Journal of Machine Learning Research 24 (2023) 1-60 Submitted 11/21; Revised 9/22; Published 1/23

The SKIM-FA Kernel: High-Dimensional Variable Selection and Nonlinear Interaction Discovery in Linear Time

Raj Agrawal R.AGRAWAL@MIT.EDU
Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139-4307, USA

Tamara Broderick TBRODERICK@MIT.EDU
Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139-4307, USA

Editor: Daniela Witten

Abstract

Many scientific problems require identifying a small set of covariates that are associated with a target response and estimating their effects. Often, these effects are nonlinear and include interactions, so linear and additive methods can lead to poor estimation and variable selection. Unfortunately, methods that simultaneously express sparsity, nonlinearity, and interactions are computationally intractable — with runtime at least quadratic in the number of covariates, and often worse. In the present work, we solve this computational bottleneck. We show that suitable interaction models have a kernel representation, namely there exists a "kernel trick" to perform variable selection and estimation in $O(\#\text{ covariates})$ time. Our resulting fit corresponds to a sparse orthogonal decomposition of the regression function in a Hilbert space (i.e., a functional ANOVA decomposition), where interaction effects represent all variation that cannot be explained by lower-order effects. On a variety of synthetic and real data sets, our approach outperforms existing methods used for large, high-dimensional data sets while remaining competitive (or being orders of magnitude faster) in runtime.

Keywords: functional ANOVA, interaction discovery, kernel ridge regression, nonlinear variable selection, sparse high-dimensional regression

1. Introduction

Many scientific and decision-making tasks require learning complex relationships between a set of p covariates and target response from N observed datapoints with $N \ll p$. For example, in genomics and precision medicine, researchers would like to identify a small set of genetic and environmental factors (out of potentially thousands or millions) associated with diseases and quantify their effects (Maher, 2008; Aschard, 2016; Slim et al., 2018; Greene et al., 2010). Estimating these effects can be challenging, however, without sufficiently flexible models. Blood sugar levels, for example, could vary sinusoidally with the time of day (e.g., depending on when an individual has breakfast, lunch, and dinner). In other instances, effects can be challenging to estimate due to multiplicative interactions. A particular drug could help individuals with certain genetic characteristics but harm others. To learn such nuances in our data for better decision-making, we need statistical methods that can model nonlinear and interaction effects. We also need computationally efficient methods that can scale to large-p settings. Unfortunately, as we detail below, existing methods suffer in at least one of these three categories.

Sparse linear regression methods (e.g., the Lasso) are typically fast but do not have the flexibility to learn nonlinear or interaction effects (Chen et al., 1998; Candes and Tao, 2007; Nakagawa et al., 2016). SpAM extends the Lasso to model nonlinear effects but assumes additive effects (Liu et al., 2008). Conversely, the hierarchical Lasso models interactions but assumes linearity, and its runtime scales quadratically with dimension (Bien et al., 2013). Recently, Agrawal et al. (2019) developed a kernel trick to learn interactions in time linear in dimension, but this method assumes linear effects. Black-box models, such as neural networks and random forests, often include interactions and nonlinear effects for the sake of prediction. However, it is unclear how to access the effects from the fitted prediction model.

The *hierarchical functional ANOVA* (Stone, 1994), which includes many of the models described above as special cases, provides a general framework to jointly model interactions and nonlinear effects through a variance decomposition of the regression function. As long as the response has finite variance and the covariates vary over a compact set, the functional ANOVA decomposition exists. Such a variance decomposition, which includes classical ANOVA decompositions of contingency tables and generalized additive models as special cases, has been widely used in applications due to desirable interpretability properties. For example, in genetic applications, biologists use ANOVA decompositions to isolate the marginal effects of particular genetic or environmental factors on disease in a population (Vitezica et al., 2013; Maher, 2008; Aschard, 2016). Unfortunately, existing functional ANOVA methods, which are primarily kernel-regression based, do not scale well with dimension (Gu and Wahba, 1993; Lin and Zhang, 2006; Gunn and Kandola, 2004); these methods use kernels that take $O(p^Q)$ time to evaluate, where Q equals the size of the highest order interaction. Hence, running kernel ridge regression for inference takes $O(p^Q N^2 + N^3)$ time.

Contributions. We consider two interconnected tasks: (1) high dimensional variable selection and (2) estimation of nonlinear additive and interaction effects. We define a new class of kernels called "model selection kernels" to simultaneously solve each of these tasks via kernel ridge regression. Model selection kernels have the flexibility to select a sparse subset of covariates that drive the response, and estimate nonlinear effects among the selected covariates. However, model selection kernels are computationally intractable to compute in general. Hence, we propose SKIM-FA kernels, a type of model selection kernel that also enjoys computational efficiency. We show how to compute SKIM-FA kernels in $O(pQ)$ time by exploiting special low-dimensional structure. We motivate this structure from the perspective of hierarchical Bayesian modeling. Then, we use equivalences between kernel ridge regression, Gaussian processes, and conjugate Bayesian regression to develop our efficient inference procedure.

Outline. We start by describing how to model nonlinear interaction effects and encode sparsity using hierarchical Bayesian modeling in Section 2. In Section 3, we define model selection kernels and develop two kernel tricks to perform inference more efficiently when the covariates are independent. Then, we extend our procedure to the general covariate case in Section 4. We defer implementation details of our final algorithm to Section 5. We conclude by discussing related work in Section 6 and benchmarking our method against other methods often used to model high-dimensional data in Section 7.

2. A Framework for Modeling Nonlinear Additive and Interaction Effects and Inducing Sparsity

Problem statement. Suppose we collect data $D = \{(x^{(n)}, y^{(n)})\}_{n=1}^{N}$ with covariates $x^{(n)} \in \mathbb{R}^p$ and continuous scalar responses $y^{(n)}$. We model $y^{(n)} = f^*(x^{(n)}) + \epsilon^{(n)}$, where $\epsilon^{(n)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2_{\text{noise}})$, $x^{(n)} \overset{\text{i.i.d.}}{\sim} \mu$, and the unknown regression function f^* belongs to some class of functions $\mathcal{H}$. Using only noisy realizations of f^*, we would like to identify which covariates f^* depends on[1] (i.e., perform variable selection), and recover main effects and interaction effects. For example, a biologist might seek to identify a small set of genes (out of tens of thousands of possible genes) associated with a disease — e.g., to design new gene-based drug targets. Understanding the relationship between the selected genes and disease response could help the biologist properly administer the drug.

To perform variable selection and estimation, existing methods often assume the majority of effects equal zero. Problematically, when there are interactions present, sparsity in effects does not guarantee that a sparse set of covariates is selected. For example, suppose we include all additive and pairwise interaction effects in our model. A method that selects p non-zero effects might be considered sparse in the effects since $p \ll p^2$. But the selected effects could correspond to p (or nearly p) selected covariates, so the burden of data collection is not reduced relative to the original problem; see Bien et al. (2013) for further motivation of sparsity in covariates, by contrast to sparsity in effects. To ensure sparsity in covariates, many interaction methods impose a "hierarchy" or "heredity" constraint (Bien et al., 2013; Radchenko and James, 2010; Haris et al., 2016). Such a constraint allows interactions to be present only among selected additive effects. If the additive effects are weak, then this constraint will lead to poor inference; see also our extended discussion in Appendix C.

To perform variable selection and estimation without requiring a heredity constraint, we use penalized regression:

$$\hat{f} = \arg\min_{f \in \mathcal{H}} \sum_{n=1}^{N} \mathcal{L}(y^{(n)}, f(x^{(n)})) + J(f), \tag{1}$$

where $\mathcal{L}(\cdot, \cdot)$ and $J(f)$ denote some loss function and penalty on model complexity, respectively. This paper focuses on four subproblems resulting from Eq. (1): (P1) picking $\mathcal{H}$ to model interactions, (P2) selecting $\mathcal{L}(\cdot, \cdot)$ and $J(f)$ to induce sparsity (i.e., to identify the small subset of covariates that influences the response), (P3) tractably solving Eq. (1) for our choice of sparsity-inducing $J(f)$, and (P4) efficiently reporting effects in $\hat{f}$.

2.1 Our Contributions: An Overview

We describe, at a high level, our solutions to subproblems P1 through P4, and what parts of our solutions are new. Our solutions to P3 and P4 are our core contributions.

[1] When the derivative exists, this set equals equals all covariates with non-zero derivatives; that is, all x_j with $j \in \{1, \dots, p\}$ such that $\|\partial f^* / \partial x_j\| \neq 0$. See also Corollary 1.

P1: Constructing $\mathcal{H}$. Our construction of $\mathcal{H}$ in Section 2.2 is based on Huang (1998). We use the hierarchical functional ANOVA introduced in Stone (1994) to make recovering interaction effects a well-defined inference task (i.e., statistically identifiable).

P2: Selecting the loss and penalty. We select the loss and penalty from a hierarchical Bayesian modeling point of view in Section 2.3. In particular, we choose the loss $\mathcal{L}$ to correspond to a negative log-likelihood of the data and the penalty $J(f)$ to correspond to a negative log prior of f. Existing sparse Bayesian interaction methods do not work at our level of generality. Nevertheless, our proposed class of priors is heavily influenced by existing sparse Bayesian interaction models.

P3: Solving Eq. (1)*.* We solve Eq. (1) in time linear in p by using two kernel tricks to (1) reduce the cost of modeling nonlinear functions and (2) avoid summing over a combinatorial number of interaction terms. The first kernel trick, described in Section 3, is based on the foundational smoothing spline ANOVA (SS-ANOVA) work by Gu and Wahba (1993). To make the connection to SS-ANOVA, we show that there exists a duality between our class of hierarchical models (see P2) and reproducing kernel Hilbert spaces induced by what we call *model selection kernels.* Our model selection kernels generalize the kernels used in Gu and Wahba (1993) by removing the requirement that all covariates be independent. Our second kernel trick, which allows us to avoid summing over a combinatorial number of interaction terms, applies to a subset of model selection kernels. We call this subset of kernels *SKIM-FA* kernels and prove that SKIM-FA kernels have desirable statistical properties from the hierarchical Bayesian modeling point of view.

P4: Reporting effects. For the case of independent covariates, we report effects using the procedure in Gu and Wahba (1993). Our new contribution, provided in Section 4, is developing an efficient algorithm to report effects for the non-independent case.

2.2 Interactions and Identifiability for Nonlinear Functions

We construct $\mathcal{H}$ in Eq. (1) by considering functions on $\mathbb{R}^p$ that can be written as a sum of lower-dimensional functions (i.e., interaction effects) that depend on at most Q covariates with $Q < p$. Our goal is to estimate these interaction effects. Unfortunately, as we detail below, such an expansion is not unique, and therefore not a valid target of inference. To make inference over $\mathcal{H}$ well-defined, we use the *hierarchical functional ANOVA* (Stone, 1994).

Modeling Interactions. Let $\mathcal{H} = \mathcal{H}_Q \coloneqq \bigoplus_{V:|V|\leq Q} \mathcal{H}_V$, where $\mathcal{H}_V$ belongs to the space of all square-integrable functions of x_V (with respect to the probability measure μ) and $V \subset [p] \coloneqq \{1, \cdots, p\}$. Then, for $f_\emptyset$ in the space of constant functions $\mathcal{H}_\emptyset = \{\theta : \theta \in \mathbb{R}\}$,

$$\begin{aligned}\bigoplus_{V:|V|\leq Q} \mathcal{H}_V &= \left\{ f : f = \sum_{V:|V|\leq Q} f_V(x_V),\ f_V \in \mathcal{H}_V \right\} \\ &= \left\{ f : f = f_\emptyset + \sum_{i=1}^{p} f_{\{i\}}(x_i) + \sum_{i<j}^{p} f_{\{i,j\}}(x_i, x_j) + \cdots + \sum_{V:|V|=Q} f_V(x_V) \right\}.\end{aligned} \tag{2}$$

Similar to additive models, $f_{\{i\}}(x_i)$ has the interpretation as the main or marginal effect of covariate x_i on y. Similarly, $f_{\{i,j\}}(x_i, x_j)$ has the interpretation as the two-way or pairwise effect of x_i and x_j on y. Unfortunately, the components in Eq. (2) are not identifiable without further constraints. For example, if $f^*(x) = f_{\{1\}}(x_1) + f_{\{2\}}(x_2) + f_{\{1,2\}}(x_1, x_2)$, then f^* also decomposes as $f_{\{1\}}(x_1) + [f_{\{2\}}(x_2) + 5] + [f_{\{1,2\}}(x_1, x_2) - 5]$.

Identifiability with the Functional ANOVA. To resolve identifiability issues, we construct a smaller space of functions $\mathcal{H}_V^o \subset \mathcal{H}_V$, where $\mathcal{H}_V^o$ includes only functions whose variation cannot be explained by lower-order effects of x_V:

$$\mathcal{H}_V^o = \{f_V \in \mathcal{H}_V : \forall A \subsetneq V,\ \forall f_A \in \mathcal{H}_A,\ \langle f_V, f_A \rangle_\mu = 0\}, \tag{3}$$

where $\langle \cdot, \cdot \rangle_\mu$ is an inner product on L^2. That is, $\langle f_A, f_B \rangle_\mu = \mathbb{E}_{x \sim \mu}[f_A(x_A) f_B(x_B)]$.

Theorem 1. *(Stone, 1994; Huang, 1998) Suppose $f \in \mathcal{H}_Q$ and μ is absolutely continuous with respect to Lebesgue measure. Further, suppose that the domain of functions in $\mathcal{H}_Q$ is $\mathcal{X}$, and $\mathcal{X}$ is a compact set of $\mathbb{R}^p$. Then, there exist (μ-almost everywhere) unique functions $f_V \in \mathcal{H}_V^o$ such that $f = \sum_{V:|V| \le Q} f_V$.*

Definition 1. Suppose $f = \sum_{V:|V| \le Q} f_V$ where $f_V \in \mathcal{H}_V^o$. Then, $\sum_{V:|V| \le Q} f_V$ is called the *functional ANOVA decomposition* of f with respect to μ.

In light of Theorem 1, we assume compactness throughout to have a well-defined target of inference (i.e., the functional ANOVA decomposition of f in Definition 1). By the orthogonality constraints in Eq. (3), the effect $f_{\{i,j\}}(x_i, x_j)$ in Definition 1 represents, for example, the variation that cannot be explained by 1D functions of x_i and x_j and an intercept. When the covariates are independent, then the signal variance decomposes as

$$\text{var}(f) = \text{var}(f_{\{\emptyset\}}) + \sum_i \text{var}(f_{\{i\}}) + \sum_{i,j} \text{var}(f_{\{i,j\}}) + \cdots \text{var}(f_{\{1,2,\cdots,p\}}(x_1, \cdots, x_p)), \tag{4}$$

where $\text{var}(f) = \langle f, f \rangle_\mu$. Hence, Eq. (4) allows us to *analyze* how the *variance* of the *function* is distributed across the interactions of different orders. Hence, the name functional analysis of variance or functional ANOVA. When all the covariates are categorical, then the functional ANOVA reduces to the classical ANOVA decomposition of a contingency table.

2.3 How to Achieve Sparsity for Nonlinear Functions

To complete our specification of Eq. (1), we still need to pick a loss and penalty function on $\mathcal{H}_Q$. We motivate our choice of loss and penalty from a Bayesian point of view. That is, we view $\mathcal{L}(\cdot, \cdot)$ as the negative log-likelihood function, $J(f)$ as the negative log prior on f, and $\hat{f}$ as the maximum a priori (MAP) estimate under our proposed Bayesian model.

Our loss. Since the noise terms are Gaussian (see "Problem Statement" at the start of Section 2), the negative log-likelihood is quadratic: $\mathcal{L}(y, f(x)) = (y - f(x))^2$ (i.e., squared-error loss).

Our penalty. We are primarily interested in the case when f^* is sparse, i.e., when f^* depends on a small number of covariates. So $J(f)$ should promote such sparsity. To that end, we first take a basis expansion of each component space, and then place a sparsity prior on the basis coefficients. We assume that for all $V \subset [p]$ and $1 \leq |V| \leq Q$, there exists a $B_V \in \mathbb{N} \cup \{\infty\}$ and feature map $\Phi_V : \mathbb{R}^{|V|} \mapsto \mathbb{R}^{B_V}$ such that the components of Φ_V form a basis of $\mathcal{H}_V^o$. Then, for any $f_V \in \mathcal{H}_V^o$, there exists $\Theta_V \in \mathbb{R}^{B_V}$ such that $f_V(x_V) = \Theta_V^T \Phi_V(x_V)$. Hence, if we can estimate Θ_V, we can estimate the functional ANOVA decomposition of f^* by Theorem 1.

To obtain a MAP estimate of Θ_V, we draw each $\Theta_V \sim \mathcal{N}(0, \theta_V \cdot I_{B_V})$, where $\theta_V \in \mathbb{R}$ is a non-negative auxiliary parameter drawn from a sparsity prior (e.g., a Laplace prior) and I_{B_V} denotes the $B_V \times B_V$ identity matrix; see Section 5 for our particular choice of prior. If $\theta := \{\theta_V\}$ is sparse, then we claim that $\{f_V\}$ is sparse. To understand why, suppose $\theta_V = 0$. Then, the prior variance of f_V equals 0. Hence, f_V will equal 0. Thus, a prior that induces sparsity in θ enables us to get sparsity in the number of effects selected. However, sparsity in *effects* does not automatically guarantee that a sparse subset of *covariates* is selected, as we discussed at the start of Section 2.

To get sparsity in covariates without requiring a heredity constraint, we draw θ from a hierarchical sparsity prior; see Section 3.2 for details. Since Φ_V is a basis of $\mathcal{H}_V^o$ and our prior on Θ_V has full support on $\mathbb{R}^{B_V}$, our choice of likelihood and prior allows us to model any $f \in \mathcal{H}_Q$ as summarized below:

$$\Theta_V \mid \theta_V \sim \mathcal{N}(0, \theta_V \cdot I_{B_V}),\ V \subset [p],\ |V| \leq Q,\ \theta_V \geq 0 \tag{5a}$$

$$y^{(n)} \mid x^{(n)}, \Theta, \sigma^2_{\text{noise}} \sim \mathcal{N}(f(x^{(n)}), \sigma^2_{\text{noise}}),\ f = \sum_{V:|V| \leq Q} \Theta_V^T \Phi_V(\cdot),\ n \in [N], \tag{5b}$$

where the likelihood in the first equation corresponds to $\exp(-J(f))$ and $\exp(-\mathcal{L}(y^{(n)}, f(x^{(n)})))$ corresponds to the likelihood in the last equation.[2] While other likelihoods and priors exist to model interactions and sparsity, many existing sparse Bayesian methods are instantiations of Eq. (5), and have desirable statistical shrinkage properties; see, for example, Wei et al. (2019); Curtis et al. (2014); Griffin and Brown (2017); Agrawal et al. (2019); Chipman (1996); George and McCulloch (1993). In the next section, we exploit the special Gaussian and interaction structure in Eq. (5) for faster inference.

3. Using Two Kernel Tricks to Reduce Computation Cost

In principle, we can analytically compute the MAP estimate of Θ_V in Eq. (5) (and hence solve Eq. (1) in closed-form); conditional on θ, Eq. (5) reduces to conjugate Bayesian regression. Unfortunately, unless p is very small or $Q = 1$, computing this closed-form solution is typically computationally intractable, for reasons we describe next. To remedy this computational

[2] While Eq. (5) has the flexibility to induce sparsity in both covariates and effects, it does not lead to sparsity in the basis expansion of a selected effect (e.g., if f_V is selected, then Θ_V will be a dense vector with probability one). Since our goal is not to learn a sparse representation of f_V, our choice of a Gaussian prior (or L_2 regularization) is not very limiting since irrelevant basis components will just be shrunk close to zero. However, if there are many irrelevant basis components in Φ_V, then a Laplace prior (or L_1 penalty) might be preferable. Other methods, such as sparse additive models, also use L_2 regularization to penalize the basis expansion coefficients (Liu et al., 2008).

intractability, we show how to make inference scale linearly with p by exploiting special model structure in Section 3.1 and Section 3.2.

Intractability of Conjugate Bayesian Regression. Our model in Eq. (5) has $B_Q \coloneqq \sum_{V:|V|\leq Q} B_V$ parameters. In general, computing the MAP estimate of these B_Q parameters requires inverting a $B_Q \times B_Q$ covariance matrix (Rasmussen and Williams, 2006, Chapter 2). So the computational cost of MAP inference scales as $O(B_Q^3 + NB_Q^2)$. B_Q may be prohibitively large for two reasons. First, B_Q is large if any basis-expansion size (i.e., any B_V) is large. For example, if $\mathcal{H}_V^o$ is infinite-dimensional (e.g., if $\mathcal{H}_V$ equals the space of all square-integrable functions of x_V), then $B_V = \infty$. Even if all the $\mathcal{H}_V^o$ are finite-dimensional (e.g., if $\mathcal{H}_V$ is generated from a finite polynomial basis), B_V typically grows exponentially with the size of $|V|$; see, e.g., Huang (1998). B_Q may also be large due to the combinatorial sum over interactions; even if all of the B_V equal 1, B_Q still has on the order of $O(p^Q)$ terms. Hence, without additional structure, the computation time for conjugate Bayesian regression is lower bounded by $\Omega(p^{3Q} + p^{2Q}N)$. Fortunately, due to unique structure in our problem, we show how to avoid the cost of explicitly generating the basis expansion ("Trick 1" in Section 3.1), and summing over all $O(p^Q)$ interactions ("Trick 2" in Section 3.2). In what follows, we assume θ is fixed. Then, we show how to estimate θ in Section 5.

3.1 Trick 1: Represent and Access Sparsity Without Basis Expansion

We show how to remove the computational dependence on the size of B_V through a kernel trick. Our kernel generalizes the one used in Gu and Wahba (1993), which assumes independent covariates, to the case of general covariate distributions. In order to prove the existence of a kernel trick, we make the following assumption:

Assumption 1. Each $\mathcal{H}_V$ is a reproducing kernel Hilbert space (RKHS).

Given that there exist reproducing kernels that can approximate any continuous function arbitrarily well, Assumption 1 is a mild condition (Micchelli et al., 2006). The non-trivial part is proving the existence of a kernel to induce $\mathcal{H}_V^o$, which is not immediate due to the orthogonality constraints in Eq. (3).

Proposition 1. *(existence of a kernel trick) Under Assumption 1, there exists a positive-definite kernel k_V such that $k_V(x, \tilde{x}) = \langle \Phi_V(x), \Phi_V(\tilde{x}) \rangle$, where the components of $\Phi_V \in \mathbb{R}^{B_V}$ form a countable basis of $\mathcal{H}_V^o$.*

We prove Proposition 1 in Appendix B.1. In Section 3.2, we show how to efficiently evaluate k_V without explicitly computing the feature maps. In light of Proposition 1, we introduce *model selection kernels* to rewrite the model in Eq. (5) as a Gaussian process. We then show how this reparametrization allows us to perform inference more efficiently.

Definition 2. A kernel k_θ is a *model selection kernel* if it can be written as $\sum_{V:|V|\leq Q} \theta_V k_V$, where k_V is the reproducing kernel for $\mathcal{H}_V^o$ and $k_\emptyset(x, \tilde{x}) = 1$ (i.e., the kernel $k_\emptyset$ induces the space of constant functions $\mathcal{H}_\emptyset$).

Lemma 1. *Let $\{y^{(n)}\}_{n=1}^N$ be generated according to the model in Eq. (5). Suppose that $\{\tilde{y}^{(n)}\}_{n=1}^N$ is generated according to the model below:*

$$\begin{aligned} f &\sim GP(0, k_\theta) \\ \tilde{y}^{(n)} \mid f, x^{(n)} &\sim \mathcal{N}(f(x^{(n)}), \sigma^2_{noise}), \quad n \in [N], \end{aligned} \tag{6}$$

where k_θ is defined in Definition 2. Then, $\{y^{(n)}\}_{n=1}^N \mid X \overset{d}{=} \{\tilde{y}^{(n)}\}_{n=1}^N \mid X$, where $\overset{d}{=}$ denotes equality in distribution.

Based on the reparametrization in Lemma 1 (see Appendix B.4 for the proof), $J(f)$ equals the penalty induced by the kernel k_θ. Hence, the solution to Eq. (1) reduces to kernel ridge regression (or equivalently equals the posterior predictive mean of the Gaussian process) by Rasmussen and Williams (2006, Chapter 2):

$$\hat{f}(x) = \bar{f}_\theta(x) \coloneqq \sum_{n=1}^N \hat{\alpha}_n k_\theta(x_n, x), \quad \hat{\alpha} = (K_\theta + \sigma^2_{\text{noise}} I_{N\times N})^{-1} Y, \tag{7}$$

where Y is a column vector with nth component $Y_n = y^{(n)}$ and $[K_\theta]_{nm} = k_\theta(x^{(n)}, x^{(m)})$.

Unlike the "weight-space view" in Section 2.3 where $f_V = \Theta_V^T \Phi_V(\cdot)$, it is not clear how to actually recover the effects f_V from the prediction function $\bar{f}_\theta$. For general kernels, accessing f_V (and consequently computing the functional ANOVA of $\bar{f}_\theta$) lacks an analytical form. Fortunately, we can easily recover f_V from $\bar{f}_\theta$ for model selection kernels:

Lemma 2. *Let k_θ be a model selection kernel and $f^{(M)}(x) = \sum_{m=1}^M \alpha_m k_\theta(x_m, x)$ for $\alpha_m \in \mathbb{R}$ and $x_m \in \mathbb{R}^p$. Then, $f^{(M)}(x) = \sum_{V:|V|\le Q} f_V$, where $f_V = \theta_V \sum_{m=1}^M \alpha_m k_V(x_m, x) \in \mathcal{H}_V^o$.*

It follows from Lemma 2 that model selection kernels enable easy variable selection; we just need to examine the sparsity pattern of θ. For general kernels, we would need to search over the entire domain of the fitted regression function (a p-dimensional space) to perform variable selection.

Corollary 1. *(nonlinear variable selection) Suppose $f^{(M)}(x) = \sum_{m=1}^M \alpha_m k_\theta(x_m, x)$. Then, $f^{(M)}(x)$ functionally depends on the set of covariates $\{i : \exists V \subset [p], i \in V \text{ s.t. } \theta_V \neq 0\}$.*

We have avoided the cost of generating the basis expansion to solve Eq. (1), but Eq. (7) is still computationally intractable; k_θ sums over $O(p^Q)$ kernels. Hence, the cost to compute the kernel matrix K_θ and invert $(K_\theta + \lambda I_{N\times N})$ takes $O(N^2 p^Q)$ and $O(N^3)$ time, respectively.

3.2 Trick 2: A Recursion to Avoid a Combinatorially Large Summation Over Interactions Given Covariate Independence

We show how to compute k_θ in $O(pQ)$ time (and hence solve Eq. (7) in $O(pQN^2 + N^3)$ time) for a particular subset of model selection kernels that we call *SKIM-FA* kernels. In what follows, we start by motivating SKIM-FA kernels from the hierarchical Bayesian characterization of model selection kernels in Eq. (5). We show that, when the covariates are independent, we can compute SKIM-FA kernels much more efficiently using a second kernel trick. In Section 4, we generalize to the non-independent covariate case by building on the procedure described in this section.

The Sparse Kernel Interaction Model for Functional ANOVA (SKIM-FA). To motivate SKIM-FA, suppose for the moment we know what covariates f functionally depends on. Let the binary vector $\kappa \in \{0,1\}^p$ encode this knowledge, where $\kappa_j = 0$ if and only if f does not depend on covariate x_j. Then, $f_V = 0$ if there there exists some $j \in V$ such that $\kappa_j = 0$. Equivalently, it suffices to check that $\prod_{j \in V} \kappa_j = 0$. Hence, if we knew κ, we should adjust our prior variance for f_V from θ_V to $\theta_V \prod_{j \in V} \kappa_j$. This update to the prior enforces support on only selected covariates, and is the same as fitting the model without the irrelevant covariates included. Since we do not know κ in advance, however, we treat κ as learnable parameter, and use it to (1) induce sparsity in effects via the product structure above, and (2) perform variable selection. We propose the following prior on Θ_V in Eq. (5), which generalizes the prior for linear pairwise interaction models in Agrawal et al. (2019):

$$\Theta_V \mid \eta, \kappa \sim \mathcal{N}\left(0, \eta_{|V|}^2 \prod_{i \in V} \kappa_i^2 \cdot I_{B_V \times B_V}\right), \tag{8}$$

for non-negative random vectors $\kappa \in \mathbb{R}_+^p$ and $\eta \in \mathbb{R}_+^{Q+1}$. We do not restrict $\kappa \in \{0,1\}^p$ so that we can leverage gradient-based techniques for learning κ more easily; see Section 5 for details.

SKIM-FA interpretation. In Eq. (8), $\eta_{|V|}^2$ quantifies the overall strength of $|V|$-way interactions by modifying the prior variance of all effects of order $|V|$. Hence, $\eta_{|V|}$ plays an analogous role to the "global scale" in sparse Bayesian linear models; see, for example, Piironen and Vehtari (2017); Carvalho et al. (2009); Agrawal et al. (2019). κ_i plays the role of a "variable importance" measure for covariate x_i by affecting the prior variance of all effects involving covariate x_i. Hence, if it turns out an effect involving x_i is strong, the posterior of κ_i will place high probability at large values (i.e., indicating that covariate x_i has high "importance"). Notice that if $\kappa_i = 0$, then the prior variance of Θ_V equals 0 whenever $i \in V$. Consequently, all effects involving x_i will equal 0. Hence, we can perform variable selection in $O(p)$ time by just examining the sparsity pattern of κ instead of in $O(p^Q)$ time using Corollary 1. In Section 5, we show how we select our sparsity prior on κ. Finally, note that while we added more structure to the prior, we have not lost modeling flexibility; as long as $\mathbb{P}(\kappa_i > 0)$ does not equal 0, then the prior variance of Θ_V will be non-zero.[3] Hence, our prior will have support on all of $\mathcal{H}_Q$.

Definition 3. A *SKIM-FA kernel* is a model selection kernel that can be written as

$$k_{\text{SKIM-FA}}(x, \tilde{x}) = \sum_{V:|V| \leq Q} \left[\eta_{|V|}^2 \prod_{i \in V} \kappa_i^2\right] k_V(x, \tilde{x}).$$

[3] SKIM-FA considers all interactions of order Q among selected covariates (i.e., does not assume sparsity in interactions between selected covariates). Specifically, suppose $\{x_1, x_2, x_3\}$ are selected and $Q = 2$. Then, SKIM-FA considers all additive and pairwise effects between the first three covariates. If $f_{\{1,2\}} = 0$, for example, then the posterior of $f_{\{1,2\}}$ is non-zero because the prior on $\Theta_{\{1,2\}}$ is drawn from a Gaussian distribution with non-zero variance. However, as N increases, the posterior of $f_{\{1,2\}}$ will be close to 0.

for some $\kappa \in \mathbb{R}^p$ and $\eta \in \mathbb{R}^{Q+1}$.

Proposition 2. *For a SKIM-FA kernel, Eq.* (5a) *can be replaced by Eq.* (8) *in Lemma 1.*

Proof. Set $\theta_V = \eta_{|V|}^2 \prod_{i \in V} \kappa_i^2$ in Lemma 1. □

Efficient evaluation of SKIM-FA kernels. Recall that k_i is the reproducing kernel for $\mathcal{H}_i^o$. Suppose, for the moment, that the reproducing kernel k_V for $\mathcal{H}_V^o$ equals $\prod_{i \in V} k_i$; we will shortly show that this condition holds when the covariates are independent. Then, by Theorem 2 and Corollary 2 below, we can compute SKIM-FA kernels orders of magnitude faster by not explicitly summing over all $O(p^Q)$ interactions in Definition 3.

Theorem 2. *Suppose $k_V(x, \tilde{x}) = \prod_{i \in V} k_i(x_i, \tilde{x}_i)$. Then,*

$$\begin{aligned} k_{\mathrm{SKIM-FA}}(x, \tilde{x}) &= \sum_{q=1}^{Q} \eta_q^2 \bar{k}_q(x, \tilde{x}) \quad \text{s.t.} \\ \bar{k}_q(x, \tilde{x}) &= \frac{1}{q} \sum_{s=1}^{q} (-1)^{s+1} \bar{k}_{q-s}(x, \tilde{x}) k^s(x, \tilde{x}), \quad \bar{k}_0(x, \tilde{x}) = 1, \\ k^s(x, \tilde{x}) &= \sum_{i=1}^{p} \kappa_i^{2s} [k_i(x_i, \tilde{x}_i)]^s. \end{aligned} \tag{9}$$

As we show in Appendix B.5, the key to proving Theorem 2 is an old recursive kernel formula provided in Vapnik (1995, pg. 199). From Theorem 2, we have two corollaries. The first requires a short inductive argument; see Appendix B.6. The second follows immediately by setting $Q = 2$ into Eq. (9).

Corollary 2. *$k_{\mathrm{SKIM-FA}}(x, \tilde{x})$ takes $O(pQ)$ time to evaluate on a pair of points.*

Corollary 3. *Suppose $Q = 2$. Then, $k_{SKIM\text{-}FA}(x, \tilde{x})$ equals*

$$0.5\eta_2^2 \left[\left(\sum_{i=1}^{p} \kappa_i^2 k_i(x_i, \tilde{x}_i) \right)^2 - \sum_{i=1}^{p} \kappa_i^4 [k_i(x_i, \tilde{x}_i)]^2 \right] + \eta_1^2 \sum_{i=1}^{p} \kappa_i^2 k_i(x_i, \tilde{x}_i) + \eta_0^2. \tag{10}$$

To see why "trick 2" in Eq. (9) indeed acts as another kernel trick, consider the linear interaction case when $\mathcal{H}_Q$ consists of interactions of the form $\prod_{i \in V} x_i$. Suppose further that κ and η are equal to the ones vector. Then, $k_{\mathrm{SKIM-FA}}(x, \tilde{x}) = \sum_{V : |V| \leq Q} \prod_{i \in V} x_i \tilde{x}_i$, which explicitly generates and sums over the interactions $\prod_{i \in V} x_i$. However, it is well known that polynomial kernels implicitly generate interactions, and hence can be used instead to avoid summing over all interactions. The core idea in Eq. (9) is similar; the kernel k^s, which sums over kernels raised to the s power in Eq. (9), implicitly generates interactions of order equal to s just like a polynomial kernel. However, instead of generating interactions of the form $\prod_{i \in V} x_i$, k^s operates on one-dimensional kernels k_i, where its product with $\bar{k}_{q-s}$ generates interactions of the form $\prod_{i \in V} k_i$. Since $k_V = \prod_{i \in V} k_i$ by assumption, these "interactions" of kernels span $\mathcal{H}_V^o$ by the product property of kernels.

To understand when $k_V = \prod_{i \in V} k_i$ in Theorem 2 holds, we provide sufficient conditions based a result from Gu and Wahba (1993). We leave our construction of k_i to Appendix D.

Assumption 2. (Tensor product space) For all $V \subset [p]$ and $1 \leq |V| \leq Q$, $\mathcal{H}_V = \bigotimes_{i \in V} \mathcal{H}_i$.

Proposition 3. *(Gu and Wahba, 1993) Suppose $\mu = \mu_\otimes$, where $\mu_\otimes(x) := \mu_1(x_1) \otimes \mu_2(x_2) \cdots \otimes \mu_p(x_p)$ and μ_j is the marginal distribution of x_j. Then, under Assumption 1 and Assumption 2, $k_V = \prod_{i \in V} k_i$.*

Since any Hilbert space of square-integrable functions of x_V can be approximated arbitrarily well by taking tensor products of one-dimensional Hilbert spaces by Stone (1994); Huang (1998), Assumption 2 is a mild assumption. The more problematic assumption is that all covariates are independent (i.e., that $\mu = \mu_\otimes$).

4. How to Get Sparsity, Interactions, and Fast Inference When Covariates Are Dependent

Here we extend to the general μ case. We start by motivating this extension in Section 4.1. In Section 4.2, we develop a change-of-basis formula to take the functional ANOVA decomposition of $\bar{f}_\theta$ with respect to $\mu_\otimes$ to one with respect to μ. In this section, we assume $Q = 2$. We defer the general Q case to Appendix F.

4.1 Practical Problems From Assuming Independent Covariates

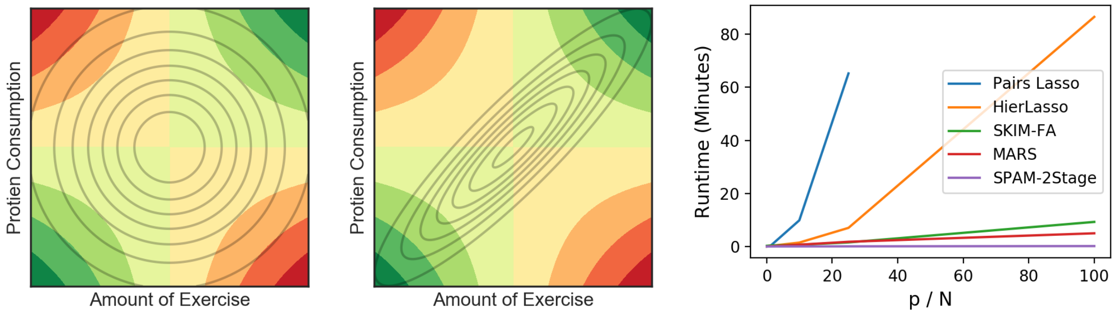
Since $\mu_\otimes = \mu_1(x_1) \otimes \mu_2(x_2) \cdots \otimes \mu_p(x_p)$, $\mu_\otimes$ has the same 1D marginal distributions as μ. Nevertheless, we prove that the functional ANOVA decomposition of an $f \in \mathcal{H}$ can be *arbitrarily* different depending on if $\mu_\otimes$ or μ is selected. We prove this claim by showing something stronger, namely that the intercepts between two functional ANOVA decompositions can be arbitrarily far apart (see Appendix B.3 for the proof of Proposition 4).

Proposition 4. *For any $\Delta > 0$, there exists a probability measure μ and square-integrable f such that the relative difference*

$$\frac{|f_\emptyset^\mu - f_\emptyset^{\mu_\otimes}|}{|f_\emptyset^\mu|} > \Delta, \quad \textit{where} \quad f_\emptyset^\mu = \mathbb{E}_\mu f(X) \quad \textit{and} \quad f_\emptyset^{\mu_\otimes} = \mathbb{E}_{\mu_\otimes} f(X).$$

To build intuition for Proposition 4, and motivate using μ instead of $\mu_\otimes$ to compute the functional ANOVA decomposition of $\bar{f}_\theta$, consider the following toy example.

Example 1. Suppose $f^*(x_1, x_2) = 100x_1x_2 - 50$, where x_1 could represent exercise, x_2 protein consumption, and $f^*(x_1, x_2)$ the expected percent decrease in body mass index after taking a weight-loss drug for an individual who consumes x_1 grams of protein and exercises x_2 minutes per week. Suppose exercise and protein consumption are positively correlated and that μ corresponds to a multivariate Gaussian distribution with mean zero, unit variance, and correlation equal to 0.9. Then, $\mu_\otimes$ corresponds to a multivariate Gaussian distribution with mean zero, unit covariance but correlation equal to 0. Suppose we report the intercept $f_\emptyset$ to summarize the typical decrease in body mass index in a population of people who might take the weight-loss drug (e.g., after drug approval). In the functional ANOVA decomposition of f^* with respect to μ, $f_\emptyset = \mathbb{E}_\mu[f^*] = \mathbb{E}_\mu[f^* + \epsilon] = \mathbb{E}_\mu[y]$= 40. Hence, this intercept says that, on average, people in this population should decrease their body mass index by 40% if they take the drug. If we instead use $\mu_\otimes$, then $f_\emptyset = \mathbb{E}_{\mu_\otimes}[f^*] = -50 \neq \mathbb{E}_\mu[f^*]$, suggesting



(a) Product measure (b) Covariate measure (c) Runtimes on Simulated Data

Figure 1: *Left and middle*: the colors denote the contour plot of the function $f^*(x_1, x_2) = 100x_1x_2 - 50$. Darker green indicates larger positive values while darker red indicates larger negative values. The gray solid lines in the left and right hand figures represent the density contours of $\mu_{\otimes}$ and μ in Example 1, respectively. *Right*: runtime comparisons of different methods as p/N increases; see Section 7 for details.

that the drug *increases* body mass index. In the $\mu_{\otimes}$ case, it is not clear how to interpret the intercept; $\mu_{\otimes}$ averages the regression surface f^* over individuals who rarely occur in the actual population (e.g., those who exercise very frequently but do not consume much protein); see also Fig. 1a and Fig. 1b for a visualization.

4.2 A Change of Basis to Handle Covariate Dependence

We generalize to the non-independent case through a change-of-basis formula provided in Theorem 3. Our formula allows us to re-express the effects estimated using the kernel in Section 3.2, which assumes independent covariates, to one with respect to the actual distribution μ. Our idea is similar to ideas in numerical linear algebra; we use one parameterization of a vector space, in our case the space of functions $\mathcal{H}_Q$, that makes computation "nice." Once we finish computation in the "nice" parameterization, we use a change-of-basis formula to report the actual quantity we care about in the original parameterization of the space, namely reporting the functional ANOVA decomposition of our fit $\bar{f}_\theta$ with respect to μ.

To make this idea mathematically precise, suppose we can write $\mathcal{H}_Q$ using two different parameterizations, one that uses $\mu_{\otimes}$ in Eq. (3) (denoted as $\mathcal{H}^o_{V,\mu_{\otimes}}$) and the other that uses μ in Eq. (3) (denoted as $\mathcal{H}^o_{V,\mu}$). Then,

$$\mathcal{H}_Q = \underbrace{\bigoplus_{V:|V|\leq Q} \mathcal{H}^o_{V,\mu_{\otimes}}}_{\text{(a)}} = \underbrace{\bigoplus_{V:|V|\leq Q} \mathcal{H}^o_{V,\mu}}_{\text{(b)}} . \tag{11}$$

If these equalities indeed hold, then we can use Theorem 2 to estimate f^* in $O(pQN^2 + N^3)$ time. Hence, it suffices to show how to take this estimate of f^* and compute its functional ANOVA decomposition with respect to μ instead of $\mu_{\otimes}$ (i.e., move from the parameterization in Eq. (11)(a) to the one in Eq. (11)(b)). We show how to compute this change-of-basis when all the $\mathcal{H}_{\{i\}}$ are finite-dimensional.

Assumption 3. For all $i \in [p]$ there exists a $B_i < \infty$ and linearly independent set of continuous functions $\{\phi_{ib}\}_{b=1}^{B_i}$ such that $\mathcal{H}_{\{i\}} = \text{span}\{1, \phi_{i1}, \cdots, \phi_{iB_i}\}$ and $\Phi_i = [\phi_{i1}, \cdots, \phi_{iB_i}]^T$.

Assumption 3 is a mild condition since we can approximate any function arbitrarily well by setting B_i sufficiently large given that $\mathcal{H}_{\{i\}}$ is separable; see Huang (1998) for rates of convergence for different finite-basis approximations. Under this assumption, Lemma 3 implies that $\mathcal{H}^o_{V,\mu_\otimes} = \mathcal{H}^o_{V,\mu}$. Hence, a change-of-basis formula exists. We provide the change-of-basis formula for $Q = 2$ in Theorem 3.

Lemma 3. *Under Assumption 2, Assumption 3, and compactness of the domain of f, any $f \in \mathcal{H}$ is square-integrable with respect to any probability measure.*

Theorem 3. *Suppose $Q = 2$ and that Assumptions 2 and 3 hold. For $f \in \mathcal{H}$, let*

$$\begin{aligned} f &= f^{\mu_\otimes}_{\emptyset} + \sum_{i=1}^{p} f^{\mu_\otimes}_{\{i\}} + \sum_{i,j=1}^{p} f^{\mu_\otimes}_{\{i,j\}} \\ &= f^{\mu}_{\emptyset} + \sum_{i=1}^{p} f^{\mu}_{\{i\}} + \sum_{i,j=1}^{p} f^{\mu}_{\{i,j\}} \end{aligned}$$

be the functional ANOVA decompositions of f with respect to $\mu_\otimes$ and μ, respectively. Then, there exist unique coefficients, $\Psi^i_{ij} \in \mathbb{R}^{1\times B_i}$, $\Psi^j_{ij} \in \mathbb{R}^{1\times B_j}$, $\Psi^0_{ij} \in \mathbb{R}$, such that

$$\begin{aligned} f^{\mu}_{\{i,j\}}(x_i, x_j) &= f^{\mu_\otimes}_{\{i,j\}}(x_i, x_j) - [\Psi^i_{ij}\Phi_i(x_i) + \Psi^j_{ij}\Phi_j(x_j) + \Psi^0_{ij}] \\ f^{\mu}_{\{i\}}(x_i) &= f^{\mu_\otimes}_{\{i\}}(x_i) + \sum_{p \geq j > i} \Psi^i_{ij}\Phi_i(x_i) + \sum_{1 \leq j < i} \Psi^i_{ji}\Phi_i(x_i) \\ f^{\mu}_{\emptyset} &= f^{\mu_\otimes}_{\emptyset} + \sum_{1 \leq i < j \leq p} \Psi^0_{ij}, \end{aligned} \tag{12}$$

where Φ_i denotes the (finite-dimensional) feature map in Definition 2.

We prove Theorem 3 in Appendix B.9. By Corollary 2, we can estimate $f^{\mu_\otimes}_{\{i\}}(x_i)$ and $f^{\mu_\otimes}_{\{i,j\}}(x_i, x_j)$ in time linear in p. Hence, it remains to show how we can actually compute each Ψ^i_{ij} in Theorem 3. In Section 5, we show how to estimate Ψ^i_{ij} arbitrarily well using a Monte Carlo approach.

5. Final Algorithm and Implementation Details

We start by describing and motivating our choice of sparsity prior on κ. Then, we show how we fit κ and η using cross-validation and our computational tools in Section 3. We conclude by showing how we compute Ψ^i_{ij} in Theorem 3 via Monte Carlo.

Our sparsity inducing prior on κ. To induce sparsity in κ for variable selection, we pick a prior on κ_i that equals the mixture of a discrete point mass at 0 and a Uniform(0, 1) random variable. Similar to a *spike-and-slab* prior (George and McCulloch, 1993), the point mass at 0 allows us to achieve exact sparsity. Unlike a spike-and-slab prior, however, we construct our prior so that we can still take gradients (and hence use continuous optimization techniques

Algorithm 1 Learn SKIM-FA Kernel Hyperparameters and Kernel Ridge Weights

```
procedure LearnHyperParams($M$, $\gamma$, $T$, $u_{\text{init}}$, $c$) where $M$ equals the cross-validation fold size, $\gamma$ the learning rate, $T$ the number of gradient descent steps, $u_{\text{init}}$ initializer for $\tilde{U}$, and $c$ the value selected in Eq. (13)
    Initialize $\tilde{U}^{(0)} = (u_{\text{init}}, \cdots, u_{\text{init}}) \in \mathbb{R}^p$
    Initialize $\eta = (1, \cdots, 1) \in \mathbb{R}^{Q+1}$    ▷ These are the global scale parameters in Eq. (8)
    $\sigma_{\text{noise}}^{(0)} = \sqrt{0.5\text{var}(Y)}$    ▷ Initialize noise variance as half of the response variance
    $\tau^{(0)} = (\tilde{U}^{(0)}, \eta^{(0)}, \sigma_{\text{noise}}^{(0)})$    ▷ Collect all parameters into a single vector
    for $t \in 1 : T$ do    ▷ Make $T$ gradient step updates
        Sample $A \sim \pi$    ▷ $\pi$ is uniform distribution over all $N - M$ subsets of $[N]$
        Collect $N - M$ covariates in $X_A \in \mathbb{R}^{(N-M)\times p}$ and responses in $Y_A \in \mathbb{R}^{(N-M)}$
        for $i \in 1 : p$ do
            $U_i^{(t-1)} = [\tilde{U}_i^{(t-1)}]^2 / \left([\tilde{U}_i^{(t-1)}]^2 + 1\right)$
            $\kappa_i^{(t-1)} = \max\left(U_i^{(t-1)} - c, 0\right)$
        end for
        Compute kernel matrix $K_\tau^A \in \mathbb{R}^{(N-M)\times(N-M)}$, where $[K_\tau^A]_{ij} = k_{\text{SKIM-FA}}([X_A]_i, [X_A]_j)$ via Eq. (9) and $[X_A]_i, [X_A]_j \in \mathbb{R}^p$
        Let $f_A$ equal the solution of Eq. (7) with $\lambda = [\sigma_{\text{noise}}^{(t)}]^2$, $K = K_\tau^A$, $Y = Y_A$
        $L = \frac{1}{M}\sum_{n\in[N]\setminus A}(y^{(n)} - f_A(x^{(n)}))^2$    ▷ Cross-validation loss in Eq. (14)
        $\tau^{(t)} = \tau^{(t-1)} - \gamma\nabla_{\tau^{(t-1)}} L$    ▷ Gradient update to parameters via autodiff library
    end for
    Compute $\alpha^{(T)}$, the kernel ridge regression weights found by solving Eq. (7) using all $N$ datapoints with SKIM-FA hyperparameters equal to $\kappa^{(T)}, \eta^{(T)}, \sigma_{\text{noise}}^{(T)}$
    return $\kappa^{(T)}, \eta^{(T)}, \sigma_{\text{noise}}^{(T)}, \alpha^{(T)}$
end procedure
```

like gradient descent). Our construction involves introducing another random variable U_i so that

$$\kappa_i = \frac{1}{1-c}\max(U_i - c, 0),\ U_i \sim \text{Uniform}(0,1). \tag{13}$$

Then, $\mathbb{P}(\kappa_i = 0) = c$. Otherwise, with probability $1 - c$, $\kappa_i \sim \text{Uniform}(0,1)$. Hence, c plays a similar role as a prior inclusion probability in a spike-and-slab prior. Since the gradient of κ_i equals 0 when $U_i < c$, this zero gradient property is key for inducing sparsity; see our proof of Proposition 5.[4]

Cross-validation loss and optimization. Given the empirical success of cross-validation and its use in other functional ANOVA methods (e.g., as in Gu and Wahba (1993); Lin and Zhang (2006)), we also use cross-validation to fit the SKIM-FA kernel hyperparameters κ and η. Specifically, we would like to pick $U, \eta, \sigma_{\text{noise}}^2$ (where $\kappa_i = \frac{1}{1-c}\max(U_i - c, 0)$) by minimizing

[4] At $U_i = c$, the derivative of $\max(U_i - c, 0)$ is undefined. Since $\max(U_i - c, 0)$ is a convex function, the set of all subgradients at $U_i = c$ is $[0, 1]$. We let the subgradient equal 0 at $U_i = c$.

a leave-M-out cross validation loss:

$$
\begin{aligned}
L(U, \eta, \sigma^2_{\text{noise}}) &= \frac{1}{\binom{N}{M}} \sum_{\substack{A: A \subset [N] \\ |A| = N-M}} \left[\frac{1}{M} \sum_{m \in A} (y^{(m)} - \bar{f}_A(x^{(m)}))^2 \right] \\
&= \mathbb{E}_{A \sim \pi} \left[\frac{1}{M} \sum_{m \in [N] \setminus A} (y^{(m)} - \bar{f}_A(x^{(m)}))^2 \right],
\end{aligned}
\tag{14}
$$

where $\bar{f}_A$ equals the kernel ridge regression fit in Eq. (7) using the subset of datapoints in A and π equals the uniform distribution over all $N - M$ sized subsets A of $[N]$.

Since the gradient of $L(U, \eta, \sigma^2_{\text{noise}})$ exists when all $U_i \neq c$, and the subgradient when some $U_i = c$, we can minimize Eq. (14) using gradient descent. However, this loss is computationally intensive; we need to solve Eq. (7) $\binom{N}{M}$ times in order to take a single gradient descent step. Instead, we approximate Eq. (14) by using stochastic gradient descent. Specifically, we randomly draw a single A from π in Eq. (14) and use the mean-squared prediction error of $\bar{f}_A$ to estimate Eq. (14). Then, this estimate leads to an unbiased estimate of Eq. (14), and hence an unbiased estimate of the gradient of $L(U, \eta, \sigma^2_{\text{noise}})$. We summarize our full procedure in Algorithm 1, and prove that it leads to sparsity below.[5]

Proposition 5. *Suppose $\kappa_i^{(t)} = 0$ at some iteration t in Algorithm 1. Then, for all subsequent iterations $t' \geq t$, $\kappa_i^{(t')} = 0$.*

Based on Proposition 5, we may view Algorithm 1 as a gradient-based analogue of backward stepwise regression; we start with the model that includes all covariates by initializing all $U_i > c$ (and consequently all $\kappa_i > 0$). Then, we keep pruning off covariates the longer we run gradient descent. We demonstrate empirically in Section 7 that the actual data-generating covariates remain while the irrelevant covariates get pruned off. Once we have found the kernel hyperparameters from Algorithm 1, Algorithm 2 and Algorithm 3 show how to perform variable selection and recover the effects, respectively. Both Algorithm 2 and Algorithm 3 follow directly from Corollary 1 and Lemma 2. In Appendix E, we discuss additional algorithmic details such as how to select c in Algorithm 1.

Estimating Ψ^i_{ij} for change-of-basis formula in Theorem 3. Our change-of-basis formula in Eq. (12) requires computing Ψ^i_{ij}. As we show in our proof of Eq. (12), Ψ^i_{ij} has the interpretation as the basis coefficients associated with an L^2 projection. Since this L^2 projection requires a high-dimensional integration, we use Monte Carlo to estimate Ψ^i_{ij} in Algorithm 4. We prove that our Monte Carlo estimate converges to the true projection coefficients in Proposition 6 below.

Proposition 6. *Let $W \to \infty$ in Algorithm 4. Then, the components returned from Algorithm 4 converge to the decomposition in Eq.* (12).

[5] Note that in Algorithm 1 we do not minimize over U but instead over $\tilde{U}$, where $U_i = \frac{\tilde{U}_i^2}{\tilde{U}_i^2 + 1}$. Since the range of $\frac{\tilde{U}_i^2}{\tilde{U}_i^2 + 1}$ equals $(0, 1)$ when $\tilde{U}_i$ varies over all of $\mathbb{R}$, we can optimize $\tilde{U}_i$ over an unconstrained domain. Since we only care about estimating the κ_i, it does not matter that U_i is not a 1-1 function of $\tilde{U}_i$.

Algorithm 2 SKIM-FA Variable Selection

```
procedure VarSelect($\kappa$)
    return $\{i : \kappa_i \neq 0\}$
end procedure
```

Algorithm 3 Estimated functional ANOVA effect $\bar{f}_V$ of $\bar{f}_\theta$ with respect to $\mu_\otimes$

```
procedure OrthEffects($V$, $\alpha$, $\kappa$, $\eta$, $\alpha$)
    $\theta_V = \eta^2_{|V|} \prod_{i \in V} \kappa_i^2$
    return $\bar{f}_V(\cdot) = \theta_V \sum_{n=1}^N \alpha_n k_V(x^{(n)}, \cdot)$
end procedure
```

6. Related Work

Below we compare SKIM-FA to existing functional ANOVA methods, and our previous work for the linear interaction case. We continue our literature review in Appendix C, where we also contrast with further methods used for interaction discovery.

Comparison with existing functional ANOVA methods. The foundational work by Gu and Wahba (1993) used a type of model selection kernel to estimate the functional ANOVA decomposition of f^* with splines. Since the method in Gu and Wahba (1993) does not lead to sparsity, Gunn and Kandola (2004); Lin and Zhang (2006) put an L_1 penalty on θ to achieve sparsity, similar to *multiple kernel learning* techniques (Lanckriet et al., 2004a). Adding an L_1 penalty does not lead to an analytical solution nor a convex optimization problem. Hence, Gunn and Kandola (2004); Lin and Zhang (2006) alternate between minimizing θ and recomputing $\bar{f}_\theta$, similar to Algorithm 1. Other approaches use cross-validation and gradient descent to iteratively select θ (Gu and Wahba, 1993). In either case, the computational bottleneck is computing and inverting $(K_\theta + \sigma^2_{\text{noise}} I_{N \times N})^{-1} Y$: k_θ takes $O(p^Q)$ time to compute on a pair of points. Hence, computing and inverting $K_\theta + \sigma^2_{\text{noise}} I_{N \times N}$ take $O(p^Q N^2)$ time and $O(N^3)$ time, respectively.

Many existing functional ANOVA techniques assume that all covariates are independent, i.e., that μ equals the *product measure*; see, for example, Gunn and Kandola (2004); Lin and Zhang (2006); Gu and Wahba (1993); Durrande et al. (2013). Hooker (2007) highlighted pathologies that arise when using $\mu_\otimes$ instead of μ. Specifically, he empirically showed on synthetic and real data that the functional ANOVA decomposition of an $f \in \mathcal{H}$ with respect to μ can be significantly different than the decomposition with respect to $\mu_\otimes$. This discrepancy arises because $\mu_\otimes$ can place high probability in regions where the actual covariate distribution μ has low probability; see also Section 4.1.

Finally, unlike our approach, some functional ANOVA methods assume sparsity in the effects rather than in the covariates the response depends on; see, for example, Gunn and Kandola (2004); Lin and Zhang (2006)). Recall from the discussion and example in Section 2 that sparsity in the covariates is useful for interpretability and downstream applications. But sparsity in the effects need not imply sparsity in the covariates. For example, suppose $Q = 2$. Then, there are on the order of p^2 interaction effects. A method that selects p non-zero effects might be considered sparse in the effects since $p \ll p^2$. But the selected effects could

Algorithm 4 Change of Basis Formula for Finite Dimensional Model Selection Kernels

```
procedure ReExpressEffect($\alpha$, $k_\theta$, $W$, $\mu$)
    Compute $f^{\mu\otimes}_{\{i,j\}}, f^{\mu\otimes}_{\{i\}}, f^{\mu\otimes}_{\emptyset}$ using Algorithm 3
    For $1 \le w \le W$ randomly sample $x^{(w)} \overset{\text{i.i.d.}}{\sim} \mu$
    Compute $X_{ij} = [\mathbf{1}\ \Phi_i(x_i^{(1)}) \cdots \Phi_i(x_i^{(W)})\ \Phi_j(x_j^{(1)}) \cdots \Phi_j(x_j^{(W)})]^T$, $\mathbf{1} = (1, \cdots, 1) \in \mathbb{R}^W$
    Compute $f^{\mu\otimes}_{ij,W} = [f^{\mu\otimes}_{\{i,j\}}(x_i^{(1)}, x_j^{(1)}) \cdots f^{\mu\otimes}_{\{i,j\}}(x_i^{(W)}, x_j^{(W)})]^T$
    Compute $[\hat{\Psi}^0_{ij}\ \hat{\Psi}^i_{ij}\ \hat{\Psi}^j_{ij}]^T = (X_{ij}^T X_{ij})^{-1} X_{ij}^T f^{\mu\otimes}_{ij,W}$        ▷ Least-squares projection
    Compute $\hat{f}^{\mu}_{\{i,j\}} = f^{\mu\otimes}_{\{i,j\}} - [\hat{\Psi}^i_{ij}\Phi_i^T(\cdot) + \hat{\Psi}^j_{ij}\Phi_j^T(\cdot) + \Psi^0_{ij}]$
    Compute $\hat{f}^{\mu}_{\{i\}} = f^{\mu\otimes}_{\{i\}} + \sum_{j>i} \hat{\Psi}^i_{ij}\Phi_i(\cdot) + \sum_{j<i} \hat{\Psi}^i_{ji}\Phi_i(\cdot)$
    Compute $\hat{f}^{\mu}_{\emptyset} = f^{\mu\otimes}_{\emptyset} + \sum_{i<j} \hat{\Psi}^0_{ij}$
    return $\hat{f}^{\mu}_{\{i,j\}}, \hat{f}^{\mu}_{\{i\}}, \hat{f}^{\mu}_{\emptyset}$
end procedure
```

correspond to p (or nearly p) selected covariates, which would not reduce the number of covariates.

Comparison with Agrawal et al. (2019). There are five main differences between this work and Agrawal et al. (2019): the method in Agrawal et al. (2019) (1) assumes linear interaction effects, (2) only considers pairwise interactions, (3) assumes strong-hierarchy (namely that interactions only occur among selected main effects), (4) does not necessarily correspond to an ANOVA decomposition, and (5) does not induce exact sparsity. See Appendix C for further discussion.

7. Experiments

Summary of experimental results. In this section, we compare our inference methods in Section 5 against existing procedures in terms of variable selection and estimation performance. We find that when the interaction effects are strong or comparable to the strength of the additive effects, our method outperforms existing methods in terms of variable selection and estimation performance. When the interaction effects are weak our method does not (uniformly) have the best performance but still performs well relative to many of the other methods.[6]

There are two immediate challenges with our empirical evaluation. The first is that existing methods estimate the functional ANOVA decomposition assuming all covariates are independent (or sometimes do not even specify the measure). Hence, we start our evaluation by assuming the covariates are independent so that we can compare against existing methods in Section 7.3 and Section 7.4. Then, in Section 7.5, we show why the assumption of independent covariates is problematic to demonstrate the practical utility of Algorithm 4.

[6] All results can be re-generated using the data and code provided in `https://github.com/agrawalraj/skimfapaper`.

The second challenge concerns our performance metrics (detailed in Section 7.1 and Section 7.2), which require knowing the ground truth effects. Since we do not know the ground truth effects in real data, we start in Section 7.3 by evaluating each method on simulated data so that we have ground truth effects. To compare methods on real data, we use a similar evaluation procedure as in Agrawal et al. (2019) to construct a synthetic ground truth for benchmarking (see Section 7.4 for details).

7.1 Benchmark Methods

We compare our method against other methods used to model high-dimensional data and interactions. We focus on the $Q = 2$ case throughout since (1) existing methods typically only work for the pairwise interaction case and (2) higher-order interactions are often difficult to interpret and estimate. Even when $Q = 2$, the functional ANOVA methods outlined in Section 7.1 take $O(p^2N^2 + N^3)$ time, making them computationally intractable for even moderate p and N settings. Instead, we focus on methods that can model interactions and actually scale to moderate-to-large p and N settings. These methods include approximate "two-stage" and greedy forward-stage regression methods, and linear interaction models. We detail these approaches in more depth in Appendix C. The list below summarizes the candidate methods (and software implementations) that we select from each category for empirical evaluation. In Appendix D and Appendix E we detail the hyperparameters used to fit SKIM-FA.

SPAM-2Stage: we perform variable selection by fitting a sparse additive model (SpAM) (Liu et al., 2008) to the data. We use the `sam` package in `R`. Since `sam` does not provide a default way to select the L_1 regularization strength, we use 5-fold cross-validation. For estimation, we generate all main and interaction effects among the subset of covariates selected by SpAM. We calculate these effects by taking pairwise products of univariate basis functions generated from a natural cubic spline basis with 5 total knots; see Appendix D for details. We estimate the basis coefficients (and hence effects) using ridge regression, where again we use 5-fold cross-validation to pick the L_2 regularization strength.

Multivariate Additive Regression Splines (MARS): we use the `python` implementation of MARS (Friedman, 1991) in `py-earth`. We consider two functional ANOVA decompositions of the fitted regression function: (1) *MARS-Vanilla* and (2) *MARS-EMP*. For MARS-Vanilla, the main effect of each covariate equals the sum of all selected univariate basis functions of that covariate (i.e., after the pruning step of MARS). Similarly, each pairwise effect equals the sum of all selected bivariate basis functions of those two covariates. This is the functional ANOVA decomposition originally proposed in Friedman (1991) and the one actually implemented in existing MARS software packages. It is unclear, however, what measure this functional ANOVA decomposition is taken with respect to. To the best of our knowledge, there currently does not exist a procedure to perform the functional ANOVA decomposition of MARS with respect to the empirical distribution of the covariates. We describe how to perform such a decomposition via *MARS-EMP*, which assumes the covariates are jointly independent. This method could be of independent interest and is outlined in

Appendix G.

Hierarchical Lasso (HierLasso): we use the implementation of HierLasso (Lim and Hastie, 2015) in the authors' `R` package `glinternet`. Since Lim and Hastie (2015) use cross-validation to pick the L_1 regularization strength, we similarly use 5-fold cross-validation.

Pairs Lasso: we fit the Lasso on the expanded set of features $\{x_i\}_{i=1}^{p}$ and $\{x_i x_j\}_{i,j=1}^{p}$. We fit the Lasso using the `python` package `sklearn`, and use 5-fold cross-validation to select the L_1 regularization strength.

7.2 Evaluation Metrics

Variable selection evaluation metrics. We consider both the power to select correct covariates and avoid incorrect ones. *# Correct Selected* counts the number of covariates correctly selected by the method. Higher is better. *# Wrong Selected* counts the number of covariates incorrectly selected by the method (i.e., Type I error). Lower is better. *# Correct Not Selected* counts the number of covariates that belong to the true model but were not selected by the method (i.e., Type II error). Lower is better.

Estimation evaluation metrics. We evaluate how well a method estimates main effects and interaction effects. Instead of looking only at the total mean squared estimation error, we break this error into multiple buckets to understand what bucket drives the majority of the error. Lower is better for all of the following quantities. *Correct Selected SSE (Main)* takes the sum of squared errors (SSE) between each estimated main effect component and true main effect component. This sum equals $\sum_{i \in S_1} \|f_i^* - \hat{f}_i\|_\mu^2$, where S_1 is the set of correctly identified main effects, $\hat{f}_i$ is the estimated main effect, and f_i^* is the true main effect. *Correct Not Selected SSE (Main)* takes the sum of squared norms of main effects not selected. This sum equals $\sum_{i \in S_2} \|f_i^*\|_\mu^2$, where S_2 is the set of correct main effects not selected. *Wrong Selected SSE (Main)* takes the sum of squared norms of main effect components incorrectly selected. This sum equals $\sum_{i \in S_3} \|\hat{f}_i\|_\mu^2$, where S_3 is the set of incorrect main effects selected. *Correct Selected SSE (Pair)*, *Correct Not Selected SSE (Pair)*, and *Wrong Selected SSE (Pair)* are the same as the analogous main effect metrics but instead considers interaction effects. *Total SSE* equals the sum of the 6 buckets above and *Total SSE / Signal Variance* equals the relative estimation error, i.e., Total SSE divided by the true signal variance.

7.3 Synthetic Data Evaluation

We randomly generate covariates and responses as follows. For the covariates, we draw each data point and covariate dimension $x_i^{(n)} \overset{\text{i.i.d.}}{\sim} \text{Uniform}([-1, 1])$. Since $[-1, 1]$ is compact, Theorem 1 ensures that the functional ANOVA decomposition is unique. We let y depend on the first 5 covariates; the remaining $p - 5$ covariates are taken as noise covariates that we do not want to select. To generate responses reflective of what we might expect in real data, we consider the 5 trends shown in Fig. A.1: linear, sine, logistic, quadratic, and exponential. We let the main effects equal the sum of these 5 trends, where the ith trend is applied to covariate i. For the interactions between the first 5 covariates, we consider all pairwise products of the 5 trends above, resulting in 10 total interactions. We select a noise variance

Method	Setting	# Correct Selected	# Wrong Selected	# Correct Not Selected
SKIM-FA	Weak Main	5	9	0
MARS	Weak Main	5	75	0
SPAM-2Stage	Weak Main	1	41	4
HierLasso	Weak Main	5	120	0
Pairs Lasso	Weak Main	5	144	0
SKIM-FA	Equal	5	0	0
MARS	Equal	5	71	0
SPAM-2Stage	Equal	5	15	0
HierLasso	Equal	5	40	0
Pairs Lasso	Equal	5	213	0
SKIM-FA	Main-Only	3	0	2
MARS	Main-Only	5	70	0
SPAM-2Stage	Main-Only	5	15	0
HierLasso	Main-Only	4	5	1
Pairs Lasso	Main-Only	4	6	1

Table 1: Synthetic Data Variable Selection Performance Results for $p = 1000$. The method with the fewest number of incorrect covariates selected is bolded.

such that the $R^2 = \frac{\sigma^2_{\text{signal}}}{\sigma^2_{\text{signal}} + \sigma^2_{\text{noise}}} = 0.8$, where $\sigma^2_{\text{signal}} = \langle f^*, f^* \rangle_\mu$. We further decompose the signal variance in terms of the total variance explained by main effects and interactions. Similar to the empirical evaluations in Lim and Hastie (2015), we consider the following three settings:

- **Weak Main Effects:** each main effect and pairwise effect has 0.01 * 1/5 and 0.99 * (1 / 10) of the total signal variance, respectively. Hence, the total main effect and pairwise effect variances equal 1% and 99% of the total signal variance, respectively.
- **Equal Main and Interaction Effects:** each main effect and pairwise effect has 0.5 * 1/5 and 0.5 * (1 / 10) of the total signal variance, respectively. Hence, the total main effect variance equals the total pairwise signal variance.
- **Main Effects Only:** each of the 5 main effects has 1/5th of the total signal variance, and each pairwise effect has 0 signal variance (i.e., no pairwise interactions).

To test the impact of increasing dimensionality on inference quality, we consider $p \in \{250, 500, 1000\}$ and keep $N = 1000$ fixed for each setting. For estimation, we compare only the nonlinear methods; linear methods will artificially perform poorly since some of the effects are highly nonlinear by construction. Evaluating estimation performance is trickier than evaluating selection performance since the functional ANOVA decomposition depends on the choice of measure. Unless otherwise stated, the target of inference is finding the functional ANOVA decomposition of f^* with respect to μ (the joint distribution of the covariates).

We summarize the variable selection and estimation performances of each method for $p = 1000$ in Table 1 and Table 2, respectively; see Appendix I for model performance results for all choices of p. As we discuss below, SKIM-FA outperforms all of the other methods (in terms of both variable selection and estimation) in the Weak Main Effects and the Equal Main and Interaction settings. For the Main Effects Only setting, SKIM-FA selects the fewest number of incorrect covariates. Since SKIM-FA does not select two of the correct covariates in this setting, however, its estimation performance is worse than some of the other benchmark methods.

Method	Setting	Correct Selected SSE (Main)	Correct Not Selected SSE (Main)	Wrong Selected SSE (Main)	Correct Selected SSE (Pair)	Correct Not Selected SSE (Pair)	Wrong Selected SSE (Pair)	Total SSE	Total SSE ÷ Signal Variance
SKIM-FA	Weak Main	0.72	0	1.37	0.61	0	0.63	3.33	0.17
SPAM-2Stage	Weak Main	0.16	0.2	6.69	0	18.33	0.31	25.69	1.28
MARS-EMP	Weak Main	0.67	0	5.86	3.37	0	5.63	15.52	0.78
MARS-VANILLA	Weak Main	23.62	0	3.18	23.16	0	15.43	65.39	3.27
SKIM-FA	Equal	1.54	0	0	0.29	0	0	1.82	0.09
SPAM-2Stage	Equal	1.67	0	1.07	0.41	0	2.16	5.31	0.27
MARS-EMP	Equal	0.61	0	3.84	1.7	0	2.52	8.67	0.43
MARS-VANILLA	Equal	454.88	0	3.16	21.46	0	13.22	492.72	24.64
SKIM-FA	Main Only	2.7	8.1	0	0	0	0.24	11.03	0.55
SPAM-2Stage	Main Only	2.67	0	0.78	0	0	0.02	3.46	0.17
MARS-EMP	Main Only	0.45	0	2.68	0	0	2.39	5.51	0.28
MARS-VANILLA	Main Only	16.14	0	1.56	0	0	10.33	28.02	1.4

Table 2: Synthetic Data Estimation Performance Results for $p = 1000$. The method with the smallest total SSE is bolded.

Weak main effects setting. In the setting of weak main effects, Spam-2Stage performs worse than the other methods in terms of the power to select correct covariates; Spam-2Stage only selects one correct covariate for $p = 500$ and $p = 1000$. This poor variable selection is expected since the signal is locked away in the interactions but SpAM assumes additive effects. In particular, only 1% of the variance is explained by additive effects (even though additive *and* interaction effects explain 80% of the variance in the response). MARS, HierLasso, and Pairs Lasso detect all 5 correct covariates but they all pick up many more incorrect covariates relative to SKIM-FA.

MARS selects many incorrect covariates because it can only form an interaction between two covariates if at least one of the covariates has an additive effect (similar to Spam2Stage). In the extreme case of no additive effects, for example, MARS randomly selects covariates to have additive effects. By random chance, MARS will eventually select a correct covariate (i.e., one the response actually depends on) to have an additive effect. Since this covariate has an interaction effect, in the next step MARS will (likely) select the correct interaction effect. Hence, MARS will need to select many incorrect covariates as additive effects before identifying the true interactions.

In terms of estimation performance, SKIM-FA has the smallest total mean-squared estimation error. Since Spam-2Stage only considers interactions between covariates selected by SpAM, its poor estimation performance is driven by not selecting many of the correct covariates. MARS-VANILLA performs a functional ANOVA decomposition with respect to an unspecified measure. Hence, it is unclear how to interpret its main and interaction effects. One might think (and truthfully what we initially thought) that MARS-VANILLA would still return a functional decomposition close to one with respect to the actual covariate distribution. Table I.6 shows that this intuition is incorrect; the relative estimation error of MARS-VANILLA always exceeds 1! This poor estimation performance stems from not specifying the measure (and hence the target of inference), not MARS's ability in finding a model with good predictive performance. In particular, MARS-EMP, which produces the *exact same predictions* as MARS-VANILLA, yields better performance because it re-orthogonalizes the fit with respect to the covariate distribution.

Equal main and interaction effects setting. In this setting, all methods are able to recover all 5 true covariates. For both estimation and variable selection, SKIM-FA performs best.

Main effects only setting. Each method selects the majority of correct covariates. However, some methods – namely Pairs Lasso and HierLasso – have a systematic bias; for all choices of p, they never select covariate 3 (the quadratic trend) since a quadratic trend has a weak linear correlation. Since the other methods can model nonlinear relationships, they can pick up this trend. Hence, they have better statistical power to detect correct covariates, improving variable selection performance. In terms of Type I error, some methods select incorrect covariates much more frequently. For example, MARS consistently selects over 50 incorrect covariates for all choices of p. A potential reason for this poor performance is that MARS induces sparsity through a greedy pruning step instead of an actual sparsity inducing penalty as in the other methods.

Runtime comparisons. We conclude this section by comparing each method in term of runtime in the high-dimensional setting. The two Lasso methods take $O(p^2N)$ time while the remaining methods depend only linearly on p. When $p > N$ and $\omega = p/N$, our method takes $O(\omega N^3)$ while the two Lasso based methods take $O(\omega^2 N^3)$ time. Hence, for higher-dimensional problems, our method will become much faster relative to the Lasso methods. For example, in genome-wide association studies, data sets can have N on the order of 10^3 and p on the order of 10^7 (1000 Genomes Project, 2015). Hence, $\omega = 10^4$, which corresponds to a potential 10^4 computational speedup factor. In Fig. 1c, we compare the runtimes of each method as we vary p/N on simulated data. We keep N fixed at 100 and vary p from 10 to 10^4. As expected, as p/N increases, our method yields substantial computational savings relative to Pairs Lasso and HierLasso. Relative to Spam-2Stage and MARS, our method does not yield better computational scaling. However, based on our synthetic evaluation above (and real data evaluation in Section 7.4), we have better statistical performance.

7.4 Evaluation on Real Data: Bike Sharing & Concrete Compressive Strength Data sets

Evaluating the methods in terms of variable selection and estimation quality is challenging because we typically do not have ground truth main and interaction effects for high-dimensional (real) data. Similar to the evaluation procedure in Agrawal et al. (2019), we instead take a low-dimensional data set where N is large and p is small. We make it high-dimensional by adding synthetic random noise covariates. These two choices have several purposes. First, by fitting a regression function on the original low-dimensional data set, standard $N^{-1/2}$ statistical convergence rates apply. Hence, for large N, a maximum-likelihood estimate of the regression function will be close to the true regression function, creating a (near) ground truth for estimation evaluation. For variable selection, the random noise covariates create a "synthetic control;" if a method selects any of the random noise covariates as a main or interaction effect, we know the method selected an incorrect covariate.

Based on these ideas, we consider the popular (low-dimensional) Bike Sharing data set, which we downloaded from the UCI Machine Learning Repository. This data set contains 17,389 datapoints and 13 covariates. We consider 4 continuous variables (hour, air

Method	# Covariates	# Original Selected	# Wrong Selected
SKIM-FA	1000	3	0
HierLasso	1000	3	5
SPAM-2Stage	1000	3	8
Pairs Lasso	1000	3	76
MARS	1000	3	119

Table 3: Variable Selection Performance for the Bike Sharing Data Set.

Method	# Noise	Correct Selected SSE (Main)	Correct Not Selected SSE (Main)	Wrong Selected SSE (Main)	Correct Selected SSE (Pair)	Correct Not Selected SSE (Pair)	Wrong Selected SSE (Pair)	Total SSE
SKIM-FA	1000	0.145	0.002	0	0.107	0.009	0	0.263
SPAM-2Stage	1000	0.149	0.002	0.027	0.081	0.009	0.000	0.269
MARS-EMP	1000	0.214	0.002	0.485	0.054	0.026	0.245	1.026
MARS-Vanilla	1000	6.556	0.002	0.796	0.947	0.026	1.882	10.209

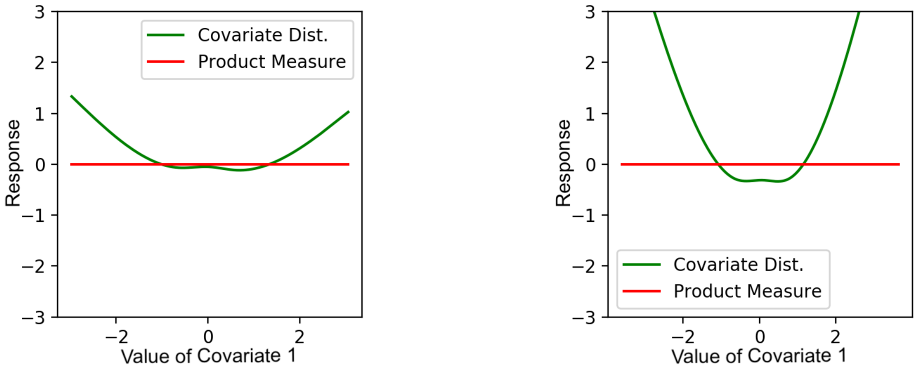
Table 4: Estimation Performance for the Bike Sharing Data Set.

temperature, humidity, windspeed) and use the total number of bikes rented as the response. We standardize the response by subtracting the mean and dividing by the standard deviation, and min-max standardize the covariates so that each covariate belongs to $[0, 1]$. For the proxy ground truth, we fit a pairwise interaction model consisting of all 4 main effects and 6 possible pairwise interactions.

Similar to our synthetic evaluation, we randomly subsample a total of $N = 10^3$ datapoints, and then train each benchmark method on this subsampled data set. To make the inference task high-dimensional we inject $p_{\text{noise}} \in \{250, 500, 1000\}$ random noise covariates, where these noise covariates are drawn iid from a Uniform(0, 1) distribution. We report on the same variable selection and estimation metrics as in the synthetic experiments for $p_{\text{noise}} = 1000$ in Table A.1 and Table A.2, respectively; see Table I.8 and Table I.9 for all choices of p_{noise}. We see that again SKIM-FA has similar or much better estimation and variable selection performance relative to the other methods. Finally, to understand the impact of correlated predictors on performance, we append correlated (real) covariates to the Bike Sharing data set (instead of synthetic ones drawn from a Uniform(0, 1) distribution) in Appendix I.1. We again find that SKIM-FA has better performance than the other methods.

7.5 Impact of Correlated Predictors on the Functional ANOVA

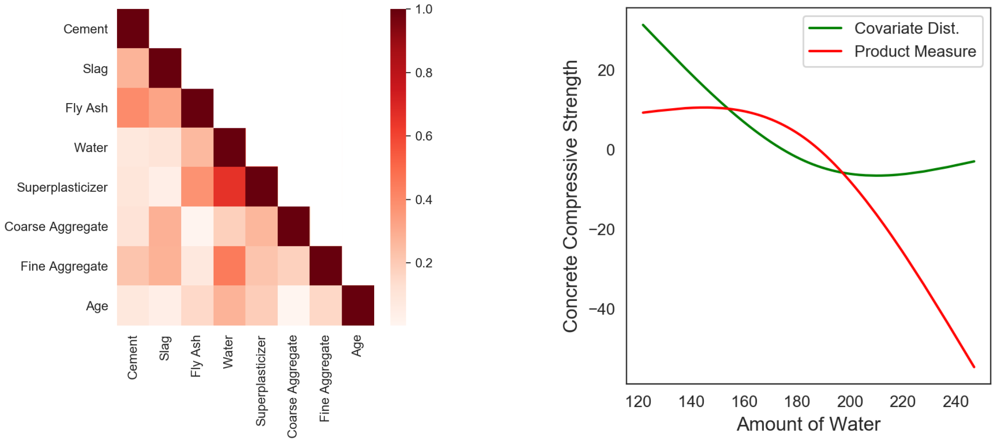
So far we have performed the functional ANOVA decomposition assuming that the covariates are jointly independent; for our synthetic data evaluation in Section 7.3 this independence held by design. Here we show the effect correlated predictors have on the resulting decomposition. Recall that previous functional ANOVA methods assume product measure, but our Algorithm 4 provides the flexibility to select different measures. We demonstrate the practical utility of this flexibility here. To this end, we consider the simplest possible regression function with interactions: $f(x_1, x_2) = x_1 x_2$. If $x_1 \perp\!\!\!\perp x_2$, then the functional ANOVA decomposition of f with respect to $\mu(x_1, x_2)$ equals $x_1 x_2$. However, if x_1 and x_2 are correlated, then the functional ANOVA decomposition no longer equals $x_1 x_2$. In particular,



(a) $\rho = 0.1$

(b) $\rho = 0.5$

Figure 2: The left hand and right hand plots show how the additive effect of x_1 (in the functional ANOVA decomposition of the function $x_1 x_2$) varies as the correlation between x_1 and x_2 increases.



(a) Absolute Value of Correlation Matrix

(b) Main Effect of Water on Strength

Figure 3: Effect of Correlated Predictors on the Concrete Compressive Strength Data Set

as the correlation between x_1 and x_2 increases, f can be explained better by additive effects (e.g., in the degenerate case when $x_1 = x_2$, then $f(x_1, x_2) = x_1^2$). To test this empirically, we randomly generate x_1, x_2 from a multivariate Gaussian distribution with marginal variances equal to 1 and pairwise correlation equal to ρ. We let $\rho \in \{0.1, 0.5\}$. Fig. 2 shows that when ρ gets stronger, the discrepancy between a functional ANOVA decomposition with respect to $\mu(x_1, x_2)$ versus product measure $\mu_{\otimes} = N(0,1) \otimes N(0,1)$ increases. As expected, as the correlation increases, a quadratic-like function of x_1 and x_2 explains f increasingly well.

We perform a similar analysis above but for real data, namely the Concrete Compressive Strength data set from the UCI machine learning repository. In Fig. 3, we plot the correlations between the 8 covariates that potentially predict the response (concrete strength). The two most correlated covariates are the amount of water and the amount of superplasticizer.

Method	# Covariates Selected	MSE	R^2
SKIM-FA	31	39.1	0.43
HierLasso	8	68.1	0.01
SPAM-2Stage	0	68.9	0.00
MARS	15	96.1	-0.39
Pairs Lasso	–	–	–

Table 5: Test Set Predictive Performance for the Obesity Gene-Expression and SNP Data Set.

Since the covariates have non-trivial correlations, the functional ANOVA decomposition with respect to μ and $\mu_{\otimes}$ might be different based on Proposition 4. In Fig. 3 we see that there indeed is a difference; the (estimated) additive effect for water on concrete strength varies substantially depending on which measure is selected to perform the functional ANOVA decomposition. In Appendix I.2, we compare how the functional ANOVA decomposition changes depending on if we use μ or $\mu_{\otimes}$ for the Bike Sharing data set. Unlike the Concrete Compressive Strength data set, however, we do not see a large difference between the two functional ANOVA decompositions for the Bike Sharing data set.

7.6 Evaluation on Real Data: Obesity Gene-Expression and SNP Data Set

We conclude by evaluating each method on a high-dimensional genomics data set where $N \ll p$. Unlike our previous evaluation, however, we do not know the ground truth effects. Hence, we are unable to compute the evaluation metrics in Section 7.2. We instead report the mean-squared prediction error of each method on a left-out test set as a proxy. As a qualitative check on inference quality, we interpret the genes selected by SKIM-FA and find that some correspond to genes already flagged as obesity-related based on previous biological studies. Below we summarize the data and our findings in more depth.

We consider the data set kindly provided by Joseph et al. (2018), which consists of the body mass index (BMI) of $N = 87$ individuals. After using the pre-processing steps in Joseph et al. (2018), we also consider 13,276 gene-expression levels, 16 single-nucleotide polymorphisms (SNPs), and a genetic risk-score feature as covariates for a total of $p =$ 13,293 covariates. Since the number of covariates is more than 100 times the number of observations, and the number of pairwise interactions almost exceeds 100 million, this data set leads to a non-trivial inference task. We report the out-of-sample mean-squared error and out-of-sample R^2 for each method in Table 5 (15% of the data is used for testing purposes). Table 5 has missing values for Pairs Lasso since the number of interactions is too large to run on a single machine. SPAM-2Stage does not select any covariates, and hence the R^2 is zero. Both HierLasso and MARS seem to overfit given the poor R^2 performance, even though each selects 8 and 15 covariates, respectively. SKIM-FA performs the best ($R^2 = 0.43$), and selects 31 covariates; see Appendix I.3 for the names of all genes and SNPs selected.

We do not know which of these 31 genes are truly associated with obesity. Nevertheless, we find that several genes SKIM-FA selects are obesity related based on previous studies. SKIM-FA selects IRS2, which is a gene associated with obesity and diabetes risk; see, for example, Butte et al. (2011). SKIM-FA says that IRS2 has a negative effect on BMI (i.e., a higher expression of IRS2 decreases BMI) which agrees with the findings in Lin et al.

(2004) based on experimental data on mice; see Appendix I.3 for details. SKIM-FA also selects a SNP (Rs2112347) and two genes (KISS1R and SKP1) which are obesity related based on Young et al. (2015); Wang et al. (2021); Geronikolou et al. (2020). Interestingly, as we discuss in Appendix I.3, SKP1 does not have strong additive effects, but it has the strongest interactions. Hence, SKP1 might be an interesting candidate for further study of its interaction properties.

8. Conclusion

In this paper, we developed a new, computationally efficient method to perform sparse functional ANOVA decompositions. The heart of our procedure relied on a new kernel trick to implicitly represent nonlinear interactions (Theorem 2), and a change-of-basis formula (Theorem 3) to re-express the fit in terms of an arbitrary measure. We compared our method against other methods often used to model high-dimensional data with interactions. We found improved performance on both simulated and real data sets by relaxing assumptions such as linearity and the presence of strong-additive effects while still remaining competitive (or being orders of magnitude faster) in terms of runtime.

There are many interesting future research directions. One involves scaling our method to both the large N and p setting; our current method takes $O(pQN^2 + N^3)$ time which becomes problematic for large N. This cubic dependence, however, is not unique to our method but rather a fundamental obstacle faced by kernel ridge regression and Gaussian processes. Fortunately, many methods already exist to help alleviate these computational challenges with respect to N; see, for example, Gardner et al. (2018); Titsias (2009); Quiñonero Candela and Rasmussen (2005). Another interesting direction involves applying our method to biological data sets. In particular, an open challenge in genomics has been detecting *epistasis*, or interaction effects between genetic variants, from genome sequencing data (Maher, 2008; Aschard, 2016; Slim et al., 2018; Greene et al., 2010). Detecting epistasis has been statistically and computationally challenging because p is in the millions, so the number of pairwise interactions is on the order of *trillions*. Since our method does not require explicitly generating all interactions, it has the potential to tractably detect interactions in such especially high-dimensional data regimes.

Acknowledgments

This work was supported in part by the DARPA I2O LwLL program, NSF Award 2029016, an ONR Early Career Grant, and MIT Lincoln Laboratory.

Supporting Materials

Appendix A. Figures and Tables Referenced in Section 7

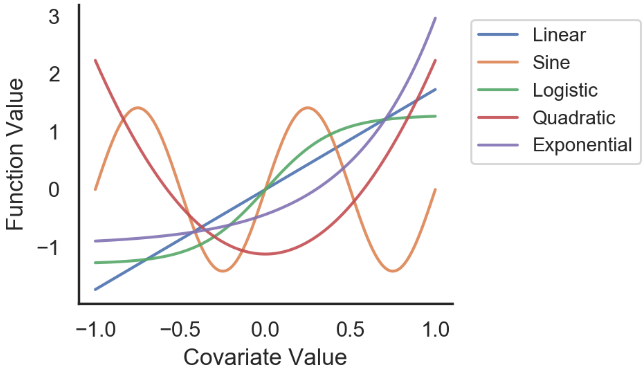


Figure A.1: Test functions used to generate synthetic data

Table A.1: Variable Selection Performance for the Bike Sharing Data Set.

Method	# Covariates	# Original Selected	# Wrong Selected
SKIM-FA	1000	3	0
HierLasso	1000	3	5
SPAM-2Stage	1000	3	8
Pairs Lasso	1000	3	76
MARS	1000	3	119

Table A.2: Estimation Performance for the Bike Sharing Data Set.

Method	# Noise	Correct Selected SSE (Main)	Correct Not Selected SSE (Main)	Wrong Selected SSE (Main)	Correct Selected SSE (Pair)	Correct Not Selected SSE (Pair)	Wrong Selected SSE (Pair)	Total SSE
SKIM-FA	1000	0.145	0.002	0	0.107	0.009	0	0.263
SPAM-2Stage	1000	0.149	0.002	0.027	0.081	0.009	0.000	0.269
MARS-EMP	1000	0.214	0.002	0.485	0.054	0.026	0.245	1.026
MARS-Vanilla	1000	6.556	0.002	0.796	0.947	0.026	1.882	10.209

Appendix B. Proofs

B.1 Proof of Proposition 1

It suffices to prove that $\mathcal{H}_V^o$ is an RKHS. First we prove that $\mathcal{H}_V^o$ is a Hilbert space. Since $\mathcal{H}_V^o \subset \mathcal{H}_V$, it suffices to show that $\mathcal{H}_V^o$ is a vector space and complete. To show that $\mathcal{H}_V^o$ is a vector space, take arbitrary $f, g \in \mathcal{H}_V^o$ and $\alpha, \beta \in R$. We want to show $\alpha f + \beta g \in \mathcal{H}_V^o$. Take an arbitrary $f_A \in \mathcal{H}_A$, $A \subsetneq V$. Then,

$$\begin{aligned}\langle \alpha f + \beta g, f_A \rangle_\mu &= \alpha \langle f, f_A \rangle_\mu + \beta \langle g, f_A \rangle_\mu \\ &= 0\end{aligned}$$

since $f, g \in \mathcal{H}_V^o$. Hence, $\mathcal{H}_V^o$ is a vector space.

Suppose towards a contradiction that $\mathcal{H}_V^o$ is not complete. Then, since $\mathcal{H}_V$ is complete, there exists an $f' \in \mathcal{H}_V \setminus \mathcal{H}_V^o$ and Cauchy sequence $\{f_n\}_{n=1}^\infty$ such that $\lim_{n\to\infty} \|f' - f_n\|_{\mathcal{H}_V}$ $= 0$, where $f_n \in \mathcal{H}_V^o$ and $\|\cdot\|_{\mathcal{H}_V}$ denotes the induced RKHS norm for $\mathcal{H}_V$. Then, there exists an $\epsilon > 0$ and $f_A \in \mathcal{H}_A$, $A \subsetneq V$ such that

$$\begin{aligned}\epsilon &= \langle f', f_A \rangle_\mu \\ &= \langle f' + f_m - f_m, f_A \rangle_\mu \\ &= \langle f' - f_m, f_A \rangle_\mu + \langle f_m, f_A \rangle_\mu \\ &= \langle f' - f_m, f_A \rangle_\mu \\ &\leq \|f' - f_m\|_\mu \|f_A\|_\mu \quad \text{(by Cauchy-Schwarz).}\end{aligned} \tag{15}$$

To reach a contraction, it suffices to show that there exists an $m < \infty$ such that $\|f' - f_m\|_\mu < \frac{\epsilon}{\|f_A\|_\mu}$. To obtain this inequality, we upper bound $\|\cdot\|_\mu$ in terms of $\|\cdot\|_{\mathcal{H}_V}$. Let r_V be the reproducing kernel for $\mathcal{H}_V$. Then, for $f \in \mathcal{H}_V$,

$$\begin{aligned}|f(x)|^2 &= |\langle f, r_V(x, \cdot) \rangle_{\mathcal{H}_V}|^2 \quad \text{(by the reproducing property)} \\ &\leq \|f\|_{\mathcal{H}_V}^2 r_V(x, x)^2 \quad \text{(by Cauchy-Schwarz).}\end{aligned} \tag{16}$$

Then,

$$\begin{aligned}\|f\|_\mu^2 &= \int |f(x)|^2 d\mu \\ &\leq \|f\|_{\mathcal{H}_V}^2 \int r_V(x, x)^2 d\mu\end{aligned} \tag{17}$$

Since $\mathcal{H}_V$ belongs to the space of square integrable functions, $\int r_V(x, x)^2 d\mu = M_V < \infty$. Hence,

$$\|f' - f_m\|_\mu \leq M_V \|f' - f_m\|_{\mathcal{H}_V}^2 < \infty. \tag{18}$$

Since $\|f' - f_m\|_{\mathcal{H}_V}^2 \to 0$, there exists an m such that $\|f' - f_m\|_\mu < \frac{\epsilon}{\|f_A\|_\mu}$. Hence, $\mathcal{H}_V^o$ is complete.

To complete the proof it suffices to show that the evaluation functional on $\mathcal{H}_V^o$ is a bounded operator. Since $\mathcal{H}_V$ is an RKHS there exists an $M_x < \infty$ such that for all $f \in \mathcal{H}_V$

$$|f(x)| \leq M_x \|f\|_{\mathcal{H}_V}. \tag{19}$$

Since $\mathcal{H}_V^o \subset \mathcal{H}_V$, then for all $g \in \mathcal{H}_V^o$,

$$|g(x)| \leq M_x \|g\|_{\mathcal{H}_V}. \tag{20}$$

B.2 Proof of Lemma 2

$$\begin{aligned} f^{(M)}(x) &= \sum_{m=1}^{M} \alpha_m k_\theta(x_m, x) \\ &= \sum_{m=1}^{M} \left(\sum_{V:|V|\leq Q} \theta_V k_V(x_m, x) \right) \\ &= \sum_{V:|V|\leq Q} \theta_V \left(\sum_{m=1}^{M} k_V(x_m, x) \right) \\ &= \sum_{V:|V|\leq Q} f_V(x). \end{aligned} \tag{21}$$

It remains to show that $f_V \in \mathcal{H}_V^o$. For all $m \in [M]$, $k_V(x_m, \cdot) \in \mathcal{H}_V^o$. Hence, $\theta_V \sum_{m=1}^{M} k_V(x_m, x) \in \mathcal{H}_V^o$ since $\mathcal{H}_V^o$ is a Hilbert space.

B.3 Proof of Proposition 4

We prove the claim using a constructive proof with $p = 2$ variables. Consider the function

$$f(x_1, x_2) = 1 + (x_1 - x_2)^{2k} I(|x_1| \leq M) I(|x_2| \leq M). \tag{22}$$

Suppose the joint distribution of (x_1, x_2) under μ equals

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \right).$$

Then, the joint distribution of (x_1, x_2) under $\mu_\otimes$ equals

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right).$$

By symmetry,

$$\begin{aligned} \mathbb{E}_{\mu_\otimes}[f(x_1, x_2)] &= 2\mu_2(x_2 < 0)\mathbb{E}_{\mu_\otimes}[f(x_1, x_2) \mid x_2 < 0] \\ &\geq 2\mu_1(x_1 > c)\mu_2(x_2 < 0)\mathbb{E}_{\mu_\otimes}[f(x_1, x_2) \mid x_1 > c, x_2 < 0] \\ &= \mu_1(x_1 > c)\mathbb{E}_{\mu_\otimes}[f(x_1, x_2) \mid x_1 > c, x_2 < 0] \\ &\geq \mu_1(x_1 > c)c^{2k} I(|c| < M). \end{aligned} \tag{23}$$

Under μ, we may assume without loss of generality that

$$\begin{aligned} x_1 &\sim \mathcal{N}(0, 1) \\ \epsilon &\sim \mathcal{N}(0, 1) \quad \text{s.t.} \quad \epsilon \perp\!\!\!\perp x_1 \\ x_2 &= \rho x_1 + \sqrt{1 - \rho^2}\epsilon. \end{aligned}$$

Then,

$$\begin{aligned}
\lim_{\rho \to 1} \mathbb{E}_\mu f(x_1, x_2) &= 1 + \lim_{\rho \to 1} \int (x_1 - x_2)^{2k} I(|x_1| \leq M) I(|x_2| \leq M) d\mu(x_1, x_2) \\
&= 1 + \lim_{\rho \to 1} \int (x_1 - \rho x_1 - \sqrt{1-\rho^2}\epsilon)^{2k} I(|x_1| \leq M) I(|x_2| \leq M) d\mu_\otimes(x_1, \epsilon) \\
&= 1 + \int \lim_{\rho \to 1} (x_1 - \rho x_1 - \sqrt{1-\rho^2}\epsilon)^{2k} I(|x_1| \leq M) I(|x_2| \leq M) d\mu_\otimes(x_1, \epsilon) \\
&= 1,
\end{aligned} \tag{24}$$

where the second to last line follows from he Dominated Convergence Theorem since $f(x_1, x_2)$ is uniformly bounded by $(2M)^{2k}$. Since $\mathbb{E}_\mu f(x_1, x_2) > 1$ for $0 \leq \rho < 1$, there exists a sequence $\{\rho_k\}_{k=1}^\infty$ such that for all $k \in \mathbb{N}$, $1 < \mathbb{E}_{\mu_k} f(x_1, x_2) < 2$ and $0 < \rho_k < 1$, where μ_k sets $\rho = \rho_k$. Pick k' large enough so that $f_{\{\emptyset\}}^{\mu_\otimes} > 2$. Then, for $k \geq k'$,

$$\begin{aligned}
\frac{|f_{\{\emptyset\}}^{\mu_\otimes} - f_{\{\emptyset\}}^{\mu_k}|}{|f_{\{\emptyset\}}^{\mu_k}|} &\geq \frac{|f_{\{\emptyset\}}^{\mu_\otimes} - f_{\{\emptyset\}}^{\mu_k}|}{2} \\
&= \frac{f_{\{\emptyset\}}^{\mu_\otimes} - f_{\{\emptyset\}}^{\mu_k}}{2} \\
&> \frac{f_{\{\emptyset\}}^{\mu_\otimes} - 2}{2}
\end{aligned} \tag{25}$$

Let $k^* = \max\left(k', \left\lceil .5 \sqrt[c]{\frac{2(\Delta+1)}{\mu_1(x_1 > c}} \right\rceil\right)$. Then, by Eq. (23) and Eq. (25), $\frac{|f_{\{\emptyset\}}^{\mu_\otimes} - f_{\{\emptyset\}}^{\mu_{k^*}}|}{|f_{\{\emptyset\}}^{\mu_{k^*}}|} > \Delta$.

B.4 Proof of Lemma 1

By equation 2.25 of Rasmussen and Williams (2006, Chapter 2), Eq. (7) equals the posterior predictive mean of the following Bayesian model:

$$\begin{aligned}
f &\sim GP(0, k_\theta) \\
y \mid f, x &\sim \mathcal{N}(f(x), \sigma^2_{\text{noise}} = \lambda).
\end{aligned}$$

We may re-write k_θ as,

$$\begin{aligned}
k_\theta(x, \tilde{x}) &= \sum_{V : |V| \leq Q} \theta_V \Phi_V^T(x) \Phi_V^T(\tilde{x}) \\
&= \sum_{V : |V| \leq Q} \Phi_V^T(x) [\theta_V I_{B^V \times B^V}] \Phi_V^T(\tilde{x}) \\
&= \sum_{V : |V| \leq Q} \Phi_V^T(x) \Sigma_V \Phi_V^T(\tilde{x}),
\end{aligned}$$

where $\Sigma_V = \theta_V I_{B^V \times B^V}$. Then, by Rasmussen and Williams (2006, Chapter 2.1.2) and the additive property of kernels, $f \sim GP(0, k_\theta)$ has the same distribution as drawing a set of regression coefficients $\Theta_V \sim \mathcal{N}(0, \Sigma_V)$ and setting $f = \sum_{V : |V| \leq Q} \Theta_V^T \Phi_V(\cdot)$. Hence, the posterior predictive mean of the Gaussian process at a point x equals $\sum_{V : |V| \leq Q} \hat{\Theta}_V^T \Phi_V(x)$.

B.5 Proof of Theorem 2

$$\begin{aligned}
k_{\text{SKIM-FA}}(x, \tilde{x}) &= \sum_{V:|V|\leq Q} \left[\eta^2_{|V|} \prod_{i\in V} \kappa_i^2\right] k_V(x, \tilde{x}) \\
&= \sum_{V:|V|\leq Q} \left[\eta^2_{|V|} \prod_{i\in V} \kappa_i^2\right] \prod_{i\in V} k_i(x_i, \tilde{x}_i) \\
&= \sum_{V:|V|\leq Q} \left[\eta^2_{|V|} \prod_{i\in V} \kappa_i^2 k_i(x_i, \tilde{x}_i)\right] \qquad (26)\\
&= \sum_{q=1}^{Q} \sum_{V:|V|=Q} \left[\eta^2_{|V|} \prod_{i\in V} \kappa_i^2 k_i(x_i, \tilde{x}_i)\right] \\
&= \sum_{q=1}^{Q} \eta_q^2 \sum_{V:|V|=q} \left[\prod_{i\in V} \kappa_i^2 k_i(x_i, \tilde{x}_i)\right]
\end{aligned}$$

Let $\tilde{k}_i(\cdot, \cdot) = \kappa_i^2 k_i(\cdot, \cdot)$. Then, Vapnik (1995, pg. 199) shows that

$$\bar{k}_q := \sum_{V:|V|=q} \prod_{i\in V} \tilde{k}_i = \frac{1}{q} \sum_{s=1}^{q} (-1)^{s+1} \bar{k}_{q-s} k^s, \qquad (27)$$

where $k^s(x, \tilde{x}) = \sum_{i=1}^{p} [\tilde{k}_i(x_i, \tilde{x}_i)]^s$ and $\bar{k}_0(x, \tilde{x}) = 1$. The result follows from Eq. (26) and Eq. (27).

B.6 Proof of Corollary 2

Computing and storing $k^1(x, \tilde{x}), \cdots, k^Q(x, \tilde{x})$ takes $O(pQ)$ time and requires $O(Q)$ memory, respectively. After computing and storing $\bar{k}_1(x, \tilde{x}), \cdots, \bar{k}_q(x, \tilde{x})$, $\bar{k}_{q+1}(x, \tilde{x})$ takes $O(q+1)$ time. Hence, computing all $\bar{k}_1(x, \tilde{x}), \cdots, \bar{k}_Q(x, \tilde{x})$ terms takes $O(Q^2)$ time given $k^1(x, \tilde{x}), \cdots, k^Q(x, \tilde{x})$. Since $Q < p$, computing $k_{\text{SKIM-FA}}(x, \tilde{x})$ takes $O(pQ)$ time.

B.7 Proof of Proposition 5

$$\begin{aligned}
\frac{\partial L}{\partial \tilde{U}_i^{(t)}} &= \frac{\partial L}{\partial \kappa_i^{(t)}} \frac{\partial \kappa_i}{\partial U_i^{(t)}} \frac{\partial U_i^{(t)}}{\partial \tilde{U}_i^{(t)}} \\
&= \frac{\partial L}{\partial \kappa_i^{(t)}} I(U_i^{(t)} > c) \frac{2\tilde{U}_i^{(t)}}{(\tilde{U}_i^{(t)} + 1)^2}.
\end{aligned}$$

Since $\kappa_i^{(t)} = 0$, that implies $U_i^{(t)} \leq c$. Hence, $\frac{\partial L}{\partial \tilde{U}_i^{(t)}} = 0$. Consequently,

$$\begin{aligned}
\tilde{U}_i^{(t+1)} &= \tilde{U}_i^{(t)} - \gamma \frac{\partial L}{\partial \tilde{U}_i^{(t)}} \\
&= \tilde{U}_i^{(t)}.
\end{aligned} \qquad (28)$$

By Eq. (28), $\kappa_i^{(t')} = 0$ for all $t' \geq t$.

B.8 Proof of Lemma 3

It suffices to prove that any $f_V \in \mathcal{H}_V$ is square-integrable with respect to any probability measure. Since ϕ_{ib} is a continuous function on a compact set, there exists a $0 < M_{ib} < \infty$ such that $|\phi_{ib}|$ is bounded by M_{ib}. Without loss of generality, assume $V = \{1, \cdots, q\}$. Then, there exists coefficients $c_{b_1,\cdots,b_q} \in \mathbb{R}$ such that

$$\begin{aligned} f_V(x_V) &= \sum_{b_1 \in [B_1]} \cdots \sum_{b_q \in [B_q]} c_{b_1,\cdots,b_q} \prod_{i=1}^{q} \phi_{ib_i}(x_i) \\ &\leq \sum_{b_1 \in [B_1]} \cdots \sum_{b_q \in [B_q]} c_{b_1,\cdots,b_q} M_*^q \\ &< \infty \end{aligned}$$

for all x_V, where $M_* = \max_{i \in [p]} \max_{b \in [B_i]} M_{ib} < \infty$ since $B_i < \infty$. Hence, for any probability measure μ,

$$\begin{aligned} \int |f_V(x_V)|^2 d\mu &< \int \left(\sum_{b_1 \in [B_1]} \cdots \sum_{b_q \in [B_q]} c_{b_1,\cdots,b_q} M_*^q \right)^2 d\mu \\ &= \left(\sum_{b_1 \in [B_1]} \cdots \sum_{b_q \in [B_q]} c_{b_1,\cdots,b_q} M_*^q \right)^2 \\ &< \infty. \end{aligned} \tag{29}$$

B.9 Proof of Theorem 3

Let

$$\begin{aligned} \tilde{f}_{ij} &= f_{\{i,j\}}^{\mu_\otimes} - [\Psi_{ij}^i \Phi_i + \Psi_{ij}^j \Phi_j + \Psi_{ij}^0] \\ \tilde{f}_i &= f_{\{i\}}^{\mu_\otimes} + \sum_{j > i} \Psi_{ij}^i \Phi_i + \sum_{j < i} \Psi_{ji}^i \Phi_i(x_i) \\ \tilde{f}_\emptyset &= f_\emptyset^{\mu_\otimes} + \sum_{i < j} \Psi_{ij}^0. \end{aligned} \tag{30}$$

We start by proving that $f = \tilde{f}_\emptyset + \sum_{i=1}^p \tilde{f}_i + \sum_{i,j=1}^p \tilde{f}_{ij}$. Expanding each component,

$$
\begin{aligned}
\tilde{f}_\emptyset + \sum_i \tilde{f}_i + \sum_{i<j} \tilde{f}_{ij} &= \tilde{f}_\emptyset + \sum_i \tilde{f}_i + \sum_{i<j} \left[f^{\mu_\otimes}_{\{i,j\}} - [\Psi^i_{ij}\Phi_i + \Psi^j_{ij}\Phi_j + \Psi^0_{ij}] \right] \\
&= f^{\mu_\otimes}_\emptyset + \sum_i \tilde{f}_i + \sum_{i<j} \left[f^{\mu_\otimes}_{\{i,j\}} - [\Psi^i_{ij}\Phi_i + \Psi^j_{ij}\Phi_j] \right] \\
&= f^{\mu_\otimes}_\emptyset + \sum_i \left[f^{\mu_\otimes}_{\{i\}} + \sum_{j>i} \Psi^i_{ij}\Phi_i + \sum_{j<i} \Psi^i_{ji}\Phi_i \right] + \\
&\qquad \sum_{i<j} \left[f^{\mu_\otimes}_{\{i,j\}} - [\Psi^i_{ij}\Phi_i + \Psi^j_{ij}\Phi_j] \right] \\
&= f^{\mu_\otimes}_\emptyset + \sum_i f^{\mu_\otimes}_{\{i\}} + \sum_{i<j} f^{\mu_\otimes}_{\{i,j\}} + \\
&\qquad \sum_i \sum_{j>i} \Psi^i_{ij}\Phi_i + \sum_i \sum_{j<i} \Psi^i_{ji}\Phi_i - \sum_{i<j} \left[\Psi^i_{ij}\Phi_i + \Psi^j_{ij}\Phi_j \right] \\
&= f + \sum_i \sum_{j>i} \Psi^i_{ij}\Phi_i + \sum_i \sum_{j<i} \Psi^i_{ji}\Phi_i - \sum_i \sum_{j>i} \left[\Psi^i_{ij}\Phi_i + \Psi^j_{ij}\Phi_j \right] \\
&= f + \sum_i \sum_{j<i} \Psi^i_{ji}\Phi_i - \sum_i \sum_{j>i} \Psi^j_{ij}\Phi_j \\
&= f + \sum_i \sum_{j<i} \Psi^i_{ji}\Phi_i - \sum_j \sum_{j<i} \Psi^i_{ji}\Phi_j \\
&= f.
\end{aligned}
$$

We now prove that there exists unique coefficients, $\Psi^i_{ij} \in \mathbb{R}^{1\times B_i}, \Psi^j_{ij} \in \mathbb{R}^{1\times B_j}, \Psi^0_{ij} \in \mathbb{R}$, such that $\tilde{f}_{ij}$ belongs to the orthogonal complement of the Hilbert space $\mathcal{H}^{\text{add}}_{\{i,j\}} := \text{span}\{1, \{\phi_{ib}\}_{b=1}^{B_i}, \{\phi_{jb}\}_{b=1}^{B_j}\} = \mathcal{H}_\emptyset \bigoplus \mathcal{H}_{\{i\}} \bigoplus \mathcal{H}_{\{j\}}$. Recall that

$$
\mathcal{H}_{\{i,j\}} = \text{span}\{1, \{\phi_{ib}\}_{b=1}^{B_i}, \{\phi_{jb}\}_{b=1}^{B_j}\}, \{\phi_{ib}\phi_{jb'}\}_{b\in[B_i], b'\in[B_j]}\}.
$$

Then, $f^{\mu_\otimes}_{\{i,j\}} \in \mathcal{H}_{\{i,j\}}$ and $\mathcal{H}^{\text{add}}_{\{i,j\}}$ is a closed convex subspace of $\mathcal{H}_{\{i,j\}}$. Therefore, by the Hilbert Projection Theorem, there exists unique $\bar{f}_{ij} \in \mathcal{H}^{\text{add}}_{\{i,j\}}$ and $f^\perp_{ij} \in \mathcal{H}_{\{i,j\}}$ such that

$$
\begin{aligned}
f^{\mu_\otimes}_{\{i,j\}} &= \bar{f}_{ij} + f^\perp_{ij} \quad \text{s.t.} \\
\langle g, f^\perp_{ij} \rangle_\mu &= 0 \quad \forall g \in \mathcal{H}^{\text{add}}_{\{i,j\}}.
\end{aligned}
\tag{31}
$$

Since $\text{span}\{1, \{\phi_{ib}\}_{b=1}^{B_i}, \{\phi_{jb}\}_{b=1}^{B_j}\}$ is a linearly independent basis of $\mathcal{H}^{\text{add}}_{\{i,j\}}$, there exists unique coefficients, $\Psi^i_{ij} \in \mathbb{R}^{1\times B_i}, \Psi^j_{ij} \in \mathbb{R}^{1\times B_j}, \Psi^0_{ij} \in \mathbb{R}$, such that $\bar{f}_{ij} = \Psi^i_{ij}\Phi_i^T(x_i) + \Psi^j_{ij}\Phi_j^T(x_j) + \Psi^0_{ij}$.

To complete the proof, we need to show that $\tilde{f}_{ij} = f^{\mu}_{\{i,j\}}, \tilde{f}_i = f^{\mu}_{\{i\}}, \tilde{f}_\emptyset = f^{\mu}_\emptyset$. It suffices to show that

$$\begin{aligned}
&\int_{x_i} \tilde{f}_i \ d\mu_i = 0 \\
&\int_{x_i,x_j} \tilde{f}_{ij} \ d\mu_i = 0 \\
&\int_{x_i,x_j} \tilde{f}_i \tilde{f}_{ij} \ d\mu(x_i, x_j) = 0.
\end{aligned} \tag{32}$$

The last two equalities in Eq. (32) follow directly from Eq. (31). For the first equality in Eq. (32), notice that

$$\begin{aligned}
\int_{x_i} \tilde{f}_i \ d\mu_i &= \mathbb{E}_{\mu_i} \tilde{f}_i \\
&= \mathbb{E}_{\mu_i} \left[f^{\mu\otimes}_{\{i\}} + \sum_{j>i} \Psi^i_{ij} \Phi_i + \sum_{j<i} \Psi^i_{ji} \Phi_i \right] \\
&= \mathbb{E}_{\mu_i} f^{\mu\otimes}_{\{i\}} + \sum_{j>i} \mathbb{E}_{\mu_i}[\Psi^i_{ij} \Phi_i] + \sum_{j<i} \mathbb{E}_{\mu_i}[\Psi^i_{ji} \Phi_i] \\
&= \sum_{j>i} \Psi^i_{ij} \mathbb{E}_{\mu_i}[\Phi_i] + \sum_{j<i} \Psi^i_{ji} \mathbb{E}_{\mu_i}[\Phi_i] \\
&= 0,
\end{aligned}$$

where the last equation follows from the fact that the components of Φ_i span $\mathcal{H}^o_{\{i\}}$ (and hence are all zero mean).

B.10 Proof of Proposition 6

As shown in the proof of Theorem 3, $\Psi^i_{ij} \in \mathbb{R}^{1 \times B_i}, \Psi^j_{ij} \in \mathbb{R}^{1 \times B_j}, \Psi^0_{ij} \in \mathbb{R}$ equal the unique set of coefficients such that $\bar{f}_{ij} = \Psi^i_{ij} \Phi_i + \Psi^j_{ij} \Phi_j + \Psi^0_{ij}$ for $\bar{f}_{ij}$ defined in Eq. (31) and also shown below:

$$\begin{aligned}
f^{\mu\otimes}_{\{i,j\}} &= \bar{f}_{ij} + f^{\perp}_{ij} \quad \text{s.t.} \\
&\langle g, f^{\perp}_{ij} \rangle_\mu = 0 \quad \forall g \in \mathcal{H}^{\text{add}}_{\{i,j\}}.
\end{aligned}$$

Let $y^{(w)}_{ij} = f^{\mu\otimes}_{\{i,j\}}(x^{(m)}_i, x^{(w)}_j)$ and $\epsilon^{(w)}_{ij} = f^{\perp}_{ij}(x^{(w)}_i, x^{(w)}_j)$, where $x^{(w)} \overset{\text{i.i.d.}}{\sim} \mu$. Then,

$$y^{(w)}_{ij} = \Psi^i_{ij} \Phi_i(x^{(w)}_i) + \Psi^j_{ij} \Phi_j(x^{(w)}_j) + \Psi^0_{ij} + \epsilon^{(w)}_{ij} \quad x^{(w)} \overset{\text{i.i.d.}}{\sim} \mu. \tag{33}$$

Then, Eq. (33) is a special case of the random design linear model under misspecification studied in Hsu et al. (2014). Hence, by Hsu et al. (2014, Theorem 11) we can consistently recover Ψ^i_{ij}, Ψ^j_{ij}, Ψ^0_{ij} by using ordinary least-squares. Hence Algorithm 4 recovers Ψ^i_{ij}, Ψ^j_{ij}, Ψ^0_{ij} as $W \to \infty$.

Appendix C. Literature Review

Finite Basis Expansion Methods. Stone (1994) introduced the hierarchical functional decomposition and derived statistical rates of convergence by approximating $\mathcal{H}$ using a finite B-spline tensor product basis. Huang (1998) later extended this result to general tensor product families such as wavelets, polynomials, etc. There have been a number of specific Bayesian and frequentist methods that fall within the general class of models described in Huang (1998); see, for example, Wei et al. (2019); Scheipl et al. (2012); Curtis et al. (2014); Ferrari and Dunson (2020b); Gustafson (2000). Unfortunately, since these methods explicitly generate the tensor product basis, they are computationally intractable as p increases beyond a few hundred or thousand covariates. In Radchenko and James (2010), the authors consider the $Q = 2$ setting, and develop the VANISH algorithm to fit nonlinear interaction models under a heredity constraint (i.e., interaction terms are only added if the main effects are selected). They authors use a finite basis to model the main and interaction effects but do not assume a tensor product basis. Unfortunately, their method does not scale well with larger p since the runtime is $O(p^2)$; see Step 0 of the VANISH algorithm on page 6. In Haris et al. (2016), the authors generalize VANISH (and several other algorithms) by using an alternating directions method of multipliers algorithm to fit interactions.

Linear models trivially fall within this class as well. For $Q = 1$, the Lasso and the many related techniques provide fast variable selection and estimation in high-dimensional linear models (Chen et al., 1998; Candes and Tao, 2007; Nakagawa et al., 2016). For $Q = 2$, the hierarchical Lasso (Bien et al., 2013) extends the Lasso to model interactions, and there have been many variants of this model; see, for example, Lim and Hastie (2015); Shah (2016). However, these methods take at least $O(p^2)$ time since they explicitly model all main and interaction effects. Other linear interaction methods assume that the interactions have a low-rank structure. This structure helps both statistically and computationally; see, for example, Rendle (2010); Ferrari and Dunson (2020a). However, this low-rank structure in the interaction effects might not always hold in practice.

Two-Stage & Forward-Stage Approaches. Instead of modeling interactions jointly, a common heuristic (similar in spirit to forward stepwise regression) is greedily adding interactions such as in multivariate additive regression splines (MARS) or GA2M (Lou et al., 2013). The iFORM algorithm proposed in Hao and Zhang (2014) is the middle-ground between MARS and fitting a model with all interactions terms included at the start. Specifically, iFORM starts with the empty model, and then proceeds by adding one more predictor at a time, where all interactions between the current active set of predictors are considered. Another common approach is performing computationally cheap variable selection methods designed for generalized additive models (e.g., Lasso or SpAM (Liu et al., 2008)) to identify a sparse set of relevant variables. By restricting to a small set of variables, one can then apply more computationally intensive interactions techniques such as RKHS ANOVA methods.

The approaches above requires some form of strong-hierarchy, namely that all interactions have non-zero main effects, to consistently identify the correct set of variables. While some problems have strong main effects, in other applications this may not be the case. For example, in genome-wide associate studies, fitting an additive-only model to predict an individual's height from genetics only has an R^2 of about 5% even though height is well-predicted by parents' heights (thought to be between $80\% - 90\%$) (Maher, 2008). This discrepancy, more generally called the problem of *missing heritability*, remains an open challenge in biology for understanding complex diseases based on genetics. One explanation for missing heritability is not modeling genetic interactions (Maher, 2008; Aschard, 2016; Slim et al., 2018; Greene et al., 2010). In other words, the main effects might be weak, or in the extreme case some genes might only have interaction effects. Hence, from a purely variable selection standpoint, modeling interactions could help better identify genes that are risk-factors for certain diseases.

In the orthogonal μ case, the statistical benefit from modeling interactions can be easily seen from the decomposition in Eq. (4). Suppose $Q = 2$ and that main effects total signal variance equals 5, the pairwise signal variance equals equals 90, and the noise variance equals 5. Then, the R^2 for an additive-only model is 5% while the R^2 for interaction model is 90%. Since the achievable signal increases (and necessarily the effective noise variance decreases), performing variable selection in a lower signal-to-noise regime might offset the statistical price of modeling more parameters.

Tree-Based Approaches. Tree-based methods (e.g., random forest, CART, gradient boosting) are often used for black-box prediction tasks. While these methods sometimes provide variable importance measures, it is unclear how to access the effects from the fitted prediction function and perform variable selection. Nevertheless, some authors have adapted tree-based methods to estimate effects and perform variable selection. For example, in Linero (2018), the authors modify the Bayesian additive regression trees method in Chipman et al. (2010) by placing Dirichlet priors on the splitting proportions of the regression tree prior to induce sparsity. While Linero (2018) perform variable selection by looking at posterior inclusion probabilities, it is unclear how to access the interaction effects from the fit. Other authors have adapted tree-based methods to estimate heterogeneous treatment effects (i.e., interactions between a treatment and set of covariates). For example, in Su et al. (2011), the authors modify the CART splitting rule to get better estimates of heterogeneous treatment effects. Similarly, in Krzykalla et al. (2020), the authors use a model-based recursive partitioning to identify covariates that most strongly interact with a particular treatment. In general, estimating treatment-by-covariate interactions is less computationally demanding since there are only $O(p)$ number of pairwise interactions between a single treatment and all covariates.

Kernel Methods. Many of the functional ANOVA methods described in Section 6 use kernels to model nonlinear interaction effects. Unfortunately, as we discuss in Section 6, these methods are computationally intractable for moderate p when $Q > 1$. Some kernel methods used to identify interactions fall under the general area of "multiple kernel learning," where the goal is to learn some weighted combination of kernels (Lanckriet et al., 2004b; Bach et al., 2004). *Hierarchical kernel learning*, for example, is a multiple kernel learning method that learns nonlinear interactions via hierarchy conditions encoded in a directed acyclic graph (Bach, 2008). This hierarchy condition translates into a method similar to the greedy forward-stage methods discussed above where higher-order interactions are added only when all lower-interactions are present. For R selected kernels, the runtime stated in Bach (2008) is $O(N^3R + N^2Rp^2 + N^2R^2p)$. The quadratic dependence on p makes this method unsuitable for larger p problems. Duvenaud et al. (2013) considers a greedy kernel search based on adding and multiplying a base set of kernels together. Since the multiplication of two kernel corresponds to an interaction, this method has the flexibility to model interaction effects. However, the focus in Duvenaud et al. (2013) is on prediction and understanding the structure of the fitted kernel instead of estimating effects and performing variable selection.

Comparison with Agrawal et al. (2019). The structure of the SKIM-FA prior in Eq. (8) generalizes the prior used in Agrawal et al. (2019) to handle non-linear effects. However, in Agrawal et al. (2019), the authors use a regularized horseshoe prior to achieve sparsity in κ. While a regularized horseshoe prior does not lead to exact sparsity in κ, the authors in Agrawal et al. (2019) were still able to develop an $O(p)$ variable selection procedure by exploiting strong-hierarchy, namely that interactions only occur among selected main effects. In the current work, we do not make any strong-hierarchy assumption. Hence, to develop an $O(p)$ variable selection procedure, we need *exact* sparsity in κ; see Section 5 for details.

In terms of computational complexity, Agrawal et al. (2019) fit linear interaction models in $O(pN^2 + N^3)$ time per iteration, which has the same asymptotic complexity as Algorithm 1. However, they use Hamiltonian Monte Carlo (HMC) to perform inference. Each HMC step requires computing and inverting an $N \times N$ kernel matrix many times. Hence, their method takes hours to complete when p and N are larger than 500. Due to this computational intensity, we do not benchmark against their method.

Appendix D. Zero Mean Kernels and Finite-Basis Functions

In this section, we show how we construct k_i, i.e., the reproducing kernel for $\mathcal{H}^o_{\{i\}}$. We construct k_i by first generating a finite-dimensional basis for $\mathcal{H}_{\{i\}}$. Then, we normalize each basis function to be zero mean and unit variance so that the normalized basis functions span $\mathcal{H}^o_{\{i\}}$. For a more general approach to construct zero mean kernels (e.g., even when $\mathcal{H}_{\{i\}}$ is infinite-dimensional) see Durrande et al. (2013).

Construction of $\mathcal{H}^o_{\{i\}}$. For each covariate dimension i, consider a set of linearly independent basis functions $\{\phi_{ib}\}_{b=1}^{B_i}$ such that

$$\mathcal{H}_{\{i\}} = \text{span}\{1, \phi_{i1}, \cdots, \phi_{iB_i}\}.$$

Let $\tilde{\phi}_{ib} = \frac{\phi_{ib} - \mathbb{E}_\mu[\phi_{ib}]}{\sqrt{\text{Var}_\mu[\phi_{ib}]}}$. Then,

$$\mathcal{H}^o_{\{i\}} = \text{span}\{\tilde{\phi}_{i1}, \cdots, \tilde{\phi}_{iB_i}\}, \quad \Phi_i := [\tilde{\phi}_{i1}, \cdots, \tilde{\phi}_{iB_i}].$$

Hence, $k_i(x_i, \tilde{x}_i) = \Phi_i(x_i)^T \Phi_i(\tilde{x}_i)$ is the reproducing kernel for $\mathcal{H}^o_i$. In many instances, we do not actually know the joint distribution of the covariates. In this case, we approximate μ

with the empirical distribution $\hat{\mu}$ of the datapoints:

$$\begin{aligned}\tilde{\phi}_{ib} &= \frac{\phi_{ib} - \mathbb{E}_{\hat{\mu}}[\phi_{ib}]}{\sqrt{\mathrm{Var}_{\hat{\mu}}[\phi_{ib}]}} \quad \text{s.t.} \\ \hat{\mu} &= \frac{1}{N}\sum_{n=1}^{N} \delta_{x^{(n)}} \\ \mathbb{E}_{\hat{\mu}}[\phi_{ib}] &= \frac{1}{N}\sum_{n=1}^{N} \phi_{ib}(x_i^{(n)}) \\ \mathrm{Var}_{\hat{\mu}}[\phi_{ib}] &= \frac{1}{N}\sum_{n=1}^{N} \phi_{ib}^2(x_i^{(n)}) - \mathbb{E}_{\hat{\mu}}[\phi_{ib}].\end{aligned}$$

Note. for the experiments in Section 7, we use a natural cubic spline basis with 5 knots at the quantiles to generate each $\mathcal{H}_{\{i\}}$ for SKIM-FA and SPAM-2Stage.

Practical Considerations for Picking Basis Functions. While any set of basis functions can be used to generate $\mathcal{H}^o_{\{i\}}$ in principle, we provide several suggestions below; see Chapter of Hastie et al. (2001) for a more in-depth review.

- If x_i is a seasonal covariate, use a wavelet basis to generate $\mathcal{H}^o_{\{i\}}$.
- If x_i is categorical, let $\mathcal{H}^o_{\{i\}}$ equal the one-hot encoding of x_i.
- if x_i is continuous, use a polynomial spline basis.
    - If extrapolation beyond the data is a concern, use a natural cubic spline basis (which enforces linearity beyond the boundary knots).

Appendix E. Additional Algorithmic Details

To fit SKIM-FA, we set the number of iterations $T = 2000$, learning rate $\gamma = 0.1$, and cross-validation batch size $M = 0.2N$ in Algorithm 1. We let the truncation level c in Algorithm 1 depend on the iteration number t for $1 \leq t \leq T$. Empirically, we find gradually increasing c as a function of t works well for accurately selecting the correct covariates and inducing sparsity in κ. As outlined in Algorithm 5, we suggest the following schedule for c:

When $t < 500$, $c = 0$ in Algorithm 5. Hence, since $\kappa_i^{(t)} = \max(U_i^{(t)} - c, 0)$, $\kappa_i^{(t)} \neq 0$ for $t \leq 500$ and $i \in [p]$. At iteration 500, we drop the bottom 25th% of covariates (determined by their importance measure $U_i^{(t)}$). Specifically, $\kappa^{(500)}$ has 25% of its entries equal to zero. For subsequent iterations, we take the previous trunction level c_t and increase that level by a factor of $(1 + r)$ until c_t reaches γ. If c_{t-1} is larger than γ, we set c_t equal to c_{t-1}.

Appendix F. SKIM-FA Extensions

F.1 Beyond Gaussian Responses

Throughout we have assumed that $y \mid x$ is drawn from a Gaussian distribution. In general, we may assume that the response belongs to an exponential family, which allows us to model,

Algorithm 5 Scheduler for Truncation Level c

```
procedure TruncScheduler($U^{(t)}$, $c_{t-1}$, $t$, $r = .01$, $\gamma = .75$)
    if $t < 500$ then return 0
    end if
    if $t = 500$ then return $q_{25}(U_1^{(t)}, \cdots, U_p^{(t)})$  ▷ Take the 25th% of the components in $U^{(t)}$
    end if
    if $t > 500$ then return $\max(\min((1+r)c_{t-1},\ \gamma),\ c_{t-1})$
    end if
end procedure
```

for example, count, binary, and exponential response data. For non-Gaussian responses, there does not exist an analytical solution to Eq. (1). Nevertheless, a combination of reweighted least squares and the Newton Raphson method can be used to iteratively solve Eq. (1) when $\mathcal{H}$ is an RKHS; see Cawley et al. (2007) for details. Since the results in Cawley et al. (2007) are not unique to the specific kernel used, we can extend SKIM-FA beyond Gaussian errors. However, from an implementation standpoint, this extension might be challenging since we must take gradients of the kernel hyperparameters during the Newton Raphson optimization steps. Hence, a similar framework used in Margossian et al. (2020) might be needed for the practical implementation of SKIM-FA to general responses.

F.2 Change-of-Basis Formula For General Q and Arbitrary Measures

We discuss how to extend the change-of-basis formula in Algorithm 4 to general Q. The general idea is as follows:

1. (PROJECT) Project each Q way interaction onto the space spanned by all lower-order interactions

2. (UPDATE) Subtract out the lower-order variation from the Q way interactions, and add back this projected variation to the lower-order interactions

3. (RECURSE) Repeat Step 1 and Step 2 for the $Q - 1$ way interactions, then $Q - 2$ interactions, until the highest order interaction is the constant term

Line 6 in Algorithm 4 is the analogue of the PROJECT step, and Lines 7-9 in Algorithm 4 is the analogue of the UPDATE STEP. Under Assumption 2, we describe how the PROJECT-UPDATE-RECURSE methodology can be used to move between two arbitrary functional ANOVA decompositions with respect to μ' and μ.

Suppose we are given the functional ANOVA decomposition with respect to μ': $f = \sum_{V:|V|\leq Q} f'_V$, where $f'_V \in \mathcal{H}^o_{V,\mu'}$. Given each f'_V, we would like re-express f as $\sum_{V:|V|\leq Q} f_V$, where $f_V \in \mathcal{H}^o_{V,\mu}$. Such a decomposition exists since $\mathcal{H}^o_{V,\mu'} = \mathcal{H}^o_{V,\mu}$ by Assumption 2 and Lemma 3. We define the projection operator of a function with interactions in $A \subset [p]$, $|A| \leq Q$ below:

$$\mathrm{proj}_{\mathcal{F}_A,\mu}[f_A] \coloneqq \sum_{V:V\subsetneq A} g_{V_A}, \quad g_{V_A} \in \mathcal{H}^o_{V,\mu}, \quad \mathcal{F}_A = \bigoplus_{V:V\subsetneq A} \mathcal{H}^o_{V,\mu} \tag{34}$$

The UPDATE step involves adding and subtracting g_{V_A} from the interactions. Since each component g_{V_A} in $\mathrm{proj}_{\mathcal{F}_A,\mu}[f_A]$ is unique by the Hilbert Projection Theorem, this procedure is well defined. We summarize our algorithm in Algorithm 6. The proof of correctness, namely that Algorithm 6 recovers the functional ANOVA decomposition of f with respect to μ follows by using the same proof strategy in Theorem 3.

Algorithm 6 Change of Basis Formula for General Q and Measures

```
procedure ReExpressANOVA($\sum_{V:|V|\leq Q} f'_V$, $\mu$)
    Let $I$ equal the highest order interaction in $\sum_{V:|V|\leq Q} f'_V$
    if $I = 0$ then return $f'_\emptyset$
    end if
    For all $A \subset [p], |A| = I$, compute $\mathrm{proj}_{\mathcal{F}_A,\mu}[f'_A]$.
    For all $A \subset [p], |A| = I$, let $f_A = f'_A - \mathrm{proj}_{\mathcal{F}_A,\mu}[f'_A]$    ▷ Update higher-order interaction effects
    For all $V \subset [p], |V| < I$, let $f'_V = f'_V + \sum_{A:|A|=I, V \subsetneq A} g_{V_A}$    ▷ Update lower-order interaction effects
    return $\sum_{A:|A|=I} f_A$ + ReExpressANOVA($\sum_{V:|V|\leq I-1} f'_V$, $\mu$)    ▷ RECURSE step
end procedure
```

Computing the Projection Operator via Monte-Carlo. We show how to compute $\mathrm{proj}_{\mathcal{F}_A,\mu}[f'_A]$ via Monte-Carlo as we do in Algorithm 4 when $Q = 2$. To this end, let the components of $\Phi_A = \bigotimes_{j \in A} \Phi_j$. Let $d^* = \max_{V:|V|\leq Q} \dim(\Phi_A)$. For $1 \leq w \leq W$ randomly sample $x^{(w)} \overset{\text{i.i.d.}}{\sim} \mu$, where $W > d^*$. Define

$$X_A = [\Phi_V(x_V^{(1)}) \cdots \Phi_V(x_V^{(W)})]^T_{V:V \subsetneq A} \tag{35}$$

where $\Phi_\emptyset = 1$. Let $f'_{A,W} = [f'_A(x_V^{(1)}) \cdots f'_A(x_V^{(W)})]^T$. Then,

$$\hat{g}_{V_A}(\cdot) = \hat{\Psi}_V^T \Phi_V(\cdot), \quad \text{where} \quad [\hat{\Psi}_V]^T_{V:V \subsetneq A} = (X_A^T X_A)^{-1} X_A^T f'_{A,W}. \tag{36}$$

By following nearly an identical proof of Proposition 6, $\hat{g}_{V_A}(\cdot) \to g_{V_A}(\cdot)$ as $W \to \infty$.

F.3 Consistency Guarantees

Proving selection consistency is beyond the scope of the current paper. Given selection consistency, however, estimation consistency follows from the work in Huang (1998), where the author examines the consistency properties of fitting functional ANOVA models via a finite-dimensional tensor product basis (i.e., as we do in SKIM-FA).

To make this connection more concrete, suppose the number of correct covariates S is fixed (and does not depend on p or N). If selection consistency holds, then SKIM-FA consistently recovers S with probability one as $N, p \to \infty$. To simplify the analysis, suppose we use sample splitting, where the first $\frac{N}{2}$ datapoints are used for selection, and the remaining $\frac{N}{2}$ datapoints are used to re-estimate the effects among the selected covariates. For any desired probability, N can be chosen sufficiently large such that SKIM-FA selects all S correct covariates exceeding the chosen probability. Since S is fixed, the results in Huang (1998) apply when estimating the effects on the held-out set of $\frac{N}{2}$ datapoints. In particular, Huang (1998) provides the rate at which N must grow as a function of the size and smoothness of the tensor product basis generated from the S selected covariates; see Theorem 3 and Corollary 2 of Huang (1998) for different rates of convergence.

Appendix G. MARS ANOVA Procedure

We show how to perform the functional ANOVA decomposition of $\hat{f}$ with respect to $\hat{\mu}_\otimes = \hat{\mu}_1 \otimes \cdots \otimes \hat{\mu}_p$, where $\hat{f}$ denotes the regression function fit from MARS and μ_i the empirical distribution of covariate i: $\hat{\mu}_i = \frac{1}{N}\sum_{n=1}^{N} \delta_{x_i^{(n)}}$. Under $\hat{\mu}_\otimes$, the functional ANOVA decomposition of $\hat{f}$ equals

$$\begin{aligned} \hat{f}_\emptyset &= \mathbb{E}_{\hat{\mu}_\otimes}[\hat{f}] \\ \hat{f}_{\{i\}}(x_i) &= \mathbb{E}_{\hat{\mu}_\otimes}[\hat{f} \mid x_i = x_i] - \hat{f}_\emptyset \\ \hat{f}_{\{i,j\}}(x_i, x_j) &= \mathbb{E}_{\hat{\mu}_\otimes}[\hat{f} \mid x_i = x_i, x_j = x_j] - \hat{f}_{\{i\}}(x_i) - \hat{f}_{\{j\}}(x_i) - \hat{f}_\emptyset, \end{aligned} \tag{37}$$

which is also shown in Durrande et al. (2013, Equation 5). We show how to compute each of the expectations in Eq. (37). The intercept $\hat{f}_\emptyset$ equals the sample average of the fitted values (i.e., $\hat{f}$ applied to each of the N training datapoints). Let X denote the $N \times p$ matrix of training data. Let X^i equal the matrix obtained by setting all values in the ith column of X equal to x_i and the remaining columns unchanged. Then,

$$\mathbb{E}_{\hat{\mu}_\otimes}[\hat{f} \mid x_i = x_i] = \frac{1}{N}\sum_{n=1}^{N} \hat{f}(X_n^i),$$

where X_n^i is the nth row of X_n^i. Similarly, let X^{ij} equal the matrix obtained by setting all values in the ith and jth columns of X equal to x_i and x_j respectively, and the remaining columns unchanged. Then,

$$\mathbb{E}_{\hat{\mu}_\otimes}[\hat{f} \mid x_i = x_i, x_j = x_j] = \frac{1}{N}\sum_{n=1}^{N} \hat{f}(X_n^{ij}).$$

Appendix H. Additional Experimental Details

H.1 Fitting benchmark methods

SPAM-2Stage: we perform variable selection by fitting a sparse additive model (SpAM) (Liu et al., 2008) to the data. We use the `sam` package in `R`. Since `sam` does not provide a default way to select the L_1 regularization strength, we use 5-fold cross-validation. For estimation, we generate all main and interaction effects among the subset of covariates selected by SpAM. We calculate these effects by taking pairwise products of univariate basis functions generated from a natural cubic spline basis with 5 total knots; see Appendix D for details. We estimate the basis coefficients (and hence effects) using ridge regression, where again we use 5-fold cross-validation to pick the L_2 regularization strength.

Multivariate Additive Regression Splines (MARS): we use the `python` implementation of MARS (Friedman, 1991) in `py-earth`.

Hierarchical Lasso (HierLasso): we use the implementation of HierLasso (Lim and Hastie, 2015) in the authors' `R` package `glinternet`. Since Lim and Hastie (2015) use cross-validation to pick the L_1 regularization strength, we similarly use 5-fold cross-validation.

Pairs Lasso: we fit the Lasso on the expanded set of features $\{x_i\}_{i=1}^p$ and $\{x_i x_j\}_{i,j=1}^p$. We fit the Lasso using the `python` package `sklearn`, and use 5-fold cross-validation to select the L_1 regularization strength.

H.2 Evaluation Criteria

Variable Selection Evaluation Metrics. We consider both the power to select correct covariates and avoid incorrect ones. *# Correct Selected* counts the number of covariates correctly selected by the method. Higher is better. *# Wrong Selected* counts the number of covariates incorrectly selected by the method (i.e., Type I error). Lower is better. *# Correct Not Selected* counts the number of covariates that belong to the true model but were not selected by the method (i.e., Type II error). Lower is better.

Estimation Evaluation Metrics. We evaluate how well a method estimates main effects and interaction effects. Instead of looking only at the total mean squared estimation error, we break this error into multiple buckets to understand what bucket drives the majority of the error. Lower is better for all of the following quantities. *Correct Selected SSE (Main)* takes the sum of squared errors (SSE) between each estimated main effect component and true main effect component. This sum equals $\sum_{i \in S_1} \|f_i^* - \hat{f}_i\|_\mu^2$, where S_1 is the set of correctly identified main effects, $\hat{f}_i$ is the estimated main effect, and f_i^* is the true main effect. *Correct Not Selected SSE (Main)* takes the sum of squared norms of main effects not selected. This sum equals $\sum_{i \in S_2} \|f_i^*\|_\mu^2$, where S_2 is the set of correct main effects not selected. *Wrong Selected SSE (Main)* takes the sum of squared norms of main effect components incorrectly selected. This sum equals $\sum_{i \in S_3} \|\hat{f}_i\|_\mu^2$, where S_3 is the set of incorrect main effects selected. *Correct Selected SSE (Pair)*, *Correct Not Selected SSE (Pair)*, and *Wrong Selected SSE (Pair)* are the same as the analogous main effect metrics but instead considers interaction effects. *Total SSE* equals the sum of the 6 buckets above and *Total SSE / Signal Variance* equals the relative estimation error, i.e., Total SSE divided by the true signal variance.

Appendix I. Additional Experimental Results

Method	**# Covariates**	**# Correct Selected**	**# Wrong Selected**	**# Correct Not Selected**
HierLasso	250	4	0	1
SKIM -FA	250	4	0	1
Pairs Lasso	250	4	5	1
SPAM-2Stage	250	5	52	0
MARS	250	5	58	0
SKIM-FA	500	5	13	0
SPAM-2Stage	500	5	28	0
Pairs Lasso	500	4	39	1
HierLasso	500	4	48	1
MARS	500	5	64	0
SKIM-FA	1000	3	0	2
HierLasso	1000	4	5	1
Pairs Lasso	1000	4	6	1
SPAM-2Stage	1000	5	15	0
MARS	1000	5	70	0

Table I.1: Variable Selection Performance for Main Effects Only Setting.

Method	**p**	**Correct Selected SSE (Main)**	**Correct Not Selected SSE (Main)**	**Wrong Selected SSE (Main)**	**Correct Selected SSE (Pair)**	**Correct Not Selected SSE (Pair)**	**Wrong Selected SSE (Pair)**	**Total SSE**	**Total SSE ÷ Signal Variance**
MARS-EMP	250	0.31	0.00	2.11	0.00	0.00	2.24	4.66	0.23
SPAM-2Stage	250	2.77	0.00	1.99	0.00	0.00	0.08	4.84	0.24
SKIM-FA	250	2.75	4.02	0.00	0.00	0.00	0.39	7.16	0.36
MARS-VANILLA	250	73.17	0.00	2.37	0.00	0.00	9.89	85.43	4.27
SPAM-2Stage	500	2.82	0.00	1.25	0.00	0.00	0.04	4.11	0.21
SKIM-FA	500	2.75	0.00	0.49	0.00	0.00	1.40	4.64	0.23
MARS-EMP	500	0.38	0.00	2.37	0.00	0.00	2.19	4.95	0.25
MARS-VANILLA	500	35.67	0.00	3.62	0.00	0.00	9.22	48.51	2.43
SPAM-2Stage	1000	2.67	0.00	0.78	0.00	0.00	0.02	3.46	0.17
MARS-EMP	1000	0.45	0.00	2.68	0.00	0.00	2.39	5.51	0.28
SKIM-FA	1000	2.70	8.10	0.00	0.00	0.00	0.24	11.03	0.55
MARS-VANILLA	1000	16.14	0.00	1.56	0.00	0.00	10.33	28.02	1.40

Table I.2: Estimation Performance for Main Effects Only Setting.

Method	# of Covariates	# Correct Selected	# Wrong Selected	# Correct Not Selected
SKIM-FA	250	5	1	0
HierLasso	250	5	25	0
SPAM-2Stage	250	5	37	0
MARS	250	5	84	0
Pairs Lasso	250	5	89	0
SKIM-FA	500	5	0	0
SPAM-2Stage	500	5	29	0
HierLasso	500	5	30	0
MARS	500	5	69	0
Pairs Lasso	500	5	182	0
SKIM-FA	1000	5	0	0
SPAM-2Stage	1000	5	15	0
HierLasso	1000	5	40	0
MARS	1000	5	71	0
Pairs Lasso	1000	5	213	0

Table I.3: Variable Selection Performance for Equal Main and Interaction Effects Setting.

Method	**p**	**Correct Selected SSE (Main)**	**Correct Not Selected SSE (Main)**	**Wrong Selected SSE (Main)**	**Correct Selected SSE (Pair)**	**Correct Not Selected SSE (Pair)**	**Wrong Selected SSE (Pair)**	**Total SSE**	**Total SSE ÷ Signal Variance**
SKIM-FA	250	1.62	0.00	0.08	0.52	0.00	0.17	2.39	0.12
SPAM-2Stage	250	1.63	0.00	1.72	8.84	0.00	0.11	12.30	0.62
MARS-EMP	250	0.71	0.00	4.44	2.17	0.00	5.69	13.01	0.65
MARS-VANILLA	250	24.91	0.00	5.28	17.13	0.00	18.03	65.35	3.27
SKIM-FA	500	1.52	0.00	0.00	0.41	0.00	0.00	1.93	0.10
SPAM-2Stage	500	1.62	0.00	3.74	2.16	0.00	5.47	12.99	0.65
MARS-EMP	500	0.71	0.00	4.69	1.63	0.96	6.57	14.56	0.73
MARS-VANILLA	500	11.36	0.00	13.22	15.62	0.96	23.55	64.71	3.24
SKIM-FA	1000	1.54	0.00	0.00	0.29	0.00	0.00	1.82	0.09
SPAM-2Stage	1000	1.67	0.00	1.07	0.41	0.00	2.16	5.31	0.27
MARS-EMP	1000	0.61	0.00	3.84	1.70	0.00	2.52	8.67	0.43
MARS-VANILLA	1000	454.88	0.00	3.16	21.46	0.00	13.22	492.72	24.64

Table I.4: Estimation Performance for Equal Main and Interaction Effects Setting.

Method	**# Covariates**	**# Correct Selected**	**# Wrong Selected**	**# Correct Not Selected**
SKIM-FA	250	5	6	0
MARS	250	5	75	0
SPAM-2Stage	250	4	77	1
Pairs Lasso	250	5	123	0
HierLasso	250	5	160	0
SKIM-FA	500	5	16	0
SPAM-2Stage	500	1	21	4
HierLasso	500	5	62	0
Pairs Lasso	500	5	85	0
MARS	500	2	132	3
SKIM-FA	1000	5	9	0
SPAM-2Stage	1000	1	41	4
MARS	1000	5	75	0
HierLasso	1000	5	120	0
Pairs Lasso	1000	5	144	0

Table I.5: Variable Selection Performance for Weak Main Effects Setting.

Method	**p**	**Correct Selected SSE (Main)**	**Correct Not Selected SSE (Main)**	**Wrong Selected SSE (Main)**	**Correct Selected SSE (Pair)**	**Correct Not Selected SSE (Pair)**	**Wrong Selected SSE (Pair)**	**Total SSE**	**Total SSE ÷ Signal Variance**
SKIM-FA	250	0.45	0.00	0.95	0.73	0.00	0.77	2.89	0.14
MARS-EMP	250	1.46	0.00	4.02	4.83	0.00	4.67	14.97	0.75
SPAM-2Stage	250	0.09	0.05	2.22	10.72	7.73	0.42	21.23	1.06
MARS-VANILLA	250	22497.35	0.00	7.31	148073.29	0.00	18.55	170596.50	8529.83
SKIM-FA	500	0.69	0.00	2.05	1.50	0.00	1.37	5.61	0.28
SPAM-2Stage	500	0.27	0.20	4.09	0.00	19.46	0.08	24.11	1.21
MARS-EMP	500	0.41	0.15	21.92	0.00	19.46	15.56	57.51	2.88
MARS-VANILLA	500	0.10	0.15	323788.65	0.00	19.46	324588.33	648396.70	32419.83
SKIM-FA	1000	0.72	0.00	1.37	0.61	0.00	0.63	3.33	0.17
MARS-EMP	1000	0.67	0.00	5.86	3.37	0.00	5.63	15.52	0.78
SPAM-2Stage	1000	0.16	0.20	6.69	0.00	18.33	0.31	25.69	1.28
MARS-VANILLA	1000	23.62	0.00	3.18	23.16	0.00	15.43	65.39	3.27

Table I.6: Estimation Performance for Weak Main Effects Setting.

Effect	Signal Variance
Hour	0.382
Air Temp.	0.104
Humidity	0.024
Windspeed	0.002
Hour x Air Temp.	0.047
Hour x Humidity	0.01
Hour x Windspeed	0.002
Air Temp. x Humidity	0.012
Air Temp. x Windspeed	0.005
Humidity x Windspeed	0.003

Table I.7: Proxy Ground Truth Effects and Signal Variances for the Bike Sharing Data Set.

I.1 Appending Irrelevant but Real Covariates to the Bike Sharing Data Set

In Section 7, we appended fake covariates drawn from a Uniform(0, 1) distribution to the Bike Sharing Data Set for various choices of p_{noise}. In many applications, however, covariates are correlated and this correlation structure might affect the performance of a method. To create a design matrix with a non-trivial correlation structure, we start by taking a completely different data set, namely the SECOM data set from the UCI Machine Learning repository which contains 591 covariates related to semi-conductor manufacturing.[7] Then, we append these covariates to the Bike Sharing data set. Since these two data sets are independent, the covariates in the SECOM data set play the same role as the synthetic fake covariates in Section 7 (i.e., should not be selected) but now have a real correlation structure. Appendix I.1 and Table I.11 summarize how each method performs in terms of variable selection and estimation, respectively.

I.2 Impact of Correlated Predictors on the Functional ANOVA for the Bike Sharing Data Set

We perform the same analysis as in Section 7.5 but for the Bike Sharing data set in Fig. I.1. Unlike the Concrete Compressive Strength data set, however, we do not see a large difference between the two functional ANOVA decompositions for the Bike Sharing data set in Fig. I.1.

I.3 Obesity Gene-Expression and SNP Data Set: Additional SKIM-FA Fit Details

Since the number of datapoints is small ($N = 87$), the number of datapoints in between any adjacent knots in a spline basis is small. Hence, we instead fit a linear interaction model for this data set using SKIM-FA. Since we fit a linear interaction model, we can examine the regression coefficients to understand the fit. Table I.12 summarizes the 31 variables selected by SKIM-FA and their estimated main effects.

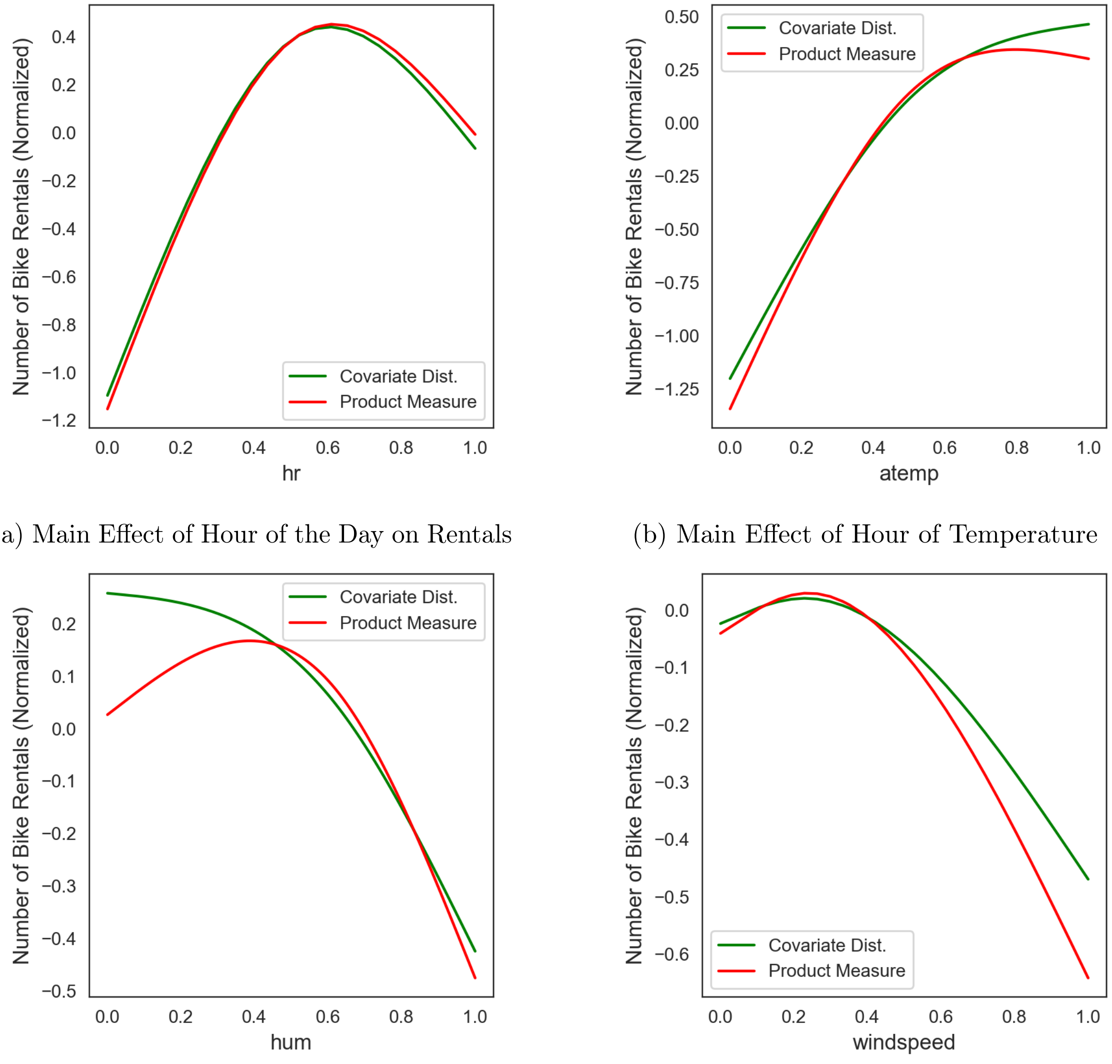
[7] We only consider 432 continuous covariates (with non-zero variance) in the SECOM data set.

Method	# Covariates	# Original Selected	# Wrong Selected
SKIM-FA	250	2	0
HierLasso	250	3	7
Pairs Lasso	250	3	29
MARS	250	3	96
SPAM-2Stage	250	4	97
SKIM-FA	500	2	0
HierLasso	500	3	8
SPAM-2Stage	500	3	22
Pairs Lasso	500	3	39
MARS	500	4	109
SKIM-FA	1000	3	0
HierLasso	1000	3	5
SPAM-2Stage	1000	3	8
Pairs Lasso	1000	3	76
MARS	1000	3	119

Table I.8: Variable Selection Performance for the Bike Sharing Data Set.

Method	# Noise	Correct Selected SSE (Main)	Correct Not Selected SSE (Main)	Wrong Selected SSE (Main)	Correct Selected SSE (Pair)	Correct Not Selected SSE (Pair)	Wrong Selected SSE (Pair)	Total SSE
SKIM-FA	250	0.15	0.027	0	0.019	0.038	0	0.233
SPAM-2Stage	250	0.149	0	0.172	0.091	0	0.01	0.422
MARS-EMP	250	0.209	0.002	0.476	0.052	0.026	0.344	1.11
MARS-Vanilla	250	6.522	0.002	1.644	1.036	0.026	2.2	11.431
SKIM-FA	500	0.148	0.027	0	0.019	0.038	0	0.231
SPAM-2Stage	500	0.15	0.002	0.057	0.081	0.009	0.002	0.302
MARS-EMP	500	0.225	0	0.529	0.052	0.026	0.3	1.131
MARS-Vanilla	500	5.564	0	0.5	1.037	0.026	2.085	9.212
SKIM-FA	1000	0.145	0.002	0	0.107	0.009	0	0.263
SPAM-2Stage	1000	0.149	0.002	0.027	0.081	0.009	0.000	0.269
MARS-EMP	1000	0.214	0.002	0.485	0.054	0.026	0.245	1.026
MARS-Vanilla	1000	6.556	0.002	0.796	0.947	0.026	1.882	10.209

Table I.9: Estimation Performance for the Bike Sharing Data Set.



(a) Main Effect of Hour of the Day on Rentals

(b) Main Effect of Hour of Temperature

(c) Main Effect of Humidity

(d) Main Effect of Windspeed

Figure I.1: Effect of Correlated Predictors on the Bike Sharing Data Set

Method	# Covariates	# Original Selected	# Wrong Selected
SKIM-FA	432	2	0
HierLasso	432	3	1
SPAM-2Stage	432	0	0
Pairs Lasso	432	3	14
MARS	432	3	97

Table I.10: Variable Selection Performance for the Bike Sharing-SECOM Data Set.

Method	# Noise	Correct Selected SSE (Main)	Correct Not Selected SSE (Main)	Wrong Selected SSE (Main)	Correct Selected SSE (Pair)	Correct Not Selected SSE (Pair)	Wrong Selected SSE (Pair)	Total SSE
SKIM-FA	432	0.137	0.026	0	0.016	0.029	0	0.208
SPAM-2Stage	432	0	0.549	0	0	0.074	0	0.623
MARS-EMP	432	0.212	0.001	7.416	0.049	0.018	6.113	13.810
MARS-Vanilla	432	3.876	0.001	85.369	1.704	0.0178	104.405	195.373

Table I.11: Estimation Performance for the Bike Sharing-SECOM Data Set.

We report the 10 strongest interaction effects in Table I.13. We see that SKP1 has the largest number of strong interaction effects follows by IRSP2.

I.4 Sensitivity to Non-Compactness and Sparse Interaction Effects

To test the sensitivity of SKIM-FA to the compactness assumption in Theorem 1 we instead draw covariates each independently from a Gaussian distribution. Since a Gaussian distribution has support on all of $\mathbb{R}$, the covariates do not belong to a compact set. On an unbounded set, the exponential function has an infinite mean. Hence, we replace the exponential trend in Section 7.3 with a cubic trend. On an unbounded set, to model a sine trend, we would need to use a wavelet basis. Since we have only have support for a spline basis in the current version of our package, we replace the sine trend in Section 7.3 with a leaky rectified linear unit (ReLU) trend, which is often used in neural networks. We keep the linear, logistic, and quadatic effects.

We let y depend on the first 5 covariates; the remaining 995 covariates are taken as noise covariates that we do not want to select. Hence, we consider a total of $p = 1000$ covariates. We let the main effects equal the sum of the 5 trends discussed above, where the ith trend is applied to covariate i. To additionally test the impact of not having all interactions present, we only consider 5 out of the 10 possible pairwise interactions: linear-logistic, leaky ReLU-linear, leaky ReLU-quadratic, logistic-quadratic, and cubic-logistic. We select a noise variance such that the $R^2 = \frac{\sigma^2_{\text{signal}}}{\sigma^2_{\text{signal}} + \sigma^2_{\text{noise}}} = 0.8$, where $\sigma^2_{\text{signal}} = \langle f^*, f^* \rangle_\mu$. We generate a total of $N = 1000$ datapoints.

In terms of variable selection performance, SKIM-FA selects all 5 true covariates and 0 incorrect covariates. We summarize the estimation performance below:

- Corrected Selected SSE (Main): .95
- Corrected Not Selected SSE (Main): 0
- Wrong Selected SSE (Main): 0
- Correct Selected SSE (Pair): 2.17
- Correct Not Selected SSE (Pair): 0
- Wrong Selected SSE (Pair): .94
- Total SSE: 4.05

Since SKIM-FA considers all pairwise interactions among selected covariates, SKIM-FA estimates 5 incorrect interactions. However, the total variance of these 5 incorrect interactions estimated by SKIM-FA is only .94. Hence, SKIM-FA shrinks all 5 incorrect interactions close to 0.

Each true main and pairwise effect has variance 2. Since all covariates are independent, the total signal variance of the main and pairwise interaction effects is each 10. Hence, the normalized SSE for the true main effects is $.95/10 = .095$ and $2.17/10 = .217$ for the true pairwise effects.

Effect	**Coeff.**
GSC	1.40
IRS2	-1.04
EOGT	1.03
SNORA71B	-1.00
SPATA20	-0.95
AK299501	-0.80
CLIC5	0.77
SKP1	0.77
SETDB2	-0.77
BTN2A3P	0.76
SAP130	0.76
KISS1R	0.69
ZBED3	-0.66
AQP11	-0.64
IYD	0.59
LOC100287177	-0.56
TMEM74B	0.47
ATP2B3	-0.44
LOC283070	0.42
IRX1	0.39
RPA4	-0.34
TSHZ3	0.31
INTS4	-0.30
ALDH1A2	-0.30
PCDH8	0.30
FBN2	0.29
KLK7	-0.28
FBXL12	-0.27
SEMA6B	0.24
SLC25A13	-0.00

Table I.12: Main Effects Selected by SKIM-FA on the Obesity Gene-Expression and SNP Data Set

Effect	**Coeff.**
(SKP1, SETDB2)	-0.14
(SKP1, ZBED3)	0.13
(RPA4, SETDB2)	0.12
(IRS2, RPA4)	0.11
(IRS2, SNORA71B)	-0.11
(SKP1, RPA4)	-0.10
(SNORA71B, RPA4)	-0.08
(RPA4, ZBED3)	0.08
(SKP1, IRS2)	-0.06
(RPA4, SAP130)	0.06

Table I.13: Interaction Effects Selected by SKIM-FA on the Obesity Gene-Expression and SNP Data Set (10 Strongest Interactions Shown)

References

Consortium 1000 Genomes Project. A global reference for human genetic variation. *Nature*, 526, 2015.

Raj Agrawal, Brian Trippe, Jonathan Huggins, and Tamara Broderick. The kernel interaction trick: Fast Bayesian discovery of pairwise interactions in high dimensions. In *International Conference on Machine Learning*, 2019.

H. Aschard. A perspective on interaction effects in genetic association studies. *Genetic Epidemiology*, 2016.

Francis Bach. Exploring large feature spaces with hierarchical multiple kernel learning. In *Neural Information Processing Systems*, 2008.

Francis R. Bach, Gert R. G. Lanckriet, and Michael I. Jordan. Multiple kernel learning, conic duality, and the SMO algorithm. 2004.

J. Bien, J. Taylor, and R. Tibshirani. A Lasso for hierarchical interactions. *The Annals of Statistics*, 41(3), 2013.

N. Butte, V. Voruganti, S. Cole, K. Haack, A. Comuzzie, D. Muzny, D. Wheeler, K. Chang, A. Hawes, and R. Gibbs. Resequencing of IRS2 reveals rare variants for obesity but not fasting glucose homeostasis in Hispanic children. *Physiological Genomics*, 43(18), 2011.

E. Candes and T. Tao. The Dantzig selector: Statistical estimation when p is much larger than n. *The Annals of Statistics*, 2007.

C. Carvalho, N. Polson, and J. Scott. Handling sparsity via the horseshoe. In *International Conference on Artificial Intelligence and Statistics*, 2009.

Gavin C. Cawley, Gareth J. Janacek, and Nicola L. C. Talbot. Generalised kernel machines. In *International Joint Conference on Neural Networks*, 2007.

S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 1998.

H. Chipman. Bayesian variable selection with related predictors. *The Canadian Journal of Statistics*, 24(1), 1996.

Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1), 2010.

S. McKay Curtis, Sayantan Banerjee, and Subhashis Ghosal. Fast Bayesian model assessment for nonparametric additive regression, 2014.

N. Durrande, D. Ginsbourger, O. Roustant, and L. Carraro. ANOVA kernels and RKHS of zero mean functions for model-based sensitivity analysis. *Journal of Multivariate Analysis*, 115(C), 2013.

David Duvenaud, James Lloyd, Roger Grosse, Joshua Tenenbaum, and Ghahramani Zoubin. Structure discovery in nonparametric regression through compositional kernel search. In *International Conference on Machine Learning*, 2013.

Federico Ferrari and David B. Dunson. Bayesian factor analysis for inference on interactions. *Journal of the American Statistical Association*, 116(525), 2020a.

Federico Ferrari and David B. Dunson. Identifying main effects and interactions among exposures using Gaussian processes. *The Annals of Applied Statistics*, 14(4), 2020b.

Jerome H. Friedman. Multivariate adaptive regression splines. *The Annals of Statistics*, 19 (1), 1991.

Jacob Gardner, Geoff Pleiss, Kilian Q Weinberger, David Bindel, and Andrew G Wilson. GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *Advances in Neural Information Processing Systems*, 2018.

Edward George and Robert McCulloch. Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 1993.

S. Geronikolou, A. Pavlopoulou, G. Lambrou, J. Koutelekos, D. Cokkinos, K. Albanopoulos, and G. Chrousos. E3 ligase FBXW2 is a new therapeutic target in obesity and atherosclerosis. *Advanced Science*, 7(20), 2020.

Casey Greene, Nicholas Sinnott-Armstrong, Daniel S. Himmelstein, Paul Park, Jason Moore, and Brent Harris. Multifactor dimensionality reduction for graphics processing units enables genome-wide testing of epistasis in sporadic ALS. *Bioinformatics*, 26(5), 2010.

J. Griffin and P. Brown. Hierarchical shrinkage priors for regression models. *Bayesian Analysis*, 12, 2017.

Chong Gu and Grace Wahba. Smoothing spline ANOVA with component-wise Bayesian “confidence intervals”. *Journal of Computational and Graphical Statistics*, 2(1), 1993.

S. Gunn and J. Kandola. Structural modelling with sparse kernels. *Machine Learning*, 2004.

Paul Gustafson. Bayesian regression modeling with interactions and smooth effects. *Journal of the American Statistical Association*, 95(451), 2000.

Ning Hao and Hao Helen Zhang. Interaction screening for ultra-high dimensional data. *Journal of the American Statistical Association*, 109(507), 2014.

Asad Haris, Daniela Witten, and Noah Simon. Convex modeling of interactions with strong heredity. *Journal of Computational and Graphical Statistics*, 25(4), 2016.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. 2001.

Giles Hooker. Generalized functional anova diagnostics for high-dimensional functions of dependent variables. *Journal of Computational and Graphical Statistics*, 16(3), 2007.

Daniel Hsu, Sham M. Kakade, and Tong Zhang. Random design analysis of ridge regression. *Foundations of Computational Mathematics*, 14(3), 2014.

Jianhua Z. Huang. Projection estimation in multiple regression with application to functional ANOVA models. *The Annals of Statistics*, 26(1), 1998.

Paule V. Joseph, Yupeng Wang, Nicolaas H. Fourie, and Wendy A. Henderson. A computational framework for predicting obesity risk based on optimizing and integrating genetic risk score and gene expression profiles. *PLoS One*, 13(5), 2018.

Julia Krzykalla, Axel Benner, and Annette Kopp-Schneider. Exploratory identification of predictive biomarkers in randomized trials with normal endpoints. *Statistics in Medicine*, 39(7), 2020.

Gert R. G. Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, and Michael I. Jordan. Learning the kernel matrix with semidefinite programming. *Journnal of Machine Learning Research*, 2004a.

Gert R. G. Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, and Michael I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5, 2004b.

M. Lim and T. Hastie. Learning interactions via hierarchical group-lasso regularization. *Journal of Computational and Graphical Statistics*, 24(3), 2015.

X. Lin, A. Taguchi, S. Park, J. Kushner, F. Li, Y. Li, and M. White. Dysregulation of insulin receptor substrate 2 in beta cells and brain causes obesity and diabetes. *Physiological Genomics*, 114(7), 2004.

Yi Lin and Hao Helen Zhang. Component selection and smoothing in multivariate nonparametric regression. *The Annals of Statistics*, 34(5), 2006.

Antonio R. Linero. Bayesian regression trees for high-dimensional prediction and variable selection. *Journal of the American Statistical Association*, 113(522), 2018.

Han Liu, Larry Wasserman, John D. Lafferty, and Pradeep K. Ravikumar. Sparse additive models. 2008.

Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Conference on Knowledge Discovery and Data Mining*, 2013.

B. Maher. Personal genomes: The case of the missing heritability. *Nature*, 2008.

Charles Margossian, Aki Vehtari, Daniel Simpson, and Raj Agrawal. Hamiltonian Monte Carlo using an adjoint-differentiated Laplace approximation: Bayesian inference for latent Gaussian models and beyond. In *Advances in Neural Information Processing Systems*, 2020.

Charles A. Micchelli, Yuesheng Xu, and Haizhang Zhang. Universal kernels. *Journnal of Machine Learning Research*, 7, 2006.

K. Nakagawa, S. Suzumura, M. Karasuyama, K. Tsuda, and I. Takeuchi. Safe pattern pruning: An efficient approach for predictive pattern mining. In *Conference on Knowledge Discovery and Data Mining*, 2016.

J. Piironen and A. Vehtari. Sparsity information and regularization in the horseshoe and other shrinkage priors. *Electronic Journal of Statistics*, 11, 2017.

Joaquin Quiñonero Candela and Carl Edward Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journnal of Machine Learning Research*, 6, 2005.

Peter Radchenko and Gareth M. James. Variable selection using adaptive nonlinear interaction structures in high dimensions. *Journal of the American Statistical Association*, 105(492), 2010.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

Steffen Rendle. Factorization machines. In *IEEE International Conference on Data Mining*, 2010.

Fabian Scheipl, Ludwig Fahrmeir, and Thomas Kneib. Spike-and-slab priors for function selection in structured additive regression models. *Journal of the American Statistical Association*, 107(500), 2012.

R. Shah. Modelling interactions in high-dimensional data with backtracking. *Journnal of Machine Learning Research*, 17(207), 2016.

L. Slim, C. Chatelain, C. Azencott, and J. Vert. Novel methods for epistasis detection in genome-wide association studies. *bioRxiv:325993*, 2018.

Charles J. Stone. The use of polynomial splines and their tensor products in multivariate function estimation. *The Annals of Statistics*, 22(1), 1994.

Xiaogang Su, Karen Meneses, Patrick McNees, and Wesley O. Johnson. Interaction trees: exploring the differential effects of an intervention programme for breast cancer survivors. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 60(3), 2011.

Michalis K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, 2009.

Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.

Z. Vitezica, L. Varona, and A. Legarra. On the additive and dominant variance and covariance of individuals within the genomic selection scope. *Genetics*, 195(4), 2013.

C. Wang, W. Xu, Y. Chao, M. Liang, F. Zhang, and K. Huang. Kisspeptin and the genetic obesity interactome. *Advances in Experimental Medicine and Biology*, 1339, 2021.

Ran Wei, Brian Reich, Jane Hoppin, and Subhashis Ghosal. Sparse Bayesian additive nonparametric regression with application to health effects of pesticides mixtures. *Statistica Sinica*, 2019.

Kristin L. Young, Misa Graff, Kari E. North, Andrea S. Richardson, Karen L. Mohlke, Leslie A. Lange, Ethan M. Lange, Kathleen M. Harris, and Penny Gordon-Larsen. Interaction of smoking and obesity susceptibility loci on adolescent BMI: The national longitudinal study of adolescent to adult health. *BMC Genetics*, 16(1), 2015.