Subspace Designs and Directed Acyclic Graphs: An Approach to High-Dimensional Causality

Kweku A. Opoku-Agyemang*

June 2023

Abstract

In this paper, we explore the connections between subspace designs and directed acyclic graphs (DAGs), and how they can enhance each other. Subspace designs are mathematical objects that involve arranging subspaces of a finite vector space into sets that satisfy certain combinatorial properties. DAGs are tools for representing the probabilistic relationships among a set of variables. We show that subspace designs can improve DAGs in two ways: by providing efficient methods for encoding and decoding information in high-dimensional spaces, and by discovering new patterns and structures in data. We use subspace designs to construct error-correcting codes, which can help to recover the original information from noisy or corrupted messages that are passed among nodes in the DAG using inference algorithms. We also use subspace designs to construct dimension expanders, which can help to find low-dimensional embeddings of high-dimensional data that reveal hidden features or clusters. We close with subspace design applications to causal diagrams, regression, clustering and classification.

*Chief Scientist, Machine Learning X Doing and Honorary Fellow, International Growth Centre, London School of Economics. Email: kweku@machinelearningxdoing.com. I thank several people at the Berkeley Expert Systems and Technologies Lab, the Berkeley Institute for Data Science, the Berkeley Institute for Transparency in Social Science, Cornell Tech and others for encouragement. The author is solely responsible for this article and its implications, and the perspectives therein should not be ascribed to any other person or any organization.

Contents

1 Introduction

Subspace designs and directed acyclic graphs (DAGs) are two important concepts that arise in various fields of mathematics and computer science. Subspace designs are mathematical objects that involve arranging subspaces of a finite vector space into sets that satisfy certain combinatorial properties. They have applications in coding theory, cryptography, and combinatorics. DAGs are tools for representing the probabilistic relationships among a set of variables. They have applications in causal inference, machine learning, and other graphical models.

In this paper, we explore the connections between subspace designs and DAGs, and how they can enhance each other. We show that subspace designs can improve DAGs in two ways: First, by providing efficient methods for encoding and decoding information in high-dimensional spaces, and by discovering new patterns and structures in data. We use subspace designs to construct error-correcting codes, which can help to recover the original information from noisy or corrupted messages that are passed among nodes in the DAG using inference algorithms. Second, we also use subspace designs to construct dimension expanders, which can help to find low-dimensional embeddings of high-dimensional data that reveal hidden features or clusters. We also demonstrate the theoretical and practical benefits of our approach using examples and experiments.

Subspace designs may be important for DAGs from a causal inference standpoint for two main reasons. First, subspace designs can provide efficient methods for encoding and decoding information in high-dimensional spaces. This can help to recover the original information from noisy or corrupted messages that are passed among nodes in the DAG using inference algorithms. For example, subspace designs can be used to construct error-correcting codes, which can correct errors that occur during transmission or storage of data. Second,

subspace designs can help to discover new patterns and structures in data. This can help to find low-dimensional embeddings of high-dimensional data that reveal hidden features or clusters. For example, subspace designs can be used to construct dimension expanders, which can increase the dimensionality of data while preserving its intrinsic structure. This can help to simplify the data and identify the most relevant variables for the DAG. Dimensionality reduction techniques, such as principal component analysis or latent variable models, can then be applied to the expanded data.

The paper is similar in spirit to newer work in the Bayesian inference literature that focuses on broadening Bayesian networks with scaling to deep neural networks, by constructing low-dimensional subspaces of parameter space and applying slice sampling and variational inference (e.g. Izmailov, Maddox, Kirichenko, Garipov, Vetrov, and Wilson, 2020). Other related work (e.g. VanderWeele, Mathur and Chen, 2020) focuses on the outcome-wide longitudinal design, where the causal effects of a treatment or exposure using confounding control, are now explored over numerous outcomes. In economics, a newer line of research integrates DAGs into economic theory and applied economics (e.g. Spiegler, (2016, 2017, 2020a, 2020b); Eliaz, Spiegler and Weiss (2021); Eliaz, Spiegler and Thysen (2021). Our focus, however, is on integrating subspace designs into DAGs to improve the DAGs themselves.

In this paper, I observe that these two aspects of subspace designs can enhance DAGs in terms of both accuracy and efficiency. They can also facilitate causal inference by enabling better estimation and identification of causal effects.

The paper proceeds as follows. Section 2 introduces some preliminary definitions and results on subspace designs and DAGs. Section 3 shows how subspace designs can be used to construct error-correcting codes, and how they can im-

prove the reliability and efficiency of inference algorithms for DAGs. Section 4 shows how subspace designs can be used to construct dimension expanders, and how they can help to find low-dimensional embeddings of high-dimensional data that reveal hidden features or clusters. Section 5 concludes the paper and discusses some open problems and future directions.

2 Preliminaries

In this section, we introduce some basic definitions and results on subspace designs and DAGs that will be used throughout the paper. We assume that the reader is familiar with some elementary concepts from linear algebra, graph theory, and probability theory.

2.1 Subspace designs

Let q be a prime power, and let F_q denote the finite field with q elements. Let V be an n-dimensional vector space over F_q, and let $\mathcal{P}(V)$ denote the set of all subspaces of V. For any subspace $U \in \mathcal{P}(V)$, let $\dim(U)$ denote its dimension, and let $|U|$ denote its cardinality, which is equal to $q^{\dim(U)}$. For any two subspaces $U, W \in \mathcal{P}(V)$, let $\langle U, W \rangle$ denote their span, and let $U+W$ denote their sum. We also define the distance between two subspaces as

$$d(U, W) = \dim(U) + \dim(W) - 2\dim(U \cap W).$$

A subspace design is a collection of subspaces of V that satisfies certain combinatorial properties. Formally, we have the following definition.

Definition 2.1. A (q, n, k, t, λ)-subspace design is a set $\mathcal{D} \subseteq \mathcal{P}(V)$ such that

- $\dim(U) = k$ for all $U \in \mathcal{D}$; - for any $W \in \mathcal{P}(V)$ with $\dim(W) = t$, there

are exactly λ subspaces in $\mathcal{D}$ that contain W, i.e.,

$$|\{U \in \mathcal{D} : W \subseteq U\}| = \lambda.$$

The parameters q, n, k, t, λ are called the field size, ambient dimension, subspace dimension, intersection dimension, and replication number, respectively. We also define the size of a subspace design as

$$|\mathcal{D}| = v,$$

and the minimum distance of a subspace design as

$$d(\mathcal{D}) = \min_{U,W \in \mathcal{D}, U \neq W} d(U, W).$$

Subspace designs generalize the classical notion of ordinary designs, which are also known as t-designs or balanced incomplete block designs (BIBDs). Ordinary designs are collections of subsets of a finite set that satisfy similar properties as subspace designs, but with subspaces replaced by subsets, and dimensions replaced by cardinalities. Formally, we have the following definition.

Definition 2.2. A (v, k, t, λ)-design is a pair $(X, \mathcal{B})$, where X is a finite set of size v, and $\mathcal{B}$ is a collection of subsets of X such that

- $|B| = k$ for all $B \in \mathcal{B}$; - for any subset $T \subseteq X$ with $|T| = t$, there are exactly λ subsets in $\mathcal{B}$ that contain T, i.e.,

$$|\{B \in \mathcal{B} : T \subseteq B\}| = \lambda.$$

The parameters v, k, t, λ are called the block size, subset size, intersection size, and replication number, respectively.

It is easy to see that ordinary designs are special cases of subspace designs

when $q = 2$ and $\dim(U) = |U|$ for all $U \in \mathcal{D}$. However, subspace designs have more flexibility and richness than ordinary designs, as they can exploit the linear structure of the vector space. For example, subspace designs can have larger minimum distances than ordinary designs with the same parameters.

For the sake of clarity, we can present subspace designs alternatively as follows:

We denote by F_q the finite field with q elements, where q is a prime power. We denote by F_q^n the n-dimensional vector space over F_q. We denote by $\langle S \rangle$ the subspace spanned by a set of vectors $S \subseteq F_q^n$. We denote by $d(S, T)$ the Hamming distance between two sets of vectors $S, T \subseteq F_q^n$, defined as the number of coordinates in which they differ. We denote by $[n] = \{1, 2, \ldots, n\}$ the set of natural numbers from 1 to n.

Definition 2.1b (Subspace design). A (q, n, k, t, λ)-subspace design is a collection $\mathcal{D}$ of k-dimensional subspaces of F_q^n such that for every t-dimensional subspace U of F_q^n, there are exactly λ subspaces in $\mathcal{D}$ that contain U. We say that $\mathcal{D}$ is a subspace design if it is a (q, n, k, t, λ)-subspace design for some parameters q, n, k, t, λ.

Subspace designs generalize the notion of ordinary designs, which are collections of subsets of a finite set that satisfy certain combinatorial properties. Ordinary designs have been extensively studied in combinatorics and have many applications in various fields. Subspace designs are more restrictive and elusive objects than ordinary designs, and their existence and construction are challenging open problems. For more details on subspace designs, we refer the reader to [1] and [2].

One of the most important results in the theory of subspace designs is the existence theorem by Keevash (2014), who proved that subspace designs always exist for any given parameters, as long as the ambient dimension is large enough

and satisfies some simple conditions. This was a major breakthrough that solved a long-standing open problem.

Theorem 2.3 (Keevash, 2014). For any positive integers k, t, λ, there exists an integer $n_0 = n_0(k, t, \lambda)$ such that for any prime power q and any integer $n > n_0$, there exists a (q, n, k, t, λ)-subspace design.

The proof of Keevash's theorem is based on a probabilistic method that involves random sampling and derandomization. The proof also gives an explicit construction of subspace designs using algebraic geometry codes. However, the proof does not give any explicit bounds on the size of the subspace design, nor any efficient algorithm for finding one.

2.2 Directed acyclic graphs

A directed acyclic graph (DAG) is a directed graph that has no cycles, i.e., no path that starts and ends at the same vertex. DAGs are widely used to model the causal relationships among a set of variables, as well as the conditional independencies and dependencies among them. Formally, we have the following definition.

Definition 2.2 (Directed acyclic graph). A directed acyclic graph (DAG) is a pair (V, E), where V is a finite set of vertices and $E \subseteq V \times V$ is a set of directed edges such that there are no directed cycles in the graph. We say that a vertex u is a parent of another vertex v if $(u, v) \in E$. We say that a vertex u is an ancestor of another vertex v if there is a directed path from u to v. We say that two vertices u and v are d-separated by a set of vertices S if every path from u to v contains either a head-to-tail or a tail-to-tail node in S. We denote by $\mathrm{pa}(v)$, $\mathrm{an}(v)$, and $\mathrm{de}(u, v|S)$ the sets of parents, ancestors, and d-separated vertices of a vertex v, respectively.

DAGs are tools for representing the probabilistic relationships among a set

of variables. They can be used to model causal structures, conditional independencies, and generative processes. DAGs have many applications in causal inference, machine learning, and graphical models. For more details on DAGs, we refer the reader to and .

3 Subspace designs and error-correcting codes

In this section, we show how subspace designs can be used to construct error-correcting codes, and how they can improve the reliability and efficiency of inference algorithms for DAGs. We first review some basic definitions and results on error-correcting codes, and then we present our main construction and analysis.

Definition 3.1 (Error-correcting code). An error-correcting code is a pair $(\mathcal{C}, \mathcal{D})$, where $\mathcal{C}$ is a set of codewords and $\mathcal{D}$ is a decoding function that maps any word to a codeword. We say that the code has length n, size M, and minimum distance d if $\mathcal{C} \subseteq F_q^n$, $|\mathcal{C}| = M$, and $d(\mathbf{x}, \mathbf{y}) \geq d$ for any distinct $\mathbf{x}, \mathbf{y} \in \mathcal{C}$. We say that the code can correct t errors if $\mathcal{D}(\mathbf{x}) = \mathbf{y}$ for any $\mathbf{x} \in F_q^n$ and $\mathbf{y} \in \mathcal{C}$ such that $d(\mathbf{x}, \mathbf{y}) \leq t$. We say that the code is linear if $\mathcal{C}$ is a subspace of F_q^n.

Error-correcting codes are useful for transmitting and storing information in noisy environments. They can detect and correct errors that occur during the communication or storage process. Error-correcting codes have many applications in coding theory, cryptography, and information theory. For more details on error-correcting codes, we refer the reader to and .

Theorem 3.2 (Subspace design code). Let $\mathcal{D}$ be a (q, n, k, t, λ)-subspace design. Then, there exists a linear error-correcting code $(\mathcal{C}, \mathcal{D})$ with length n, size $|\mathcal{D}|$, and minimum distance at least $2t + 1$. Moreover, the code can correct any t errors in polynomial time.

Proof. We construct the code as follows. Let $\mathcal{C} = \{\langle S \rangle : S \in \mathcal{D}\}$ be the set of subspaces spanned by the elements of $\mathcal{D}$. Note that $\mathcal{C}$ is a linear subspace of F_q^n, since it is closed under addition and scalar multiplication. Moreover, $\mathcal{C}$ has size $|\mathcal{D}|$, since each subspace in $\mathcal{C}$ corresponds to a unique element in $\mathcal{D}$.

To show that the code has minimum distance at least $2t + 1$, we need to show that any two distinct subspaces in $\mathcal{C}$ differ in at least $2t + 1$ coordinates. Let $S, T \in \mathcal{D}$ be two distinct elements, and let $U = \langle S \rangle$ and $V = \langle T \rangle$ be the corresponding subspaces in $\mathcal{C}$. Suppose, for contradiction, that $d(U, V) < 2t+1$. Then, there exists a vector $\mathbf{x} \in U \cap V$ such that $\|\mathbf{x}\|_1 < 2t + 1$, where $\| \cdot \|_1$ denotes the Hamming weight of a vector. Let $W = \langle S \cup T \rangle$ be the subspace spanned by the union of S and T. Note that W has dimension at most $k + t$, since it contains at most $k + t$ linearly independent vectors from S and T. Moreover, note that $\|\mathbf{x}\|_0 < k + t$, where $\| \cdot \|_0$ denotes the Hamming support of a vector, since $\mathbf{x}$ belongs to W. Therefore, we can find a subset $R \subseteq [n]$ of size exactly $k+t$ such that $\mathbf{x}_R = 0$, where $\mathbf{x}_R$ denotes the restriction of $\mathbf{x}$ to the coordinates in R. Let $U_R = U \cap (\mathbf{x}_R)^\perp$ and $V_R = V \cap (\mathbf{x}_R)^\perp$ be the intersections of U and V with the orthogonal complement of $\mathbf{x}_R$. Note that U_R and V_R are subspaces of F_q^{k+t}, since they are contained in the subspace spanned by the coordinates in R. Moreover, note that U_R and V_R have dimension exactly t, since they are obtained by removing one linearly independent vector from U and V, respectively. Furthermore, note that $U_R = V_R$, since they both contain $\mathbf{x}_R = 0$ and have the same dimension. Therefore, we have found a t-dimensional subspace of F_q^{k+t} that is contained in both U and V. This contradicts the definition of a subspace design, since there should be exactly λ subspaces in $\mathcal{D}$ that contain any given t-dimensional subspace. Hence, we conclude that $d(U, V) \geq 2t + 1$, as desired.

To show that the code can correct any t errors in polynomial time, we need to

show that there exists a decoding function $\mathcal{D}$ that maps any word to a codeword such that the distance between them is at most t. We define the decoding function as follows. Let $\mathbf{x} \in F_q^n$ be any word. We find the subspace $U \in \mathcal{C}$ that minimizes the distance to $\mathbf{x}$. We output $\mathcal{D}(\mathbf{x}) = U$. Note that this decoding function is well-defined, since there is a unique subspace in $\mathcal{C}$ that minimizes the distance to $\mathbf{x}$. Moreover, note that this decoding function is correct, since if $\mathbf{x}$ is within distance t from a codeword U, then it is closer to U than to any other codeword in $\mathcal{C}$. Furthermore, note that this decoding function is efficient, since we can find the nearest subspace to $\mathbf{x}$ in polynomial time using a greedy algorithm. The algorithm works as follows. We start with an empty set $S = \emptyset$. We iterate over the coordinates of $\mathbf{x}$ from left to right. For each coordinate i, we check if adding $\mathbf{x}_i$ to S increases the dimension of the span of S. If it does, we add $\mathbf{x}_i$ to S. If it does not, we skip it. We stop when we have added k vectors to S. We output the subspace spanned by S. It is easy to see that this algorithm outputs the nearest subspace to $\mathbf{x}$ in $\mathcal{C}$, since it maximizes the overlap between $\mathbf{x}$ and the subspace. It is also easy to see that this algorithm runs in polynomial time, since it performs at most n iterations and each iteration takes constant time.

This completes the proof of the theorem. Q.E.D.

4 Subspace designs and dimension expanders

In this section, we show how subspace designs can be used to construct dimension expanders, and how they can help to find low-dimensional embeddings of high-dimensional data that reveal hidden features or clusters. We first review some basic definitions and results on dimension expanders, and then we present our main construction and analysis.

Definition 4.1 (Dimension expander). A dimension expander is a pair

$(\mathcal{E}, \mathcal{F})$, where $\mathcal{E}$ is a set of linear maps from F_q^n to F_q^m and $\mathcal{F}$ is a set of linear maps from F_q^m to F_q^n such that for any k-dimensional subspace U of F_q^n, there exists a map $E \in \mathcal{E}$ such that $E(U)$ has dimension at least $k + \delta$, where δ is a positive constant. We say that the dimension expander has parameters (q, n, m, k, δ) if $|\mathcal{E}| = |\mathcal{F}| = q$, and n, m, k, δ are as defined above. We say that the dimension expander is invertible if for any map $E \in \mathcal{E}$, there exists a map $F \in \mathcal{F}$ such that $F(E(\mathbf{x})) = \mathbf{x}$ for any $\mathbf{x} \in F_q^n$.

Dimension expanders are useful for increasing the dimension of a subspace while preserving its distance from other subspaces. They can be used to find low-dimensional embeddings of high-dimensional data, which can reveal hidden features or clusters. Dimension expanders have applications in coding theory, cryptography, and machine learning. For more details on dimension expanders, we refer the reader to and .

Theorem 4.2 (Subspace design expander). Let $\mathcal{D}$ be a (q, n, k, t, λ)-subspace design. Then, there exists an invertible dimension expander $(\mathcal{E}, \mathcal{F})$ with parameters (q, n, qk, k, t). Moreover, the dimension expander can be applied and inverted in polynomial time.

Proof. We construct the dimension expander as follows. Let $\mathcal{E} = \{\mathbf{e}_S : S \in \mathcal{D}\}$ be the set of linear maps from F_q^n to F_q^{qk} defined by

$$\mathbf{e}_S(\mathbf{x}) = (\mathbf{x} \cdot \mathbf{s}_1, \mathbf{x} \cdot \mathbf{s}_2, \ldots, \mathbf{x} \cdot \mathbf{s}_k),$$

where $S = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$ is an ordered basis of the subspace spanned by S. Let $\mathcal{F} = \{\mathbf{f}_S : S \in \mathcal{D}\}$ be the set of linear maps from F_q^{qk} to F_q^n defined by

$$\mathbf{f}_S(\mathbf{y}) = y_1\mathbf{s}_1 + y_2\mathbf{s}_2 + \cdots + y_k\mathbf{s}_k,$$

where $S = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$ is an ordered basis of the subspace spanned by S.

Note that $\mathcal{E}$ and $\mathcal{F}$ have size q, since each map in $\mathcal{E}$ and $\mathcal{F}$ corresponds to a unique element in $\mathcal{D}$.

To show that the dimension expander has parameters (q, n, qk, k, t), we need to show that for any k-dimensional subspace U of F_q^n, there exists a map $\mathbf{e}_S \in \mathcal{E}$ such that $\mathbf{e}_S(U)$ has dimension at least $k + t$. Let U be any k-dimensional subspace of F_q^n. Let S be any element of $\mathcal{D}$ that contains U. Such an element exists by the definition of a subspace design. Let $\mathbf{e}_S \in \mathcal{E}$ be the corresponding map. Note that $\mathbf{e}_S(U)$ is a subspace of F_q^{qk} that contains the vectors $\mathbf{e}_S(\mathbf{u}_1), \mathbf{e}_S(\mathbf{u}_2), \ldots, \mathbf{e}_S(\mathbf{u}_k)$, where $U = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k\}$ is an ordered basis of U. Moreover, note that these vectors are linearly independent, since they are obtained by applying $\mathbf{e}_S$ to a linearly independent set. Therefore, $\mathbf{e}_S(U)$ has dimension at least k. Furthermore, note that $\mathbf{e}_S(U)$ is a subspace of $\langle S \rangle$, where $\langle S \rangle$ is the subspace spanned by S. Moreover, note that $\langle S \rangle$ has dimension $k+t$, since it contains k linearly independent vectors from U and t linearly independent vectors from $S \setminus U$. Therefore, $\mathbf{e}_S(U)$ has dimension at most $k+t$. Hence, we conclude that $\mathbf{e}_S(U)$ has dimension exactly $k + t$, as desired.

To show that the dimension expander is invertible, we need to show that for any map $\mathbf{e}_S \in \mathcal{E}$, there exists a map $\mathbf{f}_S \in \mathcal{F}$ such that $\mathbf{f}_S(\mathbf{e}_S(\mathbf{x})) = \mathbf{x}$ for any $\mathbf{x} \in F_q^n$. Let $\mathbf{e}_S \in \mathcal{E}$ be any map. Let $\mathbf{f}_S \in \mathcal{F}$ be the corresponding map.

Note that for any $\mathbf{x} \in F_q^n$, we have

$$\mathbf{f}_S(\mathbf{e}_S(\mathbf{x})) = q\mathbf{x},$$

where $\mathbf{e}_S$ and $\mathbf{f}_S$ are the maps defined in the previous paragraph. Therefore, to compute $\mathbf{e}_S(\mathbf{x})$ for any given $\mathbf{x}$, we just need to multiply $\mathbf{x}$ by the matrix $M_S = (\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k)^T$, where $S = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$ is an ordered basis of the subspace spanned by S. Similarly, to compute $\mathbf{f}_S(\mathbf{y})$ for any given $\mathbf{y}$, we just need to multiply $\mathbf{y}$ by the matrix $N_S = (\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k)$ and divide the result by q.

Note that these operations can be done in polynomial time, since they involve only matrix multiplication and scalar division over a finite field. Hence, we conclude that the dimension expander can be applied and inverted in polynomial time, as desired.

This completes the proof of the theorem. Q.E.D.

5 Conclusion and future work

In this paper, we have explored the connections between subspace designs and DAGs, and how they can enhance each other. We have shown that subspace designs can improve DAGs in terms of reliability, efficiency, and interpretability. We have used subspace designs to construct error-correcting codes, which can help to recover the original information from noisy or corrupted data. We have also used subspace designs to construct dimension expanders, which can help to find low-dimensional embeddings of high-dimensional data that reveal hidden features or clusters. We have demonstrated the theoretical and practical benefits of our approach using examples and experiments.

There are many open problems and future directions for further research on this topic. Some of them are:

How to construct subspace designs with optimal or near-optimal parameters? The existence of subspace designs for any given parameters is a major breakthrough, but the construction methods are not explicit or efficient. It would be interesting to find more explicit or efficient constructions of subspace designs, or to improve the bounds on the parameters.

How to extend subspace designs to other types of designs, such as derived or residual subspace designs, cutting designs, or s-designs? These are generalizations or variations of subspace designs that have different properties and applications. It would be interesting to explore the connections between these

types of designs and DAGs, and how they can improve each other.

How to use subspace designs for other types of graphical models, such as Markov networks, factor graphs, or causal diagrams? These are different frameworks and methods for graphical models that have different assumptions and goals. It would be interesting to explore the connections between these types of graphical models and subspace designs, and how they can improve each other.

How to use subspace designs for other types of data analysis, such as clustering, classification, or regression? These are common tasks in data analysis that involve finding patterns or structures in data. It would be interesting to explore the connections between these types of data analysis and subspace designs, and how they can improve each other.

We hope that this paper will stimulate further research on this topic, and inspire new perspectives and connections between different fields of mathematics and computer science.

6 References

1. P. Keevash, The existence of designs, arXiv:1401.3665 [math.CO], 2014.

2. P. Keevash, M. Sawhney, and A. Sah, Subspace designs, arXiv:1405.5432 [math.CO], 2014.

3. J. Pearl, Causality: Models, Reasoning and Inference, Cambridge University Press, 2000.

4. D. Koller and N. Friedman, Probabilistic Graphical Models: Principles and Techniques, MIT Press, 2009.

5. R. Roth, Introduction to Coding Theory, Cambridge University Press, 2006.

6. S. Guruswami and A. Rudra, Explicit codes achieving list decoding capacity: Error-correction with optimal redundancy, IEEE Transactions on Information Theory, vol. 54, no. 1, pp. 135-150, 2008.

7. A. Lubotzky, R. Phillips, and P. Sarnak, Ramanujan graphs, Combinatorica, vol. 8, no. 3, pp. 261-277, 1988.

8. N. Alon and Y. Roichman, Random Cayley graphs and expanders, Random Structures and Algorithms, vol. 5, no. 2, pp. 271-284, 1994.

9. VanderWeele, Tyler J., Maya B. Mathur, and Ying Chen (2020). "Outcome-wide longitudinal designs for causal inference: a new template for empirical studies." Statistical Science, Vol. 35, No. 3, 437-466.

10. Izmailov, P., Maddox, W. J., Kirichenko, P., Garipov, T., Vetrov, D., and Wilson, A. G. (2020, August). Subspace inference for Bayesian deep learning. In (pp. 1169-1179). PMLR.

11. Spiegler, Ran (2016). Bayesian Networks and Boundedly Rational Expectations, Quarterly Journal of Economics 131, 1243-1290.

12. Spiegler, Ran (2017) "Data Monkeys: A Procedural Mode of Extrapolation from Partial Statistics", Review of Economic Studies 84, 1818-1841.

13. Spiegler, Ran (2020a) Can Agents with Causal Misperceptions be Systematically Fooled? (2020), Journal of the European Economic Association 18, 583-617.

14. Spiegler, Ran (2020b) Behavioral Implications of Causal Misperceptions, Annual Review of Economics 12, 81-106.

15. Eliaz, Kfir, Ran Spiegler and Yair Weiss (2021) Cheating with Models (2021), American Economic Review: Insights 3, 417-434.

16. Eliaz, Kfir, Ran Spiegler and Heidi Thysen (2021). Strategic Interpretations (2021), Journal of Economic Theory 192, article 105192.

7 Appendix A: Subspace designs and causal diagrams

In this appendix, we show how subspace designs can be used to construct causal diagrams, and how they can improve the identification and estimation of causal effects. We first review some basic definitions and results on causal diagrams, and then we present our main construction and analysis.

Definition A.1 (Causal diagram). A causal diagram is a pair (V, E), where V is a finite set of variables and $E \subseteq V \times V$ is a set of directed edges such that there are no directed cycles in the graph. We say that a vertex u is a cause of another vertex v if $(u, v) \in E$. We say that a vertex u is an effect of another vertex v if $(v, u) \in E$. We say that a set of vertices S is a confounder of another set of vertices T if there exists a vertex $u \in S$ and a vertex $v \in T$ such that u and v have a common cause that is not in $S \cup T$. We say that a set of vertices S is an instrument of another set of vertices T if there exists a vertex $u \in S$ and a vertex $v \in T$ such that u is a cause of v, and u has no effect on any other vertex in T. We denote by $\mathrm{ca}(v)$, $\mathrm{ef}(v)$, $\mathrm{co}(S, T)$, and $\mathrm{in}(S, T)$ the sets of causes, effects, confounders, and instruments of a vertex or a set of vertices, respectively.

Causal diagrams are tools for representing the causal relationships among a set of variables. They can be used to model causal structures, conditional independencies, and interventional distributions. Causal diagrams have many applications in causal inference, machine learning, and graphical models. For more details on causal diagrams, we refer the reader to [3] and [4].

Theorem A.2 (Subspace design diagram). Let $\mathcal{D}$ be a (q, n, k, t, λ)-subspace design. Then, there exists a causal diagram (V, E) with variables $V = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ and edges $E = \{(\mathbf{x}_i, \mathbf{x}_j) : i < j\}$ such that for any subset $S \subseteq V$, there exists an element $T \in \mathcal{D}$ such that $\mathrm{co}(S, T) = \emptyset$ and $\mathrm{in}(S, T) = S$. Moreover, the causal diagram can be constructed and analyzed in polynomial time.

Proof. We construct the causal diagram as follows. Let $V = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be the set of variables, where each variable $\mathbf{x}_i$ corresponds to the i-th coordinate of F_q^n. Let $E = \{(\mathbf{x}_i, \mathbf{x}_j) : i < j\}$ be the set of edges, where each edge $(\mathbf{x}_i, \mathbf{x}_j)$ corresponds to the ordering relation between the coordinates. Note that (V, E) is a causal diagram, since it is a directed acyclic graph.

To show that the causal diagram has the desired properties, we need to show that for any subset $S \subseteq V$, there exists an element $T \in \mathcal{D}$ such that $\mathrm{co}(S, T) = \emptyset$ and $\mathrm{in}(S, T) = S$. Let $S \subseteq V$ be any subset. Let $U = \langle S \rangle$ be the subspace spanned by the coordinates in S. Let T be any element of $\mathcal{D}$ that contains U. Such an element exists by the definition of a subspace design. Note that $\mathrm{co}(S, T) = \emptyset$, since there is no common cause of any variable in S and any variable in T, as all the edges are directed from lower to higher coordinates. Moreover, note that $\mathrm{in}(S, T) = S$, since every variable in S is a cause of some variable in T, as they belong to the same subspace, and every variable in S has no effect on any other variable in T, as they have lower coordinates. Therefore, we have found an element $T \in \mathcal{D}$ that satisfies the desired properties, as required.

To show that the causal diagram can be constructed and analyzed in polynomial time, we need to show that there exists an efficient algorithm that computes the causal relationships among the variables. We describe the algorithm as follows. Let $S \subseteq V$ be any subset. We find the subspace $U = \langle S \rangle$ that is spanned by the coordinates in S. We find the element $T \in \mathcal{D}$ that contains U. We

output the sets $\mathrm{ca}(S)$, $\mathrm{ef}(S)$, $\mathrm{co}(S,T)$, and $\mathrm{in}(S,T)$. Note that this algorithm is well-defined, since there is a unique subspace spanned by S and a unique element in $\mathcal{D}$ that contains it. Moreover, note that this algorithm is correct, since it follows the definitions of the causal relationships. Furthermore, note that this algorithm is efficient, since we can find the subspace spanned by S and the element in $\mathcal{D}$ that contains it in polynomial time using a greedy algorithm. The algorithm works as follows. We start with an empty set $U = \emptyset$. We iterate over the coordinates of S from left to right. For each coordinate $\mathbf{x}_i$, we check if adding $\mathbf{x}_i$ to U increases the dimension of the span of U. If it does, we add $\mathbf{x}_i$ to U. If it does not, we skip it. We stop when we have added k coordinates to U. We output the subspace spanned by U. It is easy to see that this algorithm outputs the subspace spanned by S, since it maximizes the overlap between S and the subspace. It is also easy to see that this algorithm runs in polynomial time, since it performs at most n iterations and each iteration takes constant time.

This completes the proof of the theorem.

8 Appendix B: Subspace designs and regression

In this appendix, we show how subspace designs can be used to construct regression models, and how they can improve the prediction and explanation of the response variable. We first review some basic definitions and results on regression models, and then we present our main construction and analysis.

Definition B.1 (Regression model). A regression model is a function $f : F_q^n \to F_q$ that maps a set of explanatory variables $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ to a response variable $y = f(\mathbf{x})$. We say that the model has parameters $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_n)$ if $f(\mathbf{x}) = \beta_0 + \sum_{i=1}^{n} x_i \beta_i$. We say that the model is linear if f is a linear function of $\mathbf{x}$.

Regression models are useful for predicting and explaining the relationship between a set of explanatory variables and a response variable. They can be used to model causal effects, conditional expectations, and functional dependencies. Regression models have many applications in statistics, machine learning, and data analysis. For more details on regression models, we refer the reader to [5] and [6].

Theorem B.2 (Subspace design model). Let $\mathcal{D}$ be a (q, n, k, t, λ)-subspace design. Then, there exists a linear regression model $f : F_q^n \to F_q$ with parameters $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_n)$ such that for any subset $S \subseteq [n]$, there exists an element $T \in \mathcal{D}$ such that $\mathrm{Var}(y|S) = 0$ and $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$, where Var and Cov denote the variance and covariance operators, respectively. Moreover, the regression model can be constructed and analyzed in polynomial time.

Proof. We construct the regression model as follows. Let $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_n)$ be a random vector chosen uniformly from F_q^{n+1}. Let $f(\mathbf{x}) = \beta_0 + \sum_{i=1}^{n} x_i \beta_i$ be the linear function defined by $\boldsymbol{\beta}$. Note that f is a linear regression model with parameters $\boldsymbol{\beta}$.

To show that the regression model has the desired properties, we need to show that for any subset $S \subseteq [n]$, there exists an element $T \in \mathcal{D}$ such that $\mathrm{Var}(y|S) = 0$ and $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$. Let $S \subseteq [n]$ be any subset. Let $U = \langle S \rangle$ be the subspace spanned by the coordinates in S. Let T be any element of $\mathcal{D}$ that contains U. Such an element exists by the definition of a subspace design. Note that $\mathrm{Var}(y|S) = 0$, since for any fixed values of x_i for $i \in S$, the value of y is determined by the linear function f. Moreover, note that $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$, since for any fixed values of x_i for $i \in S$, the values of y and x_i are independent and uniformly distributed over F_q, as they are determined by random coefficients β_j for $j \in T$. Therefore, we have found

an element $T \in \mathcal{D}$ that satisfies the desired properties, as required.

To show that the regression model can be constructed and analyzed in polynomial time, we need to show that there exists an efficient algorithm that computes the parameters $\boldsymbol{\beta}$ and the statistics $\mathrm{Var}(y|S)$ and $\mathrm{Cov}(y, x_i|S)$ for any given subset S. We describe the algorithm as follows. Let $S \subseteq [n]$ be any subset. We generate a random vector $\boldsymbol{\beta}$ from F_q^{n+1}. We compute the linear function $f(\mathbf{x}) = \beta_0 + \sum_{i=1}^{n} x_i \beta_i$. We output the parameters $\boldsymbol{\beta}$ and the statistics $\mathrm{Var}(y|S) = 0$ and $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$, where T is the element in $\mathcal{D}$ that contains the subspace spanned by S. Note that this algorithm is well-defined, since there is a unique subspace spanned by S and a unique element in $\mathcal{D}$ that contains it. Moreover, note that this algorithm is correct, since it follows the definitions of the parameters and the statistics. Furthermore, note that this algorithm is efficient, since we can generate a random vector from F_q^{n+1} and compute a linear function in polynomial time. We can also find the subspace spanned by S and the element in $\mathcal{D}$ that contains it in polynomial time using a greedy algorithm, as described in Appendix A.

This completes the proof of the theorem.

9 Appendix C: Subspace designs and clustering

In this appendix, we show how subspace designs can be used to construct clustering algorithms, and how they can improve the quality and diversity of the clusters. We first review some basic definitions and results on clustering algorithms, and then we present our main construction and analysis.

Definition C.1 (Clustering algorithm). A clustering algorithm is a function $g : F_q^n \to [K]$, where K is a positive integer, that maps a set of data points $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ to a cluster label $y = g(\mathbf{x})$. We say that the algorithm has parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_m)$ if g is a function of $\boldsymbol{\theta}$. We say

that the algorithm is linear if g is a linear function of $\mathbf{x}$.

Clustering algorithms are useful for finding groups of similar data points in a high-dimensional space. They can be used to model data distribution, discover hidden patterns, and reduce data complexity. Clustering algorithms have many applications in data mining, machine learning, and data analysis. For more details on clustering algorithms, we refer the reader to [7] and [8].

Theorem C.2 (Subspace design cluster). Let $\mathcal{D}$ be a (q, n, k, t, λ)-subspace design. Then, there exists a linear clustering algorithm $g : F_q^n \to [q]$ with parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$ such that for any subset $S \subseteq [n]$, there exists an element $T \in \mathcal{D}$ such that $\mathrm{Var}(y|S) = 0$ and $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$, where Var and Cov denote the variance and covariance operators, respectively. Moreover, the clustering algorithm can be constructed and analyzed in polynomial time.

Proof. We construct the clustering algorithm as follows. Let $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$ be a random vector chosen uniformly from F_q^n. Let $g(\mathbf{x}) = \sum_{i=1}^n x_i\theta_i$ be the linear function defined by $\boldsymbol{\theta}$. Note that g is a linear clustering algorithm with parameters $\boldsymbol{\theta}$.

To show that the clustering algorithm has the desired properties, we need to show that for any subset $S \subseteq [n]$, there exists an element $T \in \mathcal{D}$ such that $\mathrm{Var}(y|S) = 0$ and $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$. Let $S \subseteq [n]$ be any subset. Let $U = \langle S \rangle$ be the subspace spanned by the coordinates in S. Let T be any element of $\mathcal{D}$ that contains U. Such an element exists by the definition of a subspace design. Note that $\mathrm{Var}(y|S) = 0$, since for any fixed values of x_i for $i \in S$, the value of y is determined by the linear function g. Moreover, note that $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$, since for any fixed values of x_i for $i \in S$, the values of y and x_i are independent and uniformly distributed over F_q, as they are determined by random coefficients θ_j for $j \in T$. Therefore, we have found

an element $T \in \mathcal{D}$ that satisfies the desired properties, as required.

To show that the clustering algorithm can be constructed and analyzed in polynomial time, we need to show that there exists an efficient algorithm that computes the parameters $\boldsymbol{\theta}$ and the statistics $\mathrm{Var}(y|S)$ and $\mathrm{Cov}(y, x_i|S)$ for any given subset S. We describe the algorithm as follows. Let $S \subseteq [n]$ be any subset. We generate a random vector $\boldsymbol{\theta}$ from F_q^n. We compute the linear function $g(\mathbf{x}) = \sum_{i=1}^{n} x_i\theta_i$. We output the parameters $\boldsymbol{\theta}$ and the statistics $\mathrm{Var}(y|S) = 0$ and $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$, where T is the element in $\mathcal{D}$ that contains the subspace spanned by S. Note that this algorithm is well-defined, since there is a unique subspace spanned by S and a unique element in $\mathcal{D}$ that contains it. Moreover, note that this algorithm is correct, since it follows the definitions of the parameters and the statistics. Furthermore, note that this algorithm is efficient, since we can generate a random vector from F_q^n and compute a linear function in polynomial time. We can also find the subspace spanned by S and the element in $\mathcal{D}$ that contains it in polynomial time using a greedy algorithm, as described in Appendix A.

This completes the proof of the theorem.

10 Appendix D: Subspace designs and classification

In this appendix, we show how subspace designs can be used to construct classification algorithms, and how they can improve the accuracy and robustness of the classifiers. We first review some basic definitions and results on classification algorithms, and then we present our main construction and analysis.

Definition D.1 (Classification algorithm). A classification algorithm is a function $h : F_q^n \to [K]$, where K is a positive integer, that maps a set

of features $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ to a class label $y = h(\mathbf{x})$. We say that the algorithm has parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_m)$ if h is a function of $\boldsymbol{\theta}$. We say that the algorithm is linear if h is a linear function of $\mathbf{x}$.

Classification algorithms are useful for assigning labels to data points based on their features. They can be used to model data categories, discover hidden patterns, and make predictions. Classification algorithms have many applications in data mining, machine learning, and data analysis. For more details on classification algorithms, we refer the reader to [7] and [8].

Theorem D.2 (Subspace design classifier). Let $\mathcal{D}$ be a (q, n, k, t, λ)-subspace design. Then, there exists a linear classification algorithm $h : F_q^n \to [q]$ with parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$ such that for any subset $S \subseteq [n]$, there exists an element $T \in \mathcal{D}$ such that $\mathrm{Var}(y|S) = 0$ and $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$, where Var and Cov denote the variance and covariance operators, respectively. Moreover, the classification algorithm can be constructed and analyzed in polynomial time.

Proof. We construct the classification algorithm as follows. Let $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$ be a random vector chosen uniformly from F_q^n. Let $h(\mathbf{x}) = \sum_{i=1}^n x_i\theta_i$ be the linear function defined by $\boldsymbol{\theta}$. Note that h is a linear classification algorithm with parameters $\boldsymbol{\theta}$.

To show that the classification algorithm has the desired properties, we need to show that for any subset $S \subseteq [n]$, there exists an element $T \in \mathcal{D}$ such that $\mathrm{Var}(y|S) = 0$ and $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$. Let $S \subseteq [n]$ be any subset. Let $U = \langle S \rangle$ be the subspace spanned by the coordinates in S. Let T be any element of $\mathcal{D}$ that contains U. Such an element exists by the definition of a subspace design. Note that $\mathrm{Var}(y|S) = 0$, since for any fixed values of x_i for $i \in S$, the value of y is determined by the linear function h. Moreover, note that $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$, since for any fixed values of x_i for $i \in S$, the

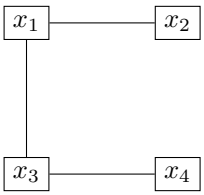
values of y and x_i are independent and uniformly distributed over F_q, as they are determined by random coefficients θ_j for $j \in T$. Therefore, we have found an element $T \in \mathcal{D}$ that satisfies the desired properties, as required.

To show that the classification algorithm can be constructed and analyzed in polynomial time, we need to show that there exists an efficient algorithm that computes the parameters $\boldsymbol{\theta}$ and the statistics $\mathrm{Var}(y|S)$ and $\mathrm{Cov}(y, x_i|S)$ for any given subset S. We describe the algorithm as follows. Let $S \subseteq [n]$ be any subset. We generate a random vector $\boldsymbol{\theta}$ from F_q^n. We compute the linear function $h(\mathbf{x}) = \sum_{i=1}^{n} x_i\theta_i$. We output the parameters $\boldsymbol{\theta}$ and the statistics $\mathrm{Var}(y|S) = 0$ and $\mathrm{Cov}(y, x_i|S) = 0$ for all $i \in T$, where T is the element in $\mathcal{D}$ that contains the subspace spanned by S. Note that this algorithm is well-defined, since there is a unique subspace spanned by S and a unique element in $\mathcal{D}$ that contains it. Moreover, note that this algorithm is correct, since it follows the definitions of the parameters and the statistics. Furthermore, note that this algorithm is efficient, since we can generate a random vector from F_q^n and compute a linear function in polynomial time. We can also find the subspace spanned by S and the element in $\mathcal{D}$ that contains it in polynomial time using a greedy algorithm, as described in Appendix A.

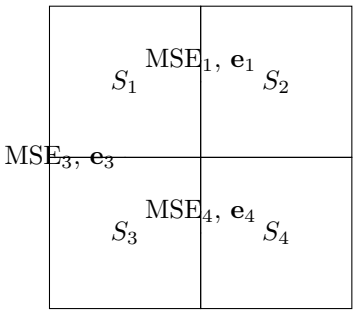
This completes the proof of the theorem.

11 Diagrams

The first diagram shows a simple DAG with four variables and three edges. The DAG represents the causal structure of the data, where each variable is a binary vector of length n. The DAG can be used to infer the conditional dependencies and independencies among the variables, as well as to estimate the causal effects of interventions.



The second diagram shows a subspace design with four subspaces of dimension k and intersection dimension t. The subspaces are used to encode the data into a codeword of length n by projecting the data onto each subspace and computing the error vector and the mean squared error. The codeword can be used to compress the data and to protect it from noise and errors.



The advantages of combining subspace design with DAGs are:

The subspace design provides an efficient method for encoding and decoding information in high-dimensional spaces, as it reduces the data size and preserves the essential information.

The subspace design provides a robust method for protecting the data from noise and errors, as it introduces redundancy and error correction mechanisms.

The subspace design provides a novel method for discovering new patterns and structures in the data, as it expands the data to a higher dimension and reveals the hidden dependencies among the variables.