2024 IEEE 63rd Conference on Decision and Control (CDC)
December 16-19, 2024. MiCo, Milan, Italy

Gram-Schmidt Methods for Unsupervised Feature Selection

Bahram Yaghooti, *Student Member, IEEE*, Netanel Raviv, *Senior Member, IEEE*, Bruno Sinopoli, *Fellow, IEEE*

***Abstract*— Unsupervised feature selection is a critical task in data analysis, particularly when faced with high-dimensional datasets and complex and nonlinear dependencies among features. In this paper, we propose a family of Gram-Schmidt feature selection approaches to unsupervised feature selection that addresses the challenge of identifying non-redundant features in the presence of nonlinear dependencies. Our method leverages probabilistic Gram-Schmidt (GS) orthogonalization to detect and map out redundant features within the data. By applying the GS process to capture nonlinear dependencies through a pre-defined, fixed family of functions, we construct variance vectors that facilitate the identification of high-variance features, or the removal of these dependencies from the feature space. In the first case, we provide information-theoretic guarantees in terms of entropy reduction. In the second case, we demonstrate the efficacy of our approach by proving theoretical guarantees under certain assumptions, showcasing its ability to detect and remove nonlinear dependencies. To support our theoretical findings, we experiment over various synthetic and real-world datasets, showing superior performance in terms of classification accuracy over state-of-the-art methods. Further, our information-theoretic feature selection algorithm strictly generalizes a recently proposed Fourier-based feature selection mechanism at significantly reduced complexity.**

I. Introduction

Feature selection, which is one of the main challenges in machine learning and reduced-order modeling, is a set of techniques used to reduce the number of features (or dimensions) in a dataset, while retaining most of the important information. The goal of feature selection is to simplify the data and make it easier to analyze or visualize, while reducing the computational complexity and memory requirements. The main approaches to feature selection involves selecting a subset of the original features based on some criteria, such as correlation analysis or mutual information, so that the selected features capture the most important information [1].

We focus on unsupervised feature selection, in which no particular future use of the data is known at the time of selecting the features (i.e., data with no labels). Feature selection can be divided into three main approaches [2]: (i) *Filter methods*, which select a subset of features as a pre-processing step without involving a specific machine learning model [3]; (ii) *Wrapper methods*, which involve the use of a specific machine learning algorithm to evaluate the relevance of feature subsets [4]; and (iii) *Embedded methods*, which integrate feature selection with the model training phase, effectively selecting the most relevant features during model optimization [5].

Information theory also plays a role in feature selection [6]. One common approach to feature selection using information theory is to use mutual information, which measures the amount of information that two variables share. To use mutual information for feature selection, one would maximize the mutual information between the data and the selected features, or equivalently, seek selected features such that the conditional entropy of the data given those features is minimized [7].

Many unsupervised feature selection algorithms exist. For instance, Laplacian Score [8], [9], Fisher Score [10], [11], and Trace Ratio [12] are similarity based techniques, which assess the importance of features by their ability to preserve data similarity; Multi-Cluster Feature Selection [13], [14], Nonnegative Discriminative Feature Selection [15], and Unsupervised Discriminative Feature Selection [16], [17] are sparse-learning based methods, which assume sparsity and employ lasso type minimization; and minimal-redundancy-maximal-relevance [18], Mutual Information based Feature Selection [19], and Fast Correlation Based Filter [20], are information theoretic techniques, which maximize mutual information between the original and selected features.

We aim at advancing the current feature selection literature by finding new principled methods for feature selection which capture *nonlinear* dependencies among the data. Specifically, given a dataset drawn i.i.d from a random variable $\mathbf{x} = (x_i)_{i=1}^d$ over $\mathbb{R}^d$, we wish to reduce it to another dataset by selecting few features (or columns), which capture as much information about $\mathbf{x}$ as possible, while mapping out nonlinear redundancies.

To this end, we propose a family of feature selection algorithms which is based on orthogonalization process of predefined family of redundancy functions. Specifically, our algorithms begin by fixing a finite family of linearly independent functions $\mathcal{F}$ in some unspecified (non-random) variables, which presumably captures the anticipated dependencies in $\mathbf{x}$. Then, in an iterative process, the variables of $\mathcal{F}$ are specified one-by-one as features of $\mathbf{x}$ (i.e, some x_i's), and the subset of $\mathcal{F}$ which depends on already-specified variables undergoes a Gram-Schmidt process over function spaces: specifying the variables of $\mathcal{F}$ as some x_i's turns the functions in $\mathcal{F}$ to random variables, on which inner-product between two functions can be defined in natural way as the expectation of their product, and then the Gram-Schmidt process can be applied.

In turn, the Gram-Schmidt process over $\mathcal{F}$ enables to

B. Yaghooti and B. Sinopoli are with the Department of Electrical and Systems Engineering, Washington University in St. Louis, St. Louis, MO, USA (Email: byaghooti@wustl.edu; bsinopoli@wustl.edu).

N. Raviv is with the Department of Computer Science and Engineering, Washington University in St. Louis, St. Louis, MO, USA (Email: netanel.raviv@wustl.edu).

979-8-3503-1632-2/24/$31.00 ©2024 IEEE

define new random variables d_j—one at each iteration j of the algorithm with $d_0 = \mathbf{x}$—by subtracting from $\mathbf{x}$ its projections on functions in $\mathcal{F}$ whose variables were already specified. These new random variables d_j, in a sense, capture whatever remains of $\mathbf{x}$ should one set to zero the part which can be described by functions discovered so far. Additionally, the d_j's give rise to *variance vectors* $\boldsymbol{\sigma}_j$ (a variance vector of a random variable is the diagonal of its covariance matrix) which provide insights into specifying the next variable of $\mathcal{F}$, naturally, as the one which maximizes some form of variance. This process is detailed in Section II.

This general framework is manifested in two complementing ways. First, we present our Gram-Schmidt Functional Selection (GFS, Section III) algorithm, which in the j^{th} step selects the most variant feature of the random variable d_j; this can be done by analyzing the elements of $\boldsymbol{\sigma}_j$. In this part of the paper we borrow ideas from the recently proposed Unsupervised Fourier Feature Selection (UFFS) due to [21] and show that the selected features reduce the conditional entropy of the data as a function of the dimension and a threshold parameter. Moreover, in the special case where $\mathcal{F}$ is chosen as the set of multilinear polynomials, our algorithm reduces to that of [21], yet at significantly lower complexity. Roughly speaking, the novelty which enables this complexity reduction is that the orthogonalization of $\mathcal{F}$ is performed variable-by-variable, rather than all at once as in [21], and stops once sufficient informativeness has been reached.

Second, to better understand the theoretical underpinnings of our first algorithm, we wish to characterize ideal settings in which similar ideas result in *perfect* feature selection, i.e., in zero conditional entropy, and present our Gram-Schmidt Feature Analysis (GFA, Section IV) algorithm. Specifically, such ideal settings occur in cases where the dependencies lie in the linear span of $\mathcal{F}$, and they are *variance reducing* in a way that will be defined formally later. In such settings, we prove that our algorithm correctly identifies and removes those dependencies. This algorithm also relies on analyzing the variance vectors $\boldsymbol{\sigma}_j$.

Finally, in section V, we apply the proposed algorithms to synthetic and real-world datasets and show their superior performance in comparison to state-of-the-art feature selection algorithms. Section VI concludes the paper.

II. Gram-Schmidt orthogonalization over Function Spaces

Throughout this paper, we make use of the well known Hadamard product, denoted by $\oplus$, where for two vectors $\mathbf{x} = (x_i)_{i=1}^n, \mathbf{y} = (y_i)_{i=1}^n$ we have $\mathbf{x} \oplus \mathbf{y} = (x_i y_i)_{i=1}^n$; similarly $\mathbf{x}^{\oplus 2} = (x_i^2)_{i=1}^n$.

In this section, we develop the Gram-Schmidt (GS) orthogonalization process over function spaces and its schematic description, upon which all our proposed algorithms are based. All algorithms are presented with respect to a generic finite family $\mathcal{F}$ of linearly independent functions chosen by the user. This $\mathcal{F}$ needs to be chosen judiciously using domain expertise; a discussion in this regard is given in the sequel, and many examples of $\mathcal{F}$'s which provide significant gains in real-world datasets are given in Section V.

Throughout the orthogonalization process we denote $\mathbf{x} = (x_1, \ldots, x_d)$, where each $x_i \in \mathbb{R}$ represents the unknown random variable from which the data is sampled. That is, each row of the data matrix $\mathbf{D}$ is sampled independently from $\mathbf{x}$. We begin by specifying a finite family of functions $\mathcal{F}$ in (non-random) variables $y_1, \ldots, y_n$, i.e., each $f \in \mathcal{F}$ is of the form $f(y_1, \ldots, y_n)$ and n is the number of selected features at the current iteration. We additionally assume that $f(y_1, \ldots, y_n) = y_i \in \mathcal{F}$ for all $i \in [n]$, i.e., $\mathcal{F}$ contains the singleton functions.

Then, the orthogonalization process creates a new set of *random* functions $\hat{\mathcal{F}}$ by

1) Taking linear combinations of the functions in $\mathcal{F}$.
2) Substituting the *non-random* variables $y_1, \ldots, y_n$ with *random* variables $z_1, \ldots, z_n$, each of which is either one of $\{x_i\}_{i=1}^d$. This makes each function in $\hat{\mathcal{F}}$ a random variable.
3) The functions in $\hat{\mathcal{F}}$ are orthonormal, i.e., $\mathbb{E}_{\mathbf{x}}[\hat{f}\hat{g}]$ is zero if $\hat{f} \neq \hat{g}$ and one otherwise, for every $\hat{f}, \hat{g} \in \hat{\mathcal{F}}$.

To clarify, every $\hat{f} \in \hat{\mathcal{F}}$ is of the form

$$\begin{aligned}\hat{f}(z_1, \ldots, z_n) &= \sum_{f \in \mathcal{F}} \alpha_{f,\hat{f}} f(z_1, \ldots, z_n) \\ &= \sum_{f \in \mathcal{F}} \alpha_{f,\hat{f}} f(x_{s_1}, \ldots, x_{s_n})\end{aligned}$$

for some real coefficients $\{\alpha_{f,\hat{f}}\}_{f \in \mathcal{F}}$, where $s_1, \ldots, s_n \in [d]$ (where $[d] \triangleq \{1, 2, \ldots, d\}$) are integers that denote the selected features; the coefficients $\alpha_{f,\hat{f}}$ may differ from one $\hat{f}$ to another, and yet the s_i's are identical for every $\hat{f} \in \hat{\mathcal{F}}$.

The s_i's are specified outside the orthogonalization process (in the algorithms GFS and GFA in Section III and Section IV, respectively), and in this section we focus on how the orthogonalization is done for any choice of those integers. Since the s_i's are chosen one-by-one, we describe the orthogonalization process in a similar fashion, i.e., variable-by-variable.

To this end, we first partition $\mathcal{F}$ to subsets according to the y_i's on which they depend, as follows. We say that a function $f \in \mathcal{F}$ *depends* on y_i if there exists $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathbb{R}^n$, different only in the i^{th} entry, such that $f(\boldsymbol{\alpha}) \neq f(\boldsymbol{\beta})$. Further, for $\mathcal{J} \subseteq [n]$ we let $\mathcal{F}_{\mathcal{J}} \subseteq \mathcal{F}$ be the set of functions which depend only on variables contained in $\{y_i\}_{i \in \mathcal{J}}$, and use the abbreviated notation $\mathcal{F}_j \triangleq \mathcal{F}_{[j]}$ (i.e., the functions in $\mathcal{F}$ which only depend on $y_1, \ldots, y_j$). Notice that $f \in \mathcal{F}_j$ can be written more compactly as $f(y_1, \ldots, y_j)$ instead of $f(y_1, \ldots, y_n)$. We partition $\hat{\mathcal{F}}$ similarly according to the dependence on the z_i's.

Using these notations, in Algorithm 1 we present our inductive orthogonalization algorithm. For some $j \geq 1$ we suppose that $s_1, \ldots, s_j$ were already chosen, and we show how to create the function family $\hat{\mathcal{F}}_j$ (i.e., the orthogonalized counterpart of $\mathcal{F}_j$) from the already orthogonalized $\hat{\mathcal{F}}_{j-1}$ and the non-random function family $\mathcal{F}_j \setminus \mathcal{F}_{j-1}$. Algorithm 1 follows a simple GS process over the function space; for

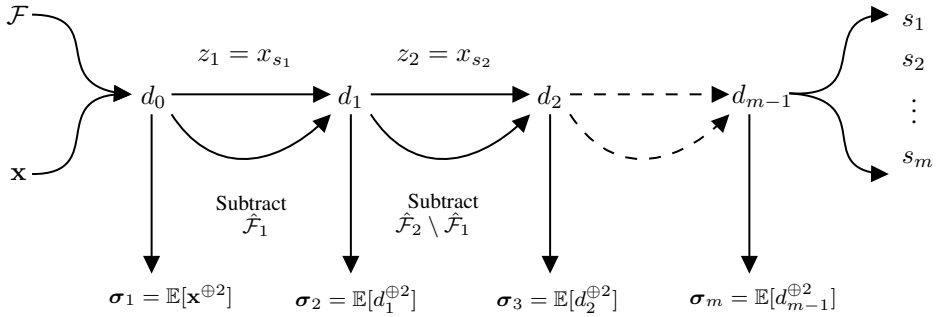

Fig. 1: A schematic description of GFS and GFA. The variance vectors $\{\boldsymbol{\sigma}_i\}_{i=1}^m$ are obtained iteratively by subtracting orthogonalized functions in $\mathcal{F}$ whose variables z_i were already specified as previously chosen features of $\mathbf{x}$, computing the new random variable d_j's, with $\boldsymbol{\sigma}_j$'s representing their variance vectors. In GFS, these variance vectors are used to select the most variant features in the new random variable d_j in each iteration while in GFA, they are used to detect and remove redundant features from the dataset.

each member of $\mathcal{F}_j \setminus \mathcal{F}_{j-1}$, the algorithm substitutes the non-random y_i's by the random variables z_i (Line 6), subtracts its projection on already-orthogonalized functions (Line 7), normalizes (Line 8), and adds the result to $\hat{\mathcal{F}}_j$ (Line 8). After the subtraction process we define a new random variable $d_j(\mathbf{x})$ from $\mathbf{x}$ (Line 10, with $d_0(\mathbf{x}) \triangleq \mathbf{x}$), whose variance vector

$$\boldsymbol{\sigma}_{j+1} = \mathrm{diag}(\mathbb{E}[d_j(\mathbf{x})d_j(\mathbf{x})^\intercal]) = \mathbb{E}[d_j(\mathbf{x})^{\oplus 2}]$$

is computed (Line 11) and returned to the calling algorithm (either GFS or GFA) to compute the next selected feature s_{j+1}.

Remark 1. *We emphasize that the data distribution* need not *be known. The random variable* $\mathbf{x}$*, as well as functions computed from it such as* $d_j(\mathbf{x})$ *and the functions in* $\hat{\mathcal{F}}$*, are merely symbolic variables. The only use of these symbolic variable is in computing expectations, which in reality, are approximated by empirical means. For instance* $\mathbb{E}[\mathbf{x}]$ *and* $\mathbb{E}[\mathbf{x}^{\oplus 2}]$ *can be approximated by* $\mathbb{E}[\mathbf{x}] \approx \frac{1}{N}\left[\sum_{i=1}^N D_{i,1}, \ldots, \sum_{i=1}^N D_{i,d}\right]^\intercal$ *and* $\mathbb{E}[\mathbf{x}^{\oplus 2}] \approx \frac{1}{N}\left[\sum_{i=1}^N D_{i,1}^2, \ldots, \sum_{i=1}^N D_{i,d}^2\right]^\intercal$.

III. Gram-Schmidt Functional Selection (GFS)

In Algorithm 2 below we propose *Gram-Schmidt Functional Selection* (GFS), which at every step j uses the variance vector $\boldsymbol{\sigma}_j$ to select the most variant feature of d_{j-1}. If the variance of this feature is less than some threshold ϵ^2, the algorithm stops, and otherwise it continues by specifying the next z_j as the highest variance feature of d_{j-1}, which enables the next call to "Orthogonalize." Formal guarantees are then given in terms of entropy reduction; that is, the conditional entropy $H(\mathbf{x}|\mathbf{z})$ is bounded by a function of ϵ and d. These guarantees require the random variable $\mathbf{x}$ to be over a finite alphabet, and yet, experiments in Section V show significant gains in real-world setting in which the data is continuous. The information-theoretic guarantee of GFS is as follows.

Theorem 1. *Let* $\mathbf{x}$ *be a random variable over a discrete domain* $\mathcal{X}^d$*, for some* $\mathcal{X} \subseteq \mathbb{R}$*, and let* $\mathbf{x}^{\mathcal{S}_\epsilon}$ *be the output of GFS over a dataset in which every datapoint is sampled independently from* $\mathbf{x}$*. Then* $H(\mathbf{x}|\mathbf{x}^{\mathcal{S}_\epsilon}) \leq dO(\epsilon)$.

Algorithm 1: Orthogonalize($\hat{\mathcal{F}}_{j-1}, \mathcal{F}_j \setminus \mathcal{F}_{j-1}, \{s_i\}_{i=1}^j, \mathbf{D}$)

1: **Input:** An orthogonalized function family $\hat{\mathcal{F}}_{j-1}$ in random variables $z_1 = x_{s_1}, \ldots, z_{j-1} = x_{s_{j-1}}$, distinct integers $\{s_i\}_{i=1}^j \subseteq [d]$, a function family $\mathcal{F}_j \setminus \mathcal{F}_{j-1}$ in non-random variables $y_1, \ldots, y_j$, and a data matrix $\mathbf{D}$ whose rows are sampled i.i.d from $\mathbf{x}$, in order to approximate expectations using empirical means.
2: **Output:** Orthogonalized functions $\hat{\mathcal{F}}_j$, and a variance vector $\boldsymbol{\sigma}_{j+1}$.
3: **Initialize:** $\hat{\mathcal{F}}_j = \hat{\mathcal{F}}_{j-1}$.
4: Denote $\mathcal{F}_j \setminus \mathcal{F}_{j-1} \triangleq \{f_a, f_{a+1}, \ldots, f_{a+\ell-1}\}$ (with $f_a = y_j$), and let $z_j = x_{s_j}$.
5: **for** $k \leftarrow 0$ **to** $\ell - 1$ **do**
6: Let $g \triangleq f_{a+k}(z_1, \ldots, z_j)$ (a random function).
7: Let $\tilde{f}_{a+k} = g - \sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1}} \mathbb{E}[g\hat{f}]\hat{f} - \sum_{r=0}^{k-1} \mathbb{E}[g\hat{f}_{a+r}]\hat{f}_{a+r}$.
8: Add $\hat{f}_{a+k} = \frac{\tilde{f}_{a+k}}{\sqrt{\mathbb{E}[\tilde{f}_{a+k}^2]}}$ to $\hat{\mathcal{F}}_j$.
9: **end for**
10: Define $d_j(\mathbf{x}) = d_{j-1}(\mathbf{x}) - \sum_{\hat{f} \in \hat{\mathcal{F}}_j \setminus \hat{\mathcal{F}}_{j-1}} \mathbb{E}[\mathbf{x}\hat{f}]\hat{f}$.
11: Define $\boldsymbol{\sigma}_{j+1} = \mathbb{E}[d_j^{\oplus 2}]$.
12: Return $\boldsymbol{\sigma}_{j+1}$, $\hat{\mathcal{F}}_j$.

To prove the theorem, let $\mathcal{S}_\epsilon$ be the features selected by GFS, and let $\{\boldsymbol{\sigma}_j\}_{j=m+1}^d$ be the variance vectors which result from completing the orthogonalization which GFS started, when the remaining variables $\{z_j\}_{j=m+1}^d$ are set to $\{x_j\}_{j \notin \mathcal{S}_\epsilon}$ in *any arbitrary* order.

Lemma 1. *The vectors* $\{\boldsymbol{\sigma}_i\}_{i=m+1}^d$ *that are defined during the completion of the orthogonalization of* $\mathcal{F}$ *satisfy that* $\|\boldsymbol{\sigma}_i\|_\infty \leq \epsilon^2$ *for all* $i \in \{m+1, \ldots, d\}$.

The following technical statement is required for the subsequent proof of Lemma 1.

Lemma 2. *In GFS, we have the following.*

Algorithm 2: Gram-Schmidt Functional Selection (GFS)

Input: Random variables $\mathbf{x} = (x_1, \ldots, x_d)^\intercal$, a function family $\mathcal{F}$ in non-random variables $y_1, \ldots, y_d$, threshold $\epsilon > 0$, and a data matrix $\mathbf{D}$ whose rows are sampled i.i.d from $\mathbf{x}$ in order to approximate expectations using empirical means.

Output: m (varying number) selected features $\mathbf{x}^{\mathcal{S}_\epsilon} = (x_j)_{j \in \mathcal{S}_\epsilon}$.

Initialize $j = 1$, $\boldsymbol{\sigma}_1 = \mathbb{E}[\mathbf{x}^{\oplus 2}]$, $d_0(\mathbf{x}) = \mathbf{x}$, and $\hat{\mathcal{F}} = \varnothing$, $\mathcal{S}_\epsilon = \varnothing$.

for $j \leftarrow 1$ **to** d **do**

  Let $s_j \triangleq \arg\max_{i \in [d]} \{\sigma_{j,i}\}_{i=1}^d$, where $\boldsymbol{\sigma}_j = (\sigma_{j,i})_{i=1}^d$.

  if $\|\boldsymbol{\sigma}_j\|_\infty \leq \epsilon^2$ **then**

    break.

  else

    Set $z_j = x_{s_j}$ and add s_j to $\mathcal{S}_\epsilon$.

  end

  $\boldsymbol{\sigma}_{j+1}, \hat{\mathcal{F}}_j =$ Orthogonalize$(\hat{\mathcal{F}}_{j-1}, \mathcal{F}_j \setminus \mathcal{F}_{j-1}, \mathcal{S}_\epsilon, \mathbf{D})$

end

(a) *For $i < j$ in $[d]$ we have* $\boldsymbol{\sigma}_j = \boldsymbol{\sigma}_i - \sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1} \setminus \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\hat{f}\mathbf{x}]^{\oplus 2}$.

(b) *The integers $s_1, \ldots, s_m$ are distinct.*

Proof of Lemma 2.(a). It is readily verified that $d_{j-1}(\mathbf{x})$ can be written as

$$d_{j-1}(\mathbf{x}) = d_{i-1}(\mathbf{x}) - \sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1} \setminus \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\mathbf{x}\hat{f}]\hat{f}, \tag{1}$$

and that $d_{i-1}(\mathbf{x})$ can be written as

$$d_{i-1}(\mathbf{x}) = \mathbf{x} - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\mathbf{x}\hat{f}]\hat{f}. \tag{2}$$

Hence, it follows from (1) that

$$\begin{aligned}
\boldsymbol{\sigma}_j &= \mathbb{E}[d_{j-1}^{\oplus 2}] \\
&= \mathbb{E}\left[\left(d_{i-1}(\mathbf{x}) - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1} \setminus \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\mathbf{x}\hat{f}]\hat{f} \right)^{\oplus 2} \right] \\
&= \boldsymbol{\sigma}_i - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1} \setminus \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\hat{f} d_{i-1}(\mathbf{x})] \oplus \mathbb{E}[\mathbf{x}\hat{f}] \\
&\quad - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1} \setminus \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\hat{f}\mathbf{x}] \oplus \mathbb{E}[d_{i-1}(\mathbf{x})\hat{f}] \\
&\quad + \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1} \setminus \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\mathbf{x}\hat{f}]^{\oplus 2},
\end{aligned} \tag{3}$$

where the last summand follows from the orthonormality of $\hat{\mathcal{F}}$. Further, it follows from (2) that every $\hat{f} \notin \hat{\mathcal{F}}_{i-1}$ satisfies

$$\mathbb{E}[\hat{f} d_{i-1}(\mathbf{x})] = \mathbb{E}[\hat{f} \cdot (\mathbf{x} - \textstyle\sum_{\hat{g} \in \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\mathbf{x}\hat{g}]\hat{g})] = \mathbb{E}[\hat{f}\mathbf{x}],$$

therefore (3) $= \boldsymbol{\sigma}_i - \sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1} \setminus \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\mathbf{x}\hat{f}]^{\oplus 2}$ as required. □

Proof of Lemma 2.(b). Notice that for every $i \in [d]$ and every $r \in [m]$ we have

$$\sigma_{r,i} = \mathbb{E}[d_{r-1}(\mathbf{x})_i^2]. \tag{4}$$

and since $d_{r-1}(\mathbf{x}) = \mathbf{x} - \sum_{\hat{f} \in \hat{\mathcal{F}}_{r-1}} \mathbb{E}[\mathbf{x}\hat{f}]\hat{f}$, it follows that

$$(4) = \mathbb{E}\left[\left(x_i - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{r-1}} \mathbb{E}[x_i \hat{f}]\hat{f} \right)^2 \right] \tag{5}$$

That is, at the beginning of the r^{th} iteration for any r, the algorithm will find the maximizer s_r over i of (5). Now, observe that if $i = s_k$ for some $k \in \{1, \ldots, r-1\}$, i.e., if the i^{th} feature was already selected in an earlier iteration k, then $z_k = x_i$. Since $z_k = \tilde{z}_k + \sum_{\hat{f} \in \hat{\mathcal{F}}_{k-1}} \mathbb{E}[z_k \hat{f}]\hat{f}$ (Line 7 of Algorithm 1), we have

$$\begin{aligned}
(5) &= \mathbb{E}\Bigg[\Bigg(\tilde{z}_k + \textstyle\sum_{\hat{g} \in \hat{\mathcal{F}}_{k-1}} \mathbb{E}[z_k \hat{g}]\hat{g} \\
&\qquad - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{r-1}} \mathbb{E}[(\tilde{z}_k + \sum_{\hat{g} \in \hat{\mathcal{F}}_{k-1}} \mathbb{E}[z_k \hat{g}]\hat{g})\hat{f}]\hat{f} \Bigg)^2 \Bigg] \\
&= \mathbb{E}\Bigg[\Bigg(\tilde{z}_k + \textstyle\sum_{\hat{g} \in \hat{\mathcal{F}}_{k-1}} \mathbb{E}[z_k \hat{g}]\hat{g} \\
&\qquad - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{r-1}} \left(\mathbb{E}[\tilde{z}_k \hat{f}] + \sum_{\hat{g} \in \hat{\mathcal{F}}_{k-1}} \mathbb{E}[z_k \hat{g}]\mathbb{E}[\hat{g}\hat{f}] \right) \hat{f} \Bigg)^2 \Bigg] \\
&= \mathbb{E}\Bigg[\Bigg(\tilde{z}_k + \textstyle\sum_{\hat{g} \in \hat{\mathcal{F}}_{k-1}} \mathbb{E}[z_k \hat{g}]\hat{g} \\
&\qquad - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{r-1}} \mathbb{E}[\tilde{z}_k \hat{f}]\hat{f} - \sum_{\hat{g} \in \hat{\mathcal{F}}_{k-1}} \mathbb{E}[z_k \hat{g}]\hat{g} \Bigg)^2 \Bigg] \\
&= \mathbb{E}\left[\left(\tilde{z}_k - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{r-1}} \mathbb{E}[\tilde{z}_k \hat{f}]\hat{f} \right)^2 \right].
\end{aligned} \tag{6}$$

Since $\tilde{z}_k$ is orthogonalized, it follows that

$$(6) = \mathbb{E}[(\tilde{z}_k - \hat{z}_k \cdot \|\tilde{z}_k\|)^2] = 0,$$

and hence the maximization problem will not select an index i that was already selected at a previous iteration. □

Proof of Lemma 1. According to the stopping criterion in GFS, it follows that

$$\begin{aligned}
&\|\boldsymbol{\sigma}_i\|_\infty > \epsilon^2 \text{ for all } i \in [m]; \text{ and} \\
&\|\boldsymbol{\sigma}_{m+1}\|_\infty \leq \epsilon^2.
\end{aligned} \tag{7}$$

Assume for contradiction that there exists $i \in \{m+1, \ldots, d\}$ such that $\|\boldsymbol{\sigma}_i\|_\infty > \epsilon^2$, and observe that $i > m+1$, since otherwise the existence of s_{m+1} contradicts (7). It follows from Lemma 2.(a) that

$$\boldsymbol{\sigma}_i = \boldsymbol{\sigma}_{m+1} - \sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1} \setminus \hat{\mathcal{F}}_m} \mathbb{E}[\hat{f}\mathbf{x}]^{\oplus 2},$$

and therefore

$$\begin{aligned}
\|\boldsymbol{\sigma}_i\|_\infty = \sigma_{i,s_i} &= \sigma_{m+1,s_i} - \sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1} \setminus \hat{\mathcal{F}}_m} \mathbb{E}[\hat{f}x_{s_i}]^2 \\
&\overset{(7)}{\leq} \epsilon^2 - \sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1} \setminus \hat{\mathcal{F}}_m} \mathbb{E}[\hat{f}x_{s_i}]^2.
\end{aligned}$$

Therefore, since $\|\boldsymbol{\sigma}_i\|_\infty > \epsilon^2$, it follows that

$$-\sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1} \setminus \hat{\mathcal{F}}_m} \mathbb{E}[\hat{f}x_{s_i}]^2 > 0,$$

which is a contradiction since $\mathbb{E}[\hat{f}x_{s_i}]^2 \geq 0$ for every $\hat{f}$. □

Proof of Theorem 1. Let $s_1, \ldots, s_m$ be the features selected by GFS, and let $\{s_{m+1}, \ldots, s_d\} = [d] \backslash \{s_1, \ldots, s_m\}$. Further, as in Lemma 1, let $\hat{\mathcal{F}}$ be the result of completing the orthogonalization of $\mathcal{F}_m$, done throughout GFS, to orthogonalization of $\mathcal{F}$ in its entirety, where the variables $\mathbf{z}^{\perp} \triangleq (z_{m+1}, \ldots, z_d)$ are specified as $z_i = x_{s_i}$ for all $i \in \{m+1, \ldots, d\}$. We have,

$$H(\mathbf{x}|\mathbf{z}) = H(\{z_i\}_{i=1}^d | \{z_i\}_{i=1}^m) = \sum_{i=m+1}^{d} H(z_i | (z_j)_{j=1}^{i-1}), \tag{8}$$

where the last equality follows from the chain rule for information entropy. Since

$$\tilde{z}_i = z_i - \sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1}} \mathbb{E}[z_i \hat{f}] \hat{f}, \tag{9}$$

it follows that

$$H(z_i | (z_j)_{j=1}^{i-1}) = H(\tilde{z}_i + \sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1}} \mathbb{E}[z_i \hat{f}] \hat{f} | (z_j)_{j=1}^{i-1}) \overset{(a)}{=} H(\tilde{z}_i | (z_j)_{j=1}^{i}) \overset{(b)}{\leq} H(\tilde{z}_i), \tag{10}$$

where (a) follows since the expression $\sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1}} \mathbb{E}[z_i \hat{f}] \hat{f}$ is uniquely determined by the variables $z_1, \ldots, z_{i-1}$ (every $\hat{f} \in \hat{\mathcal{F}}_{i-1}$ is a deterministic function of $z_1, \ldots, z_{i-1}$, and the coefficients $\mathbb{E}[z_i \hat{f}]$ are constants), and (b) follows since conditioning reduces entropy. Combining (8) with (10) we have that $H(\mathbf{x}|\mathbf{z}) \leq \sum_{i=m+1}^{d} H(\tilde{z}_i)$, and hence it remains to bound $H(\tilde{z}_i)$ for all $i \in \{m+1, \ldots, d\}$.

To this end let $i \in \{m+1, \ldots, d\}$, define $a_i \triangleq \min\{|\tilde{z}_i(\mathbf{a})| : \mathbf{a} \in \mathcal{X}^d, \tilde{z}_i(\mathbf{a}) \neq 0\}$ and $a_{\min} = \min\{a_i\}_{i=m+1}^d$, and observe that by Markov's inequality we have

$$\Pr(\tilde{z}_i(\mathbf{x}) \neq 0) = \Pr(|\tilde{z}_i(\mathbf{x})| \geq a_i) = \Pr(\tilde{z}_i(\mathbf{x})^2 \geq a_i^2) \leq \frac{\mathbb{E}[\tilde{z}_i^2]}{a_i^2} \leq \frac{\mathbb{E}[\tilde{z}_i^2]}{a_{\min}^2}. \tag{11}$$

Moreover, recall that

$$d_{i-1}(\mathbf{x}) = \mathbf{x} - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1}} \mathbb{E}[\mathbf{x} \hat{f}] \hat{f}, \text{ and } \boldsymbol{\sigma}_i = \mathbb{E}[d_{i-1}(\mathbf{x})^{\oplus 2}],$$

and therefore

$$\|\boldsymbol{\sigma}_i\|_\infty = \sigma_{i,s_i} = \mathbb{E}[d_{i-1}(\mathbf{x})_{s_i}^2] = \mathbb{E}[(x_{s_i} - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1}} \mathbb{E}[x_{s_i} \hat{f}] \hat{f})^2] = \mathbb{E}[(z_i - \textstyle\sum_{\hat{f} \in \hat{\mathcal{F}}_{i-1}} \mathbb{E}[z_i \hat{f}] \hat{f})^2] \overset{(9)}{=} \mathbb{E}[\tilde{z}_i^2].$$

Consequently, it follows that

$$(11) = \frac{\|\boldsymbol{\sigma}_i\|_\infty}{a_{\min}^2} \overset{\text{Lemma 1}}{\leq} \frac{\epsilon^2}{a_{\min}^2}.$$

Now, by using the grouping rule [6, Ex. 2.27] we have

$$H(\tilde{z}_i) \leq h_b(\epsilon^2 / a_{\min}^2) + \frac{\epsilon^2}{a_{\min}^2} \log |\mathcal{X}|,$$

where h_b is the binary entropy function. Finally, using the bound $h_b(p) \leq 2\sqrt{p(1-p)} \leq 2\sqrt{p}$, it follows that

$$\begin{aligned} H(\mathbf{x}|\mathbf{z}) &\leq \textstyle\sum_{i=m+1}^{d} H(\tilde{z}_i) \\ &\leq (d-m)(h_b(\epsilon^2/a_{\min}^2) + (\epsilon^2/a_{\min}^2) \log |\mathcal{X}|) \\ &\leq (d-m)(2\epsilon/a_{\min} + (\epsilon/a_{\min}) \log |\mathcal{X}|) = dO(\epsilon). \end{aligned}$$ □

IV. Gram-Schmidt Feature Analysis (GFA)

To provide a more comprehensive picture regarding the capabilities of our GS approach, in this section we analyze an ideal setting in which our approach can reduce the conditional entropy to zero, i.e., select the non-redundant features exactly. To this end, in Algorithm 3 below we describe *Gram-Schmidt Feature Analysis* (GFA) for detecting and removing redundant features under certain idealized conditions. In each iteration j, the algorithm begins by identifying the features of $\mathbf{x}$ in which the random variable d_j (see Algorithm 1) has no variance. These features are then subtracted from the data to produce $\bar{\mathbf{x}}$, and the most variant feature s_j is identified via the variance vector of $\bar{\mathbf{x}}$. The next call to "Orthogonalize" can then be made.

Algorithm 3: Gram-Schmidt Feature Analysis (GFA)

Data: $\mathbf{x} = (x_1, \ldots, x_d)^\intercal \in \mathbb{R}^d$, a function family $\mathcal{F}$, an integer $n \leq d$, and a data matrix $\mathbf{D}$ whose rows are sampled i.i.d from $\mathbf{x}$ in order to approximate expectations using empirical means.
Result: Selected features $\mathcal{S} \subseteq [d]$.
Initialize $\boldsymbol{\sigma}_1 = \mathbb{E}[\mathbf{x}^{\oplus 2}]$, $d_0(\mathbf{x}) = \mathbf{x}$, $\mathcal{S} = \varnothing$, and $\hat{\mathcal{F}} = \varnothing$.
for $j \leftarrow 1$ **to** n **do**
  Let $\mathcal{E} = \{i | \sigma_{j,i} = 0\}$ (or $\sigma_{j,i} < \delta$ for some small δ in the empirical variant).
  Define $\bar{\mathbf{x}} = \mathbf{x} - \sum_{i \in \mathcal{E}} x_i \mathbf{e}_i$ (where $\{\mathbf{e}_i\}_{i \in [d]}$ is the standard basis).
  Add $s_j \triangleq \arg\max_{i \in [d] \backslash \mathcal{E}} (\mathbb{E}[\bar{\mathbf{x}}^{\oplus 2}])_i$ to $\mathcal{S}$.
  $\boldsymbol{\sigma}_{j+1}, \hat{\mathcal{F}}_j = \text{Orthogonalize}(\hat{\mathcal{F}}_{j-1}, \mathcal{F}_j \backslash \mathcal{F}_{j-1}, \mathcal{S}, \mathbf{D})$
end

We now prove that under certain probabilistic assumptions, GFA correctly identifies and removes the redundant features in the data *exactly* (i.e., $H(\mathbf{x}|\mathbf{z}) = 0$, where $\mathbf{z}$ is the set of selected features). Generally speaking, suppose that $\mathbf{x}$'s features have some redundancy that can be described by $\operatorname{span} \mathcal{F}$; that is, some latent variables determine some of the features, and the remaining features are functions, from $\operatorname{span} \mathcal{F}$, of those latent variables. In such cases, it is clear that identifying the latent variables correctly yields $H(\mathbf{x}|\mathbf{z}) = 0$. The required probabilistic assumptions are that the redundancy functions are *variance reducing*, in the following sense.

Theorem 2. *For positive integers $d \geq n$, where n is the number of non-redundant features, let $\mathcal{F}$ be a set of functions in (non-random) variables $\mathbf{y} = (y_1, \ldots, y_n)$, let $h_1, \ldots, h_d \in \operatorname{span} \mathcal{F}$ with $h_i(\mathbf{y}) = y_i$ for $i \in [n]$, and let $\mathbf{w} = (w_1, \ldots, w_n)$ be random variables such that for every $j \in \{n+1, \ldots, d\}$,*

$$\operatorname{Var}\left(h_j(\mathbf{w})\right) < \min_{\{i | h_j \text{ depends on } y_i\}} \operatorname{Var}\left(w_i\right). \tag{12}$$

Let $\mathbf{x}$ in $\mathbb{R}^d$ be a random variable of the form

$$\mathbf{x} = \left[h_{\pi(1)}(\mathbf{w}), \ldots, h_{\pi(d)}(\mathbf{w})\right]^{\mathsf{T}} \tag{13}$$

for some permutation π over $[d]$, i.e., some n entries of $\mathbf{x}$ are the latent random variables $\{w_i\}_{i=1}^n$, and the remaining ones are functions of those latent variables. Then, GFA with input $\mathcal{F}$ and a data matrix drawn i.i.d from $\mathbf{x}$ outputs $\mathcal{S} = \{s_i\}_{i=1}^n$ such that $\{x_{s_j}\}_{j=1}^n = \{h_i(\mathbf{w})\}_{i=1}^n = \{w_i\}_{i=1}^n$, i.e., it identifies the latent variables $\{w_i\}_{i=1}^n$ correctly, and therefore $H(\mathbf{x}|\mathbf{z}) = 0$, where $\mathbf{z}$ is the set of selected features.

Proof. The claim is proved by induction on the iteration index j in GFA. For the base case $j = 1$, observe that[1] $\mathcal{E} = \varnothing$ and $\bar{\mathbf{x}} = \mathbf{x}$, and hence s_1 is the index of the largest entry of $\mathbb{E}[\mathbf{x}^{\oplus 2}]$. Since $\mathbb{E}[\mathbf{x}^{\oplus 2}] \overset{(13)}{=} (\operatorname{Var}(h_{\pi(1)}(\mathbf{w})), \ldots, \operatorname{Var}(h_{\pi(d)}(\mathbf{w})))$, and since each redundancy function in $\{h_{n+1}, \ldots, h_d\}$ has less variance than the latent variables on which it depends (12), it follows that $s_1 \in [d]$ satisfies

$$z_1 = x_{s_1} \overset{(13)}{=} h_{\pi(s_1)}(\mathbf{w}) = w_{\pi(s_1)}, \tag{14}$$

i.e., one latent variable is identified correctly.

Now, the induction hypothesis is that the selected $s_1, \ldots, s_{j-1} \in [d]$ are distinct integers such that $z_i = x_{s_i} = h_{\pi(s_i)}(\mathbf{w}) = w_{\pi(s_i)}$ for $i \in [j-1]$ (similar to (14)), i.e., that $j-1$ distinct latent variables were selected so far. We wish to show that $z_j = x_{s_j} \in \{w_k\}_{k=1}^n \setminus \{w_{\pi(s_i)}\}_{i=1}^{j-1}$, i.e., that the next selected feature is a latent variable that was not selected so far. We rewrite (13) as $\mathbf{x} = \sum_{i=1}^d \mathbf{e}_i h_{\pi(i)}(\mathbf{w})$, and then $d_{j-1}(\mathbf{x})$ can be written as

$$\begin{aligned} d_{j-1}(\mathbf{x}) &= \mathbf{x} - \sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1}} \mathbb{E}[\mathbf{x}\hat{f}(z_1, \ldots, z_{j-1})]\hat{f}(z_1, \ldots, z_{j-1}) \\ &= \mathbf{x} - \sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1}} \mathbb{E}[(\sum_{i=1}^d \mathbf{e}_i h_{\pi(i)}(\mathbf{w}))\hat{f}(w_{\pi(s_1)}, \ldots, w_{\pi(s_{j-1})})] \cdot \\ &\qquad \hat{f}(w_{\pi(s_1)}, \ldots, w_{\pi(s_{j-1})}). \end{aligned} \tag{15}$$

For $i \in [d]$ let ℓ_i be the smallest integer such that[2] $h_{\pi(i)} \in \operatorname{span}\{f_1, \ldots, f_{\ell_i}\}$, which implies that $h_{\pi(i)} = \sum_{a=1}^{\ell_i} \mathbb{E}[h_{\pi(i)}\hat{f}_a]\hat{f}_a$ for every i since the $\hat{f}_a$'s are orthonormal. Therefore,

$$d_{j-1}(\mathbf{x}) = \mathbf{x} - \sum_{\hat{f} \in \hat{\mathcal{F}}_{j-1}} \mathbb{E}[(\sum_{i=1}^d \mathbf{e}_i (\sum_{a=1}^{\ell_i} \mathbb{E}[h_{\pi(i)}\hat{f}_a]\hat{f}_a))\hat{f}]\hat{f}.$$

By denoting $\hat{\mathcal{F}}_{j-1} = \{\hat{f}_k\}_{k=1}^p$ we have

$$\begin{aligned} &d_{j-1}(\mathbf{x}) = \\ &= \mathbf{x} - \textstyle\sum_{k=1}^p \mathbb{E}[(\sum_{i=1}^d \mathbf{e}_i (\sum_{a=1}^{\ell_i} \mathbb{E}[h_{\pi(i)}(\mathbf{w})\hat{f}_a]\hat{f}_a))\hat{f}_k]\hat{f}_k \\ &= \mathbf{x} - \textstyle\sum_{k=1}^p \sum_{i=1}^d \sum_{a=1}^{\ell_i} \mathbf{e}_i \mathbb{E}[h_{\pi(i)}(\mathbf{w})\hat{f}_a]\mathbb{E}[\hat{f}_k\hat{f}_a]\hat{f}_k \\ &= \mathbf{x} - \textstyle\sum_{i=1}^d \sum_{a=1}^{\min\{p,\ell_i\}} \mathbf{e}_i \mathbb{E}[h_{\pi(i)}(\mathbf{w})\hat{f}_a]\hat{f}_a \\ &= \textstyle\sum_{i=1}^d \mathbf{e}_i \left(h_{\pi(i)}(\mathbf{w}) - \sum_{a=1}^{\min\{p,\ell_i\}} \mathbb{E}[h_{\pi(i)}\hat{f}_a]\hat{f}_a\right). \end{aligned} \tag{16}$$

Clearly, whenever $p > \ell_i$, the expression in the parentheses in (16) equals zero, i.e., the i^{th} entry of d_{j-1}, as well as its variance, are zero. By the definition of $\bar{\mathbf{x}}$, and by the induction hypothesis, this implies that $\bar{\mathbf{x}}$ is equal to $\mathbf{x}$ minus all $\mathbf{e}_i h_{\pi(i)}$ such that $h_{\pi(i)}$ depends on a subset of the variables $w_{\pi(s_1)}, \ldots, w_{\pi(s_{j-1})}$. By the definition of $\mathbf{x}$, the remaining entries of $\bar{\mathbf{x}}$ are either of the form w_i for $i \in [n] \setminus \{\pi(s_1), \ldots, \pi(s_{j-1})\}$, or of the form $h_i(\mathbf{w})$ for $i \in \{n+1, \ldots, d\}$, where $h_i(\mathbf{w})$ does not depends only a subset of $\{w_{\pi(s_1)}, \ldots, w_{\pi(s_{j-1})}\}$. Therefore, since each such $h_i(\mathbf{w})$ depends on at least one additional variable w_a for $a \in [n] \setminus \{\pi(s_1), \ldots, \pi(s_{j-1})\}$, Eq. (12) implies that in iteration j, the largest entry of $\mathbb{E}[\bar{\mathbf{x}}^{\oplus 2}]$ must be a latent variable, i.e., $x_{s_j} = h_{\pi(s_j)}(\mathbf{w}) = w_{\pi(s_j)}$, as required. Completing n steps of this induction implies that $\{z_i\}_{i=1}^n = \{w_{\pi(s_i)}\}_{i=1}^n = \{w_i\}_{i=1}^n$, which concludes the proof. □

[1] We assume without loss of generality that the features of $\mathbf{x}$ have positive variance.

[2] We order $\mathcal{F}$ so that $\mathcal{F}_1$ appears first, then $\mathcal{F}_2 \setminus \mathcal{F}_1$, then $\mathcal{F}_3 \setminus \mathcal{F}_2$, and so on, with y_j the appearing first among $\mathcal{F}_j \setminus \mathcal{F}_{j-1}$ for every j, see Line 4 of Algorithm 1.

V. Experimental Results

In this section, the theoretical results and performance of the proposed algorithms are evaluated through simulation studies over synthetic and real-world datasets, where expectations are approximated using empirical means. All experiments were performed on a laptop with Intel(R) Core(TM) i9-9880H CPU @ 2.30GHz and 64GB RAM.

A. Numerical Simulations for the Number of Selected Features using GFS

We begin by demonstrating that selecting higher degree multilinear polynomials as the redundant functions $\mathcal{F}$ leads to a significant reduction in maximum variance of the random variable $d_j(\boldsymbol{x})$ in each iteration. Consequently, this approach effectively detects and removes more redundant features from the datasets as the complexity of $\mathcal{F}$ increases. To support this claim, we apply GFS with $\mathcal{F}$ being multilinear polynomials of degrees 1, 2, 3, and 4, to the benchmark datasets taken from UCI repository [22]. The properties of the tested benchmark datasets are provided in Table I.

Dataset	USPS	COIL-20	Credit Approval (CA)
Features	256	1024	15
Samples	7291	1440	690

TABLE I: Properties of the tested benchmark datasets.

The number of selected features for different values of the threshold ϵ^2 and degree of the polynomials are shown in Tables II-IV. As demonstrated in the experimental results, by increasing the degree of multilinear polynomials, GFS selects fewer features given the same threshold ϵ^2.

Degree of Polynomial	ϵ^2					
	0.03	0.04	0.05	0.06	0.07	0.08
1	85	60	48	41	36	29
2	39	31	25	21	18	16
3	25	20	18	16	15	12
4	24	17	14	13	12	11

TABLE II: Number of selected features of GFS in USPS.

Degree of Polynomial	ϵ^2				
	0.01	0.0125	0.015	0.0175	0.02
1	98	76	60	50	40
2	36	21	17	15	14
3	36	20	13	11	9
4	31	19	13	10	9

TABLE III: Number of selected features of GFS in COIL-20.

B. Experimental Results for Classification Accuracy of GFS

We turn to validate the performance of GFS for classification tasks on the benchmark datasets in Table I. In Tables V-VII, we apply GFS with multilinear polynomials up to degree 4, against several well-known feature selection algorithms, i.e., Multi-Cluster Feature Selection (MCFS) [13], Nonnegative Discriminative Feature Selection (NDFS) [15], Unsupervised Discriminative Feature Selection (UDFS) [16], Laplacian Score (LS) [8], Trace Ratio (TR) [12], and Fisher Score (FS) [10]. The classification accuracy in the benchmark datasets shows superior performance of GFS in comparison to other algorithms.

We used a support vector machine classifier with radial basis function as kernel. 5-fold cross-validation on the entire datasets is used to validate the performance of the algorithms. To implement MCFS, NDFS, UDFS, LS, TR, and FS, we used `skfeature-chappers` package. The experimental results show that GFS outperforms other state-of-the-art feature selection algorithms in terms of classification accuracy.

C. Comparison of GFS and UFFS [21]

As mentioned earlier, for the special case of choosing $\mathcal{F}$ as multilinear polynomials, GFS specifies to the Unsupervised Fourier Feature Selection (UFFS) of [21], yet at significantly reduced complexity. The reduction in complexity is due to our step-by-step orthogonalization process (Algorithm 1), in contrast to [21] which orthogonalizes all multilinear polynomials. We corroborate GFS's superiority over [21] in terms of running time, the ability to capture redundant features, and classification accuracy, over synthetic datasets. The synthetic datasets we used include 10 independent normally distributed random variables, i.e. $w_1, \ldots, w_{10}$, and 20 redundant features which are randomly taken from $\{w_i w_j\}_{i,j\in[10]} \cup \{w_i w_j w_k\}_{\text{distinct } i,j,k\in[10]}$. We also considered three different dataset sizes, 5000, 10000, and 50000. In all the following comparisons between GFS and UFFS, we chose $\mathcal{F} = \{y_i\}_{i\in[10]} \cup \{y_i y_j\}_{i,j\in[10]} \cup \{y_i y_j y_k\}_{\text{distinct } i,j,k\in[10]}$ as the function family in GFS.

a) Running Time: In Table VIII, the running time ratio of the UFFS and GFS is compared for three different dataset sizes. The results show that in average GFS is almost 27 times faster than UFFS.

Degree of Polynomial	ϵ^2	
	0.1	0.4
1	14	13
2	13	12
3	13	12
4	13	12

TABLE IV: Number of Selected Features of GFS in CA.

Method	Number of selected features					
	24	17	14	13	12	11
GFS	**92.62**	**89.27**	**85.10**	**83.73**	**82.25**	**79.60**
MCFS	90.99	87.37	84.39	82.81	79.76	75.13
NDFS	89.48	82.84	78.38	73.98	72.92	69.98
UDFS	86.37	78.81	76.56	75.71	73.06	69.03
LS	85.21	80.94	76.59	75.59	74.12	73.36
TR	83.87	78.48	74.52	72.10	71.81	68.71
FS	83.87	78.93	74.39	74.02	71.81	68.71

TABLE V: Classification accuracy (%) over USPS.

Method	Number of Selected Features				
	31	19	13	10	9
GFS	**90.76**	**87.08**	**84.72**	**81.18**	**78.19**
MCFS	79.03	77.57	67.57	61.18	59.93
NDFS	87.78	83.89	81.94	77.64	75.76
UDFS	69.44	63.47	58.75	53.82	52.36
LS	68.47	65.28	57.08	49.86	48.40
TR	66.52	57.85	43.75	41.25	39.10
FS	59.23	56.32	43.02	41.20	38.90

TABLE VI: Classification accuracy (%) over COIL-20.

b) Capturing Redundant Features: To show that our proposed GFS algorithm can capture nonlinear redundancies better than UFFS, we performed the experiment reported in Table IX, which shows that GFS tends to select a redundant feature far less frequently than UFFS. UFFS selects at least one redundant feature in all the experiments because datasets contain quadratic functions, and UFFS fails to capture them.

c) Classification Accuracy: Finally, we added the labels

$$f(w_1, \ldots, w_{10}) = \text{sign}\left[\prod_{1\le j\le 3}\left(b_{0,j} + \sum_{i=1}^{10} b_{i,j} w_i\right)\right]$$

to the synthetic data, where $b_{i,j} \sim \text{Unif}(0,1)$ and mutually independent. We selected the same number of features using GFS and UFFS, and applied SVM (Table X).

D. Experimental Results for GFA

GFA's correctness is proved formally in Theorem 2 for the case that variances can be computed exactly. In this section it is shown that the impact of approximating variances from data is small. We tested GFA with $\mathcal{F} = \{y_i\}_{i\in[10]} \cup \{y_i y_j\}_{i,j\in[10]} \cup \{y_i y_j y_k\}_{\text{distinct } i,j,k\in[10]}$ on synthetic datasets with $(n, d) = (10, 20)$, where $w_1, \ldots, w_{10}$ are mutually independent zero mean normals with variance less than 1. The redundancies are either taken from $\{w_i w_j w_k\}_{\text{distinct } i,j,k\in[10]}$ (Table XI) or appropriately scaled forms of $\{w_i w_j\}_{i,j\in[10]}$ (Table XII), and as a result, it is easy to prove that the variance reduction assumption in GFA holds.

VI. Discussion

This paper presents a family of Gram-Schmidt methods for unsupervised feature selection. The presented methods,

Method	Number of Selected Features	
	13	12
GFS	84.20 (2^{nd} best)	83.91 (2^{nd} best)
MCFS	**84.34**	83.62
NDFS	83.62	83.47
UDFS	75.79	75.22
LS	84.05	**84.20**
TR	75.79	75.51
FS	75.80	75.51

TABLE VII: Classification accuracy (%) over CA.

Dataset Size	5000	10000	50000
t_{UFFS}/t_{GFS}	27.19	27.94	26.0

TABLE VIII: Running time ratio of GFS vs. UFFS.

Number of Experiments	10000		
Dataset Size	5000	10000	50000
UFFS	100	100	100
GFS	24.59	24.22	23.94

TABLE IX: Comparison of GFS vs. UFFS in capturing nonlinear redundancies. The numbers indicate the percentage of experiments (%) in which the algorithms selected at least one redundant feature.

Dataset Size	5000	10000	50000
UFFS	72.43	72.94	73.07
GFS	75.06	75.46	76.19

TABLE X: Classification accuracy (%) of GFS vs. UFFS.

Number of Experiments	10000		
Dataset Size	1000	5000	10000
Correct features (%)	94.59	98.15	98.99

TABLE XI: GFA with $(n, d) = (10, 20)$, where the redundancies are arbitrary multilinear polynomials of the w_i's.

Number of Experiments	10000		
Dataset Size	1000	5000	10000
Correct features (%)	100	100	100

TABLE XII: GFA with $(n, d) = (10, 20)$, where the redundancies are scaled quadratic polynomials of the w_i's.

based on a Gram-Schmidt orthogonalization process, can identify and remove nonlinear redundancies from the data and select a subset of informative features. The algorithms are coupled with theoretical guarantees, and experimental results show significant improvements over state-of-the-art feature selection algorithms.

We presented GFS, which requires very mild probabilistic assumptions (existence of expectations), and provides bounded entropy guarantees for discrete distributions (Theorem 1). We also presented GFA, which shows that under stricter probabilistic assumptions one can guarantee *zero* conditional entropy using similar ideas. In a sense, GFA can be viewed as "GFS in an idealized setting." For future research, it is interesting to see if one can weaken the variance reduction assumption in order to get entropy bounds between those of GFA and GFS.

The above experiments demonstrate clear competitive edge against state-of-the-art feature selection mechanisms, including [21]. The importance of choosing the right $\mathcal{F}$, however, is conspicuous. For future research we propose studying this connection further, and identifying other sets $\mathcal{F}$ which provide good performance in various data domains.

REFERENCES

[1] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in *2014 science and information conference*. IEEE, 2014, pp. 372–378.

[2] J. Cai, J. Luo, S. Wang, and S. Yang, "Feature selection in machine learning: A new perspective," *Neurocomputing*, vol. 300, pp. 70–79, 2018.

[3] A. Bommert, X. Sun, B. Bischl, J. Rahnenführer, and M. Lang, "Benchmark for filter methods for feature selection in high-dimensional classification data," *Computational Statistics & Data Analysis*, vol. 143, p. 106839, 2020.

[4] M. M. Kabir, M. M. Islam, and K. Murase, "A new wrapper feature selection approach using neural network," *Neurocomputing*, vol. 73, no. 16-18, pp. 3273–3283, 2010.

[5] S. Wang, J. Tang, and H. Liu, "Embedded unsupervised feature selection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.

[6] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.

[7] M. Bennasar, Y. Hicks, and R. Setchi, "Feature selection using joint mutual information maximisation," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8520–8532, 2015.

[8] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," *Advances in neural information processing systems*, vol. 18, 2005.

[9] R. Huang, W. Jiang, and G. Sun, "Manifold-based constraint laplacian score for multi-label feature selection," *Pattern Recognition Letters*, vol. 112, pp. 346–352, 2018.

[10] R. O. Duda, P. E. Hart *et al.*, *Pattern classification*. John Wiley & Sons, 2006.

[11] R. J. Urbanowicz, M. Meeker, W. La Cava, R. S. Olson, and J. H. Moore, "Relief-based feature selection: Introduction and review," *Journal of biomedical informatics*, vol. 85, pp. 189–203, 2018.

[12] F. Nie, S. Xiang, Y. Jia, C. Zhang, and S. Yan, "Trace ratio criterion for feature selection." in *AAAI*, vol. 2, 2008, pp. 671–676.

[13] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2010, pp. 333–342.

[14] Y. Wang, Z. Zhang, and Y. Lin, "Multi-cluster feature selection based on isometric mapping," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 3, pp. 570–572, 2021.

[15] Z. Li, Y. Yang, J. Liu, X. Zhou, and H. Lu, "Unsupervised feature selection using nonnegative spectral analysis," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 26, no. 1, 2012, pp. 1026–1032.

[16] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou, "L2, 1-norm regularized discriminative feature selection for unsupervised," in *Twenty-second international joint conference on artificial intelligence*, 2011.

[17] F. Nie, Z. Wang, L. Tian, R. Wang, and X. Li, "Subspace sparse discriminative feature selection," *IEEE transactions on cybernetics*, vol. 52, no. 6, pp. 4221–4233, 2020.

[18] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[19] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Transactions on neural networks*, vol. 5, no. 4, pp. 537–550, 1994.

[20] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proceedings of the 20th international conference on machine learning (ICML-03)*, 2003, pp. 856–863.

[21] M. Heidari, J. K. Sreedharan, G. Shamir, and W. Szpankowski, "Sufficiently informative and relevant features: An information-theoretic and fourier-based characterization," *IEEE Transactions on Information Theory*, 2022.

[22] D. Dua and C. Graff, "Uci machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml