Learning generalized depth-three arithmetic circuits in the non-degenerate case

Vishwas Bhargava*  Ankit Garg  Neeraj Kayal
Rutgers University  Microsoft Research India  Microsoft Research India
vishwas1384@gmail.com  garga@microsoft.com  neeraka@microsoft.com
Chandan Saha
Indian Institute of Science
chandan@iisc.ac.in

November 13, 2021

Abstract

Consider a homogeneous degree $d$ polynomial $f = T_1 + \cdots + T_s$, $T_i = g_i(\ell_{i,1}, \ldots, \ell_{i,m})$ where $g_i$'s are homogeneous $m$-variate degree $d$ polynomials and $\ell_{i,j}$'s are linear polynomials in $n$ variables. We design a (randomized) learning algorithm that given black-box access to $f$, computes black-boxes for the $T_i$'s. The running time of the algorithm is poly($n, m, d, s$) and the algorithm works under some non-degeneracy conditions on the linear forms and the $g_i$'s, and some additional technical assumptions $n \geq (md)^2, s \leq n^{d/4}$. The non-degeneracy conditions on $\ell_{i,j}$'s constitute non-membership in a variety, and hence are satisfied when the coefficients of $\ell_{i,j}$'s are chosen uniformly and randomly from a large enough set. The conditions on $g_i$'s are satisfied for random polynomials and also for natural polynomials common in the study of arithmetic complexity like determinant, permanent, elementary symmetric polynomial, iterated matrix multiplication. A particularly appealing algorithmic corollary is the following: Given black-box access to an $f = \text{Det}_r(L^{(1)}) + \cdots + \text{Det}_r(L^{(s)})$, where $L^{(k)} = (\ell^{(k)}_{i,j})_{i,j}$ with $\ell^{(k)}_{i,j}$'s being linear forms in $n$ variables chosen randomly, there is an algorithm which in time poly($n, r$) outputs matrices $(M^{(k)})_k$ of linear forms s.t. there exists a permutation $\pi: [s] \rightarrow [s]$ with $\text{Det}_r(M^{(k)}) = \text{Det}_r(L^{(\pi(k))})$.

Our work follows the works [KS19, GKS20] which use lower bound methods in arithmetic complexity to design average case learning algorithms. It also vastly generalizes the result in [KS19] about learning depth-three circuits, which is a special case where each $g_i$ is just a monomial. At the core of our algorithm is the partial derivative method which can be used to prove lower bounds for generalized depth-three circuits. To apply the general framework in [KS19, GKS20], we need to establish that the non-degeneracy conditions arising out of applying the framework with the partial derivative method are satisfied in the random case. We develop simple but general and powerful tools to establish this, which might be useful in designing average case learning algorithms for other arithmetic circuit models.

*Part of this work was done when the author was an intern at Microsoft Research India.
## Contents

1 **Introduction**  
   1.1 The model and our results .................................................. 1  
   1.2 Techniques and proof overview .......................................... 2  
   1.3 Related work ................................................................. 4  
   1.4 Roadmap of the paper ...................................................... 5  

2 **The learning algorithm and its analysis**  
   2.1 Non-degeneracy conditions .................................................. 5  

3 **Non-degeneracy of random circuits**  
   3.1 Non-degeneracy of random circuits: Condition 1 ..................... 8  
   3.2 Non-degeneracy of random circuits: Condition 2 ..................... 9  
   3.2.1 Direct sum structure of \((D^+_i)^{2-s}\) .............................. 11  
   3.3 Non-degeneracy condition 3: Adjoint algebra is trivial ............ 12  
   3.4 Adjoint algebra for random \(g_i\)'s .................................. 13  
   3.5 From black-box access to learning generalized depth-three circuits .......................... 15  

4 **Conclusion and open problems** ............................................. 15  

A **Adjoint algebra and vector space decomposition** ....................... 18  

B **Linear algebra with black boxes** ......................................... 20
1 Introduction

Arithmetic circuits are a natural model for computing polynomials using basic arithmetic operations like addition and multiplication. The problem of learning arithmetic circuits a.k.a. reconstruction is an important and well studied problem. Here we are given a polynomial (either explicitly as a list of coefficients or using black-box access) and the goal is to find the smallest (or approximately smallest) arithmetic circuit computing it. It can be defined for various arithmetic circuit models. Unsurprisingly, there is enough evidence to point out that the problem is likely to be hard in the worst case for most arithmetic circuit models [H˚ as90, FK09, KS09b, Shi16].

Hence, it is imperative to explore algorithms for learning arithmetic circuits that are efficient and work in the average case. One classic example of a stark contrast between the worst case and average case complexities is the tensor decomposition problem. Let us focus on $n \times n \times n$ tensors for simplicity. In the language of arithmetic complexity, tensor decomposition corresponds to learning depth-three set-multilinear circuits. We have three sets of variables $y = \{y_1, \ldots, y_n\}$, $z = \{z_1, \ldots, z_n\}$, $w = \{w_1, \ldots, w_n\}$. Then the problem is to decompose a set-multilinear polynomial $f(y, z, w) = \sum_{j,k,\ell} T_{j,k,\ell} y_j z_k w_{\ell}$ as

$$
\sum_{i=1}^{s} \ell_{i1}(y) \ell_{i2}(z) \ell_{i3}(w)
$$

for the smallest possible $s$ (here $\ell_{ij}$’s are linear forms). This is NP-hard in the worst case [H˚ as90]. However, it is possible to design efficient algorithms for tensor decomposition which work well under some mild assumptions. One such algorithm is due to Jennrich [Har70, LRA93] and states that given $f(y, z, w)$ we can find the above decomposition in polynomial time if $s \leq n$ and $\ell_{1a}, \ldots, \ell_{sa}$ are linearly independent for all $a \in [3]$. Note that the algorithm works under a bound on $s$ and also a mild assumption on the linear forms (which is satisfied when the linear forms are chosen randomly). Our algorithms will also work under such non-degeneracy conditions. Kayal and Saha [KS19] designed algorithms for learning depth-three arithmetic circuits in the non-degenerate case. That is, they design an algorithm for decomposing

$$
f(x) = \sum_{i=1}^{s} \prod_{j=1}^{d} \ell_{ij}(x)
$$

assuming a bound on $s$ and certain non-degeneracy conditions on the $\ell_{ij}$’s. Note that the above model is different from tensor decomposition or set-multilinear circuits since there is no partitioning of variables into disjoint sets and the linear forms can depend on all the variables. We prove a far-reaching generalization of the result of [KS19].

1.1 The model and our results

We study the model of generalized depth-three circuits. A circuit in this class computing a degree $d$ polynomial $f(x)$ is an expression of the following kind,

$$
f(x) = g_1(\ell_{11}, \ldots, \ell_{1m}) + \cdots + g_s(\ell_{s1}, \ldots, \ell_{sm}),
$$

where $g_i$’s are $m$-variate degree $d$ homogeneous polynomials and $\ell_{ij}$’s are linear forms in the variables $x = (x_1, \ldots, x_n)$. Our main result is an algorithm for learning decompositions of the above kind assuming certain non-degeneracy conditions.

**Theorem 1.1** (Learning generalized depth-three circuits in the non-degenerate case). There is a randomized algorithm that given black-box access to an $n$-variate degree $d$ polynomial $f = T_1 + \cdots + T_s$,  

1Despite this, there has been much success in designing worst case reconstruction algorithms. This includes reconstruction algorithms for the models of sparse polynomials [KS01], read-once algebraic branching programs (ROABPs) [BBB+00, KS06] and for models with bounded top fan-in [KS09b, KS09a, GKL12, Sin16, BSV20, BSV21].  

2This bound usually corresponds to the best known lower bounds we can prove for the corresponding model.
$T_i = g_i(\ell_{i1}, \ldots, \ell_{im})$, outputs black-boxes for the individual summands $T_i$'s. The running time of the algorithm is $\text{poly}(n, m, d, s)$. The algorithm works under certain non-degeneracy conditions and also under some additional technical assumptions such as $n \geq (md)^2$, $s \leq n^{d/4}$, $|F| \geq (nds)^3$ and $\text{char}(F) = 0$ or $\text{char}(F) > d$.

The non-degeneracy conditions are mentioned explicitly in Section 2.1. These non-degeneracy conditions are satisfied when the coefficients of the linear forms are chosen uniformly and independently at random from a large enough set and when the $g_i$'s are either random or one of the well-known polynomials in arithmetic complexity such as determinant, permanent, elementary symmetric polynomial etc. Let us mention one such appealing corollary which follows from Theorem 1.1 and the algorithms for equivalence-testing of the determinant [Kay12, GGKS19].

**Corollary 1** (Learning sums of random projections of determinants). Suppose $n, r, F, s$ be such that $n \geq s^6$, $s \leq n^{r/4}$, $|F| \geq (nrs)^3$ and $\text{char}(F) = 0$ or $\text{char}(F) > r$. There is a randomized $\text{poly}(n, r)$ time algorithm that given black-box access to an $f = \text{Det}_r(L(1)) + \cdots + \text{Det}_r(L(s))$, where $L(k) = (\ell_{ij}^{(k)})_{i,j}$ with $\ell_{ij}^{(k)}$'s being linear forms in $n$ variables whose coefficients are chosen independently and uniformly at random from an arbitrary set $S \subset F$ of size $|S| \geq (nrs)^3$, it outputs matrices of linear forms $(M^{(k)})_{ij}$ s.t. there exists a permutation $\pi : [s] \to [s]$ with $\text{Det}_r(M^{(k)}) = \text{Det}_r(L(\pi(k)))$.

**Remarks.**

1. Once we have the black-boxes for the $T_i$'s as in Theorem 1.1, it is not hard to output black-boxes for $\tilde{g}_i$'s and also $\tilde{\ell}_{i1}, \ldots, \tilde{\ell}_{im}$ s.t. $T_i = \tilde{g}_i(\tilde{\ell}_{i1}, \ldots, \tilde{\ell}_{im})$. Note that we cannot hope to exactly recover the $g_i$'s since there is some redundancy. One can always apply a linear transformation to the input variables of $g_i$'s to obtain different decompositions.

2. We get a similar result (as in Corollary 1) with $g_i$'s being the elementary symmetric polynomial, permanent, iterated matrix multiplication, monomials etc. Note that it could be a mixture of these. It might seem strange that we are able to handle permanent, but note that we are only dealing with black-boxes and hence the complexity of the permanent does not come into play. It is already known how to do equivalence-testing of the permanent efficiently [Kay12] which is similar in spirit to the $s = 1$ case.

### 1.2 Techniques and proof overview

We follow the meta framework of [KS19, GKS20] for designing learning algorithms for arithmetic circuits in the non-degenerate case via lower bounds. We note that while the meta framework is quite general, still a lot of technical work is needed to carry it out for a particular circuit class if one has lower bounds for that class. The same holds for this paper. We will not go into the full generality of the framework and refer the reader to the exposition in [GKS20]. Instead, we will explain the details for our special case.

Let us first see how one would prove a lower bound (on the number of summands $s$) for the model of generalized depth-three circuits. Consider the set of all partial differential operators of order $k$ i.e. $\mathcal{L} = \partial_x^k$. These are linear maps from $F[x]_d$ to $F[x]_{d-k}$. Note that

$$\dim(\langle \mathcal{L} \circ T_i \rangle) \leq \binom{m+k-1}{k},$$

if $T_i$ is of the form $g_i(\ell_{i1}, \ldots, \ell_{im})$. This is easy to verify if $T_i$ were equal to $g_i(x_1, \ldots, x_m)$. Then one can use the fact that the dimension of the partial derivative space doesn’t change upon an invertible linear transformation of the variables. Also note that

$$\langle \mathcal{L} \circ f \rangle \subseteq \langle \mathcal{L} \circ T_1 \rangle + \cdots + \langle \mathcal{L} \circ T_s \rangle$$

(1)
One can hope that in the generic case, one would get a lower bound \( \approx \binom{n+k-1}{k} \). It is not too hard to design an \( f \) for which \( \dim(\langle L \circ f \rangle) \approx \binom{n+k-1}{k} \) (when \( k \leq \lfloor d/2 \rfloor \)) and for such an \( f \) we get the highest lower bound.

It is natural to wonder what is the connection to learning, if there is any at all. Consider equation 1. We can choose \( k = \lfloor d/2 \rfloor \) to get the highest lower bound.

One can hope that in the generic case, one would get

\[
\langle L \circ f \rangle = \langle L \circ T_1 \rangle + \cdots + \langle L \circ T_s \rangle
\]  

(2)

That is the inclusion becomes an equality and the sum becomes a direct sum. Furthermore, let us assume that it holds for \( L' = \partial_x^{(k+1)} \) as well. That is,

\[
\langle L' \circ f \rangle = \langle L' \circ T_1 \rangle + \cdots + \langle L' \circ T_s \rangle
\]  

(3)

So we have \( U := \langle L \circ f \rangle, V := \langle L' \circ f \rangle \) and the linear maps \( \partial_x^{-1} \) from \( U \) to \( V \). Let \( U_i := \langle L \circ T_i \rangle \) and \( V_i := \langle L' \circ T_i \rangle \). Note that the linear maps \( \partial_x^{-1} \) map \( U_i \) into \( V_i \). So one is naturally led towards the following vector decomposition problem.

**Problem 1.2 (Vector space decomposition).** Given the triple \((\mathcal{M}, U, V)\) consisting of vector spaces \( U \) and \( V \) and a set of linear maps \( \mathcal{M} \) from \( U \) to \( V \), decompose \( U \) and \( V \) as

\[
U = U_1 \oplus \cdots \oplus U_s, \quad V = V_1 \oplus \cdots \oplus V_s
\]

s.t. \( \langle M \circ U_i \rangle \subseteq V_i \) for all \( i \in [s] \).

For our setting, one such decomposition is described in Equations (2) and (3). Once one has access to \( U_i \)'s (black-box access to a basis), it is not hard to obtain black-boxes for the \( T_i \)'s. So the only thing remains to prove is the uniqueness of vector space decomposition (in addition to (2) and (3) themselves). There are many efficient algorithms to solve the vector space decomposition problem. Please refer to Appendix A for specialized algorithms that work for our setting, and [GKS20] for a thorough discussion on the general problem and related work. Let us now describe our approaches to prove Equations (2) and (3) and also the uniqueness of decomposition.

For proving uniqueness of decomposition, we employ the use of the notion of an adjoint algebra, following [GKS20]. The adjoint algebra essentially captures “homomorphisms” of the triple \((\mathcal{M}, U, V)\). That is,

\[
\text{Adj}(\mathcal{M}, U, V) = \{ (D, E) : D : U \to U, \ E : V \to V \ \text{linear maps and} \ \text{LD = EL} \ \forall \ L \in \mathcal{M} \}
\]

Suppose the triple \((\mathcal{M}, U, V)\) admits a vector space decomposition \( U = U_1 \oplus \cdots \oplus U_s, \ V = V_1 \oplus \cdots \oplus V_s \). Then the projection maps \((\Pi_{U_i}, \Pi_{V_i})\) (which are identity on \( U_i, V_i \) respectively and map other vector spaces in the direct sum to 0) lie in the adjoint. We say that the adjoint algebra is *trivial* if it is spanned by these projectors. It is not hard to show that if the adjoint algebra is trivial, then the above vector space decomposition is unique (Lemma 23). Note that one can always combine blocks in an arbitrary way, but the decomposition is unique among all “finest” decompositions where one cannot decompose any block further. So we are left with proving the uniqueness of the decomposition in Equations (2) and (3). We prove that the adjoint algebra is trivial in this case (proof of Theorem 2.1) using a non-degeneracy condition on the \( g_i \)'s (Item 3 in Section 2.1; also see Section 3.3).

So now let us see how to prove Equations (2) and (3). Showing the direct sum \( U_1 + \cdots + U_s = U_1 \oplus \cdots \oplus U_s \) (and the same for \( V_i \)'s) is done in a similar way to [KS19], Schwartz-Zippel lemma yields the direct sum once one can show the existence of some set of linear forms satisfying the direct sum property. This is done using a design construction based on Nisan-Wigderson designs. This construction is inspired from [KS19] but more general. We differ significantly from previous works [KS19, GKS20] in our technique for showing that \( U = U_1 + \cdots + U_s \). The previous works relied on intricate design constructions to exhibit linear forms which satisfy this property (followed by a use of Schwartz-Zippel lemma). For our setting, one can get away
with the above design based approach, but this can become more cumbersome and challenging as the circuit models become more complicated. Hence, we devise a general way of proving statements of the form
\[ \langle L \circ f \rangle = \langle L \circ T_1 \rangle + \cdots + \langle L \circ T_s \rangle \]
for \( f = T_1 + \cdots + T_s \), which is conceptually more appealing. It is useful to have the linear maps \( L \) from a subspace of the operators (so for our case think of \( L = \langle \partial_x^{-k} \rangle \)). Since
\[ \langle L \circ f \rangle \subseteq \langle L \circ T_1 \rangle + \cdots + \langle L \circ T_s \rangle, \]
it suffices to prove that \( \langle L \circ T_i \rangle \subseteq \langle L \circ f \rangle \) for all \( i \). Let us consider the operators annihilating a particular term \( T_i \).
\[ L_{null}^i := \{ L \in L : L \circ T_i = 0 \} \]
Now note that for any \( L \in \bigcap_j \neq i L_{null}^j \), \( L \circ f = L \circ T_i \). If the subspace of operators \( \bigcap_j \neq i L_{null}^j \) was rich enough, at least to the extent relevant to \( T_i \), then we would be done. We are able to show this by moving to the duals of the vector spaces \( L_{null}^i \) (with respect to an appropriate bilinear form) and proving a direct sum property there (the proof of which turns out to be almost identical to the proof we have for the direct sum of the \( U_i \)'s!). For more details, see Section 2.

**Comparison with previous works.** Our work closely follows the papers [KS19, GKS20] on learning arithmetic circuits in the non-degenerate case via lower bounds. However, there are substantial differences as well. Firstly, as explained above, we devise a general technique for proving statements of the kind \( \langle L \circ f \rangle = \langle L \circ T_1 \rangle + \cdots + \langle L \circ T_s \rangle \). Secondly, ours is the first paper that uses the full machinery of the learning from lower bounds framework in [KS19, GKS20]. In [KS19], the framework was present in a rudimentary form and that made the analysis more cumbersome. While the framework was fully laid out in [GKS20], for their application of learning sums of powers of low degree polynomials, they eventually implement a somewhat ad hoc approach. Without this learning framework, it seems rather challenging to get such a general result as in Theorem 1.1.

### 1.3 Related work

[GKKS14] proved a lower bound for the more general model of generalized depth-four circuits (bounded bottom-fanin). [KS09a] study the worst case learning algorithms for a model which is similar to our model in many ways, but their parameters are different (they also call their model generalized depth-three circuits). There has been a lot of work on worst case reconstruction algorithms which includes reconstruction algorithms for the models of sparse polynomials [KS01], read-once algebraic branching programs (ROABPs) [BBB00, KS06] and for models with bounded top fan-in [KS09b, KS09a, GKL12, Sin16, BSV20, BSV21].

In [GKL11], a randomized polynomial-time proper learning algorithm was given for non-degenerate $3$-multilinear formulas having fan-in two. A randomized polynomial-time proper learning algorithm for non-degenerate regular formulas having fan-in two was given in [GKQ14]. An efficient randomized reconstruction for non-degenerate homogeneous ABPs of width at most \( \frac{\sqrt{n}}{2} \) is presented in [KNS19]. [KS19] designed algorithms for learning non-degenerate depth-three circuits which is a special case of our model with the \( g_i \)'s being a monomial. [GKS20], following [KS19], developed a meta framework for learning non-degenerate arithmetic circuits via lower bounds. They implemented it to learn sums of powers of low degree polynomials in the non-degenerate case.

As already mentioned, the problem of tensor decomposition is a special case for our model. Tensor decomposition is widely studied in the machine learning community as well (also known as CP decomposition), e.g. see the surveys [KB09, DL09, JGKA20]. Another kind of tensor decomposition, Tucker decomposition is also widely studied, see Section 4 in [KB09]. Tensor decomposition roughly corresponds to the $m = 1$ case.

\footnote{The papers [GKL11, GKQ14] state the results for random formulas, but it is not difficult to state the non-degeneracy conditions by taking a closer look at the algorithms.}
in our model\textsuperscript{4} Tucker decomposition roughly corresponds to $s = 1$ in our model.\textsuperscript{5} Given the wide variety of applications of these two problems in machine learning, we hope that (noise-resilient versions of) our algorithms will handle much more challenging problems in machine learning.

1.4 Roadmap of the paper

In Section 2, we present our algorithm for learning non-degenerate generalized depth-three circuits, the corresponding non-degeneracy conditions and the analysis of the algorithm assuming the non-degeneracy conditions. In Section 3, we prove that the non-degeneracy conditions are satisfied for random circuits. Section 4 contains the summary of the work and some of the open problems that arise from this work. Section A contains some basic facts about the vector space decomposition problem. Finally, Section B contains some facts about how to perform linear algebra given black boxes.

2 The learning algorithm and its analysis

In this section, we describe our algorithm for learning non-degenerate generalized depth three circuits and the analysis assuming the non-degeneracy conditions. Since we are aiming for poly$(s)$ time-complexity, we can assume that we know $s$. For a field $F$ and $d \in \mathbb{N}$, let $F[x]_d$ denote the ring of homogeneous degree $d$ polynomials in $F[x]$. Consider a homogeneous degree $d$ polynomial $f \in F[x]_d$ which is computed by a homogeneous generalized depth three circuit i.e., $f = T_1 + \cdots + T_s$, where $T_i = g_i(\ell_{i1}, \ldots, \ell_{im})$ for $i \in [s]$. Here $\ell_{ij}$’s are linear forms.

2.1 Non-degeneracy conditions

We impose the following non-degeneracy conditions on $f$ (or more precisely the circuit computing it):

1. For each $i \in [s]$, the linear forms $(\ell_{i1}, \ldots, \ell_{im})$ are linearly independent. Also the vector spaces $W_1^{(d-k)} := F[\ell_{i1}, \ldots, \ell_{im}]_{d-k}, \ldots, W_s^{(d-k)} := F[\ell_{s1}, \ldots, \ell_{sm}]_{d-k}$ form a direct sum i.e. $W_1^{(d-k)} + \cdots + W_s^{(d-k)} = W_1^{(d-k)} \oplus \cdots \oplus W_s^{(d-k)}$.

The same assumption for the vector spaces $W_i^{(d-k-1)}$’s.

2. We will use $\partial = k$ to denote the set of order-$k$ partial differential operators in the variables $x$. Consider the vector spaces $U := \langle \partial = k f \rangle$, $V := \langle \partial = (k+1) f \rangle$, $U_i := \langle \partial = k T_i \rangle$, $V_i = \langle \partial = (k+1) T_i \rangle$. We will assume that $U = U_1 \oplus \cdots \oplus U_s$ and $V = V_1 \oplus \cdots \oplus V_s$.

3. For the polynomials $g_i \in F[z]_d$, $z = (z_1, \ldots, z_m)$, the triple $\left( \partial_z^{(j)} = 1, \langle \partial_z = k g_i \rangle, \langle \partial_z = (k+1) g_i \rangle \right)$ has a trivial adjoint algebra for all $i \in [s]$ (see Definitions A.1 and A.3). That is, if $D : \langle \partial_z = k g_i \rangle \rightarrow \langle \partial_z = k g_i \rangle$ and $E : \langle \partial_z = (k+1) g_i \rangle \rightarrow \langle \partial_z = (k+1) g_i \rangle$ are linear maps s.t. $\partial_z D(p) = E(\partial_z p)$ for all $j \in [m]$ and all $p \in \langle \partial_z = k g_i \rangle$, then $D, E$ are both identity maps (up to a scalar multiplication).

The algorithm is stated in Algorithm 1. We will need the following lemma in the proof of the main theorem.

\textsuperscript{4}Strictly speaking $m = 1$ would be symmetric tensor decomposition and exactly modeling general tensor decomposition would require higher $m$ but in spirit tensor decomposition is closer to the $m = 1$ case than higher $m$.

\textsuperscript{5}Again, ignoring some symmetry considerations here.
Lemma 2. Let $h \in F[x]$ be a homogeneous degree $d$ polynomial and $\ell_1, \ldots, \ell_m \in F[x]$ be linearly independent linear forms. Then $h \in F[\ell_1, \ldots, \ell_m]$ iff $\sum_{j=1}^{n} \alpha_j \partial_{x_j} h(x) = 0$ for all $\alpha \in F^n$ s.t. $\ell_i(\alpha) = 0$ for all $i \in [m]$.

Proof. Let $\ell_i = \sum_{j=1}^{n} \ell_{ij} x_j$. In one direction, suppose $h \in F[\ell_1, \ldots, \ell_m]$ so that $h = g(\ell_1, \ldots, \ell_m)$ for $g \in F[z]$, $z = (z_1, \ldots, z_m)$. Then

$$\sum_{j=1}^{n} \alpha_j \partial_{x_j} h(x) = \sum_{j=1}^{n} \alpha_j \sum_{i=1}^{m} \ell_{ij} \partial_{z_i} g(z)|_{z=(\ell_1, \ldots, \ell_m)}$$

$$= \sum_{i=1}^{m} \ell_i(\alpha) \partial_{z_i} g(z)|_{z=(\ell_1, \ldots, \ell_m)} = 0$$

for all $\alpha \in F^n$ s.t. $\ell_i(\alpha) = 0$ for all $i \in [m]$. In the other direction, suppose $\sum_{j=1}^{n} \alpha_j \partial_{x_j} h(x) = 0$ for all $\alpha \in F^n$ s.t. $\ell_i(\alpha) = 0$ for all $i \in [m]$. Extend $\ell_1, \ldots, \ell_m$ to a full basis of $F[x]_1$, $\ell_1, \ldots, \ell_n$ (in an arbitrary way). We can write $h$ as $g(\ell_1, \ldots, \ell_n)$ for some $g \in F[w]$, $w = (w_1, \ldots, w_n)$. Our goal now is to prove that $\partial_{w_i} g(w) = 0$ for all $i \in \{m+1, \ldots, n\}$. Now

$$\sum_{j=1}^{n} \alpha_j \partial_{x_j} h(x) = \sum_{j=1}^{n} \alpha_j \sum_{i=1}^{m} \ell_{ij} \partial_{w_i} g(w)|_{w=(\ell_1, \ldots, \ell_n)}$$

$$= \sum_{i=1}^{n} \ell_i(\alpha) \partial_{w_i} g(w)|_{w=(\ell_1, \ldots, \ell_n)}$$

For $i \in \{m+1, \ldots, n\}$, we can choose an $\alpha$ s.t. $\ell_j(\alpha) = 0$ for all $j \neq i$ and $\ell_i(\alpha) \neq 0$. Then from the assumption and the above calculation we get that $\partial_{w_i} g(w)|_{w=(\ell_1, \ldots, \ell_n)} = 0$. Since $\ell_1, \ldots, \ell_n$ are linearly independent, we get that $\partial_{w_i} g(w) = 0$ for all $i \in \{m+1, \ldots, n\}$. 

Algorithm 1 Learning generalized depth-three circuits

Input: black-box access to an $f \in F[x]$ that is computed by a non-degenerate homogeneous generalized depth three circuit i.e., $f = T_1 + \cdots + T_s$, where $T_i = g_i(\ell_{i1}, \ldots, \ell_{im})$ for $i \in [s]$.

Output: $s$ black-boxes $B_1, \ldots, B_s$ such that there exists a permutation $\pi : [s] \rightarrow [s]$ s.t. $B_i$ provides black-box access to $T_{\pi(i)}$.

Subroutines:

1. Computing black-boxes for partial derivatives from the black-box for a polynomial. (Fact 22)
2. Vector space decomposition (Algorithm 2 and Corollary 26).

Parameters: The order of partial derivatives: $k$.

1. Compute black-boxes for a basis of the vector spaces $U := \langle \partial^k f \rangle$ and $V := \langle \partial^{k+1} f \rangle$ using Subroutine 1.
2. Using Subroutine 2, obtain a vector space decomposition $U = U'_1 \oplus \cdots \oplus U'_s$ and $V = V'_1 \oplus \cdots \oplus V'_s$ for the triple $(\partial^k, (U, V))$. If $s' \neq s$, then abort. Otherwise continue.
3. For each $\alpha$ s.t. $\sum_{i=1}^{n} \alpha_i = k$, write the corresponding differential operator acting on $f$, $\partial_{\alpha} f$, as $u'_{\alpha_1} + \cdots + u'_{\alpha_s}$ with $u'_{\alpha_i} \in U'_i$ (note that there is a unique such representation). We only obtain black-boxes for the polynomials $u'_{\alpha_i}$’s. This step can be carried out using Corollary 26.
4. The black-box $B_i$ on input $x$ will output $\frac{(d-k)!}{d!} \sum_{\alpha} (\alpha_1, \ldots, \alpha_n) x^\alpha u'_{\alpha_i}(x)$.

The next theorem states the correctness of Algorithm 1 assuming the non-degeneracy conditions.
Theorem 2.1. Suppose the non-degeneracy conditions stated above are satisfied. Then Algorithm 1 never aborts. Suppose $B_1, \ldots, B_s$ be the black-boxes output by the algorithm. Then there exists a permutation $\pi: [s] \rightarrow [s]$ s.t. $B_i$ is a black-box for $T_{\pi(i)}$.

Proof. It suffices to prove uniqueness of decomposition for the triple $(\vartheta = 1, U, V)$ (see Definition A.4). Assuming uniqueness of decomposition, $s' = s$ and there exists a permutation $\pi: [s] \rightarrow [s]$ s.t. $U'_i = U_{\pi(i)}$ and $V'_i = V_{\pi(i)}$. Since the $U'_i$’s form a direct sum, there is a unique way of writing each element $u \in U$ as $u = u'_1 + \cdots + u'_s$ with $u'_i \in U'_i$. For $u = \partial_\alpha f$, $u = \partial_\alpha T_{\pi(i)} + \cdots + \partial_\alpha T_{\pi(s)}$ is one such decomposition and hence the only one. Thus $u'_{\alpha i} = \partial_\alpha T_{\pi(i)}$ in which case $B_i$ computes the black-box for $T_{\pi(i)}$ by Lagrange’s formula,

$$h(x) = \frac{(d-k)!}{d!} \sum_{\alpha} \binom{k}{\alpha} x^\alpha \partial_\alpha h(x)$$

for a homogeneous degree $d$ polynomial $h$.

To prove uniqueness of decomposition, it suffices to prove that the adjoint algebra for the triple $(\vartheta = 1, U, V)$ is trivial because of Lemma 23. Consider linear maps $D: U \rightarrow U$ and $E : V \rightarrow V$ s.t. $\partial_x D(u) = E(\partial_x u)$ for all $u \in U$. Then we need to prove that $D(U_i) \subseteq U_i$, $E(V_i) \subseteq V_i$ for all $i \in [s]$ and that $(D, E)$ are scalar multiples of identity when restricted to $(U_i, V_i)$ respectively. The latter follows from Item 3 in the non-degeneracy conditions, so we only need to prove the former. To prove the former, consider $(D, E)$ in the adjoint algebra. Note that $U_i \subseteq \mathbb{F}[\ell_1, \ldots, \ell_{im}]_{d-k}$ and $V_i \subseteq \mathbb{F}[\ell_1, \ldots, \ell_{im}]_{d-k-1}$. Hence if $u \in U_i$, then

$$\sum_{j=1}^{n} \alpha_j \partial_{\ell_j} u(x) = 0$$

for all $\alpha$ s.t. $\ell_1(\alpha) = \cdots = \ell_{im}(\alpha) = 0$, by Lemma 2. Because of the relation $\partial_{\ell_j} D(u) = E(\partial_{\ell_j} u)$, we get that

$$\sum_{j=1}^{n} \alpha_j \partial_{\ell_j} D(u)(x) = 0$$

for all $\alpha$ s.t. $\ell_1(\alpha) = \cdots = \ell_{im}(\alpha) = 0$. Hence by Lemma 2, we get that $D(u) \in \mathbb{F}[\ell_1, \ldots, \ell_{im}]_{d-k}$. Hence $D(u) \in U \cap \mathbb{F}[\ell_1, \ldots, \ell_{im}]_{d-k} = U_i$ (because of the direct sum structure of the vector spaces $\mathbb{F}[\ell_1, \ldots, \ell_{im}]_{d-k}$ in Item 1). This completes the proof that $D(U_i) \subseteq U_i$. Now the space $V_i$ has a basis which consists of a subset of polynomials from $\partial_\beta T_i$ as $\beta$ varies over monomials of degree $k + 1$. We can write $\partial_\beta T_i$ as $\partial_{\ell_j} \partial_\alpha T_i$ for some $j \in [n]$ and some $\alpha$ of degree $k$. Then

$$E(\partial_\beta T_i) = E(\partial_{\ell_j} \partial_\alpha T_i) = \partial_{\ell_j} D(\partial_\alpha T_i)$$

Since $D(\partial_\alpha T_i) \in U_i$, we get that $E(\partial_\beta T_i) \in V_i$. This completes the proof that $E(V_i) \subseteq V_i$.

We will now proceed to proving theorem 1.1.

Proof of theorem 1.1. We will run algorithm 1 on the given black-box with the parameter $k$ being set to $\lceil \frac{2 \log n}{s} \rceil$. Notice that, by fact 22, the time complexity of subroutine 1 is $\text{poly}(d^k, n) = \text{poly}(s, n)$. See remark 3.1. Since theorem 2.1 guarantees the correctness of our output, we just have to verify its running time. Note that the time complexity of remaining steps is $\text{poly}(n^k, s) = \text{poly}(n, s)$, which concludes the proof.

3 Non-degeneracy of random circuits

In this section we will show that if $n > (md)^2$, $s \leq n^{d/4}$ and $k = \lceil \frac{2 \log n}{s} \rceil$, then a random $(n, d, s, \{g_i\} i \in [s])$ homogeneous generalized depth three circuit is non-degenerate with high probability. For better understanding about the regime of parameters, we record a few relations among the parameters $n, d, s$ and $k$ in the following easy to verify remark.
Remark 3.1. If \( s \leq n^{d/4}, md \leq \sqrt{n} \) and \( k = \lceil \frac{2 \log n}{\log m} \rceil \) then \( k \leq d/2 \) and \( (m+k^{-1}) \leq ns \).

We will proceed by showing that each of our non-degeneracy conditions is satisfied for random circuits, and then the result will hold directly by the union bound. We also show that \((n, d, s, \{g_i\}_{i \in [s]})\) homogeneous generalized depth three circuits are non-degenerate if the \(g_i\’s\) belong to special polynomial families like \(\text{Det}_d, \text{Perm}_d, \text{IMM}_r, \text{Sym}_r, d\) and only \(\ell_{i,j}\’s\) are chosen randomly. This is because non-degeneracy condition 1 and 2 just depend\(^6\) on \(\ell_{i,j}\’s\) and non-degeneracy condition 3 depends on the \(g_i\’s\), and we can show that aforementioned polynomial families satisfy these mild technical conditions required to show non-degeneracy condition 3.

3.1 Non-degeneracy of random circuits: Condition 1

Let’s begin by restating our first non-degeneracy condition for a generalized depth three circuit \(\sum_{i=1}^s g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m})\).

The vector spaces \(W_i^{(d-k)} := \mathbb{F}[\ell_{i,1}, \ldots, \ell_{i,m}]_{d-k}, \ldots, W_s^{(d-k)} := \mathbb{F}[\ell_{s,1}, \ldots, \ell_{s,m}]_{d-k}\) form a direct sum i.e.

\[
W_1^{(d-k)} + \cdots + W_s^{(d-k)} = W_1^{(d-k)} \oplus \cdots \oplus W_s^{(d-k)}
\]

The same assumption for the vector spaces \(W_i^{(d-k-1)}\).

We will show that if \(m \leq \sqrt{n} \) and \(s \leq (n)^{t/2}\) then a random choice of \(\{\ell_{i,j}\}_{(i,j) \in ([s],[m])}\) satisfies the equality \(\sum_{i=1}^s W_i^{(t)} = \oplus W_i^{(t)}\). In order to show this, we will need the notion of combinatorial designs.

Definition 3.2 (Nisan-Wigderson designs [NW94]). A family of sets \(\mathcal{A} = \{A_1, \ldots, A_s\}\) is said to be an \((n, m, d)\) design if \(A_i \subseteq [n]\) with \(|A_i| = m\) for all \(i \in [s]\). And, for \(i \neq j\), \(|A_i \cap A_j| < d\).

We will be using a standard construction of such designs based on the Reed-Solomon codes.

Lemma 3 (Explicit Design). Let \(m \leq \sqrt{n}\). There exists an \((n, m, d)\)-design \(\{A_1, \ldots, A_s\}\) for \(s \leq n^d\).

Lemma 4. Let \( S \subseteq \mathbb{F} \) be a finite set. If \(m \leq \sqrt{n} \) and \(s \leq n^{t/2}\) then for a random choice of \(\{\ell_{i,j}\}_{(i,j) \in ([s],[m])}\) linear forms over \(S\),

\[
\sum_{i=1}^s W_i^{(t)} = \oplus W_i^{(t)}
\]

with probability at least \(1 - \frac{mt}{|S|}\).

Proof. To show the direct sum structure of \(W_i^{(t)}\) it suffices to show that if \(w_1 + \ldots + w_s = 0\), s.t. \(w_i \in W_i^{(t)}\), then all \(w_i = 0\). By definition of \(W_i^{(t)}\), this means that for any collection \(\{g_i\}_{i \in [s]}\) of homogeneous degree-\(t\) \(m\)-variate polynomials(not all zero), \(\sum_{i=1}^s g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m}) \neq 0\). We will prove this by showing that the following polynomials

\[
\{g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m})\}_{(i \in [s]) \text{ with } g_i \neq 0}
\]

are linearly independent. Using [Kay11, Lemma 4.1], our linear independence condition boils down to showing an \(s \times s\) determinant, with entries being degree-\(t\) polynomials, is non-zero. Thus, if there exist a choice of \(\ell_{i,j}\’s\) s.t. the aforementioned determinant is non-zero then by the Schwartz-Zippel lemma a random choice from \(S\) will work with probability \(1 - \frac{mt}{|S|}\).

\(^6\)Strictly speaking, non-degeneracy condition 2 does depend on \(g_i\’s\), but we show that it holds if we just pick \(\ell_{i,j}\’s\) randomly. See 12.
The rest of the proof is devoted to coming up with a particular setting of \( \ell_{i,j} \) s.t. the polynomials 
\( g_1(\ell_{1,1}, \ell_{1,2}, \ldots, \ell_{1,m}), \ldots, g_s(\ell_{s,1}, \ell_{s,2}, \ldots, \ell_{s,m}) \) are linearly independent. We will construct this using the Nisan-Wigderson design family. By Lemma 3, we have \((n, \sqrt{m}, t)\)-design with \( s \leq (\sqrt{m})^t \). In other words, we have sets \( A_1, A_2, \ldots, A_t \) with \( A_i \subset [n] \), \( |A_i| = \sqrt{m}, |A_i \cap A_j| \leq t-1 \) and \( s \leq (\sqrt{m})^t \). Set \( \ell_{i,1} \) to sum of first \( \sqrt{m} \) variables in \( A_i \), notice all sets lie in universe \([n]\) giving a natural ordering to its elements. Similarly, \( \ell_{i,2} \) to sum of next \( \sqrt{m} \) variables of \( A_i \), and so on. Since not all \( g_i \)'s are 0, let \( g_i \neq 0 \). Note that since \( \sqrt{m} > t \), \( g_i(\ell_{i,1}, \ldots, \ell_{i,m}) \) will have a multilinear monomial (say \( M \)) of degree \( t \). Now in order to cancel \( M \), another \( g_j(\ell_{j,1}, \ldots, \ell_{j,m}) \) must compute the same monomial. But the variables in \( g_j(\ell_{j,1}, \ldots, \ell_{j,m}) \) are supported on variables in \( A_j \) and \( |A_i \cap A_j| \leq t-1 \). Thus, \( M \) can never get cancelled implying \( \sum_{i=1}^s g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m}) \neq 0 \).

As a direct consequence of Lemma 4 for \( t = d - k - 1 \) and \( t = d - k \) we get the proof of non-degeneracy condition 1 as a corollary.

**Corollary 5.** If \((md)^2 \leq n, k = \lceil \frac{2 \log n}{\log m} \rceil \) and \( s \leq n^{d/4} \) then for a random choice of \( \{ (\ell_{i,j})_{(i,j) \in ([s],[m])} \text{ linear forms over a set } S, \sum_{i=1}^s W_i^{(d-k)} = \oplus_{i \in [s]} W_i^{(d-k)} \) and \( \sum_{i=1}^s W_i^{(d-k-1)} = \oplus_{i \in [s]} W_i^{(d-k-1)} \) with probability

at least \( 1 - \frac{2nd}{|S|} \).

### 3.2 Non-degeneracy of random circuits: Condition 2

Our next non-degeneracy condition for \( f(x) = \sum_{i=1}^s g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m}) \) requires that the vector spaces 
\( U := \langle \partial = k f \rangle \) and \( V := \langle \partial = (k+1) f \rangle \) have a direct sum structure. That is,

\[
U = U_1 \oplus \cdots \oplus U_s \quad \text{and} \quad V = V_1 \oplus \cdots \oplus V_s,
\]

where \( U_i := \langle \partial = k T_i \rangle \) and \( V_i := \langle \partial = (k+1) T_i \rangle \) where \( T_i := g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m}) \). Note that as \( U_i \subseteq W_i^{(d-k)} \), the direct sum structure of \( W_i^{(d-k)} \) directly gives \( U_1 + U_2 + \ldots + U_s = U_1 \oplus U_2 \oplus \ldots \oplus U_s \). Indeed for the regime of parameters we are interested in, \( W_i^{(d-k)} \) do have a direct sum structure for random \( \ell_{i,j} \)'s by Lemma 4. Thus in order to show non-degeneracy condition 2 for random circuit, it suffices to show

\[
U = U_1 + U_2 + \ldots + U_s.
\]

Clearly, \( U \subseteq U_1 + U_2 + \ldots + U_s \). Note that, to show the other direction it suffices to show that \( U \supseteq U_i \) for all \( i \in [s] \). We show this via a novel technique of studying the space of partial derivative operators (i.e. \( \langle \partial = k \rangle \)) themselves, as opposed to the space when they are applied to a polynomial (i.e. \( \langle \partial = k f \rangle \)). Interestingly, our proof is very general and works for action of any general linear operators on a space! Thus, we state and prove it in full generality and later instantiate the setting needed for our work.

We start by elaborating on our abstract setting. Let \( f = T_1 + T_2 + \ldots + T_s \) where \( T_i \in C_i \) and \( L \) is a vector space of linear operators from \( \mathbb{F}[x] \) to \( W \). Here, \( C_i \) is a circuit class consisting of polynomials in \( \mathbb{F}[x] \). Also, let \( B : L \times L \to \mathbb{F} \) be a non-degenerate bilinear form, that is for any non-zero \( u \in L \) there exists a \( v \in L \) s.t. \( B(u,v) \neq 0 \). Furthermore, let \( L_i^+ := \{ L \in L \mid Lh = 0, \forall h \in C_i \} \). Using the bilinear product \( B \) and any subspace \( U \) of \( L \), we define \( U^{<B} \) as the linear operators (in \( L \)) s.t. for all \( u \in U \) the bilinear product is 0. Formally, \( U^{<B} := \{ L \in L \mid \forall u \in U, B(L,u) = 0 \} \).

Our next lemma shows that under a direct sum structure of

\[
\sum_{i \in [s]} (L_i^+)^{<B}, L(f) = \sum_{i \in [s]} L(T_i).
\]

**Lemma 6.** Let \( L, B, f(x) \), and \( T_i \)'s be as defined above. If \( \sum_{i \in [s]} (L_i^+)^{<B} = \oplus_{i \in [s]} (L_i^+)^{<B} \) then \( L(f) = \sum_i L(T_i) \).

**Proof.** Note that, it suffices to show that \( L(f) \supseteq L(T_i) \) for each \( i \). We will show this for \( i = 1 \) and the rest follows analogously. We will need the following claim:
Claim 7. Let $U, V \subset \mathcal{L}$ be two subspaces and $B : \mathcal{L} \times \mathcal{L} \to \mathbb{F}$ be a non-degenerate bilinear form s.t. $U^\perp + V^\perp = U^\perp \oplus V^\perp$ then $U + V = \mathcal{L}$.

Proof. Suppose $U + V \neq \mathcal{L}$ then $w \neq 0$ s.t. $B(w, w') = 0$ for all $w' \in U + V$. This implies that $w \in (U + V)^\perp = U^\perp \cap V^\perp$ which contradicts the assumption that $U^\perp$ and $V^\perp$ form a direct sum. □

Let’s apply this claim to $U := \cap_{i=2}^s \mathcal{L}^i_1$ and $V := \mathcal{L}^1_1$. Observe that $U^\perp$ and $V^\perp$ form a direct sum because of our assumption that $(\mathcal{L}^i_1)^{\perp}$ has direct sum structure. Thus, we get that $\cap_{i=2}^s \mathcal{L}^i_1 + \mathcal{L}^1_1 = \mathcal{L}$.

Note that,

$$L(f) \supset \cap_{i=2}^s \mathcal{L}_1^i(f) = \cap_{i=2}^s \mathcal{L}_1^i(T_1) + \cap_{i=2}^s \mathcal{L}_1^i(T_2 + \ldots + T_s) = \cap_{i=2}^s \mathcal{L}_1^i(T_1).$$

Also, note that,

$$L(T_1) = \cap_{i=2}^s \mathcal{L}_1^i(T_1) + \mathcal{L}_1^1(T_1) = \cap_{i=2}^s \mathcal{L}_1^i(T_1).$$

Thus, $L(f) \supset L(T_1)$. □

For our application of showing $U \supset U_i$, we set $\mathcal{L} = \langle \partial = k \rangle$, $\mathcal{C}_i$ is the class of $g_i(\ell_{i, 1}, \ell_{i, 2}, \ldots, \ell_{i, m})$ where $g_i$ is an $m$-variate degree $d$ polynomial and $\ell_{i, 1}, \ell_{i, 2}, \ldots, \ell_{i, m}$ are random $n$-variate linear forms. Also, $\mathcal{L}^i_1 = D_{1}^i := \{D \in \langle \partial = -k \rangle | Dh = 0, \forall h \in W_1^d \}$.

Note that in order to apply Lemma 6 in our setting, we have to come up with a non-degenerate bilinear map $B$, s.t. $\sum_i (D_1^i)^{\perp} = \mathcal{L}$. Let’s first note that if $(D_1^i)^{\perp}$ does satisfy the direct sum property then we are done! Observe that, on setting, $\mathcal{L} = \langle \partial = k \rangle$ and $\mathcal{L}^1_1 = D_1^1$ to $L(f) = \sum_i L(T_i)$ gives $\langle \partial = k \rangle(f) = \sum_i \langle \partial = k \rangle(T_i)$, thus implying (5).

In the rest of the section, we will focus on coming up with a bilinear form and showing that it is indeed non-degenerate. And later, via another application of Lemma 4, show the direct sum structure of $(\mathcal{D}_k^i)^{\perp}$.

For two polynomials $f$ and $g$, define

$$B(f, g) := f \left( \frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \ldots, \frac{\partial}{\partial x_n} \right) : g.$$

This inner product among two polynomials is known as apolar inner product, and is a fundamental notion with a lot of applications, see [Pra19] and the references therein. It is easy to see that $B(\ell_1, \ell_2) = v_{\ell_1} \cdot v_{\ell_2}$; where $\ell_1(x), \ell_2(x)$ are two linear forms, $v_{\ell_1}, v_{\ell_2}$ are canonical vectors associated with them and $v_{\ell_1} \cdot v_{\ell_2}$ is the standard dot product among vectors. The non-degenerate bilinear map needed for our purpose acts on $\mathcal{L} \times \mathcal{L}$ instead of polynomials as defined above. But in our case $\mathcal{L} = \langle \partial = k \rangle$ is nothing but polynomials of degree $k$ with $\frac{\partial}{\partial x_i}$ as variables, thus the definition of $B$ extends naturally.

In order to show that our bilinear map is non-degenerate, it will be convenient to work with an orthogonal basis of $\ell_{i, j}$’s. We will therefore need the following lemma.

Lemma 8. When $\text{char}(\mathbb{F}) \neq 2$, there exists an orthogonal basis of any finite dimensional vector space over $\mathbb{F}$ with respect to any non-degenerate bilinear (dot) product.

Proof. Let $V$ be a vector space over $\mathbb{F}$ s.t. $\text{char}(\mathbb{F}) \neq 2$. We will show, by induction, that we can find an orthogonal basis for $V$. Start by finding $v \in V$ such that $\langle v, v \rangle \neq 0$. Let $W$ be the orthogonal complement of $\text{span}(v)$: the set of all $w \in V$ such that $\langle v, w \rangle = 0$. Observe that, $V = \text{span}\{v\} \oplus W$. By induction, we can get an orthogonal basis for $W$, and then add on $v$ to get an orthogonal basis of $V$.

All that we have to show now is that for any linear space $V'$, if $\exists u', v' \in V'$ s.t. $\langle v', u' \rangle \neq 0$ then $\exists x \in V'$ s.t. $\langle x, x \rangle \neq 0$. Note that for the premise of the above statement, $V$ has $u', v'$ pair s.t. $\langle v', u' \rangle \neq 0$ because $\langle \cdot, \cdot \rangle$ is non-degenerate for $V$. Also, $W$(and similarly other such vector space made in induction steps) had such pair otherwise no such pair will exist in $V$ as $W$ consists of orthogonal complement of $\text{span}(v)$. Now let $u', v' \in V'$ s.t. $\langle u', u' \rangle \neq 0$ then

$$\langle u' + v', u' + v' \rangle = \langle u', u' \rangle + \langle v', v' \rangle + 2 \langle v', u' \rangle.$$

Thus not all of $\langle u' + v', u' + v' \rangle, \langle u', u' \rangle$ and $\langle v', v' \rangle$ can be zero, which concludes the proof. □
Let $V$ be the space of some linear forms in $\mathbb{F}[x]$, then by the above lemma one can assume that there exist an orthogonal basis of $V$ as long as $\text{char}(\mathbb{F}) \neq 2$. Now, we will state an observation on what is $B(f, \cdot)$ when $f$ and $g$ are expressed as polynomials in an orthogonal basis.

**Observation 9.** If $\ell_1(x), \ldots, \ell_n(x)$ is an orthogonal basis of $\mathbb{F}[x]_1$ then if $g = \sum_{\alpha} c_{\alpha} \ell^{\alpha}$ and $f = \sum_{\alpha} d_{\alpha} \ell^{\alpha}$ are degree $d$ polynomials. Then

$$B(f, g) = \sum_{\alpha} c_{\alpha} \cdot d_{\alpha} \cdot \alpha!$$

here $\alpha = \prod_{i \in [n]} \alpha_i$! and $\alpha$ as an index varies over exponent vector of $n$-variate monomials of degree exactly $d$.

The above observation follows directly by observing it when $g$ is a monomial and extending by linearity. Now, if $\text{char}(\mathbb{F}) > d$ or 0, then using this observation we directly get that $B(f, g)$ is non-degenerate.

**Lemma 10.** The bi-linear map $B(f, g)$ over $\mathbb{F}[x]_d$ is non-degenerate if $\text{char}(\mathbb{F}) > d$ or 0. That is for all non-zero $g \in \mathbb{F}[x]_d$ there exist $f \in \mathbb{F}[x]_d$ s.t. $B(f, g) \neq 0$.

**Proof.** Let $\ell_1(x), \ldots, \ell_n(x)$ be an orthogonal basis of $\mathbb{F}[x]_1$. Now, for any $g = \sum_{\alpha} c_{\alpha} \ell^{\alpha} \neq 0$, let $\alpha_o$ be an exponent vector s.t. $c_{\alpha_o} \neq 0$. On choosing $f = \ell^{\alpha_o}$ we get $B(f, g) = c_{\alpha_o} \cdot \alpha_o! \neq 0$ if $\text{char}(\mathbb{F}) > d$ or 0. □

### 3.2.1 Direct sum structure of $(D^+_{i})_{1^b}$

The only thing left to show non-degeneracy condition 2 is a direct sum structure on derivative operators $(D^+_{i})_{1^b}$. We will first study the space $D^+_{i}$, as it will help us show the required direct sum structure. Let’s assume (WLOG) that $\ell_{i,1}, \ldots, \ell_{i,m}, q_{i,1}, \ldots, q_{i,n-m}$ is an orthogonal basis of $\mathbb{F}^n$ w.r.t. $B$. That is, for $i \neq j$

$$\langle \ell_{i,1}, \ell_{i,j} \rangle = 0, \langle \ell_{i,1}, q_{i,1} \rangle = 0 \quad \text{and} \quad \langle \ell_{i,1}, \ell_{i,j} \rangle \neq 0.$$  Notice that,

$$D^+_{i} \supseteq q_{i,1} \cdot (\partial^{(k-1)}) + q_{i,2} \cdot (\partial^{(k-1)}) + \ldots + q_{i,n-m} \cdot (\partial^{(k-1)}).$$

**Claim 11.** Let $\text{char}(\mathbb{F}) > d$ or $\text{char}(\mathbb{F}) = 0$, then $W^{(k)}_{i} := \mathbb{F}[\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m}]_{k} \supseteq (D^+_{i})_{1^b}$.

**Proof.** For brevity we will denote the space $(q_{i,1}, (\partial^{(k-1)}) + q_{i,2}, (\partial^{(k-1)}) + \ldots + q_{i,n-m}, (\partial^{(k-1)}))$ by $Q$. We have that $D^+_{i} \supseteq Q$, thus $(D^+_{i})_{1^b} \subset Q_{1^b}^{1^b}$. The proof concludes by showing that $Q_{1^b}^{1^b} = W^{(k)}_{i}$. Clearly, $Q_{1^b}^{1^b} \supseteq W^{(k)}_{i}$. For the other direction, let $p \in Q_{1^b}^{1^b}$. We can write, $p = w + q$ where $w \in W^{(k)}_{i}$ and $q \in Q$. Now, since $p \in Q_{1^b}^{1^b}$, $B(p, q') = 0$ for every $q' \in Q$. Notice that for any $q' \in Q$, $B(w, q') = 0$. Thus, $B(p, q') = B(q + w, q') = B(q, q')$. Now, just like in the proof of Lemma 10 we can choose $q' \in Q$ s.t. $B(q, q') \neq 0$. That is, pick $q'$ to be any monomial in $\mathbb{F}[\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m}, q_{i,1}, \ldots, q_{i,m}]_{d}$ with non-zero coefficient in $q$ and by observation 9 we get that $B(q, q') \neq 0$. This implies $p \in Q$ and $B(p, q') \neq 0 p(\partial) \cdot p(\bar{x}) \neq 0$, thus contradicting $p \in Q_{1^b}^{1^b}$. □

**Lemma 12.** For homogeneous degree $d$ polynomials $\{g_i\}_{i \in [n]}$, let $f = \sum_{i=1}^{n} g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m}) \in \mathbb{F}[x]_d$ and $U := (\partial = k f)$, $V := (\partial = (k+1) f)$, $U_i := (\partial = k T_i)$, $V_i := (\partial = (k+1) T_i)$. If $\text{char}(\mathbb{F}) > d$ or $\text{char}(\mathbb{F}) = 0$, $(md)^2 \leq n, k = \lceil \frac{2 \log n}{\log n} \rceil$ and $s \leq n^{d/4}$ then for a random choice of $\{\ell_{i,j}\}_{(i,j) \in [n],[m]}$ linear forms over a set $S \subset \mathbb{F}$, $U = U_1 \oplus \cdots \oplus U_s$ and $V = V_1 \oplus \cdots \oplus V_s$ with probability at least $1 - \frac{2d}{|S|^3}$.

**Proof.** By Lemma 4 we get that for a finite set $S \subset \mathbb{F}$, if $m \leq \frac{|S|^2}{2d}$ and $s \leq n^{d/4}$ then for a random choice of $\{\ell_{i,j}\}_{(i,j) \in [n],[m]}$ linear forms over $S$, $\sum_{i=1}^{n} W^{(k)}_{i} = \oplus W^{(k)}_{i}$ with probability at least $1 - \frac{sk}{|S|^3}$. Now, since $W^{(k)}_{i}$ has a direct sum structure, the same will hold for their respective subspaces. Thus, $\sum_{i \in [n]} (D^+_{i})_{1^b} = \oplus_{i \in [n]} (D^+_{i})_{1^b}$. Now, using Lemma 6 with $L = (\partial = k)$ and $L^+_i = D^+_i := \{D \in (\partial = k) \mid Dh = 0, \forall h \in W^{(d)}_{i}\}$ implies that $(\partial = k)$ directly gives $U = U_1 + U_2 + \ldots + U_s$. And, as $U_i \subseteq W^{(d-k)}_{i}$, the direct sum structure of $W^{(d-k)}_{i}$ directly gives $U_1 + U_2 + \ldots + U_s = U_1 \oplus U_2 \oplus \cdots \oplus U_s$. Notice that, $W^{(d-k)}_{i}$ have direct sum structure by corollary 5 as $m \leq \frac{|S|^2}{d}$ and $s \leq n^{d/4}$. The proof for $V = V_1 \oplus \cdots \oplus V_s$ is identical. □
3.3 Non-degeneracy condition 3: Adjoint algebra is trivial

We will start by restating non-degeneracy condition 3.

For a generalized depth 3 circuit $f = \sum_{i=1}^{s} g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m})$ where $g_i \in \mathbb{F}[z]$, $z = (z_1, \ldots, z_m)$, the triple $\left( \partial_z^{-1}, \partial_z^{-k} g_i, \partial_z^{-(k+1)} g_i \right)$ has a trivial adjoint algebra for all $i \in [s]$. That is, if $D : \langle \partial_z^{-k} g_i \rangle \to \langle \partial_z^{-(k+1)} g_i \rangle$ and $E : \langle \partial_z^{-(k+1)} g_i \rangle \to \langle \partial_z^{-(k+1)} g_i \rangle$ are linear maps s.t. $\partial_z D(p) = E(\partial_z p)$ for all $p \in \langle \partial_z^{-k} g_i \rangle$, then $D, E$ are both identity maps (up to a scalar multiple).

We will show this for random $g_i$'s as well as various interesting polynomial families which includes monomials, determinant, permanent, elementary symmetric polynomial and iterated matrix multiplication. This is done by observing that under mild technical conditions on $g$, the triple $\left( \partial_z^{-1}, \partial_z^{-k} g, \partial_z^{-(k+1)} g \right)$ has a trivial adjoint algebra. Define $\{ \partial_z^{=k} g \} := \{ \partial_z^{=k} g \mid \forall m \degree k \text{ monomials in } \mathbb{F}[z] \}$. And, let $\text{var}(f)$ denote the set of variables $f$ depends on. We start by stating our technical condition:

**Technical condition 13.** Let $g \in \mathbb{F}[z]$, we need: $\text{var}(p) \neq \text{var}(p')$ for any non-zero (and distinct) $p, p' \in \{ \partial_z^{=k} \}$ and all non-zero elements of $\{ \partial_z^{=k+1} \}$ to be $\mathbb{F}$-linearly independent.

**Lemma 14.** Let $g \in \mathbb{F}[z]$ that satisfies condition 13. Then for any linear maps $D : \langle \partial_z^{=k} \rangle \to \langle \partial_z^{=k} \rangle$ and $E : \langle \partial_z^{=k+1} \rangle \to \langle \partial_z^{=k+1} \rangle$, if $\partial_z D(p) = E(\partial_z p)$ for all $p \in \langle \partial_z^{=k} \rangle$, then $D(p) = c_p \cdot p$ for all $p \in \langle \partial_z^{=k} \rangle$, where $c_p \in \mathbb{F}$ which could depend on $p$.

**Proof.** Denote the non-zero polynomials in the set $\{ \partial_z^{=k} g \}$ by $p_1, p_2, \ldots, p_i$. Say, $D(p_i) = \sum_{r \in [t]} c_{i,r} p_r$ with $c_{i,r} \in \mathbb{F}$. Pick $z \in \text{var}(g) \setminus \text{var}(p_i)$, existence of such a variable follows directly from condition 13. On differentiating $D(p_i)$ w.r.t $z$ we get,

$$\partial_z D(p_i) = \sum c_{i,r} \partial_z p_r$$

$$= E \partial_z p_i = 0$$

\[\vdots\text{adjoint condition: } \partial_z D(p) = E(\partial_z p)\]

This gives, $\sum_{r \in [t]} c_{i,r} \partial_z p_r = 0$. Since, all non-zero elements of $\{ \partial_z^{=k+1} \}$ are $\mathbb{F}$ linearly independent, we get $c_{i,r} = 0$ for all $p_r$ s.t. $z \in \text{var}(p_r)$. Now by the assumption that when $i \neq j \text{ var}(p_i) \neq \text{var}(p_j)$, and repeating the above argument for all such $z \in \text{var}(g) \setminus \text{var}(p_i)$, we get $D(p_i) = c_p p_i$. \[\square\]

Note that, the above lemma doesn’t imply that the adjoint algebra is trivial, as $c_p$ could possibly depend on $g_i$. To show that the adjoint algebra is trivial, we need to prove that $c_p$ is the same constant for all $p_i$’s. In order to do that we will need the following notion of graph associated with a polynomial.

**Definition 3.3.** For a polynomial $g \in \mathbb{F}[z]$, let $G^k_g$ be the graph whose vertices are degree $k$ multilinear monomials in $s.t. \partial_z^{=k} g \neq 0$ and the edge set consist of monomials $m_1, m_2$ with $\Delta(m_1, m_2) = 2$ and $\partial_z^{=k+1} m \neq 0$. Here $\Delta(m_1, m_2)$ is the hamming distance among the exponent vectors of $m_1$ and $m_2$.

**Lemma 15.** Let $g \in \mathbb{F}[z]$ be a polynomial s.t. it satisfies condition 13. Additionally, if $G^k_g$ is connected, then the triple $\langle \partial_z^{-1}, \partial_z^{-k} g, \partial_z^{-(k+1)} g \rangle$ has a trivial adjoint algebra.

**Proof.** As $g$ satisfies condition 13, Lemma 14 implies that for all non-zero elements $p_1, p_2, \ldots, p_i \in \{ \partial_z^{=k} \}$, $D(p_i) = c_{p_i} p_i$. All that is left to show now is that $c_{p_i}$ is the same constant for all $i \in [t]$. Let $p = \partial_z^{-k} g$ and $p' = \partial_z^{-k} g$, where $m_1$ and $m_2$ are two different degree $k$ monomials in $\mathbb{F}[z]$. Suppose, $\Delta(m_1, m_2) = 2$, and $\partial_z^{=k} g \neq 0$ where $m = \text{lcm}(m_1, m_2)$. Also, define $z$ and $z'$ by $m = z \cdot m_1$ and $m = z' \cdot m_2$. \[\square\]
Note that,

\[ c_p \partial_m^k g = c_p \partial_m^p p = \partial_m^p D(p) = E(\partial_m^p) = E(\partial_m^p p') = \partial_m^p p' \]

\[ \therefore \text{ Adjoint Condition} \]

Thus, \( c_p' = c_p \) and as \( G_g^k \) is connected, it gives \( c_p = c_p' \) for any non-zero \( p, p' \in \{ \partial = k \} \).

It is an easy exercise to see that for a random multilinear \( g \) condition 13 is satisfied and \( G_g^k \) is connected. We will now show the same holds for various polynomial families which includes monomials, determinant, permanent, elementary symmetric polynomial and iterated matrix multiplication. The argument for showing connectivity of \( G_g^k \) stems from this simple observation.

**Observation 16.** If \( m_1 \) and \( m_2 \) are degree \( k \) monomials with \( \Delta(m_1, m_2) = \delta \) and let \( m_1 = m^{(0)}, m^{(1)}, \ldots, m^{(\delta)} = m_2 \) be a path made of distance two monomials (i.e. \( \Delta(m^{i-1}, m^{i}) = 2 \) for \( i \in \{ \delta \} \)) from \( m_1 \) to \( m_2 \) s.t. \( \partial_{m^{(i)}} g \neq 0 \) (\( \text{lcm}(m^{(i)}, m^{(i+1)}) \)) then \( m_1 \) and \( m_2 \) are connected.

**Lemma 17.** If \( g \) is one of the following polynomials \( \text{Det}_d \), \( \text{Perm}_d \) (with \( 3k \leq d \)); \( \text{Sym}_{r,d} \), monomial (with \( k + 1 \leq d \)) or \( \text{IMM}_{r,d} \) (with \( 3k \leq d \)) then \( G_g^k \) is connected and condition 13 is satisfied.

**Proof.** \( g = \text{Det}_d \) or \( \text{Perm}_d \): Since, \( k \)-th order partial derivatives of determinant(permanent) is spanned by determinant(permanent) of \((n-k) \times (n-k)\) minors, it is easy to see that they satisfy condition 13. Let \( m_1 \) and \( m_2 \) be two degree \( k \) multilinear monomials s.t. \( \partial_{m_1} g \neq 0 \) and \( \partial_{m_2} g \neq 0 \). We have to show that there is a path from \( m_1 \) to \( m_2 \). Note that, by relabelling of variables, we can assume that \( m_1 = x_{11}x_{22} \cdots x_{kk} \). Say,

\[ m_2 = x_{i_1,j_1}x_{i_2,j_2} \cdots x_{i_k,j_k} \text{ s.t. } i_1 \neq i_2 \neq \cdots \neq i_k \text{ and } j_1 \neq j_2 \neq \cdots \neq j_k. \]

Define \( I := \{ i_1, i_2, \ldots, i_k \} \) and \( J := \{ j_1, j_2, \ldots, j_k \} \). Let \( I' \) be a subset of \( |d| \setminus \{|k| \cup I\} \) of size \( k \) and similarly \( J' \) be a subset of \( |d| \setminus \{|k| \cup J\} \) of size \( k \). Note that, \( I', J' \) exist because of our assumption that \( 3k \leq d \). Now, consider the monomial \( \tilde{m} = \prod_{i \in I', j \in J'} x_{ij} \). Clearly, \( \partial_{\tilde{m}} g \neq 0 \). Also, note that any \( \Delta = 2 \) path between \( m_1 \) and \( \tilde{m} \) is connected. Similarly, any path between \( m_2 \) and \( \tilde{m} \) is connected, which concludes the proof.

\( g = \text{Sym}_{r,d} \): It is easy to see that condition 13 is satisfied. Note that, the vertex set in this case consists of all degree \( k \) multilinear monomials in \( r \) variables. Let’s denote these monomials by the underlying monomial support set. That is, for a set \( S \subseteq [r] \) of size \( k \), \( m_S \) be the multilinear monomial associated with \( S \). Now, we have to show that for two sets \( S_1, S_2 \) of size \( k \), \( m_{S_1} \) and \( m_{S_2} \) are connected. Note that, for any \( m_{S_1}, m_{S_2} \) s.t. \( \Delta(m_{S_1}, m_{S_2}) = 2 \) then \( \partial_{\tilde{m}} (\text{Sym}_{r,d}) \neq 0 \) as \( k < d - 1 \). Thus, as a direct consequence of observation 16, we get \( G_g^k \) is connected for \( g = \text{Sym}_{r,d} \).

\( g = \text{IMM}_{r,d} \): It is easy to see that condition 13 is satisfied. For connectivity of \( G_g^k \), note that a derivative with respect to any variable simply results in the sum of all source-sink paths that pass through that edge. The rest of the proof is analogous to the case of \( \text{Det}_d \). Here, instead of working with disjoint minors, we will work with disjoint “paths” which exists if \( 3k \leq d \).

### 3.4 Adjoint algebra for random \( g_i \)’s

We will now show that the adjoint algebra is trivial for random \( g_i \)’s. This is done by showing that the adjoint algebra is trivial if the space spanned by \( k + 1 \) order partial derivatives applied to \( \langle g_1, g_2, \ldots, g_s \rangle \) have full dimension. Followed by observing that the aforementioned space has full dimension when \( g_i \)’s are chosen randomly.

**Lemma 18.** Let \( g_1, g_2, \ldots, g_s \in \mathbb{F}[z]_{d} \) be polynomials such that \( \dim \langle \partial^{(k+1)}(g_1, g_2, \ldots, g_s) \rangle = s \cdot (k+1)^m \) where \( \langle g_1, g_2, \ldots, g_s \rangle := \mathbb{F}\text{-span}(g_1, g_2, \ldots, g_s) \). Also, let \( U_1 = \langle \partial = k \rangle, \ V_i = \langle \partial = k+1 \rangle, \ U = U_1 \oplus \cdots \oplus U_s \) and
Lemma 4.1 boils down to showing an

\[ S \subseteq V = \text{geneous generalized depth 3 circuit}, \]

where \( \{ n \} \). We know that, \( \partial_z D(f) = E(\partial_z f) \) for all \( f \in U \). Thus for any \( z_j, z'_j, Z, Z' \) s.t. \( j \cup Z = j' \cup Z' \) we get,

\[
\partial_z \cdot D(\partial_z f) = \partial_{z'_j} \cdot D(\partial_{z'_j} f).
\]

This along with the assumption in the lemma that \( \dim(\partial = (k+1) \langle f_1, f_2, \ldots, f_s \rangle) = s \frac{(k+m)}{k+1} \) implies

\[
z_j p_i(z) = z_j' p_i(z)
\]

for all \( z_j, z'_j, Z, Z' \) s.t. \( j \cup Z = j' \cup Z' \). Note the above is a formal equality of polynomials. Consider \( Z = \{ 1, 2, \ldots, r \} \). The above equations imply that \( z_j \) divides \( p_i \) for all \( j \in [r] \). Since \( p_i \) is homogeneous of degree \( r \), this implies that \( p_i \) is \( c_i \) for some \( c_i \in F \). Now \( p_i \) is \( c_i \prod_{i \in Z} z_i \) for all \( Z \), is easy to see via a simple connectedness argument using the equations above. This finishes the proof.

We can instantiate the above lemma for random \( g_i \)'s. Note that, the condition \( \dim(\partial = (k+1) \langle g_1, g_2, \ldots, g_s \rangle) = s \frac{(k+m)}{k+1} \) is about linear independence of \( s \frac{(k+m)}{k+1} \) many polynomials of degree at most \( d \). Which using [Kay11, Lemma 4.1] boils down to showing an \( s \frac{(k+m)}{k+1} \) determinant, with entries being degree-\( d \) polynomials, is non-zero. Thus, via the Schwartz-Zippel lemma, we get the following corollary.

**Corollary 19.** For a random choice of degree \( d \) homogeneous polynomials \( \{ g_i \}_{i \in [s]} \) over a set \( S \), the triple \( \left( \partial_{z}^{1}, (\partial_{z} = k g_i), (\partial_{z} = (k+1) g_i) \right) \) has a trivial adjoint algebra for all \( i \in [s] \), with probability at least \( 1 - \frac{sd(m+k)}{|S|} \).

Now, we can combine corollary 5, 19 and Lemma 17, 12 to show that a random generalized depth 3 circuit is non-degenerate with high probability.

**Lemma 20 (Random generalized depth 3 circuits are non-degenerate).** Let \( C = \sum_{i=1}^{s} g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m}) \) be a homogeneous generalized depth 3 circuit, where \( \{ g_i \}_{i \in [s]} \) are homogeneous degree-\( d \) polynomials, and \( n \geq (md)^2 \). Suppose the coefficients of \( \ell_{i,j} \)'s are chosen uniformly and independently at random from a set \( S \subseteq F \) of size \( |S| \geq (nds)^3 \). Additionally, suppose one of the following cases is true:

1. \( g_i \)'s belong to one of the polynomial families: \( \text{Det}_d, \text{Perm}_d, \text{IMM}_{r,d}, \text{Sym}_{r,d} \), monomials with \( s \leq n^{d/6} \).
2. Coefficients of all \( g_i \)'s are chosen uniformly and independently at random from \( S \) and \( s \leq n^{d/4} \).

Then, with probability \( 1 - o(1), C \) is non-degenerate.

The proof is immediate using union bound along with remark 3.1 and hence omitted. As a direct consequence of Lemma 20 and theorem 2.1, we get the following theorem about learning “random” generalized depth circuits.

**Theorem 3.4 (Learning random generalized depth 3 circuits).** Let \( C = \sum_{i=1}^{s} g_i(\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,m}) \) be a homogeneous generalized depth 3 circuit, where \( \{ g_i \}_{i \in [s]} \) are homogeneous degree-\( d \) polynomials, and \( n \geq (md)^2 \). Suppose the coefficients of \( \ell_{i,j} \)'s are chosen uniformly and independently at random from a set \( S \subseteq F \) of size \( |S| \geq (nds)^3 \) and \( \text{char}(F) > d \) or \( \text{char}(F) = 0 \). Additionally, suppose one of the following cases is true:

1. \( g_i \)'s belong to one of the polynomial families: \( \text{Det}_d, \text{Perm}_d, \text{IMM}_{r,d}, \text{Sym}_{r,d} \), monomials with \( s \leq n^{d/6} \).
2. Coefficients of all \( g_i \)'s are chosen uniformly and independently at random from \( S \) and \( s \leq n^{d/4} \).

Then, given black-box access to \( C \) we can reconstruct it in randomized \( \text{poly}(n, m, d, s) \) time.

Note that, the \( m \) subsumes the dependence on \( r \) in the above theorem. Also, the reconstruction algorithm of theorem 3.4 is proper, i.e. it outputs a homogeneous generalized depth 3 circuit.
3.5 From black-box access to learning generalized depth-three circuits

Theorem 2.1 gives a black-box for each $g_i$'s under the technical conditions discussed. It is natural to ask if we can find $\ell_{i,j}$'s and a generalized depth 3 representation as well. This is related to the well studied equivalence-testing problem, specifically to the search version of it. The equivalence-testing question is the following: given polynomials $f$ and $g$, find an invertible linear transformation $A$ on variables such that $f = g(Ax)$, if such $A$ exists. Observe that if $g_i$ belongs to a family for which we can solve the equivalence-testing problem, then we can find $\ell_{i,j}$'s as well. This follows directly by seeing each blackbox as an instance of equivalence-testing (search version). Note that in our non-degenerate setting, $\ell_{i,j}$'s are linearly independent for each $i \in [s]$ thus satisfies the requirement that the linear transformation has to be invertible. As a direct consequence of this we get the following corollary.

Corollary 21. Suppose we are given black-box access to $f$, an $n$-variate, homogeneous degree $d$ polynomial computable by a generalized depth 3 circuit of size $s$, s.t. the non-degeneracy condition 1, 2 and 3 hold. Additionally, if each $g_i$ belongs to a family of polynomials for which there exist a poly($n,m,d$) time equivalence-testing algorithm. Then there exist a poly($s,n,d,m$) time algorithm that learns a generalized depth 3 representation of $f$.

Note that if $g_i$ is just a monomial (the special case for depth 3 circuits) then equivalence-testing follows directly from black-box factoring \cite{KT90}. Hence, when $g_i$'s are monomials the previous corollary along with Lemma 17 (monomials satisfy non-degeneracy condition 3). Gives an algorithm for learning non-degenerate homogenous depth three circuits! So, our result is a truly a generalization of the result by \cite{KS19}.

In general, equivalence-testing is considered to be a very hard problem (see \cite{Kay12, Kay11}) but it's been solved in several interesting cases, we list some of them below. For ease of representation, let us define some notation representing the complexity of the search version of the polynomial equivalence problem over a particular field. Given $m,d,r \in \mathbb{N}$ and black-box access to an $m$-variate polynomial $g$ of degree $d$, let $\text{Eqv}_F(r,d,m,f)$ denote the randomized time complexity of finding an invertible linear transformation $A$ s.t. $g(x) = f(Ax)$ if it exists, otherwise output “no-solution”.

Theorem 3.5. Following results are known for equivalence-testing of special families of polynomials:

1. $\text{Eqv}_F(r,d,m,\text{Sym}_{r,d}) = \text{poly}(r,d,m)$, if $\text{char}(\mathbb{F}) > d$ or 0. See \cite{Kay12}.
2. $\text{Eqv}_F(r,d,m,\text{Perm}_r) = \text{poly}(r,d,m)$. See \cite{Kay12}.
3. $\text{Eqv}_F(r,d,m,\text{Det}_r) = \text{poly}(r,d,m)$ if $\text{char}(\mathbb{F}) \nmid r(r-1)$ or $\mathbb{F} = \mathbb{C}$. See \cite{Kay12, GGKS19}.
4. $\text{Eqv}_F(r,d,m,\text{IMM}_{r,d}) = \text{poly}(r,d,m)$ if $\text{char}(\mathbb{F}) = 0$ or greater than $d^c$ ($c$ some fixed constant). See \cite{KNST17}.

Thus, corollary 21 along with theorem 3.5 and 3.4 gives a randomized poly$(n,d,m)$-time algorithm that outputs a generalized depth three representation, assuming $\ell_{i,j}$’s are chosen randomly, $g_i$’s belong to one of the polynomial families: Det$_d$, Perm$_d$, IMM$_{r,d}$, Sym$_{r,d}$, monomials and the corresponding assumptions on $\mathbb{F}$ holds.

4 Conclusion and open problems

We design an algorithm for learning generalized depth-three circuits in the non-degenerate case. We follow the learning from lower bounds framework of \cite{KS19, GKS20} and design new tools for proving that non-degeneracy conditions hold for random circuits, which could be useful for other such problems. Our model captures widely applicable problems such as tensor decomposition and Tucker decomposition as special cases. We are hopeful that our algorithms will find powerful applications in machine learning. We list some of the most interesting open problems next.
1. **Going beyond tensor decomposition.** Can we capture more general and powerful problems in machine learning via the model of generalized depth-three circuits?

2. **Making the algorithms noise-resilient.** Can we make our algorithms robust to noise? That is, if one is given (explicitly or black box access) \( f(x) = \sum_{i=1}^{s} g_i(\ell_1, \ldots, \ell_m) + E(x) \), for some error term \( E(x) \), can we approximately recover the summands? Such a noise-resilient version is almost a prerequisite for machine learning applications.

3. **Learning other arithmetic circuit models.** Can we learn other arithmetic circuit models in the non-degenerate case, for which we already have lower bounds? Perhaps the most appealing model to go for next is that of constant depth set-multilinear circuits. There are even new lower bounds for this model now [LST21].

**References**


A  Adjoint algebra and vector space decomposition

In this section, we prove some basic facts about the adjoint algebra and vector space decomposition, for completeness. We will start by stating that we can compute black-box access to partial derivatives of $f$ from black-box access to $f$.

**Fact 22.** Given black-box access to a $(n,d)$ polynomial $f$ and a monomial $x^\alpha$, a black-box access to $\partial_\alpha^k f$ can be computed in deterministic $\text{poly}(n,d^k)$ time.

This follows from the fact that black-box access to a first-order derivative of $f$ can be computed in deterministic polynomial time from black-box access to $f$.

Next, we define the adjoint algebra.

**Definition A.1 (Adjoint algebra).** Consider a collection of linear maps $\mathcal{L}$ from vector space $U$ to vector space $V$ (over a field $F$). The adjoint algebra for this collection of linear maps is defined as follows:

$$\text{Adj}(\mathcal{L}, U, V) = \{(D, E) \mid D : U \to U, E : V \to V \text{ are linear maps s.t. } LD = EL \text{ for all } L \in \mathcal{L}\}$$

Next we define the notion of a vector space decomposition.

**Definition A.2 (Vector space decomposition).** Consider a collection of linear maps $\mathcal{L}$ from vector space $U$ to vector space $V$. We say that $U = U_1 \oplus \cdots \oplus U_s$ and $V = V_1 \oplus \cdots \oplus V_s$ is a vector space decomposition for the triple $(\mathcal{L}, U, V)$ if $\mathcal{L}(U_i) \subseteq V_i$ for all $i \in [s]$ (and at least one of $U_i, V_i$ is a non-trivial subspace). We say that the decomposition is further indecomposable if the triples $(\mathcal{L}, U_i, V_i)$ are indecomposable for all $i$.

The next definition is about when the adjoint algebra is trivial.

**Definition A.3 (Trivial Adjoint algebra).** Consider a collection of linear maps $\mathcal{L}$ from vector space $U$ to vector space $V$. Also consider a decomposition, $U = U_1 \oplus \cdots \oplus U_s$, $V = V_1 \oplus \cdots \oplus V_s$ that is further indecomposable. We say that the adjoint algebra is trivial if

$$\text{Adj}(\mathcal{L}, U, V) = \{ (D, E) : \exists \lambda_1, \ldots, \lambda_s \text{ s.t. } D|_{U_i} = \lambda_i \mathbb{1}_{U_i}, E|_{V_i} = \lambda_i \mathbb{1}_{V_i} \text{ for all } i \in [s] \}$$

Next we define what we mean by uniqueness of decomposition.
Definition A.4 (Uniqueness of decomposition). Consider a collection of linear maps \( \mathcal{L} \) from vector space \( U \) to vector space \( V \). Also consider a decomposition, \( U = U_1 \oplus \cdots \oplus U_s = V_1 \oplus \cdots \oplus V_s \) that is further indecomposable. We say that the decomposition is unique if for any other further indecomposable decomposition, \( U = U'_1 \oplus \cdots \oplus U'_{s'} = V'_1 \oplus \cdots \oplus V'_{s'} \), it turns out that \( s = s' \) and there exists a permutation \( \pi : [s] \to [s] \) s.t. \( U'_i = U_{\pi(i)} \) and \( V'_i = V_{\pi(i)} \) for all \( i \in [s] \).

The next lemma states the uniqueness of decomposition in the case when the adjoint algebra is trivial. The uniqueness of decomposition holds in a much more general setting by a reduction to the Krull-Schmidt theorem (see [GKS20]) but here we only focus on a special case that is relevant to us.

Lemma 23. Consider a collection of linear maps \( \mathcal{L} \) from vector space \( U \) to vector space \( V \). Also consider a decomposition, \( U = U_1 \oplus \cdots \oplus U_s = V_1 \oplus \cdots \oplus V_s \) that is further indecomposable. Suppose the adjoint algebra is trivial. Then the above is the unique decomposition that is further indecomposable.

Proof. Suppose, for contradiction, there is another further indecomposable decomposition \( U = U'_1 \oplus \cdots \oplus U'_{s'} = V'_1 \oplus \cdots \oplus V'_{s'} \). Consider the linear map \( \Pi_{U_i} : U \to U \) which projects onto \( U'_i \) (i.e. is identity on \( U'_i \) maps vectors in \( U'_j, j \neq i \), to 0, extending to the rest of \( U \) linearly). Similarly, we can define \( \Pi_{V_i} : V \to V \). Since \( \mathcal{L}(U'_i) \subseteq V'_i \) for all \( i \in [s] \), it is easy to verify that \( (\Pi_{U_i}, \Pi_{V_i}) \in \text{Adj}(\mathcal{L}, U, V) \) for all \( i \in [s'] \). Because the adjoint algebra is trivial, for each \( i \in [s'] \), there exist \( (\lambda_1, \ldots, \lambda_s) \) (suppressing the dependence on \( i \) for notational convenience) s.t.

\[
\Pi_{U_i}|_{U_j} = \lambda_j \mathbb{1}_{U_j}, \Pi_{V_i}|_{V_j} = \lambda_j \mathbb{1}_{V_j}
\]

for all \( j \in [s] \). If \( \lambda_j \neq 0 \), then for all \( u \in U_j \), \( \Pi_{U_j} u = \lambda_j u \) which implies that \( u \in U'_i \). Thus \( \oplus_{j: \lambda_j \neq 0} U_j \subseteq U'_i \) and similarly \( \oplus_{j: \lambda_j \neq 0} V_j \subseteq V'_i \). If \( \lambda_j = 0 \), then for all \( u \in \oplus_{j: \lambda_j = 0} U_j \), \( \Pi_{U_j} u = 0 \) which implies that \( (\oplus_{j: \lambda_j = 0} U_j) \cap U'_i = \{0\} \). Along with

\[
U = \oplus_{j: \lambda_j \neq 0} U_j \oplus \oplus_{j: \lambda_j = 0} U_j
\]

and

\[
\oplus_{j: \lambda_j \neq 0} U_j \subseteq U'_i
\]

we get that \( \oplus_{j: \lambda_j \neq 0} U_j = U'_i \). Similarly, \( \oplus_{j: \lambda_j \neq 0} V_j = V'_i \). Since the triple \( (\mathcal{L}, U'_i, V'_i) \) is indecomposable, we get that there exists a \( j \) s.t. \( (U'_i, V'_i) = (U_j, V_j) \). Since this is true for every \( i \in [s'] \), it completes the proof of uniqueness of decomposition.

Next we state an algorithm for vector space decomposition. While an algorithm in a much more general setting follows from known algorithms for module decomposition (see [GKS20]), the algorithm we state here has the advantage that it is simpler and works for large enough fields (as opposed to algebraically closed fields). This algorithm is also present in [GKS20] but not in a very explicit form, so we restate it here for completeness as well.

Lemma 24. Algorithm 2 with parameter \( \ell \) computes the correct decomposition when the adjoint algebra is trivial, with probability at least \( 1 - \binom{s}{\ell} / \ell \).

Proof. Since the adjoint algebra is trivial,

\[
\text{Adj}(\mathcal{L}, U, V) = \{(D, E) : \exists \lambda_1, \ldots, \lambda_s \text{ s.t. } D|_{U_i} = \lambda_i \mathbb{1}_{U_i}, E|_{V_i} = \lambda_i \mathbb{1}_{V_i} \text{ for all } i \in [s]\}
\]

Let \( (\lambda^{(j)}_1, \ldots, \lambda^{(j)}_s) \) be the tuple corresponding to \( (D_j, E_j) \). Then

\[
D'_i|_{U_i} = \left( \sum_{j=1}^s \mu_j \lambda^{(j)}_i \right) \mathbb{1}_{U_i}
\]

We know that the vectors \( (\lambda^{(j)}_1, \ldots, \lambda^{(j)}_s) \), for \( j \in [s] \), are linearly independent. Hence the vectors \( (\lambda^{(1)}_i, \ldots, \lambda^{(s)}_i) \), for \( i \in [s] \), are also linearly independent. Hence for \( i \neq i' \), the linear polynomial (in the \( \mu_j \)'s) \( \sum_{j=1}^s \mu_j \lambda^{(j)}_i - \sum_{j=1}^s \mu_j \lambda^{(j)}_{i'} \)
Algorithm 2 Vector space decomposition when adjoint algebra is trivial

**Input:** Set of linear maps \( \mathcal{L} \) between vector spaces \( U \) and \( V \) s.t. the triple \((\mathcal{L}, U, V)\) admits a further indecomposable decomposition \( U = U_1 \oplus \cdots \oplus U_s, V = V_1 \oplus \cdots \oplus V_s \). Also the adjoint algebra is trivial.

**Output:** \( s \) vector spaces \( U'_1, \ldots, U'_s \) s.t. there exists a permutation \( \pi : [s] \to [s] \) s.t. \( U'_i = U_{\pi(i)} \).

**Subroutine:** Diagonalizing a diagonalizable linear map \( D : U \to U \).

**Parameters:** Randomness parameter \( \ell \).

1: Compute a basis \((D_1, E_1), \ldots, (D_s, E_s)\) of the adjoint algebra \( \text{Adj}(\mathcal{L}, U, V) \) (this is a system of linear equations). (If dimension is not \( s \), then abort).

2: Pick \( \mu_1, \ldots, \mu_s \) uniformly at random from a set of size \( \ell \). Set \( D' = \mu_1 D_1 + \cdots + \mu_s D_s \).

3: Compute the eigenvalues of \( D' \). If it has \( s \) distinct eigenvalues, call them \( \lambda_1, \ldots, \lambda_s \). If not (or it is not diagonalizable), abort.

4: Set \( U'_i \) to be the eigenspace of \( D' \) corresponding to \( \lambda_i \).

\( \lambda_{i'}(j) \) is non-zero and hence if the \( \mu_j \)'s are chosen at random from a set of size \( \ell \), then with probability at least \( 1 - 1/\ell \),

\[
\sum_{j=1}^s \mu_j (\lambda_{i'}^{(j)} - \lambda_i^{(j)}) \neq 0
\]

By a union bound, with probability at least \( 1 - (s^2)/\ell \), for any \( i \neq i' \),

\[
\sum_{j=1}^s \mu_j (\lambda_i^{(j)} - \lambda_{i'}^{(j)}) \neq 0
\]

Thus there are \( s \) distinct eigenvalues of \( D' \), one each corresponding to the eigenspace \( U_i \). This completes the proof.

B Linear algebra with black boxes

In Algorithm 1, we need to perform linear algebra given black boxes for polynomials. We give references for how to do this here. We will need the following lemma from [Kay11].

**Lemma 25** (Section A.1 in [Kay11]). Given black boxes for the polynomials \( f_1, \ldots, f_\ell \in \mathbb{F}[x]_d \), there is a randomized \( \text{poly}(n, \ell, d) \) time algorithm that computes a basis for the following vector space

\[
(f_1, \ldots, f_\ell)^\perp := \{(\alpha_1, \ldots, \alpha_\ell) : \sum_{i=1}^\ell \alpha_i f_i = 0\}.
\]

In particular, we get the following corollary.

**Corollary 26.** Given black boxes for the polynomials \( f_1, \ldots, f_\ell \in \mathbb{F}[x]_d \) which are linearly independent and for a \( p \in \mathbb{F}[x]_d \) which linearly depends on \( f_1, \ldots, f_\ell \), there is a randomized \( \text{poly}(n, \ell, d) \) time algorithm that computes \( \beta_1, \ldots, \beta_\ell \) s.t.

\[
p = \sum_{i=1}^\ell \beta_i f_i
\]

Using Corollary 26, one can compute the matrices corresponding to the linear maps \( \mathcal{L} \) in Algorithm 2 if one is given only black boxes for bases of \( U \) and \( V \). One can also carry out the Step 3 in Algorithm 1 using Corollary 26.