{i \leq n} T'_i(\overline{\mathbb{P}'}) \right) \,\middle|\, n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n) \right\}.$$

Therefore, if some trader exploits $\overline{\mathbb{P}}$, so that $\overline{\mathbb{P}}$ is not a logical inductor, then some trader exploits $\overline{\mathbb{P}'}$, so $\overline{\mathbb{P}'}$ also fails to be a logical inductor. Symmetrically, if $\overline{\mathbb{P}'}$ is not a logical inductor, then neither is $\overline{\mathbb{P}}$. □

G.8 Conditionals on Theories

Theorem 4.7.2 (Closure Under Conditioning). *The sequence $\overline{\mathbb{P}}(- \mid \psi)$ is a logical inductor over $\Gamma \cup \{\psi\}$. Furthermore, given any efficiently computable enumeration $\Psi \subseteq \mathcal{S}$ of sentences, the sequence*

$$\left(\mathbb{P}_1(- \mid \psi_1), \mathbb{P}_2(- \mid \psi_1 \wedge \psi_2), \mathbb{P}_3(- \mid \psi_1 \wedge \psi_2 \wedge \psi_3), \dots \right),$$

where the nth pricing is conditioned on the first n sentences in Ψ, is a logical inductor over $\Gamma \cup \Psi$.

Since $\overline{\mathbb{P}}$ is a logical inductor over Γ, there is some Γ-complete deductive process $\overline{D}$ over which $\overline{\mathbb{P}}$ is a logical inductor. Let $\overline{D}'$ be any deductive process. Write

$$\psi_n^\circ := \bigwedge_{\psi \in D'_n} \psi$$

for the conjunction of all sentences ψ that have appeared in $\overline{D}'$ up until time n. (We take the empty conjunction to be the sentence $\top$.) Write $\overline{\mathbb{P}}^\circ$ to mean the market $(\mathbb{P}_n(- \mid \psi_n^\circ))_{n \in \mathbb{N}^+}$.

We will show the slightly more general fact that for any e.c. $\overline{D}'$, if the theory

$$\Gamma \cup \{\psi' \mid \exists n : \psi' \in D'_n\}$$

is consistent, then $\overline{\mathbb{P}}^\circ$ is a logical inductor over the deductive process $\overline{D}^\circ$, complete for that theory, defined for any n by $D^\circ{}_n := D_n \cup D'_n$. This implies the theorem by specializing to the $\{\psi\}$-complete deductive process $(\{\psi\}, \{\psi\}, \{\psi\}, \dots)$, and to the Ψ-complete deductive process $(\{\psi_1\}, \{\psi_1, \psi_2\}, \{\psi_1, \psi_2, \psi_3\}, \dots)$ (where we pad with $\top$ to ensure this sequence is efficiently computable).

Roughly speaking, we'll take a supposed trader $\overline{T}^\circ$ that exploits $\overline{\mathbb{P}}^\circ$ and construct a trader $\overline{T}$ that exploits $\overline{\mathbb{P}}$. We'd like our trader $\overline{T}$ to mimic $\overline{T}^\circ$ "in the worlds where ψ_n° is true", and otherwise remain neutral. A first attempt would be to have our trader buy the combination

$$\phi \wedge \psi_n^\circ - \frac{\mathbb{P}_n(\phi \wedge \psi_n^\circ)}{\mathbb{P}_n(\psi_n^\circ)} \cdot \psi_n^\circ$$

whenever $\overline{T}^\circ$ buys a share in ϕ. The idea is to make a purchase that behaves like a conditional contract that pays out if ϕ is true but only has any effect in worlds where ψ_n° is true. That is, the hope is that the price of this combination is 0; in worlds where ψ_n° is false, the stock holdings from this trade are valued at 0; and in worlds where ψ_n° is true, the stock holdings have the same value as that of purchasing a ϕ-share against $\overline{\mathbb{P}}^\circ$.

There are some technical problems with the above sketch. First, the ratio of probabilities in front of ψ_n° in the above trade is not well-defined if $\mathbb{P}_n(\psi_n^\circ) = 0$; we will fix this using a safe reciprocation, after correcting the market using Lemma G.7 (closure under finite perturbations). Second, if $\mathbb{P}_n(\phi \wedge \psi_n^\circ)$ is greater than $\mathbb{P}_n(\psi_n^\circ)$, then their ratio is not the same as the conditional probability $\mathbb{P}_n(\phi \mid \psi_n^\circ) = \mathbb{P}^\circ{}_n(\phi)$ as defined in 4.7.1; in this case, our trader has stock holdings with a different value from those of the original trader exploiting $\overline{\mathbb{P}}^\circ$. If we simply replace the ratio with the conditional probability $\mathbb{P}_n(\phi \mid \psi_n^\circ)$, then when $\mathbb{P}_n(\phi \wedge \psi_n^\circ) > \mathbb{P}_n(\psi_n^\circ)$, the value of the cash holdings is different. Instead we will replace the ratio with a market feature that acts like the ratio when the trade is a sell order, but becomes the conditional probability when the trade is a buy order. We now implement this strategy.

Proof. Let $\overline{D}$, $\overline{D}^\circ$, and $\overline{\mathbb{P}}^\circ$ be defined as above.

We may assume that the collection of sentences that appear in $\overline{D}^\circ$ is consistent. If not, then no trader exploits $\overline{\mathbb{P}}^\circ$: for all sufficiently large n the set of plausible worlds $\mathcal{PC}(D^\circ{}_n)$ is empty, so the set of plausible values of any trader's holdings is a finite set, and hence bounded above.

We may further assume without loss of generality that there exists a rational $\varepsilon > 0$ such that $\mathbb{P}_n(\psi_n^\circ) > \varepsilon$ for all n. Indeed, by Theorem 4.6.3 (uniform non-dogmatism), since $\overline{D}^\circ$ is consistent, there is some $\varepsilon > 0$ such that $\mathbb{P}_\infty(\psi_n^\circ) > \varepsilon$ for all sufficiently large n. Hence by Theorem 4.2.4 (preemptive learning), we have $\liminf_{n\to\infty} \mathbb{P}_n(\psi_n^\circ) > \varepsilon$. This implies that there are only finitely many time steps n such that $\mathbb{P}_n(\psi_n^\circ) \leq \varepsilon$. Therefore by Lemma G.7 (closure under finite perturbations), the market $\overline{\mathbb{P}}'$ defined to be identical to $\overline{\mathbb{P}}$ except with $\mathbb{P}'_n(\psi_n^\circ)$ with 1 for all such n is still a logical inductor, and has the desired property. If we show that $\overline{\mathbb{P}}^{\circ}{}'$ is a logical inductor, then again by Lemma G.7, $\overline{\mathbb{P}}^\circ$ is also a logical inductor.

Now suppose some trader $\overline{T}^\circ$ exploits $\overline{\mathbb{P}}^\circ$. We will construct a trader $\overline{T}$ that exploits $\overline{\mathbb{P}}$. Consider the $\mathcal{EF}$-combination

$$\text{Sell}_n(\phi) := \phi \wedge \psi_n^\circ - \frac{(\phi \wedge \psi_n^\circ)^{*n}}{\max(\varepsilon, \psi_n^{\circ *n})} \cdot \psi_n^\circ$$

and

$$\text{Buy}_n(\phi) := \phi \wedge \psi_n^\circ - \max\left(1, \frac{(\phi \wedge \psi_n^\circ)^{*n}}{\max(\varepsilon, \psi_n^{\circ *n})}\right) \cdot \psi_n^\circ,$$

parametrized by a sentence ϕ. We write $(\text{Sell}_n(\phi))^{*n}$ for the expressible feature that computes the price of the $\mathcal{EF}$-combination $\text{Sell}_n(\phi)$ at time n, defined in the natural way by replacing sentences with their $*n$ duals, and likewise for $(\text{Buy}_n(\phi))^{*n}$.

Now define the trader $\overline{T}$ by setting

$$T_n := \sum_\phi \max(0, T_n^\circ[\phi]_\circ) \cdot (\text{Buy}_n(\phi) - (\text{Buy}_n(\phi))^{*n})$$
$$- \max(0, -T_n^\circ[\phi]_\circ) \cdot (\text{Sell}_n(\phi) - (\text{Sell}_n(\phi))^{*n})$$

where $T_n^\circ[\phi]_\circ$ is defined to be the market feature $T_n^\circ[\phi]$ with every occurrence of χ^{*i} replaced with

$$\max\left(1, \frac{(\chi \wedge \psi_i^\circ)^{*i}}{\max(\varepsilon, \psi_i^{\circ *i})}\right).$$

That is, $T_n^\circ[\phi]_\circ$ is defined so that $T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) = T_n^\circ[\phi](\overline{\mathbb{P}}^\circ)$, i.e., this market feature behaves against the market $\overline{\mathbb{P}}$ just as $T_n^\circ[\phi]$ behaves against the conditional market $\overline{\mathbb{P}}^\circ$.

We analyze the value of the trades made by $\overline{T}$ against $\overline{\mathbb{P}}$ according to each term in the above summation and by cases. First suppose that $T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \geq 0$. Then $\max(0, -T_n^\circ[\phi]_\circ(\overline{\mathbb{P}})) = 0$. Let $\mathbb{W}$ be any world. Using linearity of $\mathbb{W}$ throughout, we have

$$\mathbb{W}(\ \max(0, T_n^\circ[\phi]_\circ(\overline{\mathbb{P}})) \cdot (\mathrm{Buy}_n(\phi) - (\mathrm{Buy}_n(\phi))^{*n})(\overline{\mathbb{P}})\ )$$

$$= T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot \mathbb{W}(\mathrm{Buy}_n(\phi)(\overline{\mathbb{P}}) - (\mathrm{Buy}_n(\phi))^{*n}(\overline{\mathbb{P}}))$$

$$= \ T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot \mathbb{W}\left(\phi \wedge \psi_n^\circ - \max\left(1, \frac{(\phi \wedge \psi_n^\circ)^{*n}(\overline{\mathbb{P}})}{\max(\varepsilon, \psi_n^{\circ*n}(\overline{\mathbb{P}}))}\right) \cdot \psi_n^\circ\right)$$

$$+ T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot \mathbb{W}\left((\phi \wedge \psi_n^\circ)^{*n}(\overline{\mathbb{P}}) - \max\left(1, \frac{(\phi \wedge \psi_n^\circ)^{*n}(\overline{\mathbb{P}})}{\max(\varepsilon, \psi_n^{\circ*n}(\overline{\mathbb{P}}))}\right) \cdot \psi_n^{\circ*n}(\overline{\mathbb{P}})\right)$$

by definition of Buy;

$$= \ T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot \mathbb{W}\left(\phi \wedge \psi_n^\circ - \max\left(1, \frac{\mathbb{P}_n(\phi \wedge \psi_n^\circ)}{\mathbb{P}_n(\psi_n^\circ)}\right) \cdot \psi_n^\circ\right)$$

$$+ T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot \left(\mathbb{P}_n(\phi \wedge \psi_n^\circ) - \max\left(1, \frac{\mathbb{P}_n(\phi \wedge \psi_n^\circ)}{\mathbb{P}_n(\psi_n^\circ)}\right) \cdot \mathbb{P}_n(\psi_n^\circ)\right)$$

by definition of the ^{*n} duals and by the assumption that $\mathbb{P}_n(\psi_n^\circ) > \varepsilon$;

$$\geq \ T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot (\mathbb{W}(\phi \wedge \psi_n^\circ) - \mathbb{P}_n(\phi \mid \psi_n^\circ) \cdot \mathbb{W}(\psi_n^\circ)),$$

by definition of $\mathbb{P}_n(\phi \mid \psi_n^\circ)$, where we use the fact that $T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \geq 0$ to lower-bound the cash term by 0.

Now suppose on the other hand that $T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \leq 0$. Then $\max(0, T_n^\circ[\phi]_\circ(\overline{\mathbb{P}})) = 0$. Again let $\mathbb{W}$ be any world; using linearity throughout, we have

$$\mathbb{W}(\ -\max(0, -T_n^\circ[\phi]_\circ(\overline{\mathbb{P}})) \cdot (\mathrm{Sell}_n(\phi) - (\mathrm{Sell}_n(\phi))^{*n})(\overline{\mathbb{P}})\ )$$

$$= T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot \mathbb{W}(\mathrm{Sell}_n(\phi)(\overline{\mathbb{P}}) - (\mathrm{Sell}_n(\phi))^{*n}(\overline{\mathbb{P}}))$$

$$= \ T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot \mathbb{W}\left(\phi \wedge \psi_n^\circ - \frac{\mathbb{P}_n(\phi \wedge \psi_n^\circ)}{\mathbb{P}_n(\psi_n^\circ)} \cdot \psi_n^\circ\right)$$

$$+ T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot \left(\mathbb{P}_n(\phi \wedge \psi_n^\circ) - \frac{\mathbb{P}_n(\phi \wedge \psi_n^\circ)}{\mathbb{P}_n(\psi_n^\circ)} \cdot \mathbb{P}_n(\psi_n^\circ)\right)$$

as above, and by the definition of Sell;

$$\geq \ T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \cdot (\mathbb{W}(\phi \wedge \psi_n^\circ) - \mathbb{P}_n(\phi \mid \psi_n^\circ) \cdot \mathbb{W}(\psi_n^\circ)),$$

by definition of $\mathbb{P}_n(\phi \mid \psi_n^\circ)$, where the cash term simply cancels, and where we use the fact that $T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \leq 0$, the fact that $\mathbb{W}(\psi_n^\circ) \geq 0$, and the fact that $\mathbb{P}_n(\phi \wedge \psi_n^\circ)/\mathbb{P}_n(\psi_n^\circ) \geq \mathbb{P}_n(\phi \mid \psi_n^\circ)$ to lower-bound the value of the $-\psi_n^\circ$ term.

Since at least one of $T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \leq 0$ or $T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) \geq 0$ must hold, we have shown that for all n and all worlds $\mathbb{W} \in \mathcal{PC}(D_n)$, the value $\mathbb{W}(T_n(\overline{\mathbb{P}}))$ is at least as great as

$$\sum_\phi T_n^\circ[\phi](\overline{\mathbb{P}}^\circ) \cdot \begin{cases} 1 - \mathbb{P}^\circ{}_n(\phi) & \text{if } \mathbb{W}(\phi \wedge \psi_n^\circ) = 1 \text{ and } \mathbb{W}(\psi_n^\circ) = 1, \\ -\mathbb{P}^\circ{}_n(\phi) & \text{if } \mathbb{W}(\phi \wedge \psi_n^\circ) = 0 \text{ and } \mathbb{W}(\psi_n^\circ) = 1, \\ 0 & \text{if } \mathbb{W}(\phi \wedge \psi_n^\circ) = 0 \text{ and } \mathbb{W}(\psi_n^\circ) = 0. \end{cases}$$

where we use the fact that $T_n^\circ[\phi]_\circ(\overline{\mathbb{P}}) = T_n^\circ[\phi](\overline{\mathbb{P}}^\circ)$ and the fact that $\mathbb{P}_n(\phi|\psi_n^\circ) = \mathbb{P}^\circ{}_n(\phi)$. (Since $\mathbb{W} \in \mathcal{PC}(D_n)$ is propositionally consistent, it cannot be the case that $\mathbb{W}(\phi \wedge \psi_n^\circ) = 1$ and $\mathbb{W}(\psi_n^\circ) = 0$.)

Notice that in worlds $\mathbb{W}$ such that $\mathbb{W}(\psi_n^\circ) = 1$, the value of the trades made by $\overline{T}^\circ$ in ϕ against $\overline{\mathbb{P}}^\circ$ is precisely

$$\mathbb{W}\left(T_n^\circ[\phi](\overline{\mathbb{P}}^\circ) \cdot (\phi - \phi^{*n}(\overline{\mathbb{P}}^\circ))\right) = \begin{cases} 1 - \mathbb{P}^\circ{}_n(\phi) & \text{if } \mathbb{W}(\phi) = 1 \\ -\mathbb{P}^\circ{}_n(\phi) & \text{if } \mathbb{W}(\phi) = 0. \end{cases}$$

In particular, for any world $\mathbb{W} \in \mathcal{PC}(D^\circ{}_n)$ plausible at time n according to $\overline{D}^\circ$,

$$\mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}})\right) = \mathbb{W}\left(\sum_{i \leq n} T_i^\circ(\overline{\mathbb{P}}^\circ)\right).$$

Since $\overline{T}^\circ$ exploits $\overline{\mathbb{P}}^\circ$ over $\overline{D}^\circ$, by definition the set

$$\left\{\mathbb{W}\left(\sum_{i \leq n} T_i^\circ(\overline{\mathbb{P}}^\circ)\right) \mid n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D^\circ{}_n)\right\}$$

is bounded below and unbounded above. Therefore the set

$$\left\{\mathbb{W}\left(\sum_{i \leq n} T_i(\overline{\mathbb{P}})\right) \mid n \in \mathbb{N}^+, \mathbb{W} \in \mathcal{PC}(D_n)\right\}$$

is unbounded above, since for all n we have $D^\circ{}_n \supseteq D_n$ and hence $\mathcal{PC}(D^\circ{}_n) \subseteq \mathcal{PC}(D_n)$.

It remains to show that this set is unbounded below. Suppose for contradiction that it is not, so there is some infinite sequence $\{(\mathbb{W}_i, n_i)\}$ with $\mathbb{W}_i \in \mathcal{PC}(D_{n_i})$ on which the value $\mathbb{W}_i\left(\sum_{j \leq n_i} T_j(\overline{\mathbb{P}})\right)$ of $\overline{T}$ is unbounded below.

We may assume without loss of generality that each $\mathbb{W}_i$ is inconsistent with $\overline{D}^\circ$. Indeed, if there is no subsequence with this property and with the values of $\overline{T}$ unbounded below, then the $\mathbb{W}_i$ consistent with $\overline{D}^\circ$ have the corresponding values $\mathbb{W}_i\left(\sum_{j \leq n_i} T_j(\overline{\mathbb{P}})\right) = \mathbb{W}_i\left(\sum_{j \leq n_i} T_j^\circ(\overline{\mathbb{P}}^\circ)\right)$ unbounded below, contradicting that $\overline{T}^\circ$ exploits $\overline{\mathbb{P}}^\circ$ over $\overline{D}^\circ$. Having made this assumption, there is an infinite sequence m_i with $\mathbb{W}_i(\psi_{m_i-1}^\circ) = 1 \wedge \mathbb{W}_i(\psi_{m_i}^\circ) = 0$ for all i.

We may further assume without loss of generality that for each i, we have $n_i \leq m_i - 1$. Indeed, for any $n \geq m_i$, we have by the above analysis that $\mathbb{W}_i\left(T_n(\overline{\mathbb{P}})\right) \geq 0$; in this case replacing n_i with $m_i - 1$ would only decrease the values $\mathbb{W}_i(\sum_{j \leq n_i} T_j(\overline{\mathbb{P}}))$, and hence would preserve that this sequence is unbounded below.

In particular, it is the case that $\psi_{m_i-1}^\circ$ propositionally implies $\psi_{n_i}^\circ$. Because $\mathbb{W}_i(\psi_{m_i-1}^\circ) = 1$ and $\mathbb{W}_i \in \mathcal{PC}(D_{n_i})$, this implies $\mathbb{W}_i \in \mathcal{PC}(D^\circ{}_{n_i})$, i.e., $\mathbb{W}_i$ was plausible at time step n_i according to $\overline{D}^\circ$. But then we have that the sequence of values $\mathbb{W}_i(\sum_{j \leq n_i} T_j(\overline{\mathbb{P}})) = \mathbb{W}_i(\sum_{j \leq n_i} T_j^\circ(\overline{\mathbb{P}}^\circ))$ is unbounded below, contradicting that $\overline{T}^\circ$ exploits $\overline{\mathbb{P}}^\circ$ over $\overline{D}^\circ$.

Thus we have shown that, assuming that $\overline{T}^\circ$ exploits $\overline{\mathbb{P}}^\circ$ over $\overline{D}^\circ$, also $\overline{T}$ exploits $\overline{\mathbb{P}}$ over $\overline{D}$. This contradicts that $\overline{\mathbb{P}}$ is a logical inductor, so in fact it cannot be that $\overline{T}^\circ$ exploits $\overline{\mathbb{P}}^\circ$; thus $\overline{\mathbb{P}}^\circ$ is a logical inductor over $\overline{D}^\circ$, as desired. $\qquad\square$