M&C 2017 - International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering, Jeju, Korea, April 16-20, 2017, on USB (2017)

Sparse Bayesian Regression with Integrated Feature Selection for Nuclear Reactor Analysis

Kenneth Dayman*, Brian Ade†, Charles Weber*

*Nuclear Security Modeling Group, Oak Ridge National Laboratory, PO Box 2008 MS6170, Oak Ridge, TN 37831-6170
†Reactor Physics Group, Oak Ridge National Laboratory, PO Box 2008 MS6170, Oak Ridge, TN 37831-6170
daymankj@ornl.gov, adebj@ornl.gov, webercf@ornl.gov

Abstract - *High-dimensional-nonlinear function estimation using large datasets is a current area of interest in the machine learning community. Applications permeate throughout the analytical sciences, where ever-growing datasets are providing more information to the analyst. This paper leverages the existing relevance vector machine, a sparse Bayesian version of the well-studied support vector machine and expands the method to include integrated feature selection and automatic function shaping. These innovations produce an algorithm that can distinguish variables useful for predicting a response from variables that are unrelated or confusing. The technology has been tested using synthetic data, initial performance studies have been conducted, and a model has been developed that is capable of making position-independent predictions of the core-averaged burnup using a single specimen drawn randomly from a nuclear reactor core.*

I. INTRODUCTION

1. Need for Complex Function Estimation

Discovery of relationships between observable phenomena and prediction of future behavior or unobserved internal behavior of a physical system is the central focus of the analytical sciences. As instrumentation has become more sophisticated, measured datasets have grown. While this growth has increased the available information encoded in large, complex datasets, data interpretation and analysis has accordingly become more complex. Thus, function estimation has been passed to the arena of machine learning, where complex functions relating measured quantities to responses of interest are estimated using statistical analysis.

Machine learning and multivariate statistical analyses have been demonstrated in nuclear security applications. Orton et al. demonstrate the use of linear principal components analysis to discriminate between normal and off-normal chemistry process conditions in aqueous reprocessing scenarios using gamma-ray spectra collected of the aqueous phase following separation [1, 2]. Models implicitly correlate spectral regions to gamma-emitting fission products sensitive to nitric acid and tributyl phosphate concentrations, informing the analyst of nuclides of interest and spectral regions of interest. Further research extended this technology to predicting spent nuclear fuel burnup and reactor type classification using k-nearest neighbors, linear and quadratic discriminant analyses, support vector machines, and partial least squares regression [3].

The support vector machine (SVM) is a state-of-the-art machine learning algorithm capable of learning complex high-dimensional nonlinear relationships [4, 5]. However, the SVM has several shortcomings. First, only the best estimate of the function response is calculated when applied to new data, and it is unclear what the output of the SVM represents statistically. The lack of a probabilistic framework is especially problematic in classification problems, where the difference between class membership may be subtle, and predicting the relative probability that a test sample belongs to each possible class would more accurately represent the state of knowledge than single-point estimates. External routines have been developed to give probabilistic context to the SVM's output; however, it would be desirable to integrate uncertainty quantification natively into an algorithm without additional computational burden. Furthermore, the SVM makes no effort to identify and select variables/features that are most useful for making predictions. Separate algorithms may be employed to analyze the dataset to determine the variables with the most predictive power [6]. Such routines introduce computational burden beyond training the SVM model, and many selection algorithms are agnostic to the model when ranking and/or selecting variables. A variable that may be useful when passed to one model may not be useful when passed into a different type of analysis. Ideally, *feature selection* should be coupled to the specific prediction task and method.

2. Reactor Characterization

As a motivating example, we consider the determination of core-average burnup of a nuclear reactor core given a potentially limited sample of specimen(s) of irradiated fuel drawn randomly from the core during or after operation. The composition of irradiated fuel depends on coupled variables, including the initial fuel composition (e.g., matrix and enrichment), position within the core (i.e., the neutron flux and neutron spectrum), and time. We seek to discover multi-nuclide signatures capable of predicting one or more of these variables with no knowledge of the other variables. This paper describes the initial efforts to predict core-averaged burnup from a single fuel specimen with no knowledge of other operating conditions.

3. Proposed Improvement: Relevance Vector Machine

In this paper, we leverage an alternative to the SVM called the relevance vector machine (RVM) [7]. We describe research and development efforts to extend the capabilities of the RVM to include online basis shaping, integrated feature selection, and quantification of the uncertainty in predictions made by trained models. The results of applying the extended RVM to synthetic example data are described, along with a real-world application: learning multivariate signatures to predict the core-average burnup of irradiated nuclear fuel using the isotopic concentrations from a single specimen drawn at random from the core.

The remainder of the paper proceeds as follows: Sections II. and III. give an overview of the RVM and subsequent rapid training algorithm as developed by Tipping [7, 8] and the extension we have made to integrate feature selection and basis shaping parameter optimization, respectively. Section IV. summarizes methods to evaluate the uncertainty in predictions made by the RVM and discusses unique concerns arising from the RVM formulation. Sections V. and VI. discuss the results from applying the extended RVM for discovery of nuclide/reactor-state relations and conclusions, respectively.

4. Notation

Several conventions of notation and terminology are adopted herein. Measured inputs to a model will be vectors of d variables, $x \in \mathbb{R}^d$. When analyzing irradiated nuclear fuel, variables are concentrations of nuclides in a particular specimen drawn from some position within the core after a specified irradiation time. Each vector is called an *observation*, and sets of observations are indexed with subscripts, $\{x_i\}_{i=1}^N$ and collected into $N \times d$ dimensional matrices, $X \in \mathbb{R}^{N\times d}$, for input to the RVM. Individual variables/components of vectors are indexed using superscripts (i.e., x^i). Thus, the collection $x_1, x_2, \ldots, x_N$ are the N specimens drawn from the reactor (each of which is a collection of d nuclide measurements), $x^1, x^2, \ldots, x^d$ are the d measured nuclide concentrations in every observation, and x_i^j is the concentration of the j^{th} nuclide in the i^{th} specimen. The response (i.e., quantity to be predicted), also called the *target*, is denoted by the vector t[1].

II. RELEVANCE VECTOR MACHINES

The RVM is an alternative to the SVM that uses Bayesian statistics to derive a sparse, probabilistic model capable of learning qualitative and quantitative relationships [7]. This section briefly reviews the original development of the RVM for regression problems and the subsequent accelerated training algorithm [8].

1. Model Formulation

When applied to regression problems, the RVM aims to estimate an unknown function between predictors, x, and targets, t. The algorithm analyzes a set of training data with known predictors and target values and estimates the underlying function using a linear basis-function expansion:

$$t = f(x) + \epsilon = \sum_{m=1}^{M} w_m \phi_m(x) + \epsilon, \tag{1}$$

where each $\phi_i(x)$ is an arbitrary function, ϵ is unknown measurement noise, and w_i are weights to be determined.

If we assume independently and identically distributed Gaussian noise, $\epsilon \sim \mathcal{N}(\epsilon \,|\, 0, \sigma^2)$, we may write the probability of observing the training measurements given the training inputs (i.e., the Bayesian likelihood function) as

$$p(t \,|\, w, \sigma) = (2\pi)^{-N/2} \sigma^{-N} \exp\left(-\frac{\|t - w^T\Phi\|_2^2}{2\sigma^2}\right). \tag{2}$$

Here we have collected the basis functions evaluated at each of the training observations into a matrix $[\Phi]_{mn} = \phi_m(x_n)$ and specify the i^{th} basis function evaluated at the training points Φ^i.

Following standard hierarchical Bayesian analysis, we specify a prior distribution for the values of w. We use a mean-zero Gaussian distribution with hyperparameters $\alpha \in \mathbb{R}^N$:

$$p(w \,|\, \alpha) = (2\pi)^{-M/2} \prod_{m=1}^{M} \alpha_m^{1/2} \exp\left(-\frac{\alpha_m w_m^2}{2}\right). \tag{3}$$

The mean-zero Gaussian prior preferentially drives the weights towards zero, and the independent entries of the hyperparameter α control how strongly each entry of w is driven to zero. If some entry $w_m = 0$, the basis function $\phi_m(x)$ is not used in the function estimation shown in Equation (1), and the expansion is sparse. The basis functions retained in the final model are called *relevance vectors*.

The Gaussian prior is conjugate to the likelihood, giving an analytically tractable posterior distribution. As shown by Tipping [7], combining Equations (2) and (3) gives the posterior distribution

$$p(w \,|\, t, \alpha, \sigma) = \frac{|\Sigma|^{-1/2}}{(2\pi)^{-(N+1)/2}} \exp\left\{-\frac{(w-\mu)^T\Sigma^{-1}(w-\mu)}{2}\right\}, \tag{4}$$

where

$$\Sigma = \left(\sigma^{-2}\Phi^T\Phi + A\right)^{-1}, \tag{5}$$

$$\mu = \sigma^{-2}\Sigma\Phi^T t, \tag{6}$$

and

$$[A]_{ii} = \alpha_i, \quad [A]_{ij} = 0. \tag{7}$$

By virtue of the explicit calculation of the posterior distribution for the weights, we obtain a distribution for the estimated function. In the case of regression, we obtain predicted targets and associated uncertainty outputs. For classification problems, we obtain probabilities of class membership.

As discussed in Tipping and Faul [8], solving for the best estimate α amounts to maximizing the *marginal likelihood*

$$\mathcal{L}(\alpha) = -\frac{1}{2}\left[N \log 2\pi + \log|C| + t^T C^{-1} t\right], \tag{8}$$

[1] We use the term target to differentiate the measured value(s), which is observed with error, from the true response y, which is assumed to be perfectly described by the function to be estimated by the learning algorithm.

where $C = \sigma^2 I + \Phi A^{-1} \Phi^T$.

After optimizing Equation (8), Equations (6) and (5) are solved to specify the posterior distribution for the weights, w, and the best estimates of the weights are substituted into Equation (1) to give the estimated function.

Lastly, we define the basis functions, $\phi(x)$. While the functions are arbitrary, in the absence of problem-specific information[2], kernel functions using the input training are the most convenient and general. Kernel functions are localized to the range of predictor values in the training data (i.e., they are scaled to the problem) and have interesting connections to nonlinear mappings of the input data. Using kernel transforms allows the RVM to estimate highly nonlinear functions. There is a wealth of research in the literature concerning kernel functions, their interpretation, and connection with functional analysis (see [9, 10]). In any model, the set of basis functions is formed by evaluating the desired kernel against each of the training vectors so each training observation corresponds to a basis function: $x_m \rightarrow \phi_m(x)$.

The i^{th} basis function generated using the Gaussian kernel is

$$\phi_i(x) = K(x, x_i) = \exp\left(-\eta \sum_{j=1}^{d} \left(x^j - x_i^j\right)^2\right), \tag{9}$$

where η is a scalar shaping factor that must be tuned. The optimization of η is discussed in Section III..

2. Constructive Optimization

In the original development of the RVM, an iterative updating procedure was used to maximize Equation (8); however, two key observations allow a much faster constructive procedure. Updating the basis-shaping parameters adds complexity and computational burden (see Section III.), making fast optimization of Equation (8) a requirement for practical application.

Decomposition of $\mathcal{L}(\alpha)$ As shown in Tipping and Faul [8], the matrix C may be decomposed into a portion relating to a the i^{th} basis function, C_i, and a portion built with the remaining functions, C_{-i}.

$$C = \sigma^2 I + \Phi A^{-1} \Phi^T \tag{10}$$

$$= \sigma^2 I + \sum_{j \neq i} \Phi^j \alpha_j \left(\Phi^j\right)^T + \Phi^i \alpha_i \left(\Phi^i\right)^T \tag{11}$$

$$= C_{-i} + C_i \tag{12}$$

This decomposition is propagated through the calculation of the determinate and inverse in Equation (8):

$$|C| = |C_{-i}| \, |1 + \alpha_i^{-1} \left(\Phi^i\right)^T C_{-i}^{-1} \Phi^i| \tag{13}$$

$$C_{-1} = C_{-i}^{-1} - \frac{C_{-i}^{-1} \Phi^i \left(\Phi^i\right)^T C_{-i}^{-1}}{\alpha_i + \left(\Phi^i\right)^T C_{-i}^{-1} \Phi^i} \tag{14}$$

[2]For example, if the function to estimated were assumed to be periodic, sinusoidal function families would be a logical choice for basis functions.

Finally, Equation (8) is rewritten as

$$\mathcal{L}(\alpha) = \mathcal{L}(\alpha_{-i}) + \frac{1}{2}\left[\log \alpha_i - \log\left(\alpha_i + \left(\Phi^i\right)^T C_{-i}^{-1} \Phi^i\right) + \frac{\left(\left(\Phi^i\right)^T C_{-i}^{-1} t\right)^2}{\left(\alpha_i + \left(\Phi^i\right)^T C_{-i}^{-1} \Phi^i\right)}\right]. \tag{15}$$

This decomposition allows the change in the marginal likelihood caused by inclusion or deletion of a basis function to be calculated.

Optimal Values for α_i As shown in Faul and Tipping [11], the optimal values for α_i given in the bracketed term above may be explicitly calculated:

$$\alpha_i = \frac{\left(\left(\Phi^i\right)^T C_{-i}^{-1} \Phi^i\right)^2}{\left(\left(\Phi^i\right)^T C_{-i}^{-1} t\right)^2 - \left(\Phi^i\right)^T C_{-i}^{-1} \Phi^i}, \quad \text{if } \left(\left(\Phi^i\right)^T C_{-i}^{-1} t\right)^2 > \left(\Phi^i\right)^T C_{-i}^{-1} \Phi^i$$

$$\alpha_i = \infty, \quad \text{if } \left(\left(\Phi^i\right)^T C_{-i}^{-1} t\right)^2 \leq \left(\Phi^i\right)^T C_{-i}^{-1} \Phi^i. \tag{16}$$

The second case arises when a basis function ϕ_i explains less of the residual with that function excluded[3] than the degree of overlap with the functions currently in the model. In other words, ϕ_i does not greatly improve the prediction accuracy and carries the same information as basis functions already included in the model. In this case, the optimal value of α_i is infinite, corresponding to a delta distribution about zero for the corresponding weight–c.f., Equation (3). Thus, the i^{th} basis function is not utilized when $\alpha_i = \infty$ and may be deleted. Removing basis functions gives rise to sparsity in terms of the basis functions, and any functions used in the final trained model are called the *relevance vectors*.

Using these two observations, a constructive algorithm relying on adding and deleting basis functions while adjusting α values is used to accelerate optimization of $\mathcal{L}(\alpha)$. A flow chart for the constructive optimization of $\mathcal{L}(\alpha)$ is shown in the "Initialization" and "α-updating" portions of Figure 3. This algorithm was implemented by Tipping in a prototype Matlab code called SparseBayes [12].

III. EXTENSION TO SPARSE BASIS SHAPING

The performance of the RVM depends on the basis functions used to construct the function estimate. Consider a

[3]See the term $\left(\Phi^i\right)^T C_{-i}^{-1} t$, the inner product of the particular basis function with the existing model and vector of target/measured training responses.

function of two variables, where one variable is not important for prediction of the response, such as the function $f(x^1, x^2) = \sqrt{x^1} + \sin(x^1)$, shown in Figure 1. The ground-truth function (surface) and noisy synthetic training data sampled from this function (blue dots) with the second variable x^2 randomly distributed are shown. Figure 2, shows two estimated functions using the training data in Figure 1: $\eta = 2.0$ in Figure 2a, and $\eta = 0.05$ in Figure 2b. In Figure 2a, the model is overfit, and the estimated function closely tracks the training data but is unlikely to generalize to new data. In contrast, the estimate shown in Figure 2b does not capture the periodic behavior of the function.

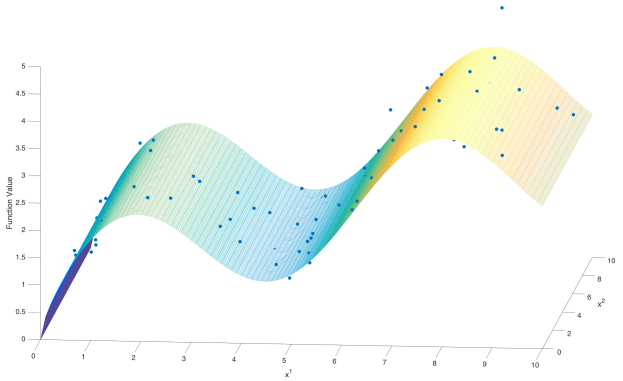


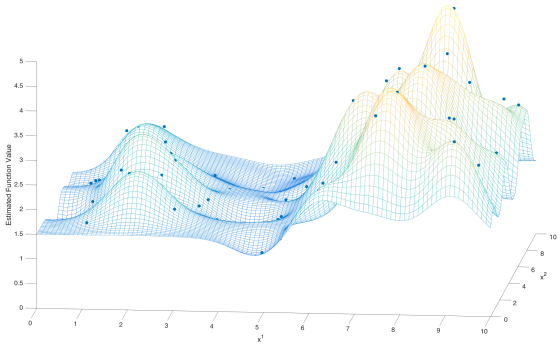
Fig. 1: Motivating example for the need of integrated basis shaping. A function of two variables, where one variable is not useful for prediction, is shown (surface) along with noisy data (blue dots) used to train RVM models shown in Figure 2.

As suggested by the results presented in Figure 2, the predictive power of the RVM depends on the proper choice of η in the formation of the basis functions. Ideally, less useful variables should be identified and removed from the model. The basis scaling factor associated with the remaining variables could then be optimized for each variable in order to maximize model fidelity.

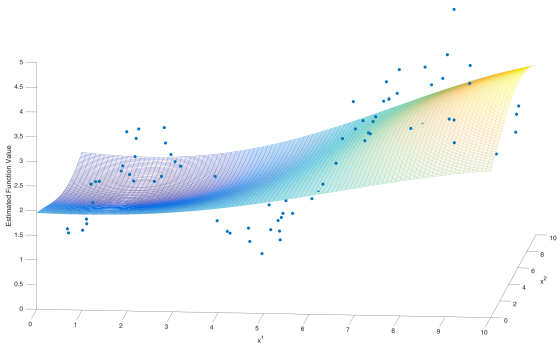
Discovery of useful predictive variables and tuning model parameters are typically performed with feature selection procedures and cross validation, respectively [6, 13]. These analyses often require additional model development, post-processing, and application of heuristic criteria. As suggested by Tipping [7], it is possible to integrate tuning of unique values for each variable during RVM training. Equation (17) modifies the kernel/basis function definition shown in Equation (9) by redefining η as a d-dimensional vector with an independent entry for each variable:

$$K(x, x_m) = \exp\left(-\sum_{k=1}^{d} \eta_k \left(x^k - x_m^k\right)^2\right). \tag{17}$$

Tipping asserted this capability to be prohibitively computationally expensive, but we have overcome this challenge by integrating a rapid approach for integrated feature selection and model tuning using a probabilistic interleaving of a partial optimization routine. A unique value of η_k is assigned to



(a) An overfitted function



(b) Fitted function fails to capture periodic behavior

Fig. 2: Two examples of poorly fitted functions as a result of choice of η. The estimated functions are shown with the surface plots against the training data, which are shown in dots. In both cases, a single value of η was used for both input variables, x^1 and x^2; however, x^2 is not useful for prediction, and retaining this variable skews results.

each input variable (e.g., nuclide concentration or channel in a gamma-ray spectrum). After each α-update loop, a second loop is entered, and the *vector* $\eta \in \mathbb{R}_+^d$ is updated using a gradient descent algorithm.

The gradient of the likelihood function in Equation (8), $\mathcal{L}(\alpha)$ with respect to η,

$$\nabla f_\eta = \left(\frac{\partial \mathcal{L}}{\partial \eta_1} \frac{\partial \mathcal{L}}{\partial \eta_2} \cdots \frac{\partial \mathcal{L}}{\partial \eta_d}\right)^T, \tag{18}$$

is computed analytically with Equation (19),

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \eta_k} &= \sum_{n=1}^{N} \sum_{m=1}^{M} \frac{\partial \mathcal{L}}{\partial \Phi_{nm}^{(i)}} \frac{\partial \Phi_{nm}^{(i)}}{\partial \eta_k} \\ &= -\sum_{n=1}^{N} \sum_{m=1}^{M} D_{nm} \Phi_{nm}^{(i)} \left(x_m^k - x_n^k\right)^2, \end{aligned} \tag{19}$$

where $D = \left(C^{-1} t t^T C^{-1} - C^{-1}\right) \Phi^{(i)} A^{-1}$, and $\Phi^{(i)}$ is the matrix of basis functions currently used in the model (i.e., the associated α values are finite), shaped by the current value of the η vector. The new index, i, denotes the current iteration of the

constructive alpha-updating iteration (the outer loop shown in Figure 3). Using this gradient, η may be updated using gradient descent [14] with a backtracking line search (BLS) to select the near-optimal step size at each η-updating iteration.

As shown in Section II.2., the optimal values for α and the associated progress of the constructive optimization algorithm depends on the basis functions, and the basis functions are determined through the static training data and (now) dynamic basis shaping factors, η. Each time η is updated, the basis functions must be reevaluated. Because of this dependence between α and η, there is substantial cross-talk between the two terms and the associated optimization routine loops (see Figure 3). Therefore, η is only partially optimized after each α-update iteration in order to preserve momentum in the optimization of α. During initial studies, several features of the algorithm were observed:

1. The algorithm is sensitive to multicollinearity,[4]
2. The entries of η converge to near their final, near-optimal values much faster than the entries of α, and
3. Entries η_k and η_j move towards similar values with opposite signs, yielding canceling values and non-unique solutions for η.

To overcome the challenges outlined above, the gradient descent algorithm was changed to gradient projection to enforce non-negativity of the entries of η, i.e.,

$$\eta_k \geq 0, \; k = 1, 2, \ldots, d. \tag{20}$$

In addition, the interleaving of the η-updating loop was changed to be probabilistic. Inspired by simulated annealing, the inner loop is entered with some probability that decreases as α-updating loops are performed. The final flowsheet for training the extended RVM, which builds on the original SparseBayes implementation, is shown in Figure 3. This new code is called SB(η).

Each time the value of η is changed, the entire basis matrix, Φ, is recalculated, and the basis vectors currently included in the model (i.e., those with associated $\alpha_i \neq \infty$) are extracted. The value of η is updated during gradient calculations, at each iteration of the iterative BLS routine used to select the step size within the gradient projection algorithm, and upon finishing the inner η-update loop shown in Figure 3.

Using SB(η), the motivating example shown in Figure 1 was analyzed again, and the results are shown in Figure 4. The algorithm was able to determine x^2 is useless for prediction, and x^2 was discarded by setting $\eta_2 = 0$. The remaining η_1 was optimized to maximize the marginal likelihood with respect to η as shown in Equation (19), and the true generating function is successfully reconstructed (see the surface in Figure 4). The estimated function tracks very well with the noisy training data (dots). The red dots are the training observations corresponding to basis functions that are not discarded by the constructive optimization algorithm and are used in the final RVM model. The final model is sparse in terms of basis functions (5 RVs are used) and variables (one half of the input variables are discarded).

[4]If some basis functions are too closely aligned, either because some subset of training observations are too similar or a subset of variables are collinear, the matrix C in Equation (19) is no longer positive definite, and the Cholesky decomposition used to robustly invert the matrix fails.

IV. EVALUATION OF UNCERTAINTY

1. Standard Approach

After training the model (i.e., optimizing Equation (8)), the posterior distribution is obtained for the weights w that define the function estimate in Equation (1), $f(x)$. Given a set of measurements collected of a test specimen, x_*, a prediction of the response is made by evaluating the function $f(x_*)$. The uncertainty associated with this prediction is estimated by summing in quadrature the irreducible error, σ (see [13]), with the uncertainty in the weights in the trained model:

$$u(f(x_*))^2 = \sigma^2 + J^T \Sigma J, \tag{21}$$

where J is the Jacobian matrix and Σ is the posterior covariance matrix implicitly evaluated at the test point, x_*. As discussed in Hastie et al. [13] and shown in Equation (3), σ quantifies the variation in the measurement noise. Evaluating the terms of the Jacobian gives

$$u(f(x_*))^2 = \sigma^2 + \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\left. \frac{\partial f}{\partial w_i} \right|_{x=x_*} \right) \left(\left. \frac{\partial f}{\partial w_j} \right|_{x=x_*} \right) \Sigma_{ij}. \tag{22}$$

From the definition of f shown in Equation (1), the partial derivatives above are equal to the basis functions (shaped with the optimized η) evaluated at the test point, $\Phi_*^m = \phi_m(x_*), \quad m = 1, 2, \ldots, M$. Thus,

$$u(f(x_*))^2 = \sigma^2 + \Phi_*^T \Sigma \Phi_*. \tag{23}$$

The typical magnitude of the entries in Φ_* is related to the distance between the test point and the training data

$$d(x_*, X) = \inf \{ \|x_* - x\|_2 \mid x \in X \}, \tag{24}$$

which quantifies the similarity between the training and test/unknown observation in the L_2 sense. As shown in Equation (17), if the test and training observations are very similar, the quantity $\sum_{k=1}^{d} \left(x^k - x_m^k \right)^2$ will be small, the basis function evaluations in Equation (23) will be large, and the uncertainty in the prediction associated with x_* will also be large. As the distance increases, the opposite situation arises, and the estimated uncertainty may decrease. Thus, predicted uncertainties are expected to increase as the training and testing data become more similar. The dependence of model performance on the size of the training set is demonstrated in Figure 5, which shows the absolute error in prediction (bias) and the prediction uncertainty as calculated using Equation (23).

Using a dataset of approximately 8000 observations, each with 90 dimensions, random subsets of data were selected and used to train a model with the extended RVM. The remaining observations were used to test the model and quantify the average absolute error in prediction and associated uncertainty. To remove effects of the training set selection, this process

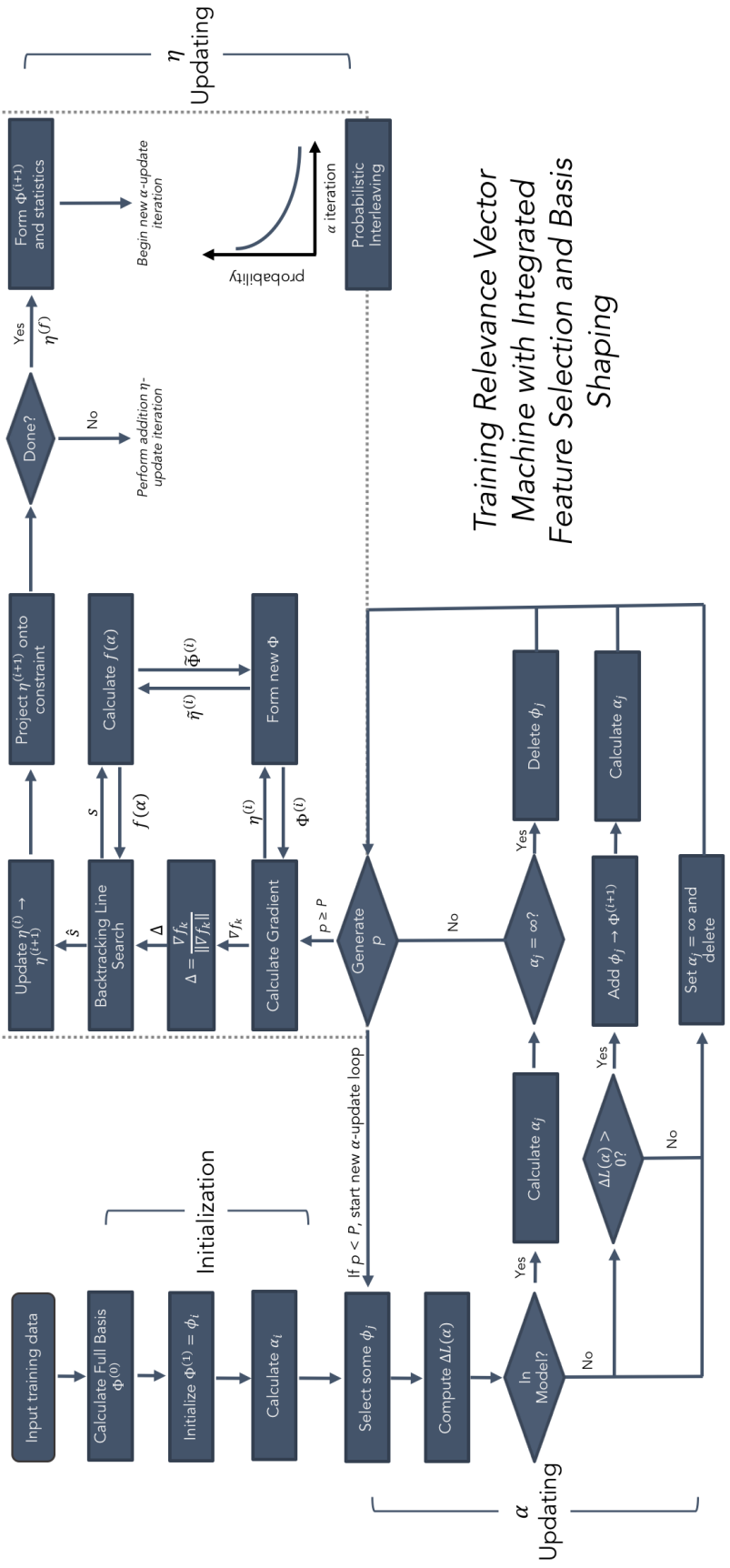


Fig. 3: Process for training the RVM with basis shaping updates with SB(η). The marginal likelihood is computed with Equation (8), the gradient with respect to η is given by Equation (19), the change in the likelihood given an inclusion or removal of a basis function is given by Equation (15), and α values are updated using Equation (16). Convergence checks and the termination procedure are not shown.

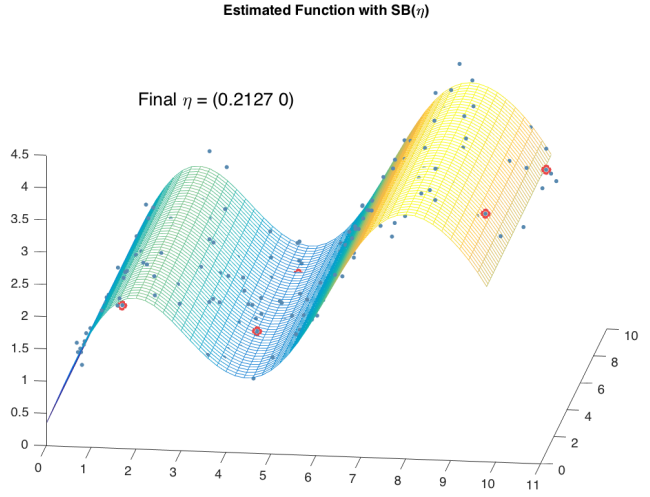


Fig. 4: Results of estimating two-dimensional function with one trivial value using SB(η). The algorithm successfully removes x^2, which is not useful for prediction and adjusts the shaping parameter associated with x^1 to optimize predictions. The surface shows the estimated function (c.f., Figure 1), and the dots show the training data. The relevance vectors are shown in red.

was repeated 50 times, and the results were averaged. Using this procedure, the effect of training set size on the bias and uncertainty of the predictions made by the extended RVM were explored, and the results are shown in Figure 5. The details of the data used in this experiment are further discussed in Section V.1..

2. Empirically Determined Uncertainty

As described above, there may be issues when using Equation (23) to compute the standard uncertainty in the predictions made using a RVM model. We consider an alternative method for computing uncertainty estimates. Given a large dataset, a subset is randomly chosen to use as training data. Once training is complete, predictions are made on the remainder of the data, the empirical cumulative distribution function for the absolute error in prediction is generated, and a bound for the prediction error is estimated such that the expected error in prediction falls below the bound with a desired probability.

Prior to training the model, the dataset is split into training and testing sets. The training set is used to find the near-optimal values of α and η using the methodology of Section III., and predictions are made on the test set. The absolute error in the predictions are binned into a histogram, $\{H_N(\delta_i)\}_i$, where δ_i is the absolute error in prediction discretized into bins. The histogram $\{H_N(\delta_i)\}_i$ is renormalized to give a discrete approximation for the distribution function for the absolute error in predictions made when using a model trained using N training observations, $f_N(\delta)$[5]:

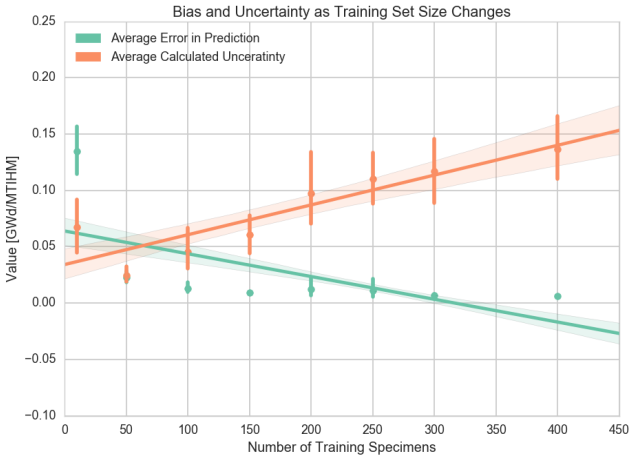


Fig. 5: The behavior of the error in prediction and uncertainty calculated with Equation (23) as a function of the number of observations included in the training set. Both quantities are averaged over the testing observations and 50 replicate trials.

$$f_N(\delta_i) = \frac{H_N(\delta_i)}{\frac{\delta_{max}}{J}\sum_{j=1}^{J}\left(H_N(\delta_{j+1}) - H_N(\delta_j)\right)} \quad i = 1, 2, \ldots, J. \tag{25}$$

The significance level α_{sig} defines a critical error rate below which predictions are expected to fall,

$$F_N(\delta_i) = \sum_{j=1}^{i} f_N(\delta_j) \tag{26}$$

$$F_N(\delta_{crit}) = 1 - \alpha_{sig} \rightarrow \delta_{crit}\left(N, \alpha_{sig}\right). \tag{27}$$

This approach has been used to define critical values and limits for analytical chemistry applications [15] and constitutes Type A uncertainty evaluation in line with the "Guide to Expression of Uncertainty in Measurements" [16].

The above routine is repeated at different values for N, and used to fit a function that estimates δ_{crit} at run time given the size of the training set used to train the model. Thus, for a model generated using N training observations, we specify a significance level α_{sig} and determine a critical value $\delta_{crit}(\alpha_{sig})$ such that the model's predictions will come within δ_{crit} of the true value with probability $1 - \alpha_{sig}$:

$$\mathbb{P}[|f(x_*) - t_*| \leq \delta_{crit}] = 1 - \alpha_{sig}. \tag{28}$$

This conclusion is only valid for test specimens drawn from reactor conditions that are consistent with the training and test sets used to develop the $\delta_{crit}(N, \alpha)$ curve.

V. REACTOR ANALYSIS WITH THE EXTENDED RVM

This section describes initial numerical studies intended to understand the performance of the extended RVM and the

[5]Note, a small abuse of notation: the probability density function $f_N(\delta)$ here is not related to the function estimated by the RVM in Equation (1).

SB(η) implementation described in Sections III. and IV.. These initial studies were performed in tandem with algorithm development; however, the final developed method and testing results are described separately to ensure clarity.

1. Data Generation

A dataset of isotopic concentrations of irradiated nuclear fuel as a function of position and irradation time within a reactor was generated with a set of depletion calculations. The model was a quarter-core of a gas-cooled, graphite-moderated reactor. The isotopics were tracked in 995 material regions (5 axial regions and a roughly 16×16 grid in the XY-plane) using the TRITON [17] routine within SCALE 6.2 and saved at 19 time points ranging from 0 to 700 days of constant-power operation. These irradiation times correspond to core-average burnup values ranging from 0 to approximately 1 GWd/MTIHM. The spatially resolved neutron spectrum and flux were calculated using continuous-energy neutron transport performed using KENO-VI [18]. Figure 6 shows the geometry of the model. The results of these simulations gave 8910 unique fuel compositions[6] to use for training and testing RVM models.

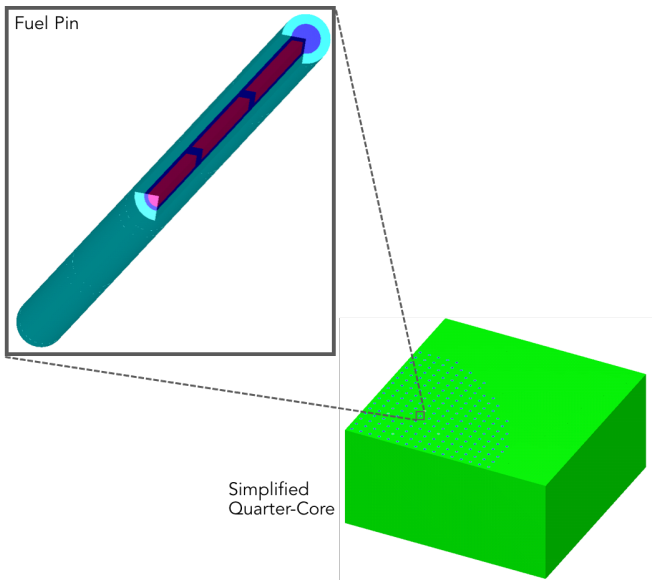


Fig. 6: Quarter-core gas-cooled, graphite-moderated reactor core model generated in TRITON/KENO to generate spatially resolved isotopics during constant-power reactor operation from 0 to 700 days, giving a core-average burnup of 0 to 1.05 GWd/MTIHM.

2. Model Training, Testing, and Profiling

For all numerical experiments, the complete dataset generated in KENO-VI was split into training and testing data. The training data were used to train the RVM model, and the trained model was applied to the testing data to predict the core-average burnup associated with each material. Prior to training, the matrix of training predictors X was standardized (i.e., each column was mean-centered and normalized to unit variance) to remove scaling effects. Without this preprocessing step, nuclides with small concentrations such as fission products with low yields[7] would be overwhelmed by nuclides with large concentrations such as ^{235}U and ^{238}U. These statistics are saved and used to standardize the test data.

Predictions were compared to the known irradiation times, and several performance measures were calculated:

1. The average absolute error in prediction made on the training set, $\frac{1}{N_t}\sum_{i=1}^{N_t}|t_i - f(x_i)|$,
2. The average absolute error in prediction made on the testing set,
3. The average relative error in prediction made on the testing set, $\frac{1}{N_{test}}\sum_{i=1}^{N_{test}}|(t_i - f(x_i))/t_i|$,
4. The number of relevance vectors (basis functions retained in the model), and
5. The number of nonzero entries of η (number of useful variables for prediction).

Several experiments were conducted to assess the behavior of the extended RVM when applied to the reactor core characterization problem, described in the following sections.

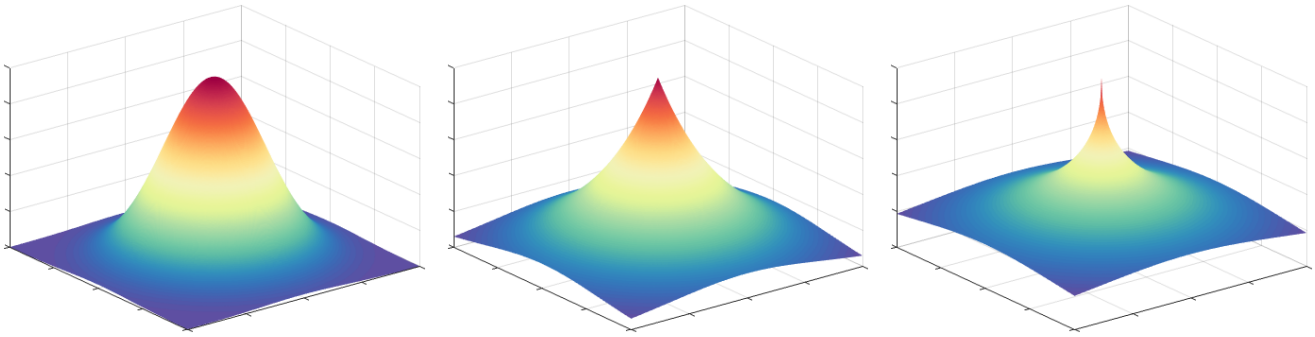
3. Testing Kernels

The dependence of the calculated uncertainty on the distance between the testing and training data as described in Section IV. and Equation (24) suggests that changing the kernel function used to develop the basis functions could improve the model by reducing the uncertainty in predictions. Three kernel functions were tested: the Gaussian kernel, the Laplacian kernel, and the heavy-tailed basis function. All three functions are similar, but two major differences are the kurtosis and the rate at which the function decreases with distance from the mean. Figure 7 shows plots of each function in $\mathbb{R}^2$. Of the three functions, the Gaussian kernel dies away with distance from the mean most rapidly. The heavy-tailed kernel has the highest kurtosis (i.e., it is the most sharply peaked).

To assess the performance of each kernel, a random subset of 300 observations was selected from the data generated in KENO-VI. The subset was selected and used to train a model with each kernel, predictions were made on the remainder of the data, and these results were used to calculate the performance metrics listed above. Using a modest number of observations accelerates model training and produces medium-fidelity results, while allowing replicates to be rapidly generated and studied. Table I summarizes the results, and three trends are apparent.

[6] Due to radial symmetry, approximately half of the $995 \times 19 = 18905$ calculated nuclide composition vectors were not unique and were discarded from analysis.

[7] These nuclides typically are found on the edges of the bimodal peaks in fission product yield curves and are more sensitive to changes in yield that may arise from changes in neutron spectrum or material composition. Accordingly, these “wing" fission products are likely to be useful in analyses.



(a) Gaussian kernel (b) Laplacian kernel (c) Heavy-tailed kernel

Fig. 7: Three basis functions used for developing RVM models.

First, when using the Gaussian kernel, predictions exhibit large variance with an average relative uncertainty of 30% in predictions on the testing data. When using the Laplacian and heavy-tailed kernels, the uncertainty calculated using Equation (23) decreases to 2.70% and 0.23%, respectively. The sharp decrease is likely due to greater kurtosis of these functions relative to the Gaussian function (i.e., there is greater large difference between the peak function value at the mean/center and the tails). The large kurtosis implies that the basis functions evaluated at the test points, Φ_* in Equation (23), are very small, and the sensitivity coefficients in Equation (23) are small.

Second, while the training error was consistently small in models using all three kernels, large increases in the testing error (error in predictions made on the testing set) relative to the training error (error in predictions made when applying the trained model to the training data) were observed when using more peak basis functions. The testing bias increased relative to the training bias by a factor of 4 and 678 when using the Laplacian and heavy-tailed kernels, respectively. These large increases in bias indicate the RVM models built using the Laplacian and heavy-tailed functions are overfitted and will likely not generalize well to new data. In contrast, the biases on the training and testing data were very consistent when the Gaussian kernel was used.

Third, models built with peaked basis functions are less sparse than the model built using Gaussian kernel basis functions. Models built with Gaussian kernels retained an average of 7 variables as indicated by the number of nonzero entries of the η vector, and 10 basis functions, which are associated with finite entries in the vector α leading to nonzero weights in Equation (1). In contrast, the models built using the Laplacian and heavy-tailed kernels used significantly more variables and basis functions. In general, sparse models are more easily interpretative, require less measured data to apply to new samples, and are more likely to generalize to new data more readily than complex models. One goal of the current research is to discover nuclides that are most useful for making different reactor characterizations independent of other unknown quantities—specifically nuclides capable of determining burnup independent of core position (which is coupled to neutron flux, spectrum, etc.). The integrated feature selection included in the extended RVM makes signature discovery possible, but using the Laplacian or heavy-tailed kernels leads to dense models that do not adequately differentiate between useful and uninformative variables.

Given these results, the Gaussian kernel is used for the remainder of the work presented herein.

4. Prediction of Core-Average Burnup

A high-fidelity RVM model to predict core-average burnup was developed using 650 training specimens extracted from the dataset described in Section V.1.. After training, all but 9 training observations were discarded (i.e., 9 relevance vectors were derived) and only 10 nuclides were retained (see Section V.4.C.). Once trained, the model was applied to the remaining synthetic specimens generated in the complete gas-cooled, graphite-moderated dataset (see Section V.1.), and the results were used to assess model performance. The burnup predictions, calculated uncertainties associated with predictions, and important predictive nuclides as identified during the integrated feature selection are discussed in the next three sections, respectively.

A. Prediction Results

Once trained, the RVM model was applied to each of 8260 testing specimens *individually* to make core-average burnup predictions along with associated uncertainty calculated with Equation (23). The burnup predictions versus the true values are shown in Figure 8. On average, the absolute error in prediction is 0.0062 GWd/MTIHM, and no major trends in the errors are observed. Since the testing specimens are sampled from all positions within the core and no associated trends in prediction error are observed, the multivariate signature autonomously developed with the extended RVM is *position-independent*. In other words, the quality of the predictions is invariant to the average neutron flux and spectrum associated with position within the core.[8]

[8]The position invariance is only valid within the bounds of variables currently considered. For example, cooling time is not considered in the data and models developed to date.

TABLE I: Summary of RVM performance with three forms for basis functions. After selecting training and testing data, a seperate model was built using basis functions derived from each kernel function, the model was used to make predictions on the testing data, and the results characeterized using several performance metrics. The results suggest models built using Laplacian and heavy-tailed kernels are overfitted (the testing error is much greater than the training error) and are less sparse in terms of the number of RV and nonzero entries in η (important variables).

	Gaussian	Laplacian	Heavy-Tailed
Relative Testing Uncertainty [%]	30.03	2.7	0.23
Training Bias [GWd/MTIHM]	0.0067	0.0019	0.0001
Testing Bias [GWd/MTIHM]	0.0072	0.0079	0.0407
No. $\eta_k \neq 0$	7	79	77
No. RV	10	96	289

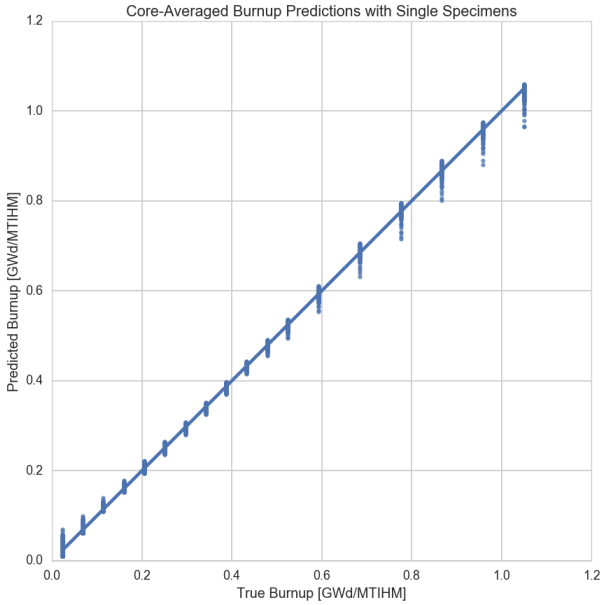


Fig. 8: Predicted versus actual core-average burnup. Predictions are made using the isotopic composition of a *single* simulated specimen. The average absolute error in prediction over the entire range of true values (0 to 1.05 GWd/MTIHM) is 0.0062 GWd/MTIHM, the average relative error is 4.47%, the average relative uncertainty is 9.39%, and the maximum error is 0.0470 GWd/MTIHM.

B. Uncertainty

During initial studies, the absolute error in prediction averaged over the testing set(s) was observed to change with the number of observations used in training the model (see Figure 5). As N increases and the set of available basis functions becomes more dense (i.e., the domain and image of the estimated function $f(x)$ is covered with finer resolution), the average absolute error in predictions made on testing data decreases while the uncertainty in those predictions increases.[9]

[9]This phenomenon is clearly related to the well-studied bias-variance tradeoff [13].

Figure 9 shows the average values of the absolute error in prediction and the uncertainty calculated with Equation (23) at each true burnup value. On average, the relative uncertainty in predictions was 9.4%; however, at every true value, the computed uncertainty (orange) is significantly larger than the difference between the predicted and actual core-average burnup (green). The consistency of the trend and size of the difference suggests an overprediction in the uncertainty as calculated by Equation (23); however, the reason for this behavior is not currently known.

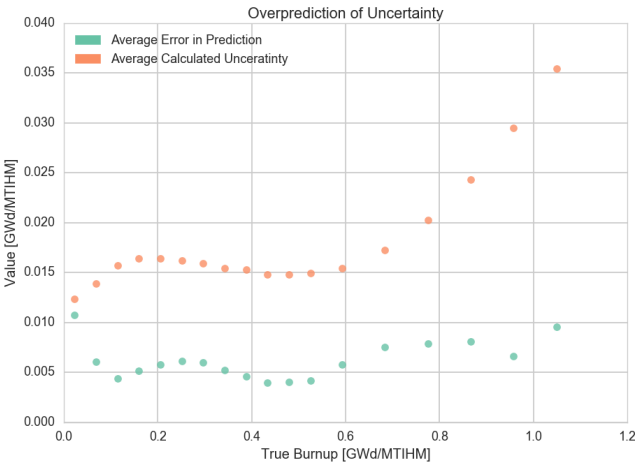


Fig. 9: The uncertainty calculated with Equation (23) and the absolute error in prediction. Points show the average value at each unique core-average burnup value.

Figure 10 shows the empirical probability density for the absolute error in predictions made using the same model used to generate Figures 8 and 9. Approximately 98% of all predictions are within 0.02 GWd/MTIHM of the true values. Using a confidence value of $\alpha_{sig} = 0.05$, the critical error in prediction value is 0.015 GWd/MTIHM (i.e., it is expected *a priori* that predictions made with this model will fall within 0.015 GWd/MTIHM of the true core-average burnup 95% of the time). Nominally this corresponds to an expanded uncertainty

with a coverage factor of 2,[10] implying the standard uncertainty is approximately 0.0075 GWd/MTIHM. This standard uncertainty estimate is significantly smaller than the values calculated with Equation (23) shown in Figure 9.

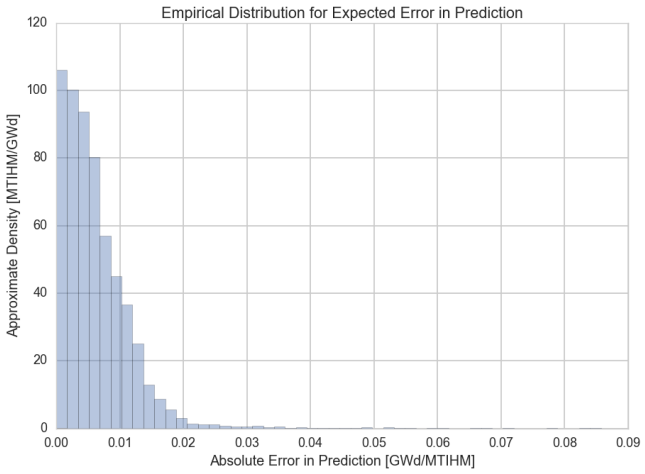


Fig. 10: Empirical distribution of absolute error in prediction.

C. Feature Selection Results: Important Nuclides

After training the extended RVM model to predict core-average burnup, only 10 of the 90 tracked nuclides were retained in the model (i.e., only 10 of the 90 entries of the η vector are nonzero once the algorithm converges and terminates). The final basis shaping factors, η_k, associated with these 10 selected nuclides are shown in Figure 11. The magnitude of each bar gives the relative importance of each nuclide for making burnup predictions. Surprisingly, the actinides and cesium isotopes traditionally used for burnup determinations in nuclear safeguards applications are not selected by the RVM. Currently, it is unclear why the selected nuclides produce a predictive signature and why the actinides and cesium are neglected. It is possible that additional nuclides input to the extended RVM for training may be informative, but the model seeks to find the sparsest multivariate signature while optimizing predictive fidelity. Therefore, if two nuclides give similar information (e.g., respond identically to neutron flux, neutron spectrum), one nuclide will be retained while the other is discarded.

VI. CONCLUSION

1. Summary

We have developed an extension to Tipping's relevance vector machine [7] that includes integrated feature selection and basis shaping via a second embedded optimization problem (see Figure 3). We have tested the technology on a two-dimensional synthetic noisy dataset, and demonstrated the extended RVM's ability to estimate a nonlinear function and distinguish predictive from uninformative variables to develop

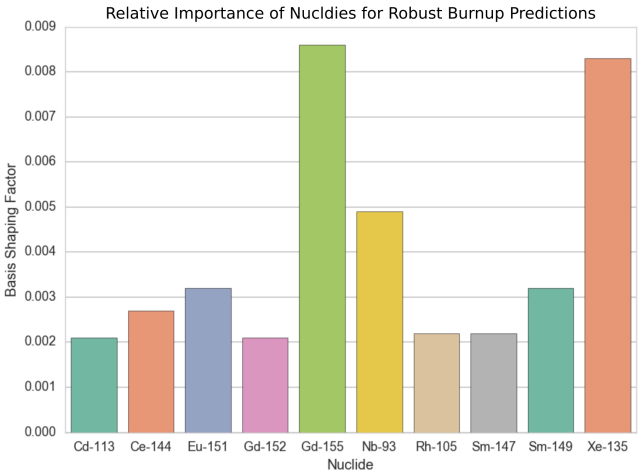


Fig. 11: Nonzero values of η_k (see Equation (17)) for a model trained with 650 training specimens to predict core-average burnup (see Figure 8). Only 10 of the 90 input nuclides were assigned nonzero values and retained in the model. Somewhat surprisingly, Cs, U, and Pu nuclides were not determined to be ideal for making position-independent predictions of core-average burnup. The magnitude of the entries shows the relative importance of the associated nuclide.

an optimal multivariate signature. The RVM's performance on the number of specimens used to train the model has been examined; an inverse relationship was shown between the prediction bias and prediction uncertainty as more training specimens are used.

Using a three-dimensional spatially-resolved depletion simulation for a gas-cooled, graphite-moderated reactor performed with SCALE 6.2, the extended RVM has been used to develop a model capable of making position-independent predictions of core-averaged burnup using a single specimen randomly drawn from the core. Predictions fell within 0.0062 GWd/MTIHM from the true value (which ranged from 0 to 1.05 GWd/MTIHM), with an average relative uncertainty of approximately 9.4%. It was observed that the models currently developed to date exhibit excess variance (calculated uncertainties far exceed observed bias), and consideration has been given to alternative methods to estimate prediction uncertainty (c.f., Section V.2.).

2. Further Work

To date, the primary focus of the research has been developing the extended RVM, synthetic reactor data, and associated code infrastructure. The application of the developed models to reactor characterization has been limited to predicting core-average burnup using ideal, noise-free data. In the future, application-specific research will be performed, which is outlined below.

[10] Using a coverage factor of 2.0 loosely corresponds to reporting 2σ uncertainties.

A. Bagging to Reduce Variance

As shown in Figures 9 and 10, the extended RVM as currently applied to reactor characterization is a relatively low-bias, high-variance method (i.e., the errors in prediction are significantly smaller than the calculated uncertainty). *Bagging* is a resampling method that is well suited for such models. The training data are repeatedly sampled to generate bootstrap training sets, and a model is trained on each bootstrap set. This ensemble of models is then applied to new testing samples, and the results are averaged together to produce a single estimated response. Assuming each model exhibits little bias (see Figure 8), the average responses from an ensemble of models will be at least as good as the results from individual models, and the variance will be reduced. If the bootstrap samples have identical variance σ and pairwise correlation $\rho \geq 0$, then the variance of the average result of an ensemble of B models will be

$$\mathbb{V}[\bar{f}(x_*)] = \rho\sigma^2 + \frac{1-\rho}{B}\sigma^2, \tag{29}$$

and the uncertainty in prediction will be reduced by a factor as large as $1/\sqrt{B}$ [13].

B. Noisy Data

In real-world applications, several types of noisy data are expected: random variation associated with measurement uncertainty (both independent and correlated), systematic biases, missing data, and gross error. Methods have been developed to simulate these types of noise in the ideal synthetic data analyzed to date. Generating replicate training data with different noise realizations will produce models that are robust to measurement noise and systematic biases. It is currently unclear how to apply models to test specimens with missing measurements. A common approach is to fill in missing values with mean values taken from the training data; however, this approach may limit the position invariance of the RVM predictions since specimens taken from the extreme positions of the core (e.g., the periphery) have isotopic compositions that differ significantly from dataset averages.

C. Simpler Signatures

Fission product signatures derived during model training by the integrated feature selection rely on 5–10 fission products, but it may not be feasible to make reliable measurements of each nuclide during real-world analyses. Therefore, ongoing research will attempt to identify the smallest subset of the identified nuclides that preserve model fidelity to determine which nuclides are the most analytically economical. The most useful nuclide to add to an existing experimental protocol depends on the nuclides already analyzed as there are significant conditional dependencies. To learn and best display this dependency, an analysis tree will be constructed that captures the best nuclide to add to any set of nuclides to best improve model fidelity. From this conditional tree, positive and negative synergistic effects between nuclides will be studied, and the optimal nuclides to analyze given experimental constraints will be identified.

D. New Predictive Models

To fully characterize a reactor core during/after operation, additional characteristics will be predicted. These include cooling time, initial enrichment of the fuel, and irradiation time (with burnup this specifies nominal power level). Research will investigate the most useful nuclides for making such predictions, as well as a method to design an analysis protocol to determine multiple variables (i.e., serial analysis with ideal models chosen by previous analyses or a single model trained on a multivariate response). Furthermore, as limited high quality fission product measurements is anticipated, reducing the number of nuclides required for analysis will be an emphasis of future research.

VII. ACKNOWLEDGMENTS

This work was funded by the Office of Defense Nuclear Nonproliferation Research and Development (NA-22), within the US Department of Energy's National Nuclear Security Administration.

REFERENCES

1. C. R. ORTON, C. G. FRAGA, R. N. CHRISTENSEN, and J. M. SCHWANTES, "Proof of concept simulations of the Multi-Isotope Process monitor: An online, nondestructive, near-real-time safeguards monitor for nuclear fuel reprocessing facilities," *Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, **629**, *1*, 209–219 (2011).
2. C. R. ORTON, C. G. FRAGA, R. N. CHRISTENSEN, and J. M. SCHWANTES, "Proof of concept experiments of the multi-isotope process monitor: An online, nondestructive, near real-time monitor for spent nuclear fuel reprocessing facilities," *Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, **672**, 38–45 (2012).
3. K. DAYMAN, J. B. COBLE, C. ORTON, and J. M. SCHWANTES, "Characterization of used nuclear fuel with multivariate analysis for process monitoring," *Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, **735**, 624–632 (2014).
4. C. CORTES and V. VAPNIK, "Support-vector networks," *Machine Learning*, **20**, *3*, 273–297 (1995).
5. H. DRUCKER, C. J. BURGES, L. KAUFMAN, A. SMOLA, and V. VAPNIK, "Support Vector Regression Machines," *Advances in neural information processing systems*, **9**, 155–161 (1997).
6. I. GUYON, S. GUNN, M. NIKRAVESH, and L. ZADEH, *Feature Extraction: Foundations and Applications* (2006).
7. M. E. TIPPING, "Sparse Bayesian learning and the relevance vector machine," *The Journal of Machine Learning Research*, **1**, 211–244 (2001).
8. M. E. TIPPING and A. C. FAUL, "Fast Marginal Likelihood Maximization for Sparse Bayesian Models," *Proceedings of the Ninth International Workshop on Artificial*

Intelligence and Statistics, pp. 1–5 (2003).
9. G. BAUDAT and F. ANOUAR, "Kernel-based methods and function approximation," in "IJCNN'01. International Joint Conference on Neural Networks. Proceedings," (2001), 2, pp. 1244–1249.
10. T. GÄRTNER, *Kernels for Structured Data*, World Scientific Publishing, New Jersey (2008).
11. A. C. FAUL and M. E. TIPPING, "Analysis of sparse Bayesian learning," *Advances in Neural Information Processing Systems*, **14**, 383–389 (2002).
12. M. E. TIPPING, "Bayesian inference: An introduction to principles and practice in machine learning," *Advanced lectures on machine Learning*, pp. 1–19 (2004).
13. T. HASTIE, R. TIBSHIRANI, and J. FRIEDMAN, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2009).
14. S. BOYD, *Convex optimization*, Cambridge University Press (2004).
15. L. CURRIE, "Limits for qualitative detection and quantitative determination. Application to radiochemistry," *Analytical Chemistry*, **40**, *3*, 586–593 (1968).
16. JOINT COMMITTEE FOR GUIDES IN METROLOGY (JCGM), "Guide to the expression of uncertainty in measurement," , *September*, 120 (2008).
17. M. D. DEHART and M. L. PRITCHARD, "Validation of SCALE and the TRITON Depletion Sequences for Gas Reactor Analysis," in "Transactions of the American Nuclear Society," (2008), pp. 683–686.
18. S. GOLUOGLU, M. DUNN, L. PETRIE, and T. SUMNER, "Development and Validations of the New Continuous Energy Capability in the Criticality Safety Code KENO," in "Proceedings of the International Conference on the Physics of Reactors (PHYSOR '08)," Interlaken (2008).