Interpretable Statistical Learning Methods

Zachary M. Jones*

Abstract

Statistical learning methods, which are a flexible class of methods capable of estimating the parameters of complex data generating functions, offer an attractive alternative to conventional methods, which often rely on strong assumptions about the assumed functional form. Statistical learning methods make fewer assumptions about functional form and can estimate a much larger class of data generating functions. A key impediment to the use of statistical learning methods is that they are usually thought of as "black box," in that they cannot be directly interpreted in general. I show that this need not be the case by explaining and implementing methods that are capable of making any method that generates predictions interpretable. This allows researchers to learn about relationships in the data without having to prespecify their functional form. I illustrate this approach using an application to predictive policing in Chicago, wherein these interpretation methods can be used to show how using historical crime data to predict future crime reinforces socio-demographic divides.

Introduction

The social world is often complex, messy, and difficult to describe with a simple set of rules or laws. In contrast, conventional methods *presume* data are generated by relatively simple processes (King 1989; Berk 2008). This hampers our ability to discover relationships in our data that we did not expect and reduces the predictive validity of our models (Miller and Page 2009; Beck, King, and Zeng 2000; Beck and Jackman 1998; Schrodt 2014; Fariss and Jones 2017). With conventional methods the data cannot tell us directly about relationships which were not prespecified in the model. Furthermore, the manner in which the data can inform our understanding of relationships which *were* prespecified in the model is constrained by the often strong assumptions made about their functional form, i.e, it is difficult to learn about nonlinear and interactive relationships from a linear additive model. One of the benefits of making strong functional form assumptions is that models can be made to be directly interpretable, by, e.g., assuming linearity, simple forms of nonlinearity (e.g., polynomial terms), and/or the absence of interactions involving more than, say, two variables.[1]. This comes at a substantial price, however. Conventional methods can miss interesting structure in data when that structure cannot be captured by the assumed model, which means that this information is not extracted from the data, damaging our ability to improve our understanding and degrading the quality of our predictions.

*Email: zmj@zmjones.com.

[1]Interpretability can be thought of in a number of ways (See, e.g., . . .), but here I use to to mean models which have additive components which can be visualized, e.g., for continuous inputs, functions of two or fewer covariates.

Advances in computing have allowed the development of statistical learning methods which allow us to make fewer assumptions about the structure in our data and consequentially to discover unexpected relationships in our data (Friedman, Hastie, and Tibshirani 2001). Statistical learning methods differ from conventional statistical methods in that the number of parameters estimated is a function of the data, rather than specified by the researcher. This gives statistical learning methods the ability to find relationships in data that were not prespecified. This ability to find the unexpected, however, frequently makes the outputs from these methods uninterpretable, which, I argue, has prevented their wide use in the social sciences, where the focus is on substantive insights in addition to prediction (Fariss and Jones 2015; Ward, Greenhill, and Bakke 2010; Daniel W Hill and Jones 2014). This is not generally a problem with conventional statistical methods because we *assume* that the function that generated the data has a relatively simple functional form which is easily interpretable.

I describe and implement general tools which can make any statistical learning method interpretable in ways that allow social scientists to answer questions about the importance of covariates, the nature of their estimated relationship with an outcome, the presence of interactions amongst said covariates (and their shape), and also explanations for predictions for specific observations or instances (Friedman 2001; Friedman and Popescu 2008; Hooker 2012; Hooker 2004; Goldstein et al. 2015). I have made these methods available in an accessible manner via four `R` packages: the **M**achine **L**earning in **R** (`mlr`) package, which provides infrastructure for fitting, tuning, and evaluating a wide variety statistical learning methods, the **E**xploratory **D**ata **A**nalysis using **R**andom **F**orests (`edarf`) package, which contains interpretation methods specific to random forests, a particularly attractive statistical learning method for social scientists, **M**onte-**C**arlo **M**ethods for **P**rediction **F**unctions (`mmpf`) which allows prediction functions to be marginalized to depend on a subset of the covariates, and is itself is used in both of the aforementioned packages, and finally **F**unctional **A**nalysis of **V**ariance (`fanova`) which projects an estimated regression model onto a set of lower dimensional functions which are interpretable (Jones and Linder 2016; Bischl et al. 2016; Jones 2017; Hooker 2012).

The reason that statistical learning methods do not generally give interpretable outputs is that the manner in which they estimate a model, which is necessitated by their ability to learn complex relationships which were not presupposed, results in an output that, even if the true model *is* human interpretable, would hide that fact: a "black box". Throughout this paper I will use a simple simulated example

$$f(w, x, y, z) = x + y^2 + xw + \sin(z) \tag{1}$$
$$y = f + \epsilon$$

where w, y, and z are uniformly distributed between -2 and 2, x is equal to $-1, 0, 1$ with equal probability, and ϵ is drawn from a standard normal distribution.

The way a regression tree estimates a model (i.e., how it arrives at predictions), is by subsetting (i.e., partitioning or grouping) y using the covariates repeatedly so that the variability of y within these subsets is minimized. This amounts to finding values of y which are similar using the covariates, similar to what ordinary least squares aims to do, but with substantially weaker assumptions about how the covariates are related to y. The result of this process is a nested set of subsets, a hierarchical structure (a tree), which describes the

series of subsets that were created: rules which describe the final subset an observation ends up in. As can be seen in Figure 1 this output is substantially more difficult to interpret than ordinary least squares. This problem is exacerbated when the estimated tree is larger (more complex), or when it is combined with other trees as in ensemble methods like boosted trees or random forests, which are generally preferable for reasons described in later sections. While this data generating function *could* be estimated with conventional methods, it would require the specification of a dummy variable for each level of x interacted with w, a third degree polynomial expansion of z to approximate the sine function between -2 and 2, and a squared term for y.[2]:

Statistical learning methods can estimate all of these terms without prespecification, and the interpretation methods I will describe herein can extract these components.

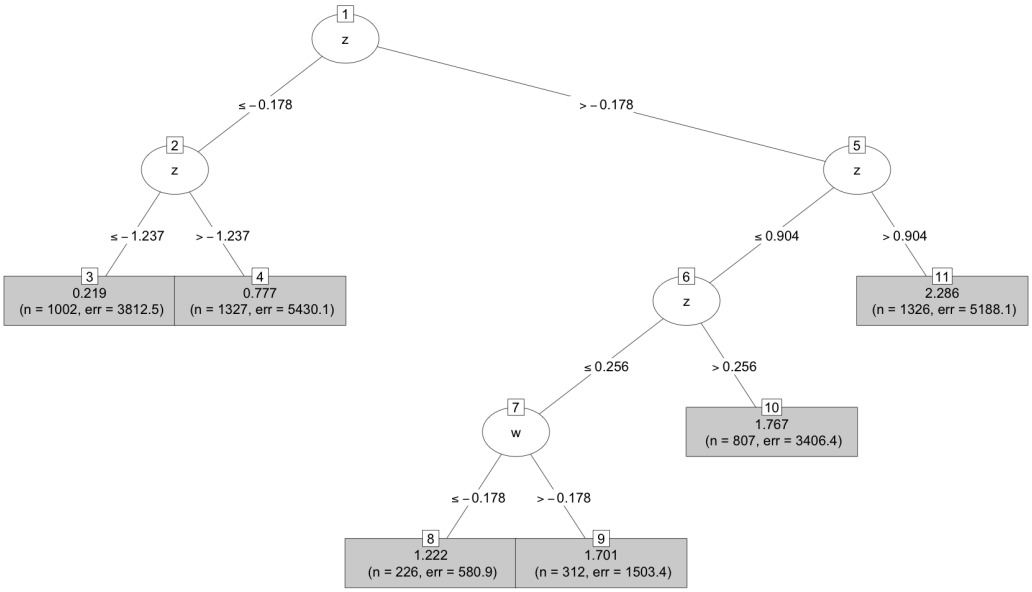

Figure 1: A model estimated via a regression tree from 5000 samples from the data generating function in Equation 1. Each circle indicates a subset or partition defined on the labeled covariate, with the rule for creating the subset listed on the line below the circle. The final subsets, or terminal nodes, shown in grey, give them predictions of the tree, the number of observations that fall into that subset, and the mean square error of the prediction on those observations. This tree was constrained to be simple for this visualization.

While there have been some applications of statistical learning in the social sciences, their use has been largely limited to instances in which forecasting is the primary goal, with a few notable exceptions (Beck, King, and Zeng 2000; Hainmueller and Hazlett 2013; Daniel W Hill and Jones 2014). While the feed-forward artificial neural networks and kernel regularized least squares used in Beck, King, and Zeng (2000) and Hainmueller and Hazlett (2013) are

[2]Specifically, a conventional linear model could estimate this model with the following terms $\mathbb{I}(x = -1)$, $\mathbb{I}(x = 0)$, $\mathbb{I}(x == 1)$, $\mathbb{I}(x = -1)w$, $\mathbb{I}(x = 0)w$, $\mathbb{I}(x == 1)w$, y^2, z, z^2, z^3.

powerful methods which could be appropriately applied to many problems they are not appropriate for *every* application, and thus it is desirable to be able to use, and interpret the results of the best statistical learning method available for your problem. The methods described herein do precisely that. Before proceeding to the details of these methods for making statistical learning methods interpretable, I will review why statistical learning methods can outperform conventional methods in common social science tasks and how they can be viewed in the same statistical framework as conventional methods.

Statistical Learning

Statistical learning methods like artificial neural networks, decision trees, random forests, boosting, and others have seen use in the social sciences, but are far from common (Beck, King, and Zeng 2000; Grimmer and Stewart 2013; Daniel W Hill and Jones 2014; Montgomery et al. 2015; Green and Kern 2012; Montgomery and Olivella 2015; Hainmueller and Hazlett 2013). Statistical learning methods can be represented in the same framework as more conventional statistical methods. Here I review these equivalences and highlight both the difficulties and advantages of using statistical learning methods as compared to conventional methods.

Assume that the $(\mathbf{X}, Y)$ are random variables drawn from some probability distribution $\mathbb{P}$ over the sample space $\mathcal{X} \times \mathcal{Y}$, which represents all combinations of X and Y that could be obtained, wherein Y represents an outcome and $\mathbf{X}$ covariates.

We typically estimate the function which maps $\mathbf{X}$ to Y by using sample data presumed drawn from $\mathbb{P}$. Often we are interested in the conditional expectation of $Y|\mathbf{X}$, and we assume that this function depends on some parameters θ.

$$\mathbb{E}\left[Y|\mathbf{X} = \mathbf{x}\right] = f(\mathbf{x}, \theta)$$

Samples of Y are denoted $\mathbf{y}$ and of $\mathbf{X}$, $\mathbf{x}$. To estimate f we need to define a set of functions which can be searched for the best approximation to f, e.g., for a linear regression model the space of all possible values of the βs. We also need a way to decide which values of these parameters are "best." Finally, we need a way to efficiently search through the set of possible models.

Defining the Space of Possible Models

A *hypothesis space*, denoted $\mathcal{H}$, is a set which defines all possible functions that could be used to approximate the true data generating function f. In the case of a linear regression the hypothesis space is the set of functions which are linear and additive in the parameters θ. Specifically, for a two parameter simple linear regression of the form $y_i = \theta_0 + \theta_1 x_i + \epsilon$, the hypothesis space is $\mathbb{R}^2$, because θ_0 and θ_1 could each be any number from negative to positive infinity. We can learn a nonlinear relationship between $\mathbf{x}$ and $\mathbf{y}$ if we expand the basis (i.e., the set of functions which describe $\mathbf{x}$) for $\mathbf{x}$ which we can denote $\phi(\mathbf{x})$, by, for example, including polynomials of $\mathbf{x}$, but this must be specified in advance rather than being automatically learned from the data. Here the dimension of θ is fixed.

$$h(\mathbf{x}, \theta) = \theta^T \mathbf{x} \tag{2}$$

Here $h \in \mathcal{H}$, that is, [eq:lm] is one of the possible models contained in the hypothesis space $\mathcal{H}$. Estimating, or learning, a model from data is searching over the hypothesis space to find the "best" candidate model.

In contrast to linear regression (and conventional methods more broadly), boosting, decision trees, random forests, and some types of neural networks, utilize a data-adaptive number of parameters θ, making the dimension of the hypothesis space larger. For example a regression tree, as in Figure 1 can be written as

$$h(\mathbf{x}) = \sum_{m=1}^{M} w_m \mathbb{I}(\mathbf{x} \in R_m) \tag{3}$$

where $\mathbb{I}(\cdot)$ is the indicator function which equals 1 if $\mathbf{x} \in R_m$ and 0 if not, R_m is the m'th of the final, disjoint, and exhaustive partitions of the data, and w_m is the mean value of $\mathbf{y}$ in this region. Each region R_m is defined by which variables, and which values in those variables, were used to create the final set of partitions in the tree: so their number is data-dependent, generally growing with the data. Again, this increases the size of $\mathcal{H}$.

Evaluating Candidate Models

How can we evaluate different candidate models in the hypothesis space? A loss function $\mathcal{L}$ tells us how good a particular h is at predicting $\mathbf{y}$. Specifically, the loss $\mathcal{L}$ is a function which maps predictions made by an approximation to f, a member of $\mathcal{H}$, which we've called h but is conventionally referred to as $\hat{f}$, and observed realizations of Y, $\mathbf{y}$, to a positive real number. Herein lies a problem though, as we are trying to approximate f, not perfectly reconstruct $\mathbf{y}$. Y is assumed herein to be random, and so contains a random and a systematic component, i.e., $\mathbf{y} = f(\mathbf{x}) + \epsilon$. If we could compute the discrepancy between $\hat{f}$ and f rather than $\hat{f}$ and $\mathbf{y}$ we would be able to distinguish between candidate hypotheses h which have overadapated to the sample data and those that have more accurately estimated f and hence will generalize better to new data. However, since we do not know f we have to substitute in $\mathbf{y}$, which conflates systematic error, when the function learned from $(\mathbf{y}, \mathbf{x})$ imperfectly represents f, and error due to the irreducible randomness in Y. Thus minimizing the loss over the observed data may result in models $\hat{f}$ which are close approximations of $\mathbf{y}$ rather than f: over-fitting. This is the opposite of the problem of picking a hypothesis space whose closest member is far from f: under-fitting. Both, however, leave us with models which can be rather useless in both learning about the parameters of f, that is, explanations about how covariates are related to some outcome, as well as predicting Y, which is sometimes the goal in and of itself, and other times is simply a minimal check on the validity of the model.

Like in the case of under-fitting, there are ways to avoid over-fitting such as minimizing the expected loss over $\mathbb{P}$. The expected loss is usually called the risk, and minimizing it can eliminate optimisim that comes from over-fitting the sample data. However, since $\mathbb{P}$ is unknown this is not directly possible. The sample-average loss is an estimator of the risk, however, it is biased downwards, that is, it is optimistic, when applied to the same data on

which the model was estimated. The most common solution to this issue is to estimate the risk by using Monte-Carlo methods like cross-validation or the bootstrap which randomly shuffle the data to make it more difficult for over-fit models to appear better than they are at generalizing beyond the sample data.[3] Penalized regression models like the least absolute shrinkage and selection operator (LASSO) or ridge regression make use of this idea to shrink the regression coefficients of a conventional model (sometimes to 0), making it simpler, to avoid overfitting.[4] Statistical learning methods use this idea to make the model as simple or complex as is necessary to minimize risk.

Choosing a loss function is an application specific task that depends on the costs of making different sorts of prediction errors. With a binary outcome, it might be the case that positive cases are more costly when they are not predicted than are negative cases that are not predicted, or, with a continuous outcome, overprediction might be more costly than underprediction (Berk et al. 2009; Muchlinski et al. 2016). Similarly, how the difference between the observed outcome and the prediction affects the loss also depends on how costly bigger or smaller mistakes are. For example mispredicting cases where persons convicted of violent crimes reoffend after parole is probably more costly than predicting recitivism amongst those convicted of non-violent crimes, and it is possible that in some cases over-predicting recitivism is less consequential than under-predicting it. While squared-loss or misclassification-loss is often used because it makes the third step, the search or optimization strategy, easier, it is not necessarily the most appropriate choice, and what loss function is used has an enormous impact on what function from the hypothesis space is chosen as the best approximation to f since it is *the* criteria for that choice.

Searching for the Best Model

In most cases, even for a simple linear regression, the hypothesis space is too big to evaluate every possible model it contains. So a way to search through the hypothesis space must be devised. In some cases this can be done with relative ease, but for many hypothesis spaces even the best available optimization methods cannot guarantee that they will find the approximation to the true function topology of loss function over the hypothesis space. However, if the best candidate in the hypothesis space for a linear, additive model is far from the truth: f, this guarantee means little, since it may be the case that a candidate taken from a larger hypothesis space, even if it is not known to be the best in said space, is much closer to f than the aforementioned optimal choice in the linear, additive hypothesis space. This is why statistical learning methods are so commonly used for pure prediction tasks.

For illustrative purposes consider a linear additive model as in eq. 2 with a squared loss function. Then the function we want to minimize is the sum of squared errors. In this simple case where we have a convex loss function like squared loss gradient descent suffices to find the parameters which minimize the loss function.

$$h(\theta, \mathbf{x}) = \sum_j \theta_j x_j$$

[3] It is also sometimes possible to derive an upper bound on the risk given the size of $\mathcal{H}$, the amount of data available, and the average loss on the sample data (Vapnik, Levin, and Le Cun 1994; Vapnik 1998).

[4] Another way of saying this is that regularization or shrinkage makes the model less sensitive to the sample data or that it makes the model smoother.

$$\mathcal{L}(\mathbf{y}, h) = \sum_{i=1}^{N} \frac{1}{2}(y_i - h)^2$$

$$\nabla\mathcal{L} = \frac{\partial\mathcal{L}}{\partial\theta} = \frac{1}{N}\sum_{i=1}^{N} -x_i(y_i - h)$$

Descending the gradient of the loss $\nabla\mathcal{L}$ given a starting value for θ (say $\mathbf{0}$), converges to the global minimum. From the starting value θ^0 the sequential update for the next step is

$$\theta^{(k+1)} = \theta^{(k)} - \gamma_k \nabla\mathcal{L}(\theta^{(k)})$$

where γ controls the size of the steps along the gradient, which can itself be estimated by gradient descent. The procedure stops when $\nabla\mathcal{L}$ is sufficiently small, that is, when the loss function is not decreasing any longer.

In contrast consider boosting, a statistical learning method. Like with the aforementioned linear additive model we want to minimize the loss $\mathcal{L}(\mathbf{y}, h) = \frac{1}{2}\sum_{i=1}^{N}(y_i - h)^2$, however now each candidate h has more parameters. In this case it is not possible to h will be estimated sequentially, that is, it is composed of a sequence of models which are combined in the last step. We denote the first m components of h as h_m.

$$h_m(\gamma, \theta, \mathbf{x}) = \sum_{m=1}^{M} \gamma_m \phi_m(\mathbf{x}, \theta_m) \tag{4}$$

Here ϕ_m is a model itself, such as a regression tree like is shown in Figure 1. For each m the gradient of $\mathcal{L}$ with respect to h_m is

$$\nabla\mathcal{L}(h_m) = \frac{\partial\mathcal{L}}{\partial h_m} = \sum_{i=1}^{N}(y_i - h_m)$$

and so, like in the case of gradient descent for linear regression

$$h_{m+1} = h_m - \gamma_m \nabla\mathcal{L}(h_m)$$

.

The scaling parameters γ are estimated by a separate application of gradient descent.[5]

So at each step m boosting using this loss function amounts to fitting $\phi_m(\mathbf{x}, \theta_m)$ to $\mathbf{y} - h_{m-1}$: repeatedly refitting the residuals. This amounts to giving higher weight to observations where the sequence of models thus far has performed poorly, and by combining these models as in eq. 4 nonlinear relationships between $\mathbf{x}$ and $\mathbf{y}$ can be estimated without specifying ϕs which can estimate nonlinearity. It should also be clear that the resulting output h_M, which is a weighted sum of models ϕ_m, which are themselves estimated on residuals which depend on h_{m-1} is not directly interpretable, even if the data generating function it is approximating is.

[5]Specifically, $\gamma_m = \arg\min_\gamma \mathcal{L}(\mathbf{y}, h_{m-1} + \gamma h_m)$.

Again, this property is a necessary consequence of statistical learning methods' capacity for automatically finding patterns (e.g., nonlinearity and/or interaction amongst the covariates) that were not prespecified.

Predicting Burglary from Crime Data in Chicago

police departments have limited resources and so want to expend them in an optimal fashion

there are a number of companies which make predictions about the incidence of crime

conventional methods are ill-suited to this prediction task

statistical learning methods cannot be "inspected"

there are good reasons to want to inspect models in this instance

since there are sociodemographic biases in the data it is possible that the any model learned from these data can effectively learn about these biases, even if sociodemographic covariates are not included

local inspection might be desirable: why does the forecasting model predict crime incidence or absence in particular times and places

global inspection might be desirable: how does crime vary seasonally (month to month, over the course of a day), geographically, or dynamically (i.e., is burglary "streaky?")

random forests are an easy to use statistical learning method because they can be scaled to "big" data (how big are these data), they can accomodate covariates of all different types, and they don't have very many tuning parameters.

some discussion of forecast errors, and/or a comparison?

interpretation of the output of the model will accompany explanations of the different methods for doing so.

Interpreting Models

I argue that there are four main types of interpretation which are commonly sought. Functional form describes the shape of the relationship between covariates and the model. Magnitude describes how important covariates are in determining the model's predictions. Interaction detection relates to the estimation of the functional form and magnitude of the relationship between groups of covariates and the model. Reliability relates to how different sources of uncertainty could possibly change the estimated effect.

Functional Form

Interpreting the functional form of the relationship between the covariates and the model is one of the most common tasks. For a linear and additive model this is described by a single number which is obtained by taking the partial derivative of the model with respect to said covariate; e.g., $\frac{\partial \hat{f}}{\partial \mathbf{x_j}} \sum_j \beta_j \mathbf{x}_j = \beta_j$. The situation is complicated for models like logistic

regression where the formulation of the model forces this derivative to depend on $\mathbf{x}_{-j}$, but the principle remains the same. For models which are nonlinear in $\mathbf{x}_j$ this derivative is also not constant, but does not necessarily depend on $\mathbf{x}_{-j}$. For example if $\hat{f} = \beta_1 \mathbf{x}_1 + \beta_2 \mathbf{x}_1^2 + \beta_3 \mathbf{x}_2$ then $\frac{\partial \hat{f}}{\partial \mathbf{x}_1} = \beta_1 + \beta_2 \mathbf{x}_1$ and to interpret the relationship between $\mathbf{x}_1$ and $\hat{f}$ it is necessary to plot $\mathbf{x}_1$ and this derivative at a grid of values on $\mathbf{x}_1$.

Magnitude

Magnitude describes how important covariates are in determining a models' predictions. In the case of a linear model this is simply the absolute size of the associated regression coefficients. However, for a nonlinear model this isn't so simple. If the derivative of the model with respect to a covariate is not constant in that model then there is no longer a single number summary of that covariate's influence on the model's predictions. Yet, this might be what is desired, as in Daniel W. Hill and Jones (2014) where the goal was to sift through a large set of potentially relevant covariates in order to focus on ones that were identified as useful predictors by the model.

Interaction Detection

The identification and description of interactions is a common task as well (Brambor, Clark, and Golder 2006). In the case of a linear model, again, this is fairly straightforward, however, the product terms commonly used to estimate interactions in these models are commonly misused in testing nonlinear hypotheses (Hainmueller, Mummolo, and Xu 2016; Pepinsky 2017). Using statistical learning methods avoids this problem by automatically detecting interactions whilst making few assumptions about their possible functional form. This allows researchers to ask these common questions, like, whether or not the effect of one variable changes as a function of another in a general (i.e., not necessarily monotonically) fashion.

Reliability

Most of the above, when computed using conventional methods, are or can be accompanied by estimates of sampling variability. In some cases this can be a useful measure of reliability but in many cases the assumptions necessary for these reliability estimates to be valid are grossly violated, most commonly, independence. This is particularly problematic in social data, where dependence is both complex and ubiquitous, so much so that it is often of primary interest, as is in analyses of social networks. Another difficulty is other sources of uncertainty which are more difficult to measure and adjust for. It may also be the case that uncertainty from non-random measurement error is substantially more important than random measurement error (i.e., sampling error) in many cases, and instances where non-random measurement error is identified or where multiple measures of the same (or similar) concepts produce substantially different conclusions constitute anecdotal evidence that this is the case (E.g., Lupu 2017; Baum and Zhukov 2015; Jerven 2013). All of this suggests that many measures of uncertainty should be viewed with skepticism and so I have not focused on measuring reliability herein. However, all of the software referenced herein is capable of using estimates of the variability of a models' predictions to produce

reliability bounds for the aforementioned types of interpretation tasks which correspond to the quantities that typically accompany the equivalent outputs from conventional methods.

Approximating and Interpreting Functional Form

When the most complex functional form capable of learned from the data is simple, as in a linear additive model, the parameters of said model are often directly interpretable. Some statistical learning methods have similar properties (See, e.g., Hainmueller and Hazlett 2013; Beck, King, and Zeng 2000). However, many such methods cannot be directly interpreted but may be otherwise desirable for a given application: due to, e.g., their ability to deal with large N or many covariates, their predictive performance in this or similar applications, etc.

One of the most common types of interpretation for conventional methods is functional form; i.e., for a linear model what is the sign and magnitude of the coefficient for a covariate, defining the slope of the line representing its relationship with the models predictions. For conventional models where one or more of the covariates have a nonlinear relationship in the model, e.g., a logistic regression, the effect of a covariate on the models' predictions depends on the other covariates. To understand the effect of a covariate then, it is necessary to compute predictions at particular values of the other covariates. An average marginal effect gives the effect of a particular covariate computed at, rather than fixed values for the other covariates, every observed combination of the other covariates. This is a form of Monte-Carlo integration which also works with statistical learning methods, where it is referred to as partial dependence. It allows the recovery of the parital relationship between particular covariates and $\hat{f}$ regardless of how $\hat{f}$ is estimated (Friedman 2001).

Partial Dependence

If we knew the joint distribution $\mathbb{P}$ of $\mathbf{X}$ we could visualize the dependence between some some subset of $\mathbf{X}$, say $\mathbf{X}_u$, by integrating out the compliment $\mathbf{x}_{-u}$ and computing, e.g., the conditional expectation of $Y|X_u$. However, since we do not know $\mathbb{P}$ and have instead estimated this expectation we can instead use Monte-Carlo integration using the sample data. We would compute

$$\mathbb{E}(\hat{f}(\mathbf{x})|\mathbf{x}_u) = \int_{x_{-u}} \hat{f}(\mathbf{x}_u, \mathbf{x}_{-u})\mathbb{P}(\mathbf{x}_u|\mathbf{x}_{-u})\mathbb{P}(\mathbf{x}_{-u})d\mathbf{x}_{-u}. \tag{5}$$

However the conditional probability of $X_u|X_{-u}$ introduces into the desired description of the relationship between $\mathbf{x}_u$ and $\hat{f}$ dependencies between $\mathbf{x}_u$ and $\mathbf{x}_{-u}$. If this term is dropped, we get the *partial dependence* of $\hat{f}(\mathbf{x})$ on $\mathbf{x}_u$, which can be estimated from the sample data (Friedman 2001).[6]:

$$\mathbb{E}_{X_{-u}}[f(X)] = \int_{X_{-u}} f(X)d\mathbb{P}(X_{-u}) \approx \frac{1}{N}\sum_{i=1}^{N} \hat{f}(\mathbf{x}_u, \mathbf{x}_{-u}^{(i)}) = \bar{f}_{\mathbf{x}_u}(\mathbf{x}_u) \tag{6}$$

[6]The effect of the inclusion of this conditional probability on the estimate of the effect of how $\hat{f}$ depends on $\mathbf{X}_u$ can be substantial even when there is no interaction between $\mathbf{X}_u$ and $\mathbf{X}_{-u}$ (Jones 2017).

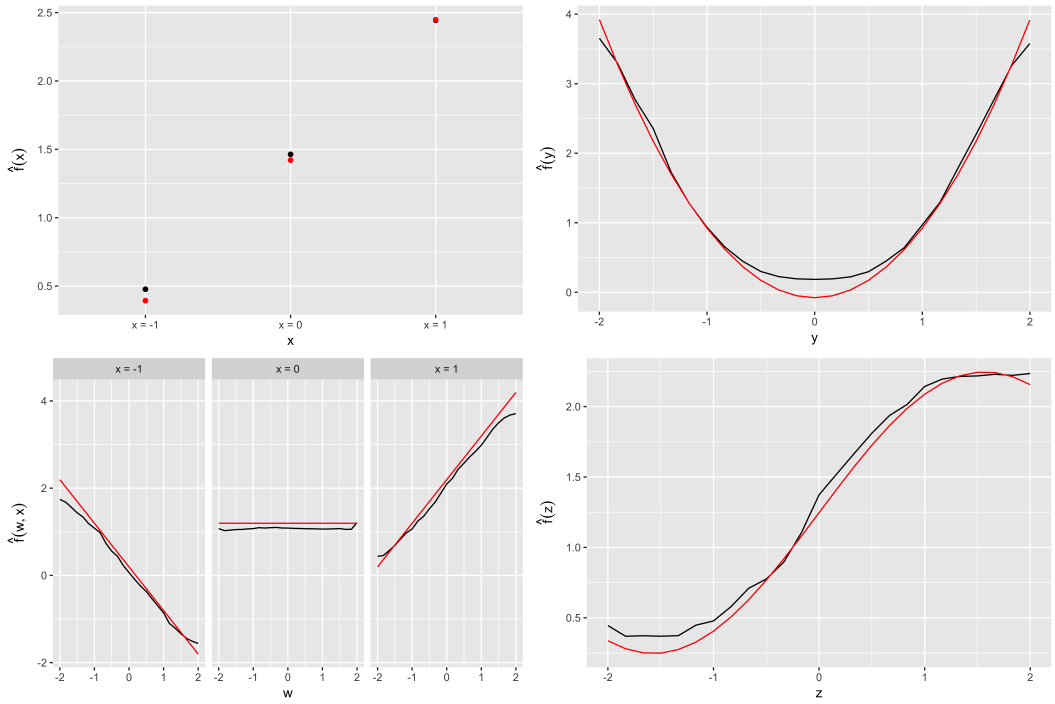

Figure 2: This figure shows the partial dependence: the marginalized prediction function, for each component of the model. The black lines, or dots, show the partial dependence estimated from a random forest estimate of the data generating function of Equation 1, while the red line shows the 'truth', the partial dependence of the indicated covariates on the data generating function without any noise. In the top left panel is shown the partial dependence of the discrete covariate **x**, and in the bottom left its interaction with the continuous covariate **w**. In the upper and lower right panels the partial dependence of the covariates which have nonlinear effects, **y** and **w** are shown. While a conventional method could estimate Equation 1, the random forest estimated these relationships without prespecification (albeit in a black-box manner), and with partial dependence we are able to recover them.

This gives the relationship between the prediction made by $\hat{f}(\mathbf{x})$ and $\mathbf{x}_u$ averaged across the observed values of $\mathbf{x}_{-u}$. A suitable grid of points on $\mathbf{x}_u$ must be chosen. This can be done by taking a Monte-Carlo sample from the empirical distribution of $\mathbf{x}_u$ or by selecting values of interest.

How closely $\bar{f}_{\mathbf{x}_u}(\mathbf{x}_u)$ corresponds to $\hat{f}(\mathbf{x}_u)$ depends on the structure of $\hat{f}$. In the case $\hat{f}(\mathbf{x})$ is additive or multiplicative in $\{\mathbf{x}_u, \mathbf{x}_{-u}\}$, that is, $\hat{f}(\mathbf{x}) = \bar{f}_{\mathbf{x}_u}(\mathbf{x}_u) + \bar{f}_{\mathbf{x}_{-u}}(\mathbf{x}_{-u})$ or $\hat{f}(\mathbf{x}) = \bar{f}_{\mathbf{x}_u}(\mathbf{x}_u)\bar{f}_{\mathbf{x}_{-u}}(\mathbf{x}_{-u})$ computing the partial dependence gives an approximation to $\hat{f}(\mathbf{x})$ that is accurate to an additive or multiplicative constant, respectively (Friedman 2001; Jones 2017). Discovering when such (additive or multiplicative) factorizable structure is *not* present is a question about interaction, that is, non-separable dependence between $\mathbf{x}_u$ and $\mathbf{x}_{-u}$ in $\hat{f}$.

One way to detecting interaction amongst the covariates using partial dependence is to estimate the variability of eq. 6. This can be done again using Monte-Carlo methods by substituting the sample variance for the mean. Another possibility is to avoid Monte Carlo integration over the distribution of $\mathbf{x}_{-u}$ and to instead visualize the expected value of the outcome at $\mathbf{x}_u$ for each observation conditional on $\mathbf{x}_{-u}$. Goldstein et al. (2015) propose using this approach.

$$\mathbb{E}_{\mathbf{x}_{-u}^{(i)}}[\hat{f}(\mathbf{x})] \approx \hat{f}(\mathbf{x}_u, \mathbf{x}_{-u}^{(i)}) = \bar{f}_{\mathbf{x}_u}^{(i)}(\mathbf{x}_u) \tag{7}$$

This is the partial dependence of $\mathbf{x}_u$ on $\hat{f}(\mathbf{x}^{(i)})$, which has some advantages over eq. 6. When there is interaction between $\mathbf{x}_u$ and $\mathbf{x}_{-u}$ in $\hat{f}(\mathbf{x})$ eq. 7 will show variation in the estimated individual conditional expectation since the relationship between $\mathbf{x}_u$ and $\hat{f}(\mathbf{x})$ depends on $\mathbf{x}_{-u}$. Hence, regions of interaction can be discovered by observing regions of high variance in eq. 7. Although this can be visualized directly by directly estimating the Monte Carlo variance rather than the mean, visualizing the individual conditional expectation allows for simultaneous visualization of regions of interaction and the the functional form of the relationship between $\hat{f}(\mathbf{x})$ and $\mathbf{x}_u$. This comes at the cost of less interpretability. For each curve in eq. 6, eq. 7 produces N curves. One approach making these visualizations more interpretable is to center the individual conditional expectaton curves, by, e.g., subtracting the individual conditional expectation of the smallest $\mathbf{x}_u$. Then variation in the curves relative to this baseline may be observed.

It is often the case that rather than the expected value, an "effect," a change in the expected value of the outcome for a change in the covariates, is of direct interest. The partial derivative of each curve in eq. 7 or eq. 6, the partial derivative of $\hat{f}(\mathbf{x})$ with respect to $\mathbf{x}_u$ can also be estimated using numerical methods. This serves as another way to discover interactions between $\mathbf{x}_u$ and $\mathbf{x}_{-u}$. If there is interaction then the partial derivative will depend on both $\mathbf{x}_u$ and $\mathbf{x}_{-u}$, while if there is no interaction the derivative will not. Hence variation in the derivative of the individual conditional expectation indicates interaction.

All of the above can be quickly and easily estimated for any prediction function $\hat{f}$, regardless of how it was estimated (i.e., both conventional and statistical learning methods) from the sample data using **M**onte **C**arlo **M**arginalization of **P**rediction **F**unctions (`mmpf`) which is a publically available `R` package. The **M**achine **L**earning in **R** (`mlr`) package also makes all of this functionality available, and, additionally, makes available a wide variety of statistical learning methods available for use in a consistent manner, with a wide variety of tools for evaluating and tuning their performance (Bischl et al. 2016; Jones 2017). Lastly, **E**xploratory

Data **A**nalysis using **R**andom **F**orests (`edarf`) again provides similar functionality to what is described above along with additional interpretation methods specific to random forests, which are beyond the scope of this paper (Jones and Linder 2016).

Variable Importance

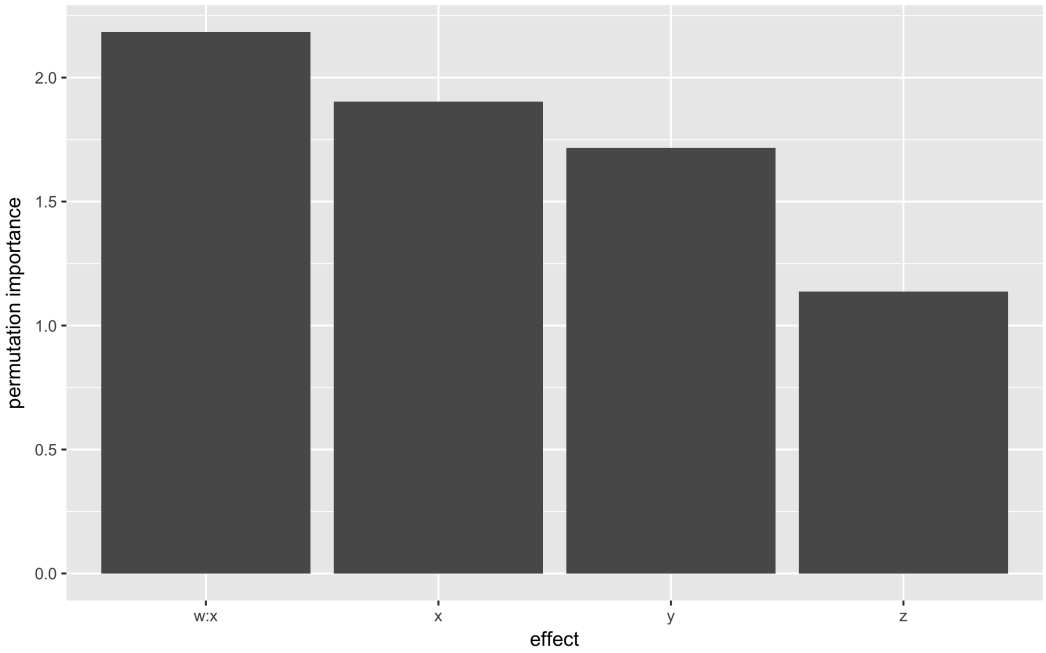

Figure 3: Permutation importance estimated from the data generating function in Equation 1. This indicates that the most important component of the model is the interaction of $\mathbf{x}$ and $\mathbf{w}$. Since $\mathbf{x}$ and $\mathbf{w}$ interact in Equation 1 and the random forest estimated this interaction, as can be seen in Figure 2, permuting either or both covariates will increase prediction error substantially, as they introduce error from the individual components as well as their interaction. In this way permutation importance gives weight to covariates like $\mathbf{w}$ which may only have a conditional effect. Components with less variability, like $\sin(\mathbf{z})$ (which has a sample variance of 0.592) have less importance than those with more variability (e.g., $\mathbf{y^2}$, which has a sample variance of 1.418).

The question of which and how important covariates are is a question that arises with predictive and exploratory tasks. In case of the former it is often advantageous to remove covariates which are not useful in predicting the outcome and in the latter case computing the importance of covariates is useful in finding strong predictive relationships for further study. For example these methods were used by Daniel W. Hill and Jones (2014) to rank covariates according to their importance to accurately predicting different measures of human rights abuses. As previously mentioned in the case where the derivative of the model with respect to a covariate is constant the importance of that covariate in that model is summarized by a single number whose magnitude is often of primary interest. With models where this

derivative is not constant there is no straightforward way to summarize the importance of a covariate with a single number; fortunately, there are methods which have generalized the notion of importance to models of an arbitrary form. Variable importance methods define importance as a covariate's contribution to decreasing prediction (in particular generalization) error.

Permutation importance is a general method for estimating variable importance which is agnostic to the learning method used. It is the Monte Carlo expectation of the change in prediction error that occurs when $\mathbf{X}_u$ is permuted. If $\mathbf{X}_u$ was not useful to $\hat{f}(\mathbf{x})$ in making predictions then we would expect no change or a decrease in prediction error, whereas if $\mathbf{X}_u$ is important there should be an increase in prediction error when it is permuted. Due to Monte-Carlo error from the permutation, this is repeated M times to give the expectation of the change in the loss $\mathcal{L}$ from permuting $\mathbf{X}_u$.

$$I_{\mathbf{X}_u} = \frac{1}{M} \sum_{j=1}^{M} \mathcal{L}\left(\hat{f}(\mathbf{X}_{u\pi^{(j)}}, \mathbf{X}_{-u}), \hat{f}(\mathbf{X})\right)$$

Take for example a linear model of the form $\hat{f}(\mathbf{X}) = \beta_u \mathbf{X}_u + \beta_{-u}\mathbf{X}_{-u}$. Say that β_{-u} was 0 for each column of $\mathbf{X}_{-u}$. Then if we randomly permuted $\mathbf{X}_{-u}$ M times, and, for each one of those times we computed $\hat{f}$ using this randomly shuffled version of $\mathbf{X}_{-u}$, we would find that the prediction error did not change at all, because $\mathbf{X}_{-u}$ does not contribute to the prediction made by $\hat{f}(\mathbf{X})$. Conversely, if we permuted $\mathbf{X}_u$, and β_u was non-zero, perhaps relatively large, we would be multiplying a randomized matrix of covariates by large numbers β_u, and, unsurprisingly, this will make $\hat{f}$ also basically random, and consequently its predictions awful, which would allow us to conclude that $\mathbf{X}_u$ is important in determining $\hat{f}$. Clearly this is unecessary in this simple case where we could simply inspect the parameters β, but with statistical learning methods this is not usually possible.

Measuring the change in the loss that results from permuting certain covariates as a way of estimating predictive importance is agnostic to the choice of loss function. With categorical (ordered or unordered) outcome variables category-specific or overall (e.g., average agreement across categories) importance can be computed, which can be useful in cases where categories are imbalanced or the costs of making prediction errors differs across the categories, e.g., as may be the case when predicting violence, where an absence of violence is more common than violence but violence is often costly when not anticipated (Berk et al. 2009). With a continuous outcome the permutation importance can be computed for the mean, i.e., how permuting a covariate affects the conditional expectation function, or any other summary of the distribution of outcomes, such as a quantile. Lastly the permutation importance may not be aggregated at all, giving an observation specific measure of importance. This can, for example, be used to estimate the importance of covariates across the distribution of the outcome (a continuous analogue to category-specific importance).

Permutation importance is certainly not the only method for computing predictive importance with statistical learning methods. There are many method specific measures for Random Forests have been subject to study (Strobl et al. 2007; Strobl et al. 2008; Altmann et al. 2010; Nicodemus et al. 2010; Schwarz, König, and Ziegler 2010; Janitza, Strobl, and Boulesteix 2013; Grömping 2009; Louppe et al. 2013). Friedman (2001) proposed permutation importance as well as other generic methods such as that of Laan (2006) which may prove useful for

political scientists but which are beyond the scope of this paper. At this point the theoretical study of this method is limited: Louppe et al. (2013) investigates the properties of a random forest specific method with a random forest like method which is tractable for his purposes but would not be used in practice.

Laan (2006) Louppe et al. (2013) Friedman (2001)

Permutation importance as described above is implemented in `edarf`, `mmpf`, and `mlr`.

Functional ANOVA

Another approach to making models interpretable is the Analysis of Variance applied to functions: the functional ANOVA (Hooker 2004; Hooker 2012). The functional ANOVA finds the best possible decomposition of a model in terms of additive components, where the additive components depend on a number of covariates set by the user.

That is, $\hat{f}$ can be represented as a constant plus functions of one variable, functions of two variables, up to a number fixed by the user.

$$\hat{f}(\mathbf{x}) = g_0 + \sum_{i=1}^{p} g_i(X_i) + \sum_{i \neq j} g_{ij}(X_{ij}) + \dots$$

Another way to write this is by using the subset operator. Here u is some proper subset of the covariate indices, e.g., $\{1, 2\}$ or $\{2, 3\}$. Each g_u here is the projection of $\hat{f}$ onto the space of functions (the g) additive in u.[7] Each of the components g is constrained to be hierarchically orthogonal, that is, unrelated to components which are proper subset of u. $\hat{f}$ can be completely decomposed as

$$\hat{f}(\mathbf{X}) \approx g_u(\mathbf{x}_u) + \sum_{v \subset u} g_v(\mathbf{x}_v) + g_{-u}(\mathbf{x}_{-u}) + \sum_{i \in u \subset v' \subseteq -i} g_{v'}(\mathbf{x}_{v'})$$

The components of this decomposition are the solution to the optimization problem:

$$\arg\min_{g} \left(g_0 + g_u(\mathbf{x}_u) + \sum_{v \subset u} g_v(\mathbf{x}_v) + g_{-u}(\mathbf{x}_{-u}) + \sum_{i \in u \subset v' \subseteq -i} g_{v'}(\mathbf{x}_{v'}) - \hat{f}(\mathbf{x}) \right)^2$$

which can be estimated using a pointwise estimation scheme described in Hooker (2012) and implemented in the `fanova` package and also available in `mlr`. As described in Hooker (2004) this can be used to, not only estimate the aformentioned components, but also to measure the loss that occurs when excluding them from the decomposition, giving another method of variable importance.

The functional ANOVA is closely related to other variance decomposition methods like ... (cite french papers).

[7] When u is a single covariate, $g_u(\mathbf{x}_u) = \bar{f}_u(\mathbf{x}_u)$. So each g_u can be estimated by eq. 6.

Extrapolation

- existing approaches (convex hull)
- multivariate density estimation

A key issue with the abovementioned methods is extrapolation (See, e.g., King and Zeng (2006) for a discussion of the issue in political science). Regions of eq. 6 and eq. 7, i.e., combinations of $\mathbf{x}_u$ and $\mathbf{x}_{-u}$ or $\mathbf{x}_{-u}^{(i)}$ may have low probability under the joint distribution. Any estimate of the behavior of $\hat{f}(\mathbf{x})$ in this region will have high variance, i.e., it may be interpreted less reliably. It is therefore desireable to detect such regions visually, and, further, to avoid constructing approximations to $\hat{f}(\mathbf{x})$ that are strongly influenced by such regions.

Regions of extrapolation can be detected visually with eq. 7 by, for each curve, adding a point which indicates the observed value of $\mathbf{x}_u$. The density of the points at a given point in $\mathbf{x}_u$ gives an estimate of the marginal density of $\mathbf{x}_u$ at said point. This can be particularly useful when $\mathbf{x}_u$ is multidimensional.

Summary and Future Work

References

Altmann, André, Laura Toloşi, Oliver Sander, and Thomas Lengauer. 2010. "Permutation Importance: A Corrected Feature Importance Measure." *Bioinformatics* 26 (10). Oxford Univ Press: 1340–7.

Baum, Matthew A, and Yuri M Zhukov. 2015. "Filtering Revolution: Reporting Bias in International Newspaper Coverage of the Libyan Civil War." *Journal of Peace Research* 52 (3). Sage Publications Sage UK: London, England: 384–400.

Beck, Nathaniel, and Simon Jackman. 1998. "Beyond Linearity by Default: Generalized Additive Models." *American Journal of Political Science* 42. University of Texas Press: 596–627.

Beck, Nathaniel, Gary King, and Langche Zeng. 2000. "Improving Quantitative Studies of International Conflict: A Conjecture." *American Political Science Review* 94 (01). Cambridge Univ Press: 21–35.

Berk, Richard A. 2008. *Statistical Learning from a Regression Perspective*. Springer.

Berk, Richard, Lawrence Sherman, Geoffrey Barnes, Ellen Kurtz, and Lindsay Ahlman. 2009. "Forecasting Murder Within a Population of Probationers and Parolees: A High Stakes Application of Statistical Learning." *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 172 (1). Wiley Online Library: 191–211.

Bischl, Bernd, Michel Lang, Lars Kotthoff, Julia Schiffner, Jakob Richter, Erich Studerus, Giuseppe Casalicchio, and Zachary M. Jones. 2016. "Mlr: Machine Learning in R." *Journal of Machine Learning Research* 17 (170): 1–5. http://jmlr.org/papers/v17/15-066.html.

Brambor, Thomas, William Roberts Clark, and Matt Golder. 2006. "Understanding

Interaction Models: Improving Empirical Analyses." *Political Analysis* 14 (1). SPM-PMSAPSA: 63–82.

Fariss, Christopher J, and Zachary M Jones. 2015. "Enhancing Validity in Observational Settings When Replication Is Not Possible." *Forthcoming at Political Science Research and Methods*.

Fariss, Christopher J., and Zachary M. Jones. 2017. "Enhancing Validity in Observational Settings When Replication Is Not Possible." *Political Science Research and Methods*. Cambridge University Press, 1–16. doi:10.1017/psrm.2017.5.

Friedman, Jerome H. 2001. "Greedy Function Approximation: A Gradient Boosting Machine." *Annals of Statistics*. JSTOR, 1189–1232.

Friedman, Jerome H, and Bogdan E Popescu. 2008. "Predictive Learning via Rule Ensembles." *The Annals of Applied Statistics*. JSTOR, 916–54.

Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. 2001. *The Elements of Statistical Learning*. Vol. 1. Springer series in statistics Springer, Berlin.

Goldstein, Alex, Adam Kapelner, Justin Bleich, and Emil Pitkin. 2015. "Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation." *Journal of Computational and Graphical Statistics* 24 (1). Taylor & Francis: 44–65.

Green, Donald P, and Holger L Kern. 2012. "Modeling Heterogeneous Treatment Effects in Survey Experiments with Bayesian Additive Regression Trees." *Public Opinion Quarterly*. AAPOR, nfs036.

Grimmer, Justin, and Brandon M Stewart. 2013. "Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts." *Political Analysis*. SPM-PMSAPSA, mps028.

Grömping, Ulrike. 2009. "Variable Importance Assessment in Regression: Linear Regression Versus Random Forest." *The American Statistician* 63 (4).

Hainmueller, Jens, and Chad Hazlett. 2013. "Kernel Regularized Least Squares: Reducing Misspecification Bias with a Flexible and Interpretable Machine Learning Approach." *Political Analysis*. SPM-PMSAPSA, mpt019.

Hainmueller, Jens, Jonathan Mummolo, and Yiqing Xu. 2016. "How Much Should We Trust Estimates from Multiplicative Interaction Models? Simple Tools to Improve Empirical Practice."

Hill, Daniel W, and Zachary M Jones. 2014. "An Empirical Evaluation of Explanations for State Repression." *American Political Science Review* 108 (03). Cambridge Univ Press: 661–87.

Hill, Daniel W., and Zachary M. Jones. 2014. "An Empirical Evaluation of Explanations for State Repression." *American Political Science Review* 108 (3): 661–87.

Hooker, Giles. 2004. "Discovering Additive Structure in Black Box Functions." In *Proceedings of the Tenth Acm Sigkdd International Conference on Knowledge Discovery and Data Mining*, 575–80. ACM.

———. 2012. "Generalized Functional Anova Diagnostics for High-Dimensional Functions of

Dependent Variables." *Journal of Computational and Graphical Statistics*. Taylor & Francis.

Janitza, Silke, Carolin Strobl, and Anne-Laure Boulesteix. 2013. "An Auc-Based Permutation Variable Importance Measure for Random Forests." *BMC Bioinformatics* 14 (1). BioMed Central Ltd: 119.

Jerven, Morten. 2013. *Poor Numbers: How We Are Misled by African Development Statistics and What to Do About It*. Cornell University Press.

Jones, Zachary. 2017. "Mmpf: Monte-Carlo Methods for Prediction Functions."

Jones, Zachary M., and Fridolin J. Linder. 2016. "Edarf: Exploratory Data Analysis Using Random Forests." *The Journal of Open Source Software* 1 (6). The Open Journal. doi:10.21105/joss.00092.

King, Gary. 1989. *Unifying Political Methodology: The Likelihood Theory of Statistical Inference*. University of Michigan Press.

King, Gary, and Langche Zeng. 2006. "The Dangers of Extreme Counterfactuals." *Political Analysis* 14 (2). SPM-PMSAPSA: 131–59.

Laan, Mark J van der. 2006. "Statistical Inference for Variable Importance." *The International Journal of Biostatistics* 2 (1).

Louppe, Gilles, Louis Wehenkel, Antonio Sutera, and Pierre Geurts. 2013. "Understanding Variable Importances in Forests of Randomized Trees." In *Advances in Neural Information Processing Systems*, 431–39.

Lupu, Jones, Yonatan. 2017. "Is There More Violence in the Middle?"

Miller, John H, and Scott E Page. 2009. *Complex Adaptive Systems: An Introduction to Computational Models of Social Life*. Princeton university press.

Montgomery, Jacob M, and Santiago Olivella. 2015. "Tree-Based Models for Political Science Data."

Montgomery, Jacob M, Santiago Olivella, Joshua D Potter, and Brian F Crisp. 2015. "An Informed Forensics Approach to Detecting Vote Irregularities." *Political Analysis*. SPM-PMSAPSA, mpv023.

Muchlinski, David, David Siroky, Jingrui He, and Matthew Kocher. 2016. "Comparing Random Forest with Logistic Regression for Predicting Class-Imbalanced Civil War Onset Data." *Political Analysis* 24 (1). Cambridge University Press: 87–103.

Nicodemus, Kristin K, James D Malley, Carolin Strobl, and Andreas Ziegler. 2010. "The Behaviour of Random Forest Permutation-Based Variable Importance Measures Under Predictor Correlation." *BMC Bioinformatics* 11 (1). BioMed Central Ltd: 110.

Pepinsky, Thomas B. 2017. "Visual Heuristics for Marginal Effects Plots."

Schrodt, Philip A. 2014. "Seven Deadly Sins of Contemporary Quantitative Political Analysis." *Journal of Peace Research* 51 (2). SAGE Publications: 287–300.

Schwarz, Daniel F, Inke R König, and Andreas Ziegler. 2010. "On Safari to Random Jungle: A Fast Implementation of Random Forests for High-Dimensional Data." *Bioinformatics* 26

(14). Oxford Univ Press: 1752–8.

Strobl, Carolin, Anne-Laure Boulesteix, Thomas Kneib, Thomas Augustin, and Achim Zeileis. 2008. "Conditional Variable Importance for Random Forests." *BMC Bioinformatics* 9 (1). BioMed Central Ltd: 307.

Strobl, Carolin, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. 2007. "Bias in Random Forest Variable Importance Measures: Illustrations, Sources and a Solution." *BMC Bioinformatics* 8 (1). BioMed Central Ltd: 25.

Vapnik, Vladimir Naumovich. 1998. *Statistical Learning Theory*. 2nd ed. Wiley New York.

Vapnik, Vladimir, Esther Levin, and Yann Le Cun. 1994. "Measuring the Vc-Dimension of a Learning Machine." *Neural Computation* 6 (5): 851–76.

Ward, Michael D, Brian D Greenhill, and Kristin M Bakke. 2010. "The Perils of Policy by P-Value: Predicting Civil Conflicts." *Journal of Peace Research* 47 (4). Sage Publications: 363–75.