Towards a Unified Framework for Uncertainty-aware Nonlinear Variable Selection with Theoretical Guarantees

Anonymous Author(s)
Affiliation
Address
email

Abstract

We develop a simple and unified framework for nonlinear variable selection that incorporates uncertainty in the prediction function and is compatible with a wide range of machine learning models (e.g., tree ensembles, kernel methods, neural networks, etc). In particular, for a learned nonlinear model $f(\mathbf{x})$, we consider quantifying the importance of an input variable $\mathbf{x}^j$ using the integrated partial derivative $\Psi_j = \|\frac{\partial}{\partial \mathbf{x}^j} f(\mathbf{x})\|^2_{P_\mathcal{X}}$. We then (1) provide a principled approach for quantifying variable selection uncertainty by deriving its posterior distribution, and (2) show that the approach is generalizable even to non-differentiable models such as tree ensembles. Rigorous Bayesian nonparametric theorems are derived to guarantee the posterior consistency and asymptotic uncertainty of the proposed approach. Extensive simulations and experiments on healthcare benchmark datasets confirm that the proposed algorithm outperforms existing classic and recent variable selection methods.

1 Introduction

Variable selection is often of fundamental interest in many data science applications, providing benefits in prediction error, interpretability, and computation by excluding unnecessary variables. As datasets grow in complexity and size, it is crucial that variable selection methods can account for complex dependencies among variables while remaining computationally feasible. Furthermore, as the number of approaches to model such datasets have increased, it is crucial that the importance of each variable can be compared across model classes and extended to new ones as they are developed.

While there are established approaches for variable selection in linear models (e.g., LASSO regression [29]), there is little consensus in methodology or theory for variable selection in nonlinear models. Generalized additive models [30] use similar variable selection methods as their linear counterparts [66], but the additivity assumption for nonlinear functions of the variables is too restrictive in many applications. Random Forests (RF) [7] measure variable importance using an impurity measure, which is based on the average reduction of the loss function were a given variable be removed from the model. [22] extended this method to boosting, where the variable importance is generalized by considering the average over all of the decision trees. Deep neural networks (DNNs) are widely-used for many artificial intelligence applications, and a substantial effort has been invested into developing DNNs with variable selection capabilities. Typically, this class of models involves manipulating the input layer, for example by imposing an L_1 penalty [11, 19], using backward selection [11], or knockoffs [44]. Unfortunately, each model class based on DNNs requires a tailored variable selection procedure, which limits comparability across different model formulations.

Bayesian variable selection methods provide principled uncertainty quantification in variable importance estimates as well as a complete characterization of their dependency structure. These methods

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

allow the variable selection procedure to tailor its decision rule with respect to the correlation structure [42]. Yet, as in frequentist models, each method has a different definition of a variable's importance. For example, in Bayesian additive regression trees (BART), a variable's importance can be measured by the proportion of trees that use it [13], while in Gaussian process (GP) models, a variable's importance can be measured by the frequency of the fluctuations of the resulting function (e.g., the length-scale parameter as controlled by the automatic relevance determination) in the direction of the variable [48, 68]. Furthermore, the traditional Bayesian modeling procedures tend to be computationally burdensome, making them less feasible for large-scale applications [1].

Our work starts with the observation that many machine learning models can be written as kernel methods by constructing a corresponding feature map. For example, random forests can be written as kernel methods by partitions [16], and deep neural networks can be written as kernel methods by using the last hidden layer as the feature map [62, 31, 9]. Each of these feature maps can be constructed before the Bayesian learning of the Gaussian process (e.g., by pre-training on the same or a separate dataset), providing additional modeling expressiveness and representational capacity. The ability of a GP model to incorporate these adaptive feature maps becomes especially important in high-dimension applications, where effective dimension reduction is necessary to circumvent the curse of dimensionality and to ensure good finite-sample performance [2].

Contributions. We propose a unified variable selection framework that is compatible with a wide range of machine learning models that can be defined by, or be closely approximated by, a differentiable feature map. Notable members include neural networks and random forests (Appendix B). Our approach defines variable importance as the norm of the function's partial derivative, as was previously studied in the context of frequentist nonparametric regression [58]. We extend it to apply to a much wider class of models than previously considered (Section 2), propose a principled Bayesian approach to quantify the variable selection uncertainty in finite data (Section 3.1), and derive rigorous Bayesian nonparametric theorems to guarantee the method's consistency and asymptotic optimality (Section 3.2). To incorporate powerful non-differentiable models into our framework, we also show how to apply this approach to partition-based methods (e.g., decision trees) by leveraging its (soft) feature representation (Appendix F). This leads to the first derivative-based Bayesian variable selection approach for tree-type models that is both theoretically grounded and empirically powerful, strongly outperforming other variable selection approaches tailor-designed for random forest (e.g., impurity or random-forest knockoff [8, 10]). We conduct extensive empirical validation of our approach and compare its performance to that of many existing methods across a wide range of data generation scenarios. Results show a clear advantage of the proposed approach, especially in complex scenarios or when the input is a mixture of discrete and continuous features (Section 4).

2 Preliminaries

Problem Setup. We consider the classic nonparametric regression setting with d-dimensional features $\mathbf{x} = (\mathbf{x}^1, \ldots, \mathbf{x}^d) \in \mathcal{X} = \mathbb{R}^d$ and a continuous response $y \in \mathbb{R}$. The features $\mathbf{x}$ are allowed to have a flexible nonlinear effect on y, such that:

$$y = f_0(\mathbf{x}) + e_i, \quad \text{where } e_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2), \tag{1}$$

with homoscedastic noise level σ^2. The data dimension d is allowed to be large but assumed to be constant and does not grow with the sample size n. Here the data-generating function f_0 is a flexible nonlinear function that resides in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}_0 : \mathcal{X}_0 \to \mathbb{R}$ induced by a certain positive definite kernel function k_0, and the input space of the true function $\mathcal{X}_0$ spans only a small subset of the input features $(\mathbf{x}^1, \ldots, \mathbf{x}^d)$, i.e., $\mathcal{X}_0 \subset \mathcal{X}$.

To this end, the goal of *global* variable selection is to produce a variable importance score ψ_j for each of the input features $(\mathbf{x}^1, \ldots, \mathbf{x}^d)$ such that it can be used as a classification signal for whether $\mathbf{x}^j \in \mathcal{X}_0$. As a result, the variable selection decision can be made by threhsolding $\psi_j > s$ with a predefined threshold s. The quality of a variable selection signal ψ_j can be evaluated comprehensively using a standard metric such as the *area under the receiver operating characteristic* (AUROC), which measures the Type-I and Type-II errors of variable selection decision $I(\psi_j > s)$ over a range of thresholds s.

2.1 Quantifying Model Uncertainty via Featurized GP

In the nonlinear regression scenario given by Equation (1), a classic approach to uncertainty-aware model learning is the Gaussian process (GP). Specifically, assuming that f_0 can be described by a

flexible RKHS $\mathcal{H}_k$ governed by the kernel function k, the GP model imposes a Gaussian process prior $f \sim \mathcal{GP}(0, k)$, such that the function evaluated at any collection of examples follows a multivariate normal ($\mathcal{MVN}$) distribution

$$\mathbf{f} \equiv (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n))^\top \sim \mathcal{MVN}(\mathbf{m}_{n\times 1}, \mathbf{K}_{n\times n}),$$

with mean $\mathbf{m}_i = m(\mathbf{x}_i)$ and covariance matrix $\mathbf{K}_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. The choice of the prior mean $\mathbf{m}$ and kernel k enable prior specification directly in function space. For example, the Matérn kernel places a prior over $\lceil \nu \rceil - 1$ times differentiable functions, with length-scale l^2 and amplitude variance σ^2. As $\nu \to \infty$, this reduces to the common radial basis function (RBF) kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp(\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 / l^2)$.

Under the above construction, the posterior predictive distribution of f evaluated at new observations $\mathbf{x}_1^*, \ldots, \mathbf{x}_{n^*}^*$ is also a multivariate normal,

$$\mathbf{f}^* | \{\mathbf{x}_i\, y_i\}_{i=1}^n \sim \mathcal{MVN}(\mathbb{E}[\mathbf{f}^*], \mathrm{Cov}[\mathbf{f}^*]), \quad \text{where} \tag{2}$$

$$\mathbb{E}[\mathbf{f}^*] = \mathbf{m}^* + \mathbf{K}^*(\mathbf{K} + \sigma^2 \mathbf{I}_n)^{-1}(\mathbf{y} - \mathbf{m}); \quad \mathrm{Cov}[\mathbf{f}^*] = \mathbf{K}^{**} - \mathbf{K}^*(\mathbf{K} + \sigma^2 \mathbf{I}_n)^{-1}\mathbf{K}^{*\top},$$

with $\mathbf{m}_i^* = m(\mathbf{x}_i^*)$, $\mathbf{K}_{ij}^* = k(\mathbf{x}_i^*, \mathbf{x}_j)$, and $\mathbf{K}_{ij}^{**} = k(\mathbf{x}_i^*, \mathbf{x}_j^*)$. Equation (2) is known as the kernel-based representation (or dual representation) of f [55]. Although mathematically elegant, the posterior (2) is expensive to compute due to the need to invert the $n \times n$ matrix $(\mathbf{K} + \sigma^2 \mathbf{I})^{-1}$.

Feature-based Representation of GP. Alternatively, Mercer's theorem [15] states that as long as the kernel function $k(\cdot, \cdot)$ can be written as the inner product of a set of basis functions $\phi(\mathbf{x}) = \{\phi_k(\mathbf{x})\}_{k=1}^D$, such that $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \phi(\mathbf{x}')$, then elements of the RKHS $f \in \mathcal{H}_k$ can be written in terms of a linear expansion of basis functions [55]:

$$f(\mathbf{x}) = \sum_{k=1}^D \boldsymbol{\beta}_k \phi_k(\mathbf{x}) = \phi(\mathbf{x})^\top \boldsymbol{\beta}, \text{ where } \boldsymbol{\beta} \sim \mathcal{MVN}(\boldsymbol{\mu}, \mathbf{I}_D). \tag{3}$$

This is known as the feature-based representation (or primal representation) of a Gaussian process. Notice that (3) is not an approximation method but an *exact* reparametrization of the Gaussian process model whose kernel function is induced by feature representation $\phi(\mathbf{x})$.

Scalable Posterior Computation via Minibatch Updates. The above feature-based representation is powerful in that it reduces the GP posterior inference into a Bayesian linear regression problem for $\boldsymbol{\beta}$. This brings two concrete benefits. First, the posterior of $\boldsymbol{\beta}$ in Equation (3) adopts a closed-form:

$$\boldsymbol{\beta} \sim \mathcal{MVN}(\mathbb{E}[\boldsymbol{\beta}], \mathrm{Cov}[\boldsymbol{\beta}]), \quad \text{where} \tag{4}$$

$$\mathbb{E}[\boldsymbol{\beta}] = \boldsymbol{\mu} + \Sigma_{\boldsymbol{\beta}} \Phi^\top (\mathbf{y} - \Phi \boldsymbol{\mu})/\sigma^2; \quad \mathrm{Cov}[\boldsymbol{\beta}] = \Sigma_{\boldsymbol{\beta}} = (\Phi^\top \Phi/\sigma^2 + \mathbf{I})^{-1},$$

where $\Phi = (\phi(\mathbf{x}_1)^\top, \ldots, \phi(\mathbf{x}_n)^\top)^\top \in \mathbb{R}^{n \times D}$ is the feature matrix evaluated on the training data [55]. For large-scale applications, Equation (4) enables us to compute the exact posterior of $\boldsymbol{\beta}$ in a mini-batch fashion. For example, the posterior matrix $\mathrm{Cov}[\boldsymbol{\beta}] = \Sigma_{\boldsymbol{\beta}}$ can be updated using the Woodbury identity:

$$\Sigma_{\boldsymbol{\beta},t+1} = \Sigma_{\boldsymbol{\beta},t} - \Sigma_{\boldsymbol{\beta},t} \Phi_m^\top (\sigma^2 \mathbf{I} + \Phi_m \Sigma_{\boldsymbol{\beta},t} \Phi_m^\top)^{-1} \Phi_m \Sigma_{\boldsymbol{\beta},t}. \tag{5}$$

where Φ_m is the D-dimension batch-specific feature matrix evaluated on the mini-batch. Similarly, the posterior mean $\mathbb{E}[\boldsymbol{\beta}]$ can be computed by accumulating the $D \times 1$ vector $\Phi^\top(\mathbf{y} - \Phi\boldsymbol{\mu}) = \sum_m \Phi_m^\top (\mathbf{y}_m - \Phi_m \boldsymbol{\mu})$, and compute the posterior mean according to Equation (4) at the end.

The posterior distribution of $\boldsymbol{\beta}$ induces a Gaussian process posterior for the prediction function $\mathbf{f}^* = \Phi^* \boldsymbol{\beta}$, where Φ^* is the feature map evaluated on the test data, with mean $\mathbb{E}[\mathbf{f}^*] = \Phi^* \boldsymbol{\mu} + \Phi^* \Sigma_{\boldsymbol{\beta}} \Phi^\top (\mathbf{y} - \Phi \boldsymbol{\mu})/\sigma^2$ and covariance $\mathrm{Cov}[\mathbf{f}^*] = \Phi^* \Sigma_{\boldsymbol{\beta}} \Phi^{*\top}$. This distribution is equivalent to the kernel-based representation (2) but reduces the computational complexity from cubic time $O(n^3)$ to a linear time $O(n)$ and is minibatch compatible (i.e., Equation (5)).

Incorporating Modern ML Model Classes. The second key advantage of the feature-based representation (3) is its generality: a wide range of machine learning models can be written in term of the feature-based form $f(\mathbf{x}) = \phi(\mathbf{x})^\top \boldsymbol{\beta}$ [52, 16, 39], making the Gaussian process a unified framework for quantifying model uncertainty for a wide array of modern ML models. Appendix B summarizes important examples including GAMs, decision trees, random-feature models, deep neural networks and their ensembles. Furthermore, when a deterministically-trained $\hat{\boldsymbol{\beta}}$ is available (e.g., via a sophisticated adaptive shrinkage procedure that is not available in Bayesian context), we can incorporate this as prior knowledge into GP modeling by setting $\boldsymbol{\mu} = \hat{\boldsymbol{\beta}}$ (Equation (3)).

2.2 Bayesian Nonparametric Guarantees for Probabilistic Learning

The quality of a Bayesian learning procedure is commonly measured by the learning rate of its posterior distribution $\Pi_n = \Pi(\cdot \mid \{\mathbf{x}_i, y_i\}_{i=1}^n)$. Intuitively, the rate of this convergence is measured by the size of the smallest shrinking balls around f_0 that contain most of the posterior probability. Specifically, we consider the size of the set $A_n = \{g \mid \|g - f_0\|_n^2 \leq M\epsilon_n\}$ such that $\Pi_n(A_n) \to 1$ [25, 51]. The concentration rate ϵ_n here indicates how fast the small ball A_n concentrates towards f_0 as the sample size increases. Below we state the formal definition of posterior convergence [25].

Definition 1 (Posterior Convergence)**.** For $f_0 : \mathcal{X} \to \mathbb{R}$ where $\mathcal{X} = \mathbb{R}^d$, denote $\mathcal{H}_0$ the true RKHS induced by a kernel function k_0, and denote $\mathcal{H}_\phi$ induced by the feature function $\phi : \mathcal{X} \to \mathbb{R}^D$. Let $f_0 \in \mathcal{H}$ be the true function, and let $\mathbb{E}_0$ denote the expectation with respect to the true data-generation distribution. Assuming $\mathcal{H}_\phi$ is dense in $\mathcal{H}$, then, the posterior distribution $\Pi_n(f)$ concentrates around f_0 at the rate ϵ_n if there exists an $\epsilon_n \to 0$ such that, for any $M_n \to \infty$,

$$\mathbb{E}_0 \Pi_n(f : \|f - f_0\|_n^2 \geq M_n \epsilon_n) \to 0. \tag{6}$$

Notice that we allow the model space $\mathcal{H}_\phi$ and the true function space $\mathcal{H}$ to be different, but the prior must be *dense* in the later for the convergence to happen. Fortunately, this condition is shown to hold for a wide variety of ML models, including random features, random forests, and neural networks [3, 33, 53, 61, 57]. The notion of posterior convergence can also be used to discuss the learning quality of other probabilistic estimates (e.g., variable importance ψ_j). In that case, we can simply replace (f, f_0) in (6) by their variable importance counterparts. This is the focus of Section 3.2.

3 Methods

3.1 Quantifying Variable Importance under Uncertainty

In this work, we consider quantifying the *global* importance of a variable based on the norm of the corresponding partial derivative. This is motivated by the observation that, if a function f is differentiable, the relative importance of a variable $\mathbf{x}^j$ at a point $\mathbf{x}$ can be captured by the magnitude of the partial derivative function, $|\frac{\partial}{\partial \mathbf{x}^j} f(\mathbf{x})|$ [58]. This proposed quantity requires the consideration of two issues. First, instead of quantifying the relevance of a variable on a single input point, we need to define a proper global notion of variable importance. Therefore, it is natural to integrate the magnitude of the partial derivative over the input space $\mathbf{x} \in \mathcal{X} : \Psi_j(f) = \|\frac{\partial}{\partial \mathbf{x}^j} f\|_{P_\mathcal{X}}^2 = \int_{\mathbf{x} \in \mathcal{X}} |\frac{\partial}{\partial \mathbf{x}^j} f(\mathbf{x})|^2 \, dP_\mathcal{X}(\mathbf{x})$. Second, since $P_\mathcal{X}(\mathbf{x})$ is not known from the training observations, $\Psi_j(f)$ can be approximated by its empirical counterpart,

$$\psi_j(f) = \|\frac{\partial}{\partial \mathbf{x}^j} f\|_n^2 = \frac{1}{n} \sum_{i=1}^n |\frac{\partial}{\partial \mathbf{x}^j} f(\mathbf{x}_i)|^2. \tag{7}$$

Notice that $\psi_j(f)$ is an estimator that is derived from the prediction function f estimated using finite data. Consequently, to make a proper decision regarding the importance of an input variable $\mathbf{x}^j$, it is important to take into account uncertainty in f. To this end, by leveraging the featurized GP representation introduced in Section 3.1, we show that this can be done easily for a wide range of ML models $f(\mathbf{x}) = \phi(\mathbf{x})^\top \boldsymbol{\beta}$ by studying the posterior distribution of ψ_j.

Posterior Distribution of Variable Importance. After we obtain the posterior distribution of $\boldsymbol{\beta}$ (4), the posterior distribution of variable importance can be derived according to Equation (7):

$$\psi_j(f) = \frac{1}{n} |\frac{\partial}{\partial \mathbf{x}^j} f(\mathbf{X})|^\top |\frac{\partial}{\partial \mathbf{x}^j} f(\mathbf{X})| = \frac{1}{n} \boldsymbol{\beta}^\top \frac{\partial \Phi}{\partial \mathbf{x}^j} \frac{\partial \Phi}{\partial \mathbf{x}^j}^\top \boldsymbol{\beta}, \tag{8}$$

where $\frac{\partial \Phi}{\partial \mathbf{x}^j} \in \mathbb{R}^{D \times n}$ is the derivative of the feature map with respect to $\mathbf{x}^j$, across n training samples. The posterior distribution of $\psi_j(f)$ adopts a closed form as a generalized chi-squared distribution (see Appendix A.2 for derivation). In practice, we can sample ψ_j conveniently from its posterior distribution by computing $\frac{\partial}{\partial \mathbf{x}^j} f(\mathbf{X}) = \left(\frac{\partial \Phi}{\partial \mathbf{x}^j}\right)^\top \boldsymbol{\beta}^{(s)}$, where $\boldsymbol{\beta}^{(s)}$ are Monte Carlo samples from the closed-form posterior (4).

There are two ways in which uncertainty aids the model selection process. First, the posterior survival function $P(\psi_j(f) > s)$ of the variable importance utilizes the full posterior distribution of $\psi_j(f)$ to identify the probability that the variable $\mathbf{x}^j$ exceeds a given threshold s. By increasing $s \in (0, \infty)$, $P(\psi_j > s)$ provides a intuitive sense of how model's belief about the importance of variable $\mathbf{x}^j$

changes as the criteria s becomes more stringent, similar to the regularization path in LASSO methods [20] but with the incorporation of posterior uncertainty about the variable importance. See Appendix I for an application to a Bangladesh birth cohort study. Second, by integrating the survival function over the threshold, i.e., $\int_{s>0} P(\psi_j(f) > s)\, ds$, we obtain the posterior mean of $\psi_j(f)$, and this too incorporates uncertainty in f. To see this, notice by using the "trace trick" we can write

$$\mathbb{E}[\psi_j(f)] = \mathbb{E}\left[\text{tr}\left(\boldsymbol{\beta}^\top \frac{\partial \Phi}{\partial \mathbf{x}^j}\frac{\partial \Phi}{\partial \mathbf{x}^j}^\top \boldsymbol{\beta}\right)\right] = \mathbb{E}[\boldsymbol{\beta}]^T \frac{\partial \Phi}{\partial \mathbf{x}^j}\frac{\partial \Phi}{\partial \mathbf{x}^j}^\top \mathbb{E}[\boldsymbol{\beta}] + \text{tr}\left(\frac{\partial \Phi}{\partial \mathbf{x}^j}\frac{\partial \Phi}{\partial \mathbf{x}^j}^\top \text{Cov}[\boldsymbol{\beta}]\right), \quad (9)$$

where all expectations are taken with respect to the posterior. Therefore, the posterior mean of $\psi_j(f)$ depends on the covariance structure of $\boldsymbol{\beta}$, and how it interacts with the eigenspace of the partial derivative functions (encoded by $\frac{\partial \Phi}{\partial \mathbf{x}^j}\frac{\partial \Phi}{\partial \mathbf{x}^j}^\top$). In Section 4 we provide an extensive investigation of AUROC scores using the posterior mean of $\psi_j(f)$ for variable selection.

In Appendix A.3, we summarize the algorithms for computing the posterior distributions of the featurized Gaussian process (Equation (4)) and for the posterior distributions of variable importance (Equation (8)), and discuss their space and time complexity.

3.2 Theoretical Guarantees

From a theoretical perspective, the variable importance measure ψ_j introduced in (7) can be understood as a quadratic functional of the Gaussian process model f [18]. To this end, rigorous Bayesian nonparametric guarantees can be obtained for ψ_j's ability in learning the true variable importance in finite samples (i.e., posterior convergence, Theorem 1) and its statistical optimality from a frequentist perspective, in providing a low-variance estimator that attains the Cramér-Rao bound (i.e., Bernstein von-Mises phenomenon, Theorem 2).

Posterior Convergence. We first show that, for a ML model f that can learn the true function f_0 with rate ϵ_n (in the sense of Definition 1), the entire posterior distribution of the variable importance measure $\psi_j(f)$ converges consistently to a point mass at the true $\Psi_j(f_0)$ at a speed that is equal or faster than ϵ_n.

Theorem 1 (Posterior Convergence of Variable Importance ψ_j)**.** *Suppose $y_i = f_0(\mathbf{x}_i) + e_i,\ e_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$, and denote as $\mathbb{E}_0$ the expectation with respect to the true data-generation distribution centered around f_0. For the RKHS $\mathcal{H}_\phi$ induced by the feature function $\phi : \mathcal{X} \to \mathbb{R}^D$ and $f \in \mathcal{H}_\phi$, if:*

(1) The posterior distribution $\Pi_n(f)$ converges toward f_0 at a rate of ϵ_n;

(2) The differentiation operator $D_j : f \to \frac{\partial}{\partial \mathbf{x}^j} f$ is bounded: $\|D_j\|_{op}^2 = \inf\{C \geq 0 : \|D_j f\|_2^2 \leq C\|f\|_2^2$, for all $f \in \mathcal{H}_\phi\}$;

Then the posterior distribution for $\psi_j(f) = \|\frac{\partial}{\partial \mathbf{x}^j} f\|_n^2$ contracts toward $\Psi_j(f_0) = \|\frac{\partial}{\partial \mathbf{x}^j} f_0\|_{P_\mathcal{X}}^2$ at a rate not slower than ϵ_n. That is, for any $M_n \to \infty$,

$$\mathbb{E}_0 \Pi_n \left[\sup_{j \in \{1,\ldots,d\}} |\psi_j(f) - \Psi_j(f_0)| \geq M_n \epsilon_n\right] \to 0.$$

Proof is in Appendix C. Theorem 1 is a generalization of the classic quadratic functional convergence result to a much wider range of ML models in the context of Bayesian variable selection [18, 42, 67]. It confirms the important fact that, for a ML model f that can accurately learn the true function f_0 under finite data, despite the potential lack of identifiablity in the model parameters (e.g., weights in a neural network), we can consistently recover the true variable importance at a fast rate by using the proposed variable importance estimate $\psi_j(f)$.

From a practical point of view, Theorem 1 reveals that the finite-sample performance of variable importance $\psi_j(f)$ depends on two factors: (1) the finite-sample generalization performance of the prediction function f, and (2) the mathematical property of f in terms of its Lipschitz condition. Therefore, to ensure effective variable selection in practice, the practitioner should take care to select a model class f that has a theoretical guarantee in capturing the target function f_0, empirically delivers strong generalization performance under finite data, and is well-conditioned in terms of the behavior of its partial derivatives. To this end, we note that, under the featurized Gaussian process

$f = \phi(\mathbf{x})^\top \boldsymbol{\beta}$ discussed in this work, users are free to choose a performant model class (e.g., random forest, random-feature or DNN) whose feature representation spans an RKHS $\mathcal{H}_\phi$ that is *dense* in the infinite-dimensional function space (therefore f enjoys a convergence guanrantee) [3, 33, 53, 61, 57], and is empirically more effective than the classic GP methods. (We discuss the Lipschitz condition of these models in Appendix E.1) Indeed, as we will verify in experiments (Section 4), there does not exist an "optimal" model class that performs universally well across all data settings (i.e., no free lunch theorem [69]). This highlights the importance of having a general-purpose framework for variable selection that can flexibly incorporate the most effective model for the task at hand.

Statistical Efficiency & **Uncertainty Quantification.** Next, we verify the uncertainty quantification ability of the variable importance measure $\psi_j(f)$ under featurized GP, by showing that it exhibits the *Bernstein-von Mises* (BvM) phenomenon. That is, its posterior measure $\Pi_n(\psi_j(f))$ converges towards a Gaussian distribution that is centered around the truth $\Psi_j(f_0)$, so that its $(1-\alpha)\%$ level credible intervals achieve the nominal coverage probability for the true variable importance. More importantly, the BvM theorem verifies that the posterior distribution of $\psi_j(f)$ is *statistically optimal*, in the sense that its asymptotic variance attains the Cramér-Rao bound (CRB) that cannot be improved upon [5].

Theorem 2 (Bernstein-von Mises Theorem for Variable Importance ψ_j)**.** *Suppose* $y_i = f_0(\mathbf{x}_i) + e_i,\ e_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2),\ i = 1, \ldots, n$. *Denote* $D_j : f \to \frac{\partial}{\partial \mathbf{x}^j} f$ *the differentiation operator and* $H_j = D_j^\top D_j$ *the inner product of* D_j*, such that:*

$$\psi_j(f) = \|D_j(f)\|_n^2 = \frac{1}{n}\langle D_j f, D_j f\rangle = \frac{1}{n} f^\top H_j f. \tag{10}$$

Assuming conditions (1)-(2) in Theorem 1 hold, and additionally:

(3) f_0 *is square-integrable over the support* $\mathcal{X}$ *and* $\|f_0\|_2 = 1$*;*

(4) $rank(H_j) = o_p(\sqrt{n})$*;*

Then

$$\sqrt{n}(\psi_j(f) - \psi_j(f_0)) \xrightarrow{d} \mathcal{N}(0, 4\sigma^2 \|H_j f_0\|_n^2).$$

Proof is in Appendix D. Theorem 2 provides a rigorous theoretical justification for $\psi_j(f)$'s ability to quantify its uncertainty about the variable importance. More importantly, it verifies that $\psi_j(f)$ has the good frequentist property that it quickly converging to a minimum-variance estimator at a fast speed, which is important in practice. Compared to the previous BvM results that tend to focus on a specific Bayesian ML model, Theorem 2 is considerably more general (i.e., applicable to a much wider range of models) and comes with a simpler set of conditions [56, 67, 42]. Specifically, (3) is a standard assumption in nonparametric analysis. It ensures the true function f_0 does not diverge towards infinity and makes learning impossible [12]. The unit norm assumption $\|f_0\|_2 = 1$ is only needed to simplify the exposition of the proof, and the theorem can be trivially extended to $\|f_0\|_2 = C$ for any $C > 0$. The most interesting condition is (4). Let's denote $\mathcal{H}_j$ the space of partial derivatives functions $\frac{\partial}{\partial \mathbf{x}^j} f, f \in \mathcal{H}$. Then intuitively, (4) says to attain the BvM phenomenon, the effective dimensionality of the derivative function space $\mathcal{H}_j$ (as measured by $rank(H_j) = rank(D_j)$) cannot be too large. Since effective dimensionality of the derivative space is bounded above by that of the original RKHS $f \in \mathcal{H}_\phi$, (4) essentially states that the effective dimensionality of the model space $\mathcal{H}_\phi$ cannot grow too fast with data ($o_p(\sqrt{n})$). Fortunately, this condition is satisfied by a wide range of ML models including trees and deep networks [56, 67]. See Appendix E.2 for further discussion.

4 Experiment Analysis

In this section, we investigate the finite-sample performance of the derivative norm metric ψ_j for variable selection (7) under a wide variety of ML methods. We illustrate the breath of our framework by applying it to tree ensembles (Appendix F), where a principled and gradient-based uncertainty-aware variable selection approach has been previously unavailable. We also apply it to linear models and (approximation) kernel machines, which are standard approaches to variable selection in data science practice [65, 6]. Over a wide range of complex and realistic data scenarios (e.g., discrete features, interactions, between-feature correlations) derived from socioeconomic and healthcare

datasets, we investigate the method's statistical performance in accurately recovering the ground-truth features (in terms of the Type I and Type II errors), and compare it to other well-established approaches in each of the model classes (Table 1). Our main observations are:

O1: **Importance of generality**. There does not exist a model class that performs universally well across all data scenarios (i.e., no free lunch theorem [69], Figure 1, 6-15). This highlights the importance of an unified variable selection framework that incorporates a wide range of models, so that practitioners have the freedom of choosing the most suitable model class for the task at hand.

O2: **Good prediction translates to effective variable selection**. Comparing between different model classes, the ranking of models' predictive accuracy is generally consistent with the ranking of their variable selection performance under ψ_j (i.e., better prediction translates to better variable selection, as suggested in Theorem 1).

O3: **Statistical efficiency of** ψ_j. Comparing within each model class, the derivative norm metric ψ_j generally outperforms other measures of variable importance. The advantage is especially pronounced in small samples and for correlated features. This empirically verifies that ψ_j has good finite-sample statistical efficiency even under complex data scenarios (as suggested in Theorem 2).

Model Class	(Ours)	Baselines
Tree Ensembles	RF-FDT	RF-Impurity, RF-Knockoff, BART
(Appr.) Kernel Methods	RFNN	BKMR, BAKR
Linear Models	GAM	BRR, BL

Table 1: Summary of methods considered in the experiments.

Models & **Methods**. We consider three main classes of models (Table 1): (I) **Random Forests (RF)**. Given a trained forest, we quantify variable importance using ψ_j by translating it to an ensemble of featurized decision trees (**FDT**s) (Appendix F), and compare it to three baselines: *impurity* (**RF-impurity**) [8], *RF-based kernel knockoff* (**RF-knockoff**) [10], and *Bayesian additive regression trees* (**BART**). (II) **(Approximate) Kernel Methods**. We apply ψ_j to a random-feature model that approximates a Gaussian process with a RBF kernel [52], and set the number of features to $\sqrt{n}\,log(n)$ to ensure proper approximation of the exact RBF-GP [59]. We term this approach *Random-feature Neural Networks* (**RFNN**), and compared it to both *Bayesian Kernel Machine Regression* (**BKMR**) [6] based on a GP with exact RBF kernel and spike-and-slab prior and *Bayesian Approximate Kernel Regression* (**BAKR**) based on random-feature model with a projection-based feature importance measure and an adaptive shrinkage prior [14]. (III) **Linear Models**. We apply ψ_j to a featurized GP representation of the *Generalized Additive Model* (**GAM**), with the prior center $\boldsymbol{\beta}_0$ set at the frequentist estimate of the original GAM model obtained from a sophiscated REML procedure [70]. We compare it to two baselines: *Bayesian Ridge Regression* (**BRR**) [32] and *Bayesian LASSO* (**BL**) [50]. Appendix G provides further detail. Previously, [42] studied the specialization of our framework to the deep neural networks (DNNs), we don't repeat that work here as DNN is not yet a standard data science model for tabular data.

To quantify variable importance while accounting for posterior uncertainty the variable importance $\psi_j(f)$, we examine its posterior survival function $\int_{s>0} P(\psi_j(f) > s)\, ds$ (i.e., the posterior likelihood of $\psi_j(f)$ greater than the threshold s) integrated over the full range of thresholds s. For other methods, we use their default metrics to quantify variable importance (e.g., variable inclusion probabilities in **BART**, **BKMR**. Appendix G).

Datasets and Tasks. We consider two synthetic benchmarks and three real-world socio-economic and healthcare datasets, encapsulating challenging phenomena such as between-feature correlations and interaction effects. For the synthetic benchmark, we generate data under the Gaussian noise model $y \sim \mathcal{N}(f_0, 1)$ for four types of outcome-generation functions f_0 (`linear`, `rbf`, `matern32` and `complex`, see Appendix G.2 for a full description) with number of causal variables $d^\star = 5$. Two types of feature distribution are considered: (1) **synthetic-continuous**: all features follow $\mathbf{x}^j \sim Unif(-2, 2)$; (2) **synthetic-mixture**: two of the causal features and two of the non-causal features are distributed as $Bern(0.5)$ and the rest are distributed as $Unif(-2, 2)$; We vary sample size $n \in \{100, 200, 500, 1000\}$ and data dimension $d \in \{25, 50, 100\}$, leading to 96 total scenarios.

For real-world data, we consider (1) **adult**: 1994 U.S. census data of 48842 adults with 8 categorical and 6 continuous features [38]; (2) **heart**: a coronary artery disease dataset of 303 patients from Cleveland clinic database with 7 categorical and 6 continuous features [17]; and (3) **mi**: disease

records of myocardial infarction (MI) of 1700 patients from Krasnoyarsk interdistrict clinical hospital during 1992-1995, with 113 categorical and 11 continuous features [27]. All datasets exhibit non-trival correlation structure among features (Appendix Figure 3-5). Since the ground-truth causal features on these datasets are not known, in order to rigorously evaluate variable selection performance, we follow the standard practice in causal ML to simulate the outcome based on causal features selected from data [72]. We use the four outcome-generating functions as described previously and evaluate over the same data size $\times$ dimension combinations, leading to 144 total scenarios[1]. We repeat the simulation 20 times for each scenario, and use AUROC to measure the variable selection performance (in terms of Type I and II errors) of each method.

In the Appendix I, we further evaluate the method on a well-studied environmental health dataset (Bangladesh birth cohort study [37]) with respect to the real outcome (infant development scores). We visualize the "Bayesian" regularization path as introduced earlier. The selected variables correspond well with the established toxicology pathways in the literature [26].

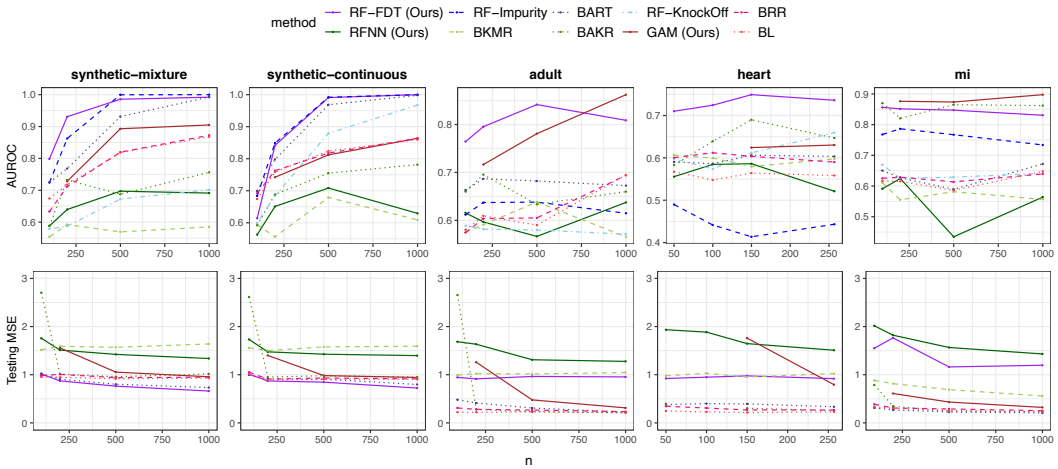

Figure 1: Method performance in variable selection (measured by AUROC, row 1) and prediction (measured by test MSE, row 2) under `matern32` data-generation function and with input dimension 100 (5 causal features). The ranking of the variable **FDT**s (solid purple) outperforms other methods in most of the data settings, and **GAM** outperforms in the setting of large data size and high percentage of categorical features (**adult** and **mi**). The rankings of performance are roughly consistent between prediction and variable selection.)

4.1 Results

Figure 1 shows the methods' performance in variable selection **(Row 1)** and prediction **(Row 2)**[2] in an exemplary setting, where the true function f_0 is `matern32` with an input dimension of 100. It represents the tabular data setting that we are the most interested in: nonlinear feature-response relationship with interaction effects and high input dimension. This is because f_0 is sampled from a RKHS induced by Mátern $\frac{3}{2}$ kernel, which contains a large space of continuous and at least once differentiable functions [55]. We delay complete visualizations for all 240 scenarios to Appendix H.

Recalling the three observations introduced earlier:

O1 ("No free lunch"): No method performs universally well. For example, **BAKR** performs robustly in correlated datasets (**heart** and **mi**), but poorly otherwise. Kernel approaches (**RFNN** and **BKMR**) performs competitively in low dimension, but their performance deteriorates quickly as dimension increases (Figure 6-7). **FDT**s generally is the strongest method in small samples and high dimensions, but can be outperformed by **GAM** in large samples and data with high percentage of categorical features (**adult** and **mi**). This highlights the importance of an unified framework that allows users to select the most appropriate model for variable selection depending on the data setting.

[1]In the setting where required data dimension is higher than that of the real data, we generate additional synthetic features from $Unif(-2, 2)$. We use $n \in \{50, 100, 150, 257\}$ for **heart** due to data size restriction.

[2]For the prediction plots, a method will not be visualized if they share the model fit with another method (**RF-impurity** and **RF-knockoff**, or if the method does not produce valid result due to small sample size (**GAM**).

O2 ("Good prediction implies effective variable selection"): Comparing among the gradient-based methods under each model class (i.e., **FDT**s, **GAM** and **RFNN**. Solid lines in Figure 1), we see that their rankings in prediction (row 2) is largely consistent with rankings in variable selection. It's worth noting that this pattern is occasionally violated (e.g., **GAM** in **adult**, n=500 and **heart**, n=250), but that does not contradict our conclusion (Theorem 1) since the convergence rate of the prediction function only forms an *upper bound* for the convergence rate of ψ_j.

O3 (Statistical efficiency of ψ_j): Comparing between variable selection methods from the same class (especially for tree models. i.e., **FDT**s v.s. **RF-impurity / RF-knockoff / BART**), we see that **FDT**s is competitive or strongly outperforms its baselines in variable selection, despite being based on exactly the same fitted model (**RF-impurity / RF-knockoff**), or not accounting for the uncertainty in the tree growing process (**BART**). This pattern is consistent in most data settings, and the advantage is especially pronounced in high dimensions, small data sizes, and correlated datasets (Appendix H, Figure 6-10). This provides strong empirical evidence for the fact that ψ_j is a statistically efficient estimator for variable selection with good finite-sample behavior (as suggested in Theorem 2), and can deliver strong variable selection performance for tabular data when combined with a performant ML model like random forest.

Appendix H contains further discussion.

5 Discussion and Future Directions

The modern data analysis pipeline typically involves fitting multiple models, comparing their performance, and iterating as necessary. When variable selection is involved, the practitioner may ask *are the variable importances across models measuring the same behavior?* And, *what if the most suitable model does not have a satisfactory variable selection procedure?* By framing model choice as kernel choice — which we emphasize includes kernels corresponding machine learning methods like neural networks and random forests in addition to the long list of traditional kernels — we propose a unified variable selection procedure that is compatible across models and we prove strong guarantees for this procedure.

Limitations. Our proposed framework provides principled uncertainty quantification by performing exact Bayesian inference on the weights $\boldsymbol{\beta}$ of a feature map $\phi(\mathbf{x})$. We do not consider uncertainty in the feature map itself. This means, for example, that if the feature map is given by the last hidden layer of a neural network trained by maximizing the posterior, then our model class corresponds to the *neural linear model*. This model is different from a fully Bayesian neural network, which performs posterior inference also on the kernel hyperparameters (i.e., the hidden weights) [49, 62, 64]. Likewise, the kernel induced by the featurized decision tree studied here does not consider uncertainty in the tree's partitioning process. Yet, this does not seem to be a significant limitation in our experiments (e.g., **FDT**s outperforms **BART**), although this point still merits further investigation in the future.

In our experiments, we focused on kernels based on tree ensembles, kernel methods and linear models. In the future, it would be worth expanding this framework to other model classes (e.g., MARS [21] or neural network) and estimating the importance of interaction effects and higher-order terms. We would also like to apply this method to large-scale scientific studies (e.g., epidemiology study based on extremely large EHR datasets) where an uncertainty-aware nonlinear variable selection method is typically impossible due to challenges with scalability.

Societal Impacts. The method proposed in this paper provides a theoretically-grounded approach for quantifying variable importance that is applicable to a wide range of ML models. We expect it to provide a set of powerful tools for practitioners to understand the importance of input variables in their ML models with limited data, which is especially important for scientific investigations such as epidemiology and computational biology. However, we recognize that this approach can potentially be utilized by bad actors to probe the input-variable uncertainty of an existing ML system, and use it to engineer more targeted white-box adversarial attacks. To this end, we recommend system developers to incorporate this approach into the formal verification procedure of a ML system, so as to monitor and understand the model uncertainty with respect to input variables, and devise proper improvement and prevention strategies (e.g., data augmentation or randomized smoothing targeted at specific variables) accordingly.

References

[1] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan. An introduction to mcmc for machine learning. *Machine learning*, 50(1):5–43, 2003.

[2] F. Bach. Breaking the Curse of Dimensionality with Convex Neural Networks. *arXiv:1412.8690 [cs, math, stat]*, Oct. 2016. URL http://arxiv.org/abs/1412.8690.

[3] G. Biau. Analysis of a random forests model. *The Journal of Machine Learning Research*, 13 (1):1063–1095, 2012.

[4] G. Biau, E. Scornet, and J. Welbl. Neural Random Forests. *Sankhya A*, 81(2):347–386, Dec. 2019. ISSN 0976-8378. doi: 10.1007/s13171-018-0133-y. URL https://doi.org/10.1007/s13171-018-0133-y.

[5] P. J. Bickel and B. J. Kleijn. The semiparametric bernstein–von mises theorem. *The Annals of Statistics*, 40(1):206–237, 2012.

[6] J. F. Bobb, L. Valeri, B. Claus Henn, D. C. Christiani, R. O. Wright, M. Mazumdar, J. J. Godleski, and B. A. Coull. Bayesian kernel machine regression for estimating the health effects of multi-pollutant mixtures. *Biostatistics*, 16(3):493–508, July 2015. ISSN 1465-4644. doi: 10.1093/biostatistics/kxu058.

[7] L. Breiman. Random Forests. *Machine Learning*, 45(1):5–32, Oct. 2001. ISSN 1573-0565. doi: 10.1023/A:1010933404324. URL https://doi.org/10.1023/A:1010933404324.

[8] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and Regression Trees*. Taylor & Francis, Jan. 1984. ISBN 978-0-412-04841-8.

[9] R. Calandra, J. Peters, C. E. Rasmussen, and M. P. Deisenroth. Manifold gaussian processes for regression. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 3338–3345. IEEE, 2016.

[10] E. Candes, Y. Fan, L. Janson, and J. Lv. Panning for Gold: Model-X Knockoffs for High-dimensional Controlled Variable Selection. *arXiv:1610.02351 [math, stat]*, Dec. 2017. URL http://arxiv.org/abs/1610.02351. arXiv: 1610.02351.

[11] G. Castellano and A. M. Fanelli. *Variable Selection Using Neural-Network Models*. 2000.

[12] I. Castillo and J. Rousseau. A bernstein–von mises theorem for smooth functionals in semiparametric models. *The Annals of Statistics*, 43(6):2353–2383, 2015.

[13] H. A. Chipman, E. I. George, and R. E. McCulloch. BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298, Mar. 2010. ISSN 1932-6157, 1941-7330. doi: 10.1214/09-AOAS285. Publisher: Institute of Mathematical Statistics.

[14] L. Crawford, K. C. Wood, X. Zhou, and S. Mukherjee. Bayesian Approximate Kernel Regression With Variable Selection. *Journal of the American Statistical Association*, 113(524):1710–1721, Oct. 2018. ISSN 0162-1459. doi: 10.1080/01621459.2017.1361830. URL https://doi.org/10.1080/01621459.2017.1361830.

[15] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, Cambridge, 2000. ISBN 978-0-521-78019-3. doi: 10.1017/CBO9780511801389.

[16] A. Davies and Z. Ghahramani. The random forest kernel and other kernels for big data from random partitions. *arXiv preprint arXiv:1402.4293*, 2014.

[17] R. Detrano, A. Janosi, W. Steinbrunn, M. Pfisterer, J. J. Schmid, S. Sandhu, K. H. Guppy, S. Lee, and V. Froelicher. International application of a new probability algorithm for the diagnosis of coronary artery disease. *Am J Cardiol*, 64(5):304–310, Aug. 1989. ISSN 0002-9149. doi: 10.1016/0002-9149(89)90524-9.

[18] S. Efromovich and M. Low. On optimal adaptive estimation of a quadratic functional. *The Annals of Statistics*, 24(3):1106–1125, 1996.

[19] J. Feng and N. Simon. Sparse-Input Neural Networks for High-dimensional Nonparametric Regression and Classification. *arXiv:1711.07592 [stat]*, June 2019. URL http://arxiv.org/abs/1711.07592.

[20] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.

[21] J. H. Friedman. Multivariate Adaptive Regression Splines. *The Annals of Statistics*, 19(1):1 – 67, 1991. doi: 10.1214/aos/1176347963. URL https://doi.org/10.1214/aos/1176347963.

[22] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232, Oct. 2001. ISSN 0090-5364, 2168-8966. doi: 10.1214/aos/1013203451. URL https://projecteuclid.org/euclid.aos/1013203451.

[23] N. Frosst and G. Hinton. Distilling a Neural Network Into a Soft Decision Tree. *arXiv:1711.09784 [cs, stat]*, Nov. 2017. URL http://arxiv.org/abs/1711.09784. arXiv: 1711.09784.

[24] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Mach Learn*, 63(1):3–42, Apr. 2006. ISSN 1573-0565. doi: 10.1007/s10994-006-6226-1. URL https://doi.org/10.1007/s10994-006-6226-1.

[25] S. Ghosal and A. v. d. Vaart. Convergence rates of posterior distributions for noniid observations. *The Annals of Statistics*, 35(1):192–223, Feb. 2007. ISSN 0090-5364, 2168-8966. doi: 10.1214/009053606000001172.

[26] K. Gleason, J. P. Shine, N. Shobnam, L. B. Rokoff, H. S. Suchanda, M. O. S. Ibne Hasan, G. Mostofa, C. Amarasiriwardena, Q. Quamruzzaman, M. Rahman, et al. Contaminated turmeric is a potential source of lead exposure for children in rural bangladesh. *Journal of Environmental and Public Health*, 2014, 2014.

[27] S. E. Golovenkin, J. Bac, A. Chervov, E. M. Mirkes, Y. V. Orlova, E. Barillot, A. N. Gorban, and A. Zinovyev. Trajectories, bifurcations, and pseudo-time in large clinical datasets: applications to myocardial infarction and diabetes data. *GigaScience*, 9(11):giaa128, Nov. 2020. ISSN 2047-217X. doi: 10.1093/gigascience/giaa128. URL https://doi.org/10.1093/gigascience/giaa128.

[28] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

[29] T. Hastie, R. Tibshirani, and M. Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman & Hall/CRC, 2015. ISBN 978-1-4987-1216-3.

[30] T. J. Hastie and R. J. Tibshirani. *Generalized Additive Models*. Chapman and Hall/CRC, Boca Raton, Fla, 1st edition edition, June 1990. ISBN 978-0-412-34390-2.

[31] G. E. Hinton and R. R. Salakhutdinov. Using deep belief nets to learn covariance kernels for gaussian processes. *Advances in neural information processing systems*, 20, 2007.

[32] A. E. Hoerl and R. W. Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1):55–67, Feb. 1970. ISSN 0040-1706. doi: 10.1080/00401706.1970.10488634.

[33] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[34] O. Irsoy, O. T. Yildiz, and E. Alpaydin. Soft decision trees. *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, 2012.

[35] A. Jacot, B. Simsek, F. Spadaro, C. Hongler, and F. Gabriel. Implicit regularization of random feature models. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4631–4640. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/jacot20a.html.

[36] A. Karthikeyan, N. Jain, N. Natarajan, and P. Jain. Learning accurate decision trees with bandit feedback via quantized gradient descent. *arXiv preprint arXiv:2102.07567*, 2021.

[37] M. L. Kile, E. G. Rodrigues, M. Mazumdar, C. B. Dobson, N. Diao, M. Golam, Q. Quamruzzaman, M. Rahman, and D. C. Christiani. A prospective cohort study of the association between drinking water arsenic exposure and self-reported maternal health symptoms during pregnancy in bangladesh. *Environmental Health*, 13(1):1–13, 2014.

[38] R. Kohavi. Scaling Up theaADceccuisriaocny-TorfeNeaHivyeb-Bridayes Classi ers:. page 6.

[39] J. Lee, Y. Bahri, R. Novak, S. S. Schoenholz, J. Pennington, and J. Sohl-Dickstein. Deep neural networks as gaussian processes. *arXiv preprint arXiv:1711.00165*, 2017.

[40] F. Liang, Q. Li, and L. Zhou. Bayesian Neural Networks for Selection of Drug Sensitive Genes. *Journal of the American Statistical Association*, 113(523):955–972, July 2018. ISSN 0162-1459. doi: 10.1080/01621459.2017.1409122. URL https://doi.org/10.1080/01621459.2017.1409122.

[41] F. Liu, X. Huang, Y. Chen, and J. A. Suykens. Random features for kernel approximation: A survey on algorithms, theory, and beyond. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (01):1–1, 2021.

[42] J. Liu. Variable selection with rigorous uncertainty quantification using deep bayesian neural networks: Posterior concentration and bernstein-von mises phenomenon. In *International Conference on Artificial Intelligence and Statistics*, pages 3124–3132. PMLR, 2021.

[43] J. Z. Liu. Gaussian Process Regression and Classification under Mathematical Constraints with Learning Guarantees. *arXiv:1904.09632 [cs, math, stat]*, Apr. 2019. URL http://arxiv.org/abs/1904.09632.

[44] Y. Y. Lu, Y. Fan, J. Lv, and W. S. Noble. DeepPINK: reproducible feature selection in deep neural networks. *arXiv:1809.01185 [cs, stat]*, Sept. 2018. URL http://arxiv.org/abs/1809.01185.

[45] J. Mairal and B. Yu. Complexity analysis of the lasso regularization path. *arXiv preprint arXiv:1205.0079*, 2012.

[46] S. Mei and A. Montanari. The generalization error of random features regression: Precise asymptotics and the double descent curve. *arXiv: Statistics Theory*, 2019.

[47] C. A. Micchelli, Y. Xu, and H. Zhang. Universal kernels. *Journal of Machine Learning Research*, 7(12), 2006.

[48] R. M. Neal. *Bayesian Learning for Neural Networks*. Springer-Verlag, Berlin, Heidelberg, 1996. ISBN 978-0-387-94724-2.

[49] S. W. Ober and C. E. Rasmussen. Benchmarking the neural linear model for regression. In *2nd Symposium on Advances in Approximate Bayesian Inference*, 2019.

[50] T. Park and G. Casella. The Bayesian Lasso. *Journal of the American Statistical Association*, 103(482):681–686, June 2008. ISSN 0162-1459. doi: 10.1198/016214508000000337. URL https://doi.org/10.1198/016214508000000337.

[51] N. Polson and V. Rockova. Posterior Concentration for Sparse Deep Learning. *arXiv:1803.09138 [cs, stat]*, Mar. 2018. URL http://arxiv.org/abs/1803.09138.

[52] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, pages 1177–1184, Red Hook, NY, USA, Dec. 2007. Curran Associates Inc. ISBN 978-1-60560-352-0.

[53] A. Rahimi and B. Recht. Uniform approximation of functions with random bases. In *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, pages 555–561. IEEE, 2008.

[54] A. Rahimi and B. Recht. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc., 2009. URL https://proceedings.neurips.cc/paper/2008/file/0efe32849d230d7f53049ddc4a4b0c60-Paper.pdf.

[55] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning series)*. The MIT Press, hardcover edition, 11 2005. ISBN 978-0262182539. URL https://lead.to/amazon/com/?op=bt&la=en&cu=usd&key=026218253X.

[56] V. Rockova. On semi-parametric inference for bart. In *International Conference on Machine Learning*, pages 8137–8146. PMLR, 2020.

[57] V. Ročková and S. van der Pas. Posterior concentration for bayesian regression trees and forests. *The Annals of Statistics*, 48(4):2108–2131, 2020.

[58] L. Rosasco, S. Villa, S. Mosci, M. Santoro, and A. Verri. Nonparametric sparsity and regularization. *The Journal of Machine Learning Research*, 14(1):1665–1714, Jan. 2013. ISSN 1532-4435.

[59] A. Rudi and L. Rosasco. Generalization Properties of Learning with Random Features. *arXiv:1602.04474 [cs, stat]*, Jan. 2018. URL http://arxiv.org/abs/1602.04474.

[60] G. Santin and R. Schaback. Approximation of eigenfunctions in kernel-based spaces. *Advances in Computational Mathematics*, 42(4):973–993, 2016.

[61] J. Schmidt-Hieber. Nonparametric regression using deep neural networks with ReLU activation function. *The Annals of Statistics*, 48(4):1875–1897, Aug. 2020. ISSN 0090-5364, 2168-8966. doi: 10.1214/19-AOS1875.

[62] J. Snoek, O. Rippel, K. Swersky, R. Kiros, N. Satish, N. Sundaram, M. M. A. Patwary, Prabhat, and R. P. Adams. Scalable bayesian optimization using deep neural networks. In *Proceedings of the 32nd International Conference on Machine Learning*, 2015.

[63] R. Tanno, K. Arulkumaran, D. Alexander, A. Criminisi, and A. Nori. Adaptive Neural Trees. In *Proceedings of the 36th International Conference on Machine Learning*, pages 6166–6175. PMLR, May 2019. URL https://proceedings.mlr.press/v97/tanno19a.html. ISSN: 2640-3498.

[64] S. Thakur, C. Lorsung, Y. Yacoby, F. Doshi-Velez, and W. Pan. Uncertainty-aware (una) bases for deep bayesian regression using multi-headed auxiliary networks. *arXiv preprint arXiv:2006.11695v4*, 2021.

[65] R. Tibshirani. Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996. ISSN 2517-6161. doi: https://doi.org/10.1111/j.2517-6161.1996.tb02080.x.

[66] L. Wang, L. Xue, A. Qu, and H. Liang. Estimation and model selection in generalized additive partial linear models for correlated data with diverging number of covariates. *Annals of Statistics*, 42(2):592—624, 2014.

[67] Y. Wang and V. Rocková. Uncertainty quantification for sparse deep learning. In *International Conference on Artificial Intelligence and Statistics*, pages 298–308. PMLR, 2020.

[68] D. Wipf and S. Nagarajan. A new view of automatic relevance determination. *Advances in neural information processing systems*, 20, 2007.

[69] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82, 1997.

[70] S. N. Wood. *Generalized additive models: an introduction with R*. chapman and hall/CRC, 2006.

[71] Y. Yang, I. G. Morillo, and T. M. Hospedales. Deep Neural Decision Trees. *arXiv:1806.06988 [cs, stat]*, June 2018. URL http://arxiv.org/abs/1806.06988. arXiv: 1806.06988.

[72] L. Yao, Z. Chu, S. Li, Y. Li, J. Gao, and A. Zhang. A survey on causal inference. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(5):1–46, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See line 53-69.
   (b) Did you describe the limitations of your work? [Yes] See **Limitations** paragraph in Section 5.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See **Societal Impacts** paragraph in Section 5.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See line 203-205 and line 241-242.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix C and Appendix D.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Instructions for downloading and processing data are in the Appendix G. Code repository is in preparation at GitHub and will be released in camera-ready version.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix G.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Appendix G.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix G.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you citet the creators? [Yes] See line 314-318.
   (b) Did you mention the license of the assets? [Yes] See Appendix G.2.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See Appendix G.2.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] All data involved in this work are already previously open-sourced. They all can be downloaded via external URL linked in Appendix G.2.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] See Appendix G.2.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]