Journal of Machine Learning Research 23 (2022) 1-32 Submitted 11/20; Revised 12/21; Published 8/22

Getting Better from Worse: Augmented Bagging and A Cautionary Tale of Variable Importance

Lucas Mentch LKM31@PITT.EDU
Siyu Zhou SIZ25@PITT.EDU
Department of Statistics
University of Pittsburgh
Pittsburgh, PA 15260, USA

Editor: Isabelle Guyon

Abstract

As the size, complexity, and availability of data continues to grow, scientists are increasingly relying upon black-box learning algorithms that can often provide accurate predictions with minimal *a priori* model specifications. Tools like random forests have an established track record of off-the-shelf success and even offer various strategies for analyzing the underlying relationships among variables. Here, motivated by recent insights into random forest behavior, we introduce the simple idea of augmented bagging (AugBagg), a procedure that operates in an identical fashion to classical bagging and random forests, but which operates on a larger, augmented space containing additional randomly generated noise features. Surprisingly, we demonstrate that this simple act of including extra noise variables in the model can lead to dramatic improvements in out-of-sample predictive accuracy, sometimes outperforming even an optimally tuned traditional random forest. As a result, intuitive notions of variable importance based on improved model accuracy may be deeply flawed, as even purely random noise can routinely register as statistically significant. Numerous demonstrations on both real and synthetic data are provided along with a proposed solution.

Keywords: Variable Importance, Random Forests, Regularization

1. Introduction

As data continues to become larger and more complex, scientists and analysts are increasingly relying upon adaptive learning methods in lieu of the more traditional parametric statistical models that require *a priori* model specification. Among these flexible alternatives, bagging (Breiman, 1996) and random forests (Breiman, 2001) have proven among the most popular and robust tools available with successful application in nearly every scientific field; for just a few select examples, see Díaz-Uriarte and De Andres (2006); Cutler et al. (2007); Bernard et al. (2007); Mehrmohamadi et al. (2016); Coleman et al. (2020). In a recent study, Fernández-Delgado et al. (2014) compared the performance of 179 classification methods across all datasets then available in the UCI Machine Learning Repository (Dua and Graff, 2017) and found random forests to be the top overall performer. In the previous two decades since their inception, numerous studies have sought to establish their important statistical properties including consistency (Biau and Devroye, 2010; Scornet et al., 2015; Klusowski, 2021), asymptotic normality (Mentch and Hooker, 2016; Wager

and Athey, 2018), and rates of convergence (Peng et al., 2019) as well as means by which standard errors (Sexton and Laake, 2009), confidence intervals (Wager et al., 2014; Mentch and Hooker, 2016), and hypothesis testing procedures (Mentch and Hooker, 2016, 2017; Coleman et al., 2022) can be obtained.

Bagging, first introduced in a tree-based setting by Breiman (1996), involves drawing B bootstrap samples from the original training data, refitting the base model (tree) on each, and averaging the individual outputs to obtain the final predictions. When base models are traditional classification or regression trees (Breiman et al., 1984), at each internal node, the optimal empirical split point is chosen by searching over all features and potential splits. Random forests can thus be seen as a less-greedy alternative, whereby eligible features for splitting are randomly selected at each internal node.

Despite the abundance of forest-related work in recent years, substantially less effort has been devoted to principled studies of the inner workings of random forests that might more fully explain their robust record of success. Recently however, Mentch and Zhou (2020) suggested that the additional randomness utilized in random forests was simply an implicit form of regularization. The `mtry` parameter in random forests that dictates the number of available features at each split could therefore be seen as akin to the λ shrinkage penalty in explicit regularization methods like ridge regression (Hoerl and Kennard, 1970) and lasso (Tibshirani, 1996). Mentch and Zhou (2020) suggested that the random subsampling of features helped the trees to avoid overfitting and that this was particularly beneficial in low signal-to-noise ratio settings. LeJeune et al. (2020) demonstrated a similar effect for ensembles consisting of linear model base learners fit via ordinary least squares (OLS).

The idea that the randomness in random forests serves as a means of regularization not only eliminates some of the mystery of their sustained success but also suggests that alternative modifications to the standard bagging procedure that also induce some means of regularization may produce similar gains in accuracy. In this work, we introduce one such alternative idea we refer to as *augmented bagging* (AugBagg) wherein the original feature space is augmented with additional noise features generated conditionally independent of the response, after which the standard bagging procedure is carried out. Very recent work by Kobak et al. (2020) showed that under certain conditions, including particular forms of additional random noise features in the regression can also improve the performance of linear models. As a result, performing minimum-norm least squares on an augmented design with increasingly many features each with increasingly small variance can be seen as equivalent to ridge regression on the original design.

Our work here in the context of bagging and random forests uncovers findings that are arguably even more surprising and troubling. First, unlike the somewhat strict requirements in Kobak et al. (2020), the presence of additional noise features seems to often help regularize the model regardless of their individual variance or dependence on each other. Most alarmingly, in many instances, we show that this simple act of adding extra random noise features to the model can greatly *improve* its out-of-sample predictive accuracy *over even the most optimally tuned model on the original design.* Rather than making a bad model worse as many would naturally presume, the addition of otherwise predictively useless random noise features can have precisely the opposite effect.

This finding has crucial implications for the ways in which we measure and test feature importance. In black-box contexts where traditional measures like p-values may be

unavailable or difficult to obtain, numerous recent studies have formally proposed methods to evaluate feature importance by measuring the change in accuracy when the features of interest are dropped from the model (Mentch and Hooker, 2016, 2017; Lei et al., 2018; Coleman et al., 2022; Williamson et al., 2021). The implicit logic in such procedures feels intuitive and obvious: if the response can be more accurately predicted when a supplemental collection of features are included in the model, then those additional features must hold some information about the response beyond whatever is offered by the original collection of features. This work, however, demonstrates that this need not be the case. Rather, in some instances, particularly when the data itself are quite noisy, independent random noise features can improve predictions when added to a model. This can thus lead to situations that feel almost paradoxical in which, depending on the assumptions made and the type of test deployed, noise features that are completely independent of the response may routinely register as statistically significant. Much further discussion on the implications of this finding is included in the latter sections of this work along with a proposed solution.

The remainder of this paper is laid out as follows. In Section 2 we formally introduce the AugBagg procedure and in Section 3 we provide numerous simulations and real-data experiments to demonstrate its surprisingly competitive predictive performance. In Section 4 we provide theoretical motivation for the AugBagg procedure, building upon very recent results established for other learning procedures. Implications for measuring and testing variable importance are discussed in Section 5, where we also suggest a more robust alternative testing framework in which tests for feature importance maintain the nominal level for noise features, even when such features are capable of producing non-trivial gains in accuracy.

2. Augmented Bagging

Throughout the remainder of this paper, we assume data of the form $\mathcal{D}_n = \{\boldsymbol{Z}_1, ..., \boldsymbol{Z}_n\}$ where each ordered pair $\boldsymbol{Z}_i = (\boldsymbol{X}_i, Y_i)$ consists of a feature vector $\boldsymbol{X}_i = (X_{1,i}, ..., X_{p,i})$ and response $Y_i \in \mathbb{R}$. Given B bootstrap samples of the data, the bagging procedure (Breiman, 1996) generates a prediction at $\boldsymbol{x}$ of the form

$$\hat{y}_{\text{Bagg}} = \frac{1}{B}\sum_{b=1}^{B} T(\boldsymbol{x};\, \omega_b, \mathcal{D}_n) \tag{1}$$

where the randomness ω_b serves only to select the bootstrap sample on which the b^{th} model T is trained. The augmented bagging (AugBagg) procedure we introduce here represents a straightforward extension of classical bagging. Beginning with the original dataset $\mathcal{D}_n$, we create an augmented dataset $\mathcal{D}_n^*$ consisting of additional noise features generated conditionally independent of Y. This augmented dataset thus takes the form $\mathcal{D}_n^* = \{\boldsymbol{Z}_1^*, ..., \boldsymbol{Z}_n^*\}$ where each $\boldsymbol{Z}_i^*$ now denotes an ordered triplet $(\boldsymbol{X}_i, \boldsymbol{N}_i, Y_i)$ consisting of the original features $\boldsymbol{X}_i$ and response Y_i, but also an additional set of noise features $\boldsymbol{N}_i = (N_{1,i}, ..., N_{q,i})$. The original bagging procedure is then performed on this augmented feature space so that the AugBagg output produces predictions of the form

$$\hat{y}_{\text{AugBagg}} = \frac{1}{B}\sum_{b=1}^{B} T((\boldsymbol{x}, \boldsymbol{n});\, \omega_b, \mathcal{D}_n^*) \tag{2}$$

where $\boldsymbol{n}$ can be filled in with random draws from the additional noise features. Note here that the additional noise features are sampled first so that the same draw $\boldsymbol{n}$ is used across each of the B base models rather than being drawn again for each tree.

Importantly, we insist only that $\boldsymbol{N}$ be generated conditionally independent of Y given $\boldsymbol{X}$. This thus allows for additional noise features to be correlated with the original features. The noise features, however, are still sampled at random so that even if duplicate observations $\boldsymbol{x}_i = \boldsymbol{x}_j$ appear in the original data, it need not be the case that $(\boldsymbol{x}_i, \boldsymbol{n}_i) = (\boldsymbol{x}_j, \boldsymbol{n}_j)$. As demonstrated in the following sections, the manner in which noise features are generated can greatly impact performance.

Many of the simulations and experiments carried out below follow the classical definitions and settings of bagging and random forests in which base learners are assumed to be full-depth CART-style trees, as these are the kinds of models most frequently employed in practice and available by default in software. Note that whenever the randomness ω is assumed to select both the bootstrap sample as well as the `mtry` $< p$ eligible features at each internal node as in the case of random forests, the resulting prediction $\hat{y}_{\text{RF}}$ can be written in the same general form as (1). The invariance of CART-style trees to feature scaling presents an additional benefit here, as somewhat less precision is needed in generating the additional noise features for the augmented bagging procedure. We stress, however, that our findings to come are not tree-based and in particular, that the regularization effect offered via augmenting with noise features should be seen ultimately as a by-product of model averaging rather than the specific kinds of base learners that are utilized.

3. Simulations and Real Data Examples

We now present a number of simulation studies to demonstrate the effectiveness of the AugBagg procedure in practice. To begin, we consider a standard linear model of the form $Y = \boldsymbol{X}\beta + \epsilon$ with $n \times p$ design matrix, the rows of which are i.i.d. multivariate normal $\mathcal{N}_p(\boldsymbol{0}, \Sigma)$ where $\Sigma \in \mathbb{R}^{n \times p}$ has entry $(i, j) = \rho^{|i-j|}$ with $\rho = 0.35$. The form of this covariance corresponds to that utilized frequently in the recent work by Mentch and Zhou (2020) and to the 'beta-type 2' setup utilized in Hastie et al. (2020). The original data includes $n = 100$ observations, $p = 5$ original signal features with $\beta_1 = \cdots = \beta_5 = 1$, and q additional i.i.d. noise features sampled from $\mathcal{N}(0, 1)$ independent of $\boldsymbol{X}$ are then added with q ranging from 1 to 250. As in Mentch and Zhou (2020) and Hastie et al. (2020), the noise term ϵ is sampled from $\mathcal{N}(0, \sigma_\epsilon^2)$ with σ_ϵ^2 chosen to satisfy a particular signal-to-noise ratio (SNR), given in this context by $\beta^T \Sigma \beta / \sigma_\epsilon^2$.

Figure 1 shows the performance of the AugBagg procedure where bagging is performed with unpruned trees utilizing both the original p signal features as well as the q additional noise features. Horizontal lines in the background of each plot correspond to random forests at different levels of `mtry` built using only the original $p = 5$ features. Each plot corresponds to a different SNR (0.01, 0.05, 0.09, or 0.14) and shows the relative test error, defined as the test MSE calculated on an independent, randomly generated test set of 1000 observations, scaled by σ_ϵ^2. Each point in each plot corresponds to the error averaged over 500 iterations with error bars showing ± 1 standard deviation. Note that in each case, the random forest error grows as `mtry` increases and so in particular, bagging on only the original 5 features (i.e. a random forest with `mtry` $= 5$) is the worst-performing model. At the lowest SNR of

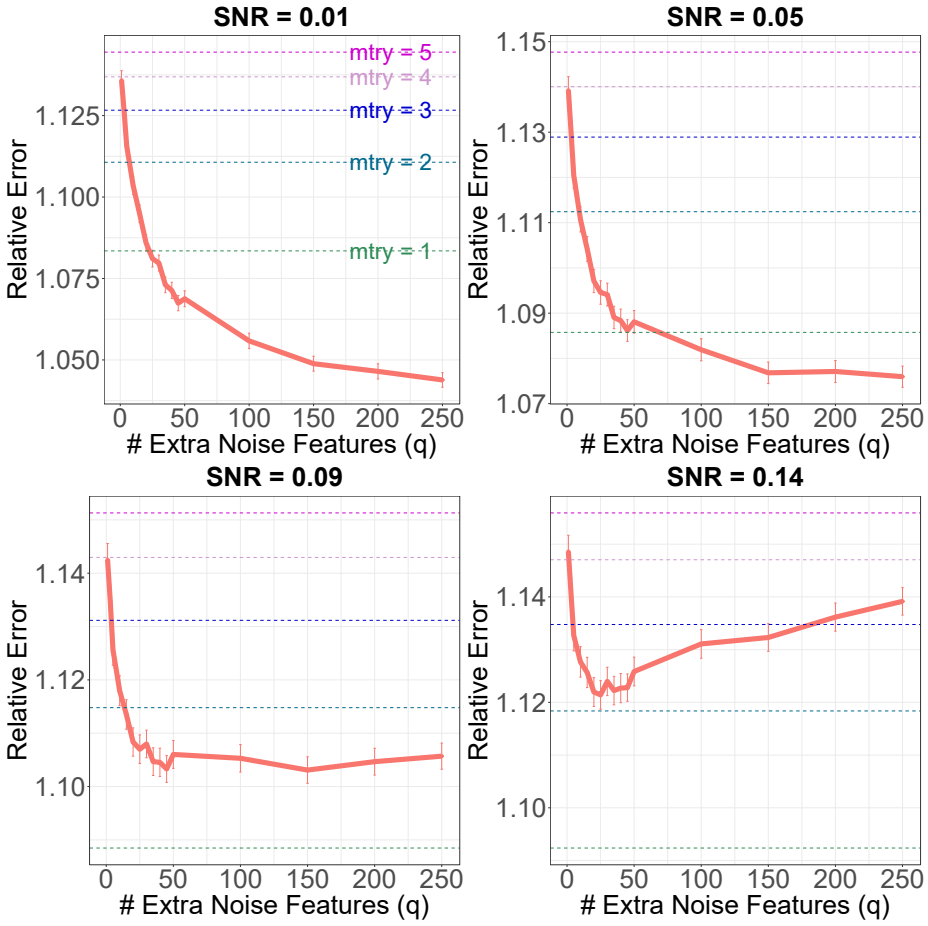

Figure 1: Performance of Augmented Bagging as q additional independent noise features are added to the model as compared with random forests and traditional bagging (`mtry` $= 5$) built on the original data. Each point in each plot corresponds to the average error after repeating the experiment 500 times with error bars showing ± 1 standard deviation.

0.01, however, *augmented* bagging appears to continually improve with q, easily surpassing even the best random forest once approximately $q = 25$ additional noise features are added into the model. Thus, the act of simply adding additional noise features into the model transforms the least accurate model (bagging, or, a random forest with `mtry` $= 5$) into one better than the best model built on the original data (a random forest with `mtry` $= 1$). The results are similar, though less dramatic, when the SNR is increased to 0.05. When the SNR is increased to 0.09, the performance of AugBagg appears to level-off around $q = 50$, never achieving that of the optimal random forest with `mtry` $= 1$. Finally, when the SNR is 0.14, the additional noise features appear to help until approximately $q = 50$, after which point the performance begins to deteriorate.

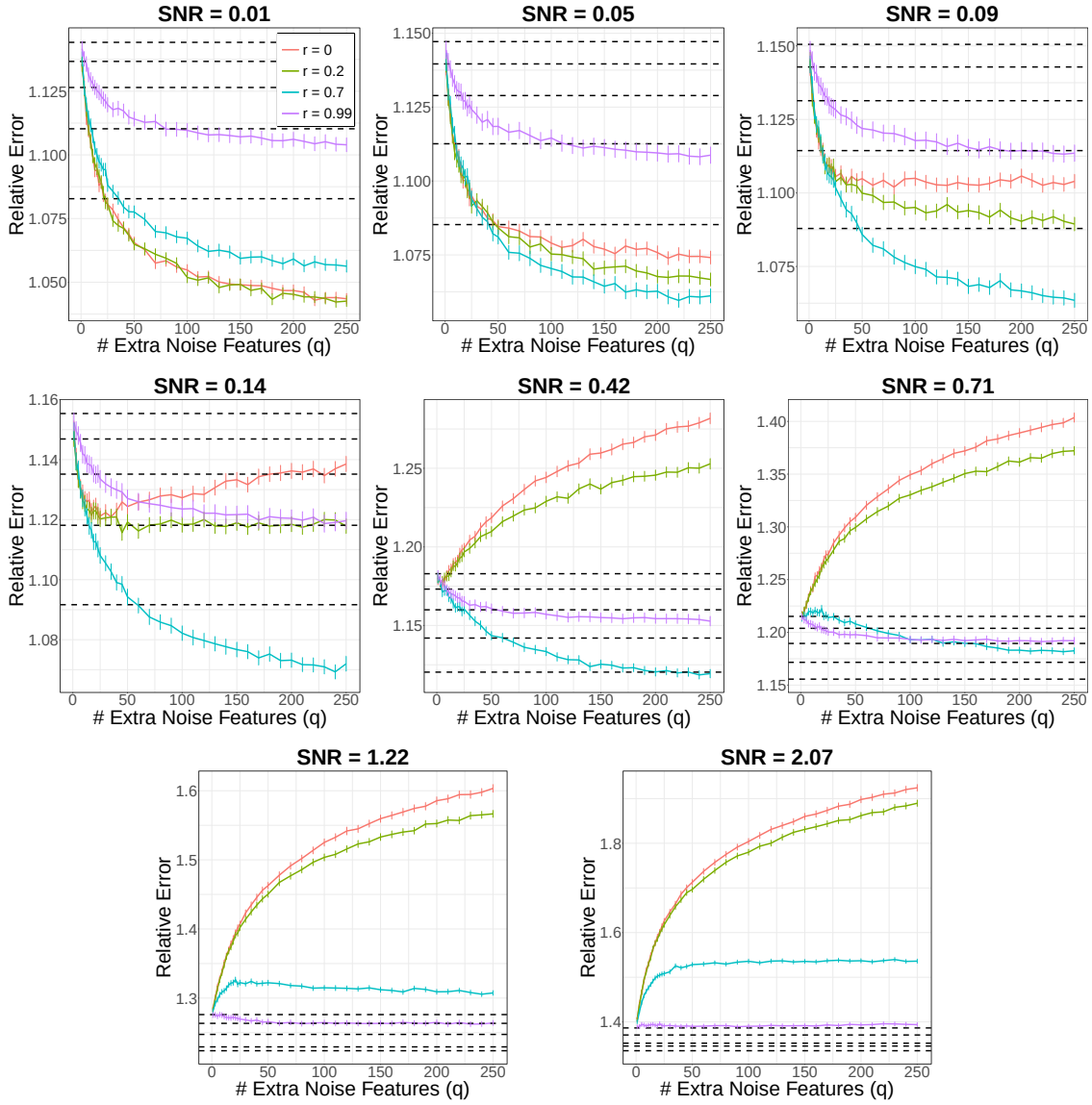

Figure 2: Performance of AugBagg compared against random forests as additional noise variables are added to the model. Different colored lines in each plot correspond to different correlation strengths between the original and noisy additional features. Each black horizontal dashed line in each plot corresponds to the performance of a random forest model built on the original data (without extra noise features) at a fixed value of `mtry`.

The results in Figure 2 expand these simulations. The data and model setup remain the same but the results are explored over a wider range of 8 SNRs, starting at 0.01 and then ranging from 0.05 to 2.07, equally spaced on the log scale. With the exception of the lowest SNR of 0.01, the remaining sequence of SNRs is the same as was recently employed by Hastie et al. (2020) and correspond to a proportion of variance explained (PVE), defined as $\text{SNR}/(1+\text{SNR})$, of 0.01 on the low end (SNR=0.01) and 0.67 on the high end (SNR=2.07). In addition to the additional noise features sampled independently of $\boldsymbol{X}$, here we consider the addition of noisy features that are correlated with one of the first 5 signal features. In a similar fashion to knockoffs (Barber et al., 2015; Candes et al., 2018), such noise features are thus independent of the response Y given $\boldsymbol{X}$. To generate such features, we first select an original feature X at random and generate a standard normal $Z \sim \mathcal{N}(0,1)$. For a given level of correlation r, each of the additional q features then take the form

$$N = rX + \sqrt{1-r^2}Z. \tag{3}$$

In each of the plots in Figure 2 we consider correlations of $r = 0, 0.2, 0.7$, and 0.99 for batches of additional features ranging in size from 1 to 250. Performance is measured in the same fashion and estimates are averaged over 500 replications for each point in each plot. In the following discussion, we will use the shorthand $AB(q,r)$ to denote an AugBagg model with q additional noise features, each of which has correlation r with one of the features in the original dataset.

Figure 2 presents a very interesting and telling story in terms of how the additional noise features are influencing performance and how that influence changes across different SNR levels. Looking only at Figure 1 where the noise features are independent of both the response and the original features, one might suspect that this phenomenon occurs only at very low SNRs. Looking at Figure 2 however, we see that when the noise features are correlated with the original features, improvements in model accuracy are seen even at relatively high SNRs.

At the lowest SNRs of 0.01 and 0.05, we see that in every case, the AugBagg models are improving with the number of extra noise features q. Once $q > 50$, all AugBagg models begin to outperform even the best random forest, with the exception of $AB(q, 0.99)$ where very highly correlated noise features are added. At SNR = 0.09, much the same story is present but now only $AB(q, 0.7)$ outperforms the optimal random forest and again this transition happens around $q = 50$. At SNR = 0.14 we begin to see an interesting shift where the performance of the independent noise model $AB(q, 0)$ begins to deteriorate with q. When the SNR grows to 0.42 and 0.71, this effect is much more pronounced with $AB(q, 0)$ and $AB(q, 0.2)$ both deteriorating with q. At the largest SNRs of 1.22 and 2.07, $AB(q, 0.99)$ is now the only model not deteriorating substantially with q.

3.1 Experiments on Real World Data

The previous simulations demonstrate that the AugBagg procedure can lead to substantial gains in accuracy over the baseline bagging procedure on synthetic datasets. Following a very similar setup to Mentch and Zhou (2020), we now investigate its performance on a variety of real-world datasets. Data summaries are provided in Table 1; a total of nine low-dimensional ($p < n$) and five high-dimensional ($p > n$) datasets are included.

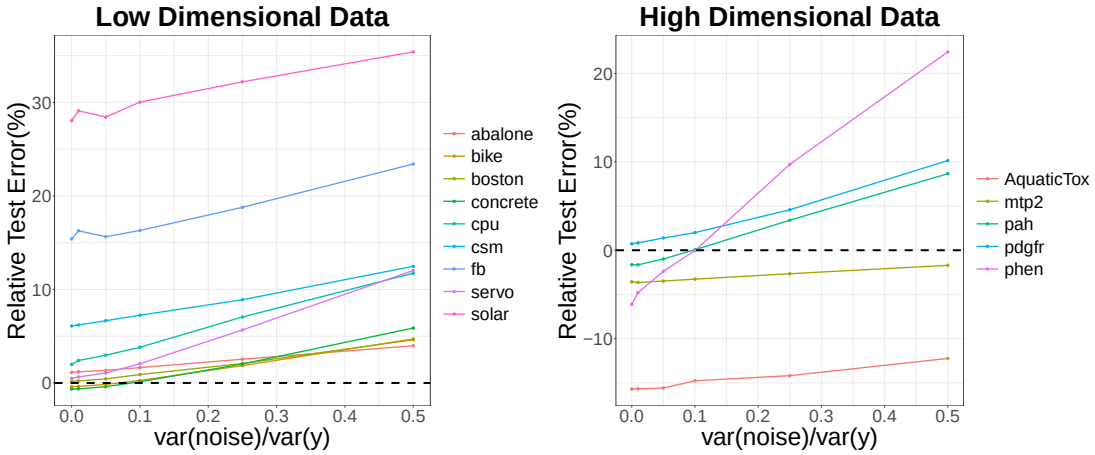

Figure 3: Relative test error (RTE) on real datasets with additional noise added onto the response. Left: low-dimensional datasets. Right: high-dimensional datasets.

In implementing the AugBagg procedure, we consider tuning both the number of additional noise features q as well as the level of correlation r. Since different datasets have different numbers of original features, q is tuned over $p/2$, p, $3p/2$ and $2p$. The correlation strength r is tuned over 0, 0.1, 0.4, 0.7 and 0.9. In datasets with mixed feature types, each additional noise feature is chosen to be correlated with one randomly selected continuous feature from the original data. As in Mentch and Zhou (2020), because the true SNR of real-world data is unknown, we inject further noise of the form $\epsilon \sim N(0, \sigma_\epsilon^2)$ into the response in order to observe trends in changes in model performance when the amount of noise grows larger relative to that in the original data. The variance of the noise σ_ϵ^2 is chosen as some proportion of $\hat{\sigma}_y^2$, the estimated variance of the original response Y. Performance is measured in terms of relative test error (RTE), defined as

$$\text{RTE} = \frac{\widehat{Err}(bagging) - \widehat{Err}(AugBag)}{\hat{\sigma}_y^2} \times 100\% \tag{4}$$

with positive values indicating superior performance by AugBagg. Here $\widehat{Err}$ is obtained via 10-fold cross validation.

Results are shown in Figure 3. In every case, the performance of the tuned AugBagg procedure increases as more noise is added to the response, as demonstrated by the positive slope displayed for each dataset. In 12 of the 14 datasets, AugBagg quickly begins to outperform bagging on the original data with substantial improvements occurring as more noise is injected. Furthermore, it is interesting to note that in the two cases where traditional bagging remains superior, both datasets (`AquaticTox` and `mtp2`) are high-dimensional and, in fact, contain the largest number of original features out of all datasets considered ($p = 468$ and 1142, respectively). In these cases, it is quite possible that many of the original features are themselves noisy and thus the additions we make are of no further benefit. Indeed, an optimally tuned lasso model built on the `AquaticTox` and `mtp2` datasets selects only (approximately) 17% and 4% of the features, respectively.

Dataset	p	n
Abalone Age [`abalone`] Waugh (1995)	8	4177
Bike Sharing [`bike`] Fanaee-T and Gama (2014)	11	731
Bioston Housing [`boston`] Harrison Jr and Rubinfeld (1978)	13	506
Concrete Compressive Strength [`concrete`] Yeh (1998)	8	1030
CPU Performance [`cpu`] Ein-Dor and Feldmesser (1987)	7	209
Conventional and Social Movie [`csm`] Ahmed et al. (2015)	10	187
Facebook Metrics [`fb`] Moro et al. (2016)	7	499
Servo System [`servo`] Quinlan (1993)	4	167
Solar Flare [`solar`] Li et al. (2000)	10	1066
Aquatic Toxicity [`AquaticTox`] He and Jurs (2005)	468	322
Molecular Descriptor Influencing Melting Point [`mtp2`] Bergström et al. (2003)	1142	274
Weighted Holistic Invariant Molecular Descriptor [`pah`] Todeschini et al. (1995)	112	80
Adrenergic Blocking Potencies [`phen`] Cammarata (1972)	110	22
PDGFR Inhibitor [`pdgfr`] Guha and Jurs (2004)	320	79

Table 1: Summary of datasets utilized.

4. Theoretical Motivation and Analogous Results

In the following three subsections, we draw upon recent results on interpolation and implicit regularization in order to provide some theoretical motivation for the practical success of the AugBagg procedure.

4.1 Randomization as Regularization

In very recent work, Mentch and Zhou (2020) argue that the success of random forests is due in large part to a kind of implicit regularization offered by the `mtry` parameter governing the number of features available for splitting at each node. The authors demonstrate that the lower the signal-to-noise ratio (SNR) of the data, the smaller the optimal value of `mtry`. Moreover, the authors demonstrate that this behavior is not tree-specific, but holds for any ensemble consisting of forward-selection-style base learners in which the available features are randomly restricted at each step. Specifically, the authors consider a type of randomized forward selection (RandFS) that proceeds in the same fashion as a standard linear model forward selection process, but where only a randomly selected subset of the remaining features are eligible to be added to the model at each step.

Given data of the form described above, consider a generic regression relationship of the form $Y = f(\boldsymbol{X}) + \epsilon$ and consider an estimate $\hat{f}_{RFS}$ formed by averaging over B individual RandFS models $\hat{f}_{RFS,1}, ..., \hat{f}_{RFS,B}$. Each of the individual RandFS models can be written as

$$\hat{\beta}^{(b)}_{\text{RFS}} = \hat{\beta}^{(b)}_0 + X^{(b)}_{(1)}\hat{\beta}^{(b)}_{(1)} + \cdots + X^{(b)}_{(d)}\hat{\beta}^{(b)}_{(d)}$$

where $X^{(b)}_{(j)}$ is the feature selected at the j^{th} step in the b^{th} model and $\hat{\beta}^{(b)}_{(j)}$ is the corresponding coefficient estimate. Given an orthogonal design matrix, the authors note that for any given feature X_j, the corresponding coefficient estimate in each model is either 0 if X_j is not included in the model or it equals the ordinary least squares estimate $\hat{\beta}_{j,OLS}$ if X_j is selected. Averaging across B models of this form thus yields a term in the final model of

the form $\gamma_j \cdot \hat{\beta}_{j,OLS}$ where γ_j corresponds to the proportion of individual RandFS models in which X_j was included.

In this sense, the ensemblized RandFS procedure can be seen as producing shrinkage and the amount of shrinkage γ_j on each feature depends on both the probability that the feature is made eligible and the probability that the feature is actually selected if made available. While the latter probability depends on the particular modeling technique and loss function employed, the probability of being made eligible is a direct function of only `mtry`.

But the previous statement is only valid under the typical "fixed p" setup where the dimensionality of the feature space is assumed fixed. Suppose instead that `mtry` is held fixed and that the procedure is repeated on an augmented feature space where more noise variables are added. Then under the same setup as above, it's clear that γ_j decreases as a function of the number of extra noise features q since each original feature will thus have a lower probability of being made eligible. However, even for large values of `mtry`, we argue further that the probability of being selected once eligible also decreases as q increases and that such a decrease can be particularly dramatic for features only weakly related to the response. Indeed, given an original feature X_j not perfectly correlated with the response Y in this linear model setting, if we generate additional independent random noise features, eventually some will appear more correlated with Y just by random chance and the weaker the correlation between X_j and Y, the fewer the number of noise features we would expect to need to generate in order to see this. Put simply, as more noise features are added to the model, the probability that some of those new features will appear at least as important as X_j grows with q. Thus, even for large values of `mtry` where the procedure begins to resemble that of bagging, the augmented version of the procedure may produce a similar kind of regularization and shrinkage to that offered by traditional random forests.

4.2 AugBagg and OLS Ensembles

While the recent work of Mentch and Zhou (2020) utilized linear model forward selection settings in order to better illustrate the regularization effect of random forests, in work appearing around the same time, LeJeune et al. (2020) provided an in-depth analysis focused on ensembles where each base learner is simply a linear model constructed on a subsample of features and observations with coefficients estimated via ordinary least squares. As in Mentch and Zhou (2020), the authors observe that feature subsampling at the base-learner stage produces a regularization effect, concluding that for optimally-tuned subsampling rates, the asymptotic risk of the OLS ensemble is equal to the asymptotic risk of ridge regression, an explicit regularization procedure. Here we review the setup utilized in LeJeune et al. (2020) and demonstrate that the same procedure applied to an augmented design is equivalent to one in which more shrinkage is applied to the original data.

Assume now that we have data of the form $\boldsymbol{Z}_1, ..., \boldsymbol{Z}_n$ where each $\boldsymbol{Z}_i = (\boldsymbol{X}_i, Y_i)$ and

$$Y_i = \boldsymbol{X}_i'\beta + \epsilon_i$$

where $Y_i \in \mathbb{R}$ denotes the response, the features $\boldsymbol{X}_i \in \mathbb{R}^p$ are drawn i.i.d. from $\mathcal{N}_p(0_{p\times 1}, \Sigma)$, and the ϵ_i are i.i.d. with mean 0 and variance σ_ϵ^2 and are independent of $\boldsymbol{X}$.

To build OLS ensembles, we draw B submatrices by applying row subsampling to the observations and column subsampling on $\boldsymbol{X} = [\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n]'$. Let S_b and T_b denote the

sets of column and row indices, respectively, selected in the b^{th} model, while $\boldsymbol{S}_b$ and $\boldsymbol{T}_b$ denote the subsampling matrices obtained by selecting the the columns from I_p and I_n corresponding to the indices in S_b and T_b. Let $\mathcal{S}$ and $\mathcal{T}$ denote the entire collections of all possible S_b and T_b, respectively. For each base learner, the OLS minimum-norm estimator is given by

$$\hat{\beta}^{(b)} = \boldsymbol{S}_b \left(\boldsymbol{T}_b' \boldsymbol{X} \boldsymbol{S}_b\right)^+ \boldsymbol{T}_b Y$$

where $(\cdot)^+$ denotes the Moore-Penrose pseudoinverse, so that the estimated coefficients of the ensemble are thus given by

$$\hat{\beta}^{ens} = \frac{1}{B} \sum_{b=1}^{B} \boldsymbol{S}_b \left(\boldsymbol{T}_b' \boldsymbol{X} \boldsymbol{S}_b\right)^+ \boldsymbol{T}_b Y.$$

The risk of $\hat{\beta}^{ens}$

$$R(\hat{\beta}^{ens}) \triangleq \mathbb{E}_{\boldsymbol{x}} \left[\left\langle \boldsymbol{x}, \beta - \hat{\beta}^{ens} \right\rangle^2 \right] = \left\langle \beta - \hat{\beta}^{ens}, \Sigma(\beta - \hat{\beta}^{ens}) \right\rangle$$

is defined as the expected squared error at an independent point $\boldsymbol{x}$, where the $\langle \cdot, \cdot \rangle$ notation denotes the Frobenius inner product. LeJeune et al. (2020) then employ the following assumptions to allow for a more precise evaluation of the risk.

Assumption 1 *(Finite Subsampling) The subsets in the collections $\mathcal{S}$ and $\mathcal{T}$ are selected at random such that $|S_b| < |T_b| - 1$ and that the following hold:*

1. *$Pr(j \in S_b) = \frac{|S_b|}{p}$ for all $j \in [p] = \{1, 2, \ldots, p\}$*
2. *$Pr(m \in T_b) = \frac{|T_b|}{n}$ for all $m \in [n]$*
3. *The subsets $S_1, S_2, \ldots, S_B, T_1, \ldots, T_B$ are conditionally independent given the row subsample sizes $(|T_b|)_{b=1}^B$.*

Assumption 2 *(Asymptotic Subsampling) For some $\alpha, \eta \in [0, 1]$, the subsets in the collections $\mathcal{S}$ and $\mathcal{T}$ are selected randomly such $|S_b|/p \xrightarrow{a.s.} \alpha$ as $p \to \infty$ and $|T_b|/n \xrightarrow{a.s.} \eta$ as $n \to \infty$ for all $b \in [B]$.*

Furthermore, it is assumed that $\Sigma = I_p$, that $\|\beta\|_2 = 1$, and that $p/n \to \gamma$ with $\eta > \alpha\gamma$ as $n, p \to \infty$.

Under these assumptions, conditional on the subset sizes, the expected risk of the bias and variance over $\boldsymbol{X}$, $\mathcal{S}$ and $\mathcal{T}$ converge almost surely as follows:

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{X},\mathcal{S},\mathcal{T}} \left[\text{bias}(\hat{\beta}^{ens}) \right] &\xrightarrow[a.s.]{p,n\to\infty} \frac{B-1}{B} \left(\frac{(1-\alpha)^2}{1-\alpha^2\gamma} \right) + \frac{1}{B} \left(\frac{\eta(1-\alpha)}{\eta - \alpha\gamma} \right) \\
&\xrightarrow{B\to\infty} Bias(\alpha, \gamma) := \frac{(1-\alpha)^2}{1-\alpha^2\gamma} \\
\mathbb{E}_{\boldsymbol{X},\mathcal{S},\mathcal{T}} \left[\text{var}(\hat{\beta}^{ens}) \right] &\xrightarrow[a.s.]{p,n\to\infty} \frac{B-1}{B} \left(\frac{\sigma^2\alpha^2\gamma}{1-\alpha^2\gamma} \right) + \frac{1}{B} \left(\frac{\sigma^2\alpha\gamma}{\eta - \alpha\gamma} \right) \\
&\xrightarrow{B\to\infty} Var(\alpha, \gamma) := \frac{\sigma^2\alpha^2\gamma}{1-\alpha^2\gamma}
\end{aligned}$$

Thus, for an OLS ensemble built with subsamples drawn such that $|S_b| = \lfloor \alpha p \rfloor$ and $|T_b| = \lfloor \eta n \rfloor$ with $p/n \to \gamma$ and ensemble size $B \to \infty$, $\mathbb{E}_{\boldsymbol{X},\mathcal{S},\mathcal{T}}\left[\text{bias}(\hat{\beta}^{ens})\right]$ and $\mathbb{E}_{\boldsymbol{X},\mathcal{S},\mathcal{T}}\left[\text{var}(\hat{\beta}^{ens})\right]$ will converge almost surely to $Bias(\alpha,\gamma)$ and $Var(\alpha,\gamma)$ respectively. Notice that for fixed γ, $Bias(\alpha,\gamma)$ is decreasing in α while $Var(\alpha,\gamma)$ is increasing in α.

Now suppose that the same kind of subsampled OLS ensemble is constructed on an augmented feature space where $\boldsymbol{X}$ is augmented with $\boldsymbol{N} = [\boldsymbol{N}_1, \ldots, \boldsymbol{N}_n]' \in \mathbb{R}^{n \times q}$, and where the $\boldsymbol{N}_i$ are drawn i.i.d. from $N_q(0_{q\times 1}, I_q)$. Let $S_b^\star$ and $T_b^\star$ denote the subsampling indices on the b^{th} model constructed on this augmented design $[\boldsymbol{X}\ \boldsymbol{N}]$ and suppose that the subsampling sizes remain the same as in the OLS ensemble constructed on the original data so that $|S_b^\star| = |S_b|$ and $|T_b^\star| = |T_b|$. Furthermore, suppose that the number of additional features $q \to \infty$ as $p \to \infty$ such that $\frac{q}{p} \to \theta$ for some constant $\theta > 0$. Under these assumptions,

$$\frac{|S_b|}{p+q} \to \frac{\alpha}{1+\theta} = \alpha^\star$$
$$\frac{p+q}{n} \to (1+\theta)\gamma = \gamma^\star,$$

and so $\mathbb{E}_{\boldsymbol{X},\mathcal{S},\mathcal{T}}\left[\text{bias}(\hat{\beta}^{ens})\right]$ and $\mathbb{E}_{\boldsymbol{X},\mathcal{S},\mathcal{T}}\left[\text{var}(\hat{\beta}^{ens})\right]$ converge to $Bias(\alpha^\star,\gamma^\star)$ and $Var(\alpha^\star,\gamma^\star)$, respectively. More specifically,

$$Var(\alpha^\star,\gamma^\star) = \frac{\sigma^2\alpha^{\star 2}\gamma^\star}{1-\alpha^{\star 2}\gamma^\star} = \frac{\sigma^2\alpha^2\gamma}{1+\theta-\alpha^2\gamma}$$

is decreasing with $\theta \geq 0$, so $Var(\alpha^\star,\gamma^\star) \leq Var(\alpha,\gamma)$. Similarly, under the assumption that $\eta > \alpha\gamma$,

$$Bias(\alpha^\star,\gamma^\star) = \frac{(1-\alpha^\star)^2}{1-\alpha^{\star 2}\gamma^\star} = \frac{(1+\theta-\alpha)^2}{(1+\theta)^2-(1+\theta)\alpha^2\gamma}$$

is increasing with $\theta \geq 0$, so $Bias(\alpha^\star,\gamma^\star) \geq Bias(\alpha,\gamma)$. Thus, constructing an OLS ensemble on an augmented design leads to a more regularized estimator with increased bias and decreased variance – the same effect as would be found by constructing the ensemble on the original design with the same η but a smaller subsampling rate.

4.3 Implicit Regularization and Ridge Regression

In addition to the work described above, an intriguing collection of work has emerged in recent years on the so-called "double-descent" phenomenon coined by Belkin et al. (2019), whereby the generalizability error of models may sometimes continue to improve beyond the point of interpolation where training error vanishes. Hastie et al. (2019) followed up this work with an impressive and thorough analysis on the behavior of minimum norm interpolation for high-dimensional least squares estimators. While this work focused on the "ridgeless" setting, interesting related results have also been established for ridge and kernel ridge regression. Kobak et al. (2020) showed that for a standard ridge estimator of the form

$$\hat{\beta}_\lambda = (\boldsymbol{X}'\boldsymbol{X} + \lambda I)^{-1}\boldsymbol{X}'Y$$

the optimal penalty λ can be 0 or negative even when $p \gg n$. In particular, this may happen when the majority of signal comes from a small subset of high-variance features due to an implicit regularization effect offered by a larger collection of relatively low-variance noise features. In very recent work, Jacot et al. (2020) consider ridge estimators acting on a (possibly larger) transformed feature space consisting of Gaussian random features and show that such an estimator with ridge penalty λ is close to a kernel ridge regression estimator with effective penalty $\tilde{\lambda}$ where $\tilde{\lambda} > \lambda$. d'Ascoli et al. (2020) consider a similar random feature setup in investigating the double descent behavior of neural networks and provide a thorough review of much of the recent work on interpolation where we would refer interested readers.

In motivating the AugBagg procedure proposed above, we turn to a key result from Kobak et al. (2020). As above, assume we have (original) training data of the form $(\boldsymbol{X}, Y)$ where $y = \boldsymbol{x}'\beta + \epsilon$ and let $\hat{\beta}_\lambda$ denote the ridge estimator of $\beta \in \mathbb{R}^p$. Now consider a new estimator $\hat{\beta}_q$ formed by performing minimum norm least squares and taking only the first p elements after augmenting $\boldsymbol{X}$ with q additional i.i.d. noise features, each with mean 0 and variance λ/q. The theorem below shows that augmenting the original design with low-variance noise features produces an equivalent regularization effect to ridge regression.

Theorem 1 [Kobak et al. (2020)] *Under the setup described above,*

$$\hat{\beta}_q \xrightarrow[q\to\infty]{a.s.} \hat{\beta}_\lambda.$$

Furthermore, for any $\boldsymbol{x}$, let $\hat{y}_\lambda = \boldsymbol{x}'\hat{\beta}_\lambda$ denote the ridge prediction and let $\hat{y}_{Aug}$ be the prediction generated by the augmented model that includes the additional q parameters using $\boldsymbol{x}$ extended with q random elements generated in the same fashion. Then

$$\hat{y}_{Aug} \xrightarrow[q\to\infty]{a.s.} \hat{y}_\lambda.$$

Kobak et al. (2020) go on to note that a direct but surprising consequence of this result is that "*adding random predictors with some fixed small variance could in principle be used as an arguably bizarre but viable regularization strategy similar to ridge regression.*" Furthermore, the final statement in Theorem 1 implies that the expected MSE of the augmented model (i.e. the non-truncated model that includes the q additional noise features) converges to the MSE of the ridge estimator as $q \to \infty$. In particular, note that when the optimal λ is non-zero, the augmented model with noise features generated according to the procedure outlined above will outperform the model that utilizes only the original data.

Figure 4 gives a demonstration of this surprising result. Here we utilize the same linear model setup described in previous sections with $n = 100$ observations, $p = 75$ features, the first $s = 5$ of which are signal with a coefficient equal to 1. For each SNR, we begin by generating 100 independent datasets and perform cross-validation on each to obtain 100 estimates of the optimal value of λ; the final estimate $\hat{\lambda}_{opt}$ is taken as the median across these. Then, for each combination of SNR and q, we generate an independent training set where the q additional noise features are sampled i.i.d. from $\mathcal{N}(0, \hat{\lambda}_{opt}/q)$. The minimum-norm OLS estimator is then calculated via the singular value decomposition and the relative test error is recorded on an independent test set with 100 observations. The entire process

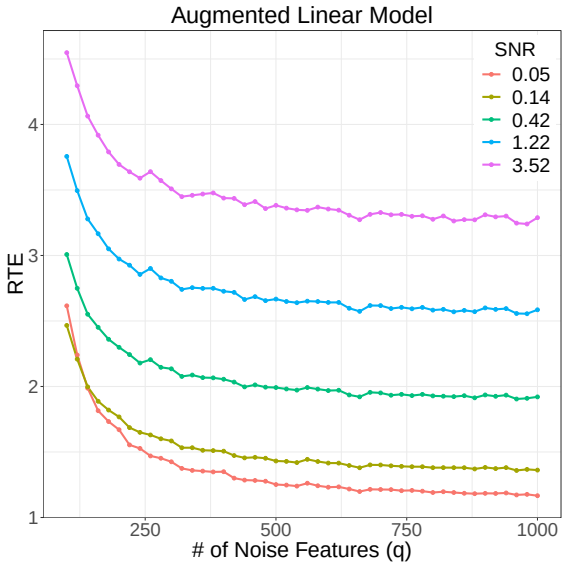

Figure 4: Performance of augmented linear model across different SNRs as increasingly many noise features are added to the model.

is repeated 100 times and the average relative test error is shown in Figure 4. In each case, we see clearly that the model error decreases as more noise features are added into the model.

Suppose now that we build ensembles of estimators of the kind in Theorem 1 by drawing B subsamples, constructing the estimators on each subsample, and averaging. Similar to the setup used above in LeJeune et al. (2020), let $T_b \subseteq [n]$ be the set of indices of selected observations in the b^{th} subsample and let $\boldsymbol{T}_b$ be the $n \times |T_b|$ matrix obtained by selecting columns from I_n corresponding to the indices in T_b. Construct $\hat{\beta}_q^{(b)}$ as above based on $\boldsymbol{T}_b'\boldsymbol{X}$ and $\boldsymbol{T}_b'Y$, which denote the design matrix and response, respectively, corresponding to the observations selected in b^{th} subsample. The final ensemble coefficient estimate formed by averaging the augmented minimum norm estimators is given by

$$\hat{\beta}^{ens} = \frac{1}{B}\sum_{b=1}^{B}\hat{\beta}_q^{(b)}$$

where, by Theorem 1,

$$\hat{\beta}^{ens} = \frac{1}{B}\sum_{b=1}^{B}\hat{\beta}_q^{(b)} \xrightarrow[q\to\infty]{a.s.} \frac{1}{B}\sum_{b=1}^{B}\hat{\beta}_\lambda^{(b)}$$

with

$$\hat{\beta}_\lambda^{(b)} = (\boldsymbol{X}'\boldsymbol{T}_b\boldsymbol{T}_b'\boldsymbol{X} + \lambda I_p)^{-1}\boldsymbol{X}'\boldsymbol{T}_b\boldsymbol{T}_b'Y.$$

Now consider an orthogonal setting where $\boldsymbol{X}\boldsymbol{X}' = I_n$ and let η denote the subsampling rate so that $|T_b|/n \to \eta \in (0, 1]$. Let C be a $n \times n$ diagonal matrix where C_{ii} is the number

of times that the i^{th} observation appears in the B subsamples and let $\lambda_q = \frac{1+\lambda-\eta}{\eta} \geq \lambda$. Using the Woodbury matrix identity, a ridge estimator with penalty λ can be rewritten as

$$\hat{\beta}_\lambda = \frac{1}{1+\lambda}\boldsymbol{X}'Y,$$

and

$$\begin{aligned}
\hat{\beta}^{ens} \xrightarrow[q\to\infty]{a.s.} \frac{1}{B}\sum_{b=1}^{B}\hat{\beta}_\lambda^{(b)} &= \frac{1}{B}\sum_{b=1}^{B}(\boldsymbol{X}'\boldsymbol{T}_b\boldsymbol{T}_b'\boldsymbol{X} + \lambda I_p)^{-1}\boldsymbol{X}'\boldsymbol{T}_b\boldsymbol{T}_b'Y \\
&= \frac{1}{B}\sum_{b=1}^{B}(\lambda^{-1}I_p - (\lambda(\lambda+1))^{-1}\boldsymbol{X}'\boldsymbol{T}_b\boldsymbol{T}_b'\boldsymbol{X})\boldsymbol{X}'\boldsymbol{T}_b\boldsymbol{T}_b'Y \\
&= \frac{1}{B}\sum_{b=1}^{B}\frac{1}{1+\lambda}\boldsymbol{X}'\boldsymbol{T}_b\boldsymbol{T}_b'Y \\
&= \frac{1}{1+\lambda}\frac{1}{B}\boldsymbol{X}'CY \\
&\xrightarrow{B\to\infty} \frac{\eta}{1+\lambda}\boldsymbol{X}'Y \\
&= \frac{1}{1+\lambda_q}\boldsymbol{X}'Y \\
&= \hat{\beta}_{\lambda_q}.
\end{aligned}$$

Thus, in this simple case, an ensemble of minimum-norm least squares estimators constructed on an augmented design produces an estimate equivalent to one produced via ridge regression on the original design. Furthermore, the shrinkage produced by the ensemble is stronger than that of each individual base model.

On a final note, we stress that the purpose of producing this result is not to advocate for this kind of augmented bagging over ridge regression. Indeed, given the equivalence just described paired with the fact that ridge regression is both well-established and naturally motivated, it's difficult to imagine practical settings in which augmented bagging would offer any distinct advantage. Rather, we offer the above results primarily to make explicit the shrinkage that is produced by augmented bagging – a fact that has crucial implications for measuring and testing variable importance.

5. Implications for Variable Importance

Within any kind of black-box supervised learning framework, establishing a valid means of measuring the importance of features is of utmost importance. Indeed, in such non-parametric regimes where model fit and behavior remain largely hidden from view, understanding how features contribute information to the prediction is paramount for scientists and practitioners. In the context of bagging and random forests specifically, Breiman's original out-of-bagg (oob) (Breiman, 2001) importance scores are one such popular measure, though many issues such as a preference for correlated features and those with many categories have been noted in the years following their introduction (Strobl et al., 2007;

Algorithm 1 Random Forest Permutation Test (Coleman et al., 2022)

Require: Original training set $\mathcal{D}_n$, test set $\mathcal{D}_{\text{test}}$, number of permutations P
  Create alternative data $\mathcal{D}_n^*$
  Build ensemble RF with $\mathcal{D}_n$ and predict at $\mathcal{D}_{\text{test}}$
  Build ensemble RF^* with $\mathcal{D}_n^*$ and predict at $\mathcal{D}_{\text{test}}$
  Compute difference in errors $d_0 = MSE(RF^*) - MSE(RF)$
  for i in $1 : P$ **do**
    Randomly shuffle base models between ensembles to form RF_i and RF_i^*
    Compute permuted difference in errors $d_i = MSE(RF_i^*) - MSE(RF_i)$
  Calculate p-value $p = \frac{1}{P+1}\left[1 + \sum_{i=1}^{P} I(d_0 > d_i)\right]$

Nicodemus et al., 2010; Toloşi and Lengauer, 2011). As a result, various formal hypothesis testing procedures have recently been developed to more accurately assess the importance of features in such ensembles. Unfortunately, as demonstrated in the following subsection, even these more rigorous tests are sometimes vulnerable to highly misleading results due to the potentially beneficial effects of noisy features described in the previous sections.

5.1 Hypothesis Tests for Importance

Recently, Mentch and Hooker (2016) proposed a formal hypothesis testing procedure for measuring feature importance in random forests. Given a generic relationship of the form $y = f(\boldsymbol{x}) + \epsilon$, the authors consider partitioning the original set of features $\boldsymbol{X}$ into two groups, $\boldsymbol{X}_0$ and $\boldsymbol{X}_{\text{test}}$, where the latter group contains the features of interest so that a null hypothesis of the form

$$H_0 : g(\boldsymbol{X}_0, \boldsymbol{X}_{\text{test}}) = g_0(\boldsymbol{X}_0) \tag{5}$$

may be rejected whenever the features in $\boldsymbol{X}_{\text{test}}$ make a significant contribution to predicting the response. Under strict assumptions and conditions, the functions g and g_0 may be replaced by the (true) regression functions f and f_0, though in general they are taken to represent the true mean predictions generated by the respective forests. The particular form of these hypotheses relates to the idea of intrinsic vs extrinsic testing discussed at length in Section 5.2. The authors propose to evaluate the hypothesis in (5) by constructing two separate random forest models: one constructed on the original data and one constructed on an altered dataset where the features in $\boldsymbol{X}_{\text{test}}$ are either substituted for randomized replacements independent of the response or dropped from the model entirely. Predictions from each forest are then computed at a number of test points and the differences are combined to form an appropriate test statistic. Coleman et al. (2022) recently proposed a permutation-based alternative to this test. Here again, two forests are constructed in the same fashion as just described, but trees are then randomly permuted across forests and the new difference in accuracy between forests is recorded. That process is then repeated many times to form the null distribution of accuracy differences to which the original difference in accuracy can be compared. An outline of this test is given in Algorithm 1. Note that this nonparametric test avoids the need for explicit variance calculation and as a result is far more computationally efficient and scalable. Also note that these tests are carried out below in the context of subsampled bagging (i.e. with non-random trees), though this can

be seen as merely a special case of random forests with the `mtry` value set equal to the total number of features available.

Crucially, these tests ultimately rely on measuring the difference between either raw predictions or predictive accuracy between two tree-based ensembles constructed on different training sets. Both papers advocate for replacing the features under investigation with randomized alternatives, noting that the tests can potentially produce spurious results when features are instead dropped from the second model, though neither provides a detailed explanation as to why this occurs. Elsewhere in the literature, alternative tests specifically propose to evaluate feature importance by measuring the drop in performance when the features in question are removed from the model. Such is the case, for example, with the **L**eave-**O**ut-**Co**variates (LOCO) measure proposed by Lei et al. (2018) in the context of conformal inference and most recently in the tests proposed by Williamson et al. (2021). Furthermore, though often done informally, it remains common throughout the broader scientific literature for authors to argue for the importance of particular variables based on decreases in model performance when such variables are excluded.

The results presented in the sections above present a substantial concern with such measures. In particular, if model performance can be improved simply by adding randomly generated features that are (at least conditionally) independent of the response, then observing a significant improvement in accuracy when a particular set of features is included does not imply that any relationship to the response or even the other covariates need exist.

To emphasize this point, we implement the test for variable importance recently developed in Coleman et al. (2022) and investigate its behavior under simulated settings. We utilize the same linear model setup as in previous sections with $p = 5$ original signal features sampled from $\mathcal{N}_p(\mathbf{0}, \Sigma)$ with $\Sigma_{ij} = \rho^{|i-j|}$ and $\rho = 0.35$ and consider adding q additional noise features to test for importance. These noise features are either independent of the original five features or are correlated with a randomly selected signal feature with correlation strength r. Thus, relative to the sort of generic null hypothesis specified in (5), our default set of features consist of the original signals so that $\boldsymbol{X}_0 = (X_1, ..., X_5)$ and the features under investigation are those additional noise features being added, $\boldsymbol{X}_{\text{test}} = (N_1, ..., N_q)$. As done previously, the error in the model is adjusted to produce a pre-specified SNR.

To carry out the procedure in Coleman et al. (2022), for each test, we create a training set $\mathcal{D}_n$ with $n = 500$ observations and a test set $\mathcal{D}_{Test}$ with 1000 observations. Let $\boldsymbol{X}$ denote the original $n \times (p + q)$ design matrix and $\boldsymbol{X}^*$ denote the design matrix where the q noise features of interest are either dropped or replaced with a random substitute. Thus, for "drop tests", $\boldsymbol{X}^*$ will be of dimension $n \times p$ whereas for "replacement tests", $\boldsymbol{X}^*$ will be of dimension $n \times (p+q)$. We construct two decision tree ensembles, each with 100 trees. Each tree in the first ensemble is built on a subsample of size 100 from the original training data $(Y, \boldsymbol{X})$; each tree in the second ensemble is built on a subsample of size 100 from the modified data $(Y, \boldsymbol{X}^*)$. Each ensemble here is thus constructed via *subbagging*, though trees are still non-random built to full depth. After recording the original error difference between the two ensembles, trees are randomly shuffled between ensembles a total of 1000 times and each time this new permuted error difference is recorded to form the null distribution. The null hypothesis that the q noise features are not important is rejected whenever the original error difference lies in the upper quantile of the null distribution of permuted error

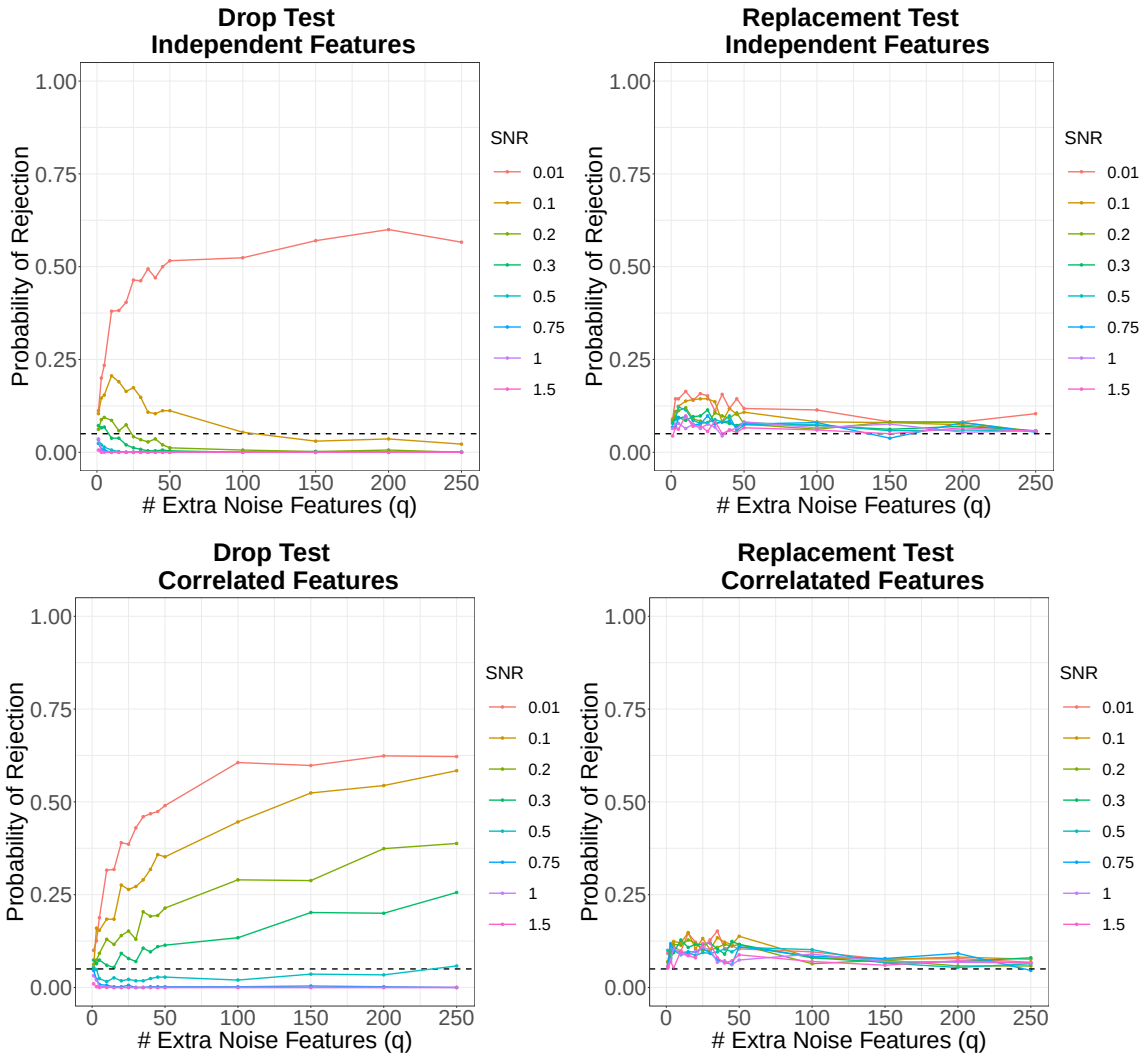

Figure 5: Probability of rejecting the null hypothesis and concluding an additional independent set of noise features are important when dropping the features in question (left column) vs replacing the features in question (right column) when those features are independent (top row) vs correlated (bottom row).

differences. This entire procedure is then repeated 500 times to form empirical rejection probabilities.

Figure 5 shows the probability of rejecting H_0 and concluding the additional noise features are important across various SNRs and numbers of additional features when those features are either dropped or replaced by null substitutes. For these as well as each of the tests deployed below, we set the nominal level to the standard $\alpha = 0.05$ so that if the tests are performing as intuitively expected, we should only see the null hypothesis to be rejected (indicating that the noise features are significant) about 5% of the time. However,

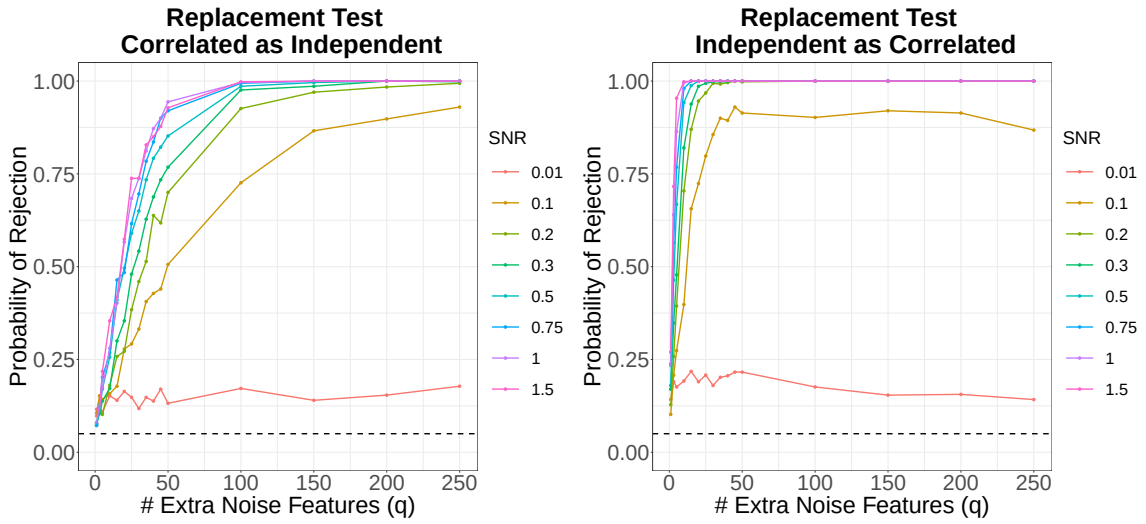

Figure 6: Probability of rejecting the null hypothesis using replacement tests where correlated features are replaced with independent features (left) and where independent features are replaced with correlated features (right).

it is readily apparent that for the drop tests (Figure 5 Left Column), rejections routinely happen well over 5% of the time. This is particularly evident at low SNRs when many additional noise features correlated with the original five features are added where we see (Figure 5 Bottom Left) rejection rates surpassing even 50%.

In previous work both in Mentch and Hooker (2016) and Coleman et al. (2022), the authors claim that the testing procedures developed within are more robust whenever the features under investigation are replaced by randomly generated substitutes rather than being dropped from the model entirely. And indeed, from the right column of Figure 5 it is readily observed that regardless of the SNR or the dependence structure of the noise features on the original features, these replacement tests appear to be far better behaved. Note that these rejection rates do lie very slightly above the nominal rate of 5%, however. This is because the tests developed in Coleman et al. (2022) are valid only asymptotically and in particular, rely on a notion of asymptotic independence between the base models (in this case, trees), which can be achieved asymptotically by subsampling at sufficiently slow rates.

Unfortunately, carrying out accurate replacement-style tests in practice is easier said than done. In the plots shown in the right-hand column of Figure 5, the replacement noise features are sampled from exactly the same distribution as the original noise features being tested for importance. In practice, of course, the distribution of the features in question is unknown. Figure 6 compares the performance of these replacement tests whenever noise features of one kind are replaced by noise features of another kind. On the left, the original noise features are randomly correlated with an original signal feature at $r = 0.7$ and these features are replaced with independent noise features. Here we again notice quite a troubling trend: the test has a very high probability of rejecting across all but the lowest SNRs

and this probability appears to increase with q. Perhaps even worse is the fact that the rejection probabilities appear to be increasing at a faster rate at higher SNRs. Thus, even in "good data" settings, it appears that such tests are very likely to cause correlated noise features to appear important whenever testing against the performance of a model using only independent noise (or, for example, permutations of the original features) as a substitute. While this setting is likely most representative of what might often happen in practice, for completeness, we also consider the opposite setting in the plot on the right of Figure 6 where independent noise features are replaced with ones correlated with a randomly selected feature in $\boldsymbol{X}_0$. Here again we see the same kind of troubling results. These results highlight the potential issues with replacing features by randomized replacements from a different distribution and thus might suggest some promise for procedures involving knockoff variables (Barber et al., 2015; Candes et al., 2018) that explicitly attempt to generate randomized replacements from the same distribution as the original copies. Indeed, recent work by Hooker et al. (2021) suggests such approaches can sometimes offer a drastic improvement, even in low SNR settings.

5.2 Intrinsic vs Extrinsic Testing

Though troubling, these results above should not be at all surprising given the empirical results in Section 3 that showed strong evidence of improved performance when additional noise features are added to the model. These tests simply make clear that such improvements are routinely large enough to register statistical significance. We caution readers from drawing too much from the particular rejection probabilities shown in the left column of Figure 5. These empirical results should in no way be seen as guidelines for how often or under what settings such tests will produce inflated rejection proportions. Rather, the amount by which these kinds of tests inflate the anticipated rejection proportion will depend entirely on the relationships within the data as well as the power of the particular testing procedure employed. Indeed, similar testing procedures with higher power could potentially reject even more often than shown in Figure 5 for the same datasets. By the same reasoning, ensembles consisting of base learners other than trees may also reject more or less often.

The tests carried out above from Coleman et al. (2022) are what a recent work by Williamson et al. (2021) referred to as *extrinsic* tests in that the results are model-specific. Formally, when MSE is the measure of error employed, the hypotheses in Coleman et al. (2022) can be written as

$$\begin{aligned} H_0 &: \mathbb{E}(MSE_{RF}(\mathcal{D}_{\text{test}})) = \mathbb{E}(MSE_{RF^*}(\mathcal{D}_{\text{test}})) \\ H_1 &: \mathbb{E}(MSE_{RF}(\mathcal{D}_{\text{test}})) < \mathbb{E}(MSE_{RF^*}(\mathcal{D}_{\text{test}})) \end{aligned} \tag{6}$$

where the test set $\mathcal{D}_{\text{test}}$ is assumed fixed and the expectation is taken across the training data and any additional randomness involved with the construction of the base learners. By contrast, Williamson et al. (2021) recently put worth a framework for testing *intrinsic* or *population-level* (model-agnostic) notions of variable importance. In particular, the authors consider defining the importance of a collection of features $\mathcal{S}$ as the amount of *oracle predictiveness* lost when those features are excluded. While the framework is flexible enough so as to allow for various forms of importance measures, in our context here, the most natural

corresponding hypotheses for this kind of intrinsic test can be written as

$$\begin{aligned} H_0 &: E(Y - E(Y|X))^2 = E(Y - E(Y|X_{-\mathcal{S}}))^2 \\ H_1 &: E(Y - E(Y|X))^2 < E(Y - E(Y|X_{-\mathcal{S}}))^2. \end{aligned} \tag{7}$$

Comparing the hypotheses in (6) to those in (7), one may wonder why we applied the extrinsic tests in Coleman et al. (2022) rather than the intrinsic tests in Williamson et al. (2021). Indeed, given that the extrinsic tests reject so often, the intrinsic alternative may appear to be the natural solution and even the more direct way of addressing the question really of interest in most practical settings. Unfortunately, while this may be true in theory, valid application of such intrinsic testing procedures requires several strong assumptions, including, for example, that the estimators converge to the true conditional expectations at a rate of $n^{-1/4}$. This is, of course, difficult to guarantee for flexible learning procedures like the bagging and random forest procedures that we employ here consisting of CART-style trees as base learners.

Appendix A contains more detail on the mechanics of how the intrinsic tests in Williamson et al. (2021) can be carried out. We also demonstrate that in this case, nearly identical steps can be taken to produce an analogous extrinsic test. Note from the plots in Appendix A that this extrinsic analogue of the test in Williamson et al. (2021) produces results very qualitatively similar to those in Section 5.1 that utilize the extrinsic test in Coleman et al. (2022).

The fact that extrinsic tests exist that can be carried out in nearly identical fashion to those of analogous intrinsic tests highlights the slipperiness of this issue. Indeed, put simply, it would seem that the primary difference between intrinsic and extrinsic tests largely boils down to the assumptions one is willing to make. Practitioners should thus take extreme care in considering the necessary assumptions before claiming to have conducted a valid intrinsic test. Likewise, readers should always regard claims that a valid intrinsic test was carried out with guarded skepticism and an eye toward whether the necessary assumptions are truly met in the context at hand. Suppose, for example, that one carries out a particular test and finds a significant result; that is, the test rejects the null hypothesis and therefore suggests that a particular collection of features is important. If the test was a valid intrinsic test where all necessary assumptions are met, then one can conclude that there is evidence that those features really do hold unique predictive power for the response not contained in the other features. On the other hand, if those assumptions are not met, the test should therefore be seen as only an extrinsic test and thus, just as we have seen throughout this paper, it is possible that those features may be improving the predictive accuracy of the model and yet may be totally or at least conditionally independent of the response.

Finally, we close this section with a brief discussion on the role of model selection and tuning. In developing their framework for intrinsic testing, Williamson et al. (2021) note the importance of considering a sufficiently rich class of predictive models and carefully tuning across that model class in order to find the optimal predictive model before one should consider moving forward with formal inference. On this point we certainly agree. Indeed, our overarching point in this paper was to show that the predictive accuracy of some supervised learning models could be improved by merely including additional irrelevant noise features. While we focused primarily on bagging to demonstrate this point, it is likely that these troubling effects would have been less severe had we, for example, considered an entire

class of random forests and tuned across the `mtry` parameter and depth of trees. Indeed, as noted in the sections above, the additional noise features here are simply serving as a means of implicit regularization. If sufficient regularization can be accomplished via other means (limiting tree depth or decreasing the `mtry` parameter), the additional noise features may no longer be of additional benefit and may in fact start to degrade performance as one would expect. (See recent work in Zhou and Mentch (2021) for a more detailed discussion on tuning random forests with respect to depth and `mtry`.) This highlights the crucial importance of carefully tuning a random forest via some form of external or cross validation before undertaking any kind of inference. On the other hand, we also want to stress that tuning across a large class is not necessarily sufficient to guarantee the kind of fast convergence needed for intrinsic testing. In recent work by Hastie et al. (2020), for example, the authors repeatedly demonstrate that at low SNRs, best subset selection (in which every possible linear model is constructed) performs quite poorly even when tuned on a large external validation set.

5.3 Bad Tests or Bad Interpretations?

Given the results in Section 5.1, one may be tempted to conclude that procedures of this style that assign relevance to features based on the improvement in predictive accuracy seen when they are included are simply "bad" because the outcomes are "wrong" far too often. Indeed, if rejecting the null hypothesis in a test of this sort is taken to mean that the features in question are "important" and "important" is taken to mean that those features possess some unique explanatory power for the response not captured by the other features available, then certainly such tests would appear to be highly problematic as the rejection rates in the above settings very often lie far above the nominal level of $\alpha = 0.05$.

In our view, however, such an understanding is too naive. The demonstrations above do not necessarily imply that anything is wrong with the tests themselves. Rejecting the null hypotheses in such tests means only that there is evidence that the features in question improve model performance when included. The simulations in Section 3, however, suggest that even the inclusion of additional noise features can improve model performance, sometimes to a dramatic degree. As discussed in the previous subsection, while intrinsic tests can theoretically overcome these model-specific defects, it's difficult to say in general when the necessary assumptions would be met in practical settings when flexible learning models are being employed.

This situation highlights the crucial need for precise language in discussions of feature importance. While "predictive improvement" intuitively feels like a natural proxy, it seems quite unlikely that features independent of the response (at least conditionally) ought to ever be considered "important" for most practical purposes. Certainly this is the case whenever scientists argue that particular features must be collected in order to construct the optimal predictive model or when arguing that features generated by a new piece of technology can lead to further improved model performance over those that were previously available.

In situations such as these, it seems that what is really being sought is not a measure of how "important" certain features may be, but rather how "essential" they are. Even when

additional variables improve model performance, we really seek to determine whether they do so meaningfully or significantly more than randomized alternatives. Interested readers are also invited to see a similar discussion on *model class reliance* appearing recently in Fisher et al. (2019). Finally, as alluded to also in Williamson et al. (2021), practitioners should always have in mind a notion of relevant effect size when conducting tests for importance such as these. While small upticks in predictive accuracy may sometimes be sufficient to achieve statistical significance for certain features, in practice those improvements may still not justify the cost of their collection and inclusion in the model.

6. Discussion

The work in the preceding sections introduced the idea of augmented bagging (AugBagg), a simple procedure identical to traditional bagging except that additional noise features, conditionally independent of the response, are first added to the feature space. Surprisingly, we showed that this simple modification to bagging can lead to drastic improvements in model performance, sometimes even outperforming well-established alternatives like an optimally-tuned random forest. Performance gains appear most dramatic at low SNRs, though the introduction of correlated noise features can continue to improve performance even at higher SNRs. The fact that performance can sometimes be dramatically improved by simply adding conditionally-independent features into the model has important implications for variable importance measures and especially in interpreting the results from tests of variable importance.

On one hand, this work fits well within the rapidly expanding collection of work that explores the potential benefits of excess noisy features. While some earlier papers experimented with the presence of additional noise either added to or multiplied across the original features prior to training (Bishop, 1995; Srivastava et al., 2014), a more popular recent trend has been to analyze models built with random features generated from transforms of the original predictors obtained, for example, via Gaussian Processes or the Random Fourier Features model (see, e.g., Rahimi et al. (2007); Rudi and Rosasco (2017); Belkin et al. (2019); Mei and Montanari (2019); Hastie et al. (2019); Jacot et al. (2020)). Much of this recent work has focused on the idea of the "double descent", demonstrating both empirically and mathematically that purposeful over-parameterization – building models that contain more (random) features than observations – can sometimes be beneficial.

On the other hand, we are not aware of other work specifically defining a procedure by simply augmenting the original data with additional pure noise features to potentially achieve superior predictive accuracy. The fact that models constructed on larger and noisier feature collections are sometimes preferable would seem to run counter to much of traditional statistical thinking. Countless procedures have been proposed in recent decades that assume $\boldsymbol{X} = (\boldsymbol{X}_{\text{Signal}}, \boldsymbol{X}_{\text{Noise}})$ and attempt to uncover the subset of signal features $\boldsymbol{X}_{\text{Signal}}$ with a minimal 'false positive' rate. Indeed, many may intuitively believe that the setting where all available features are signal is something of a 'gold standard' for regression. While there may be good inferential reasons why separating signal and noise is important, this work suggests that such a task is unnecessary and perhaps even detrimental (at least for some models) whenever predictive accuracy is the primary objective.

Along those lines, though AugBagg may sometimes produce predictions substantially more accurate than an alternative baseline like random forests, we stress that the procedure should not be seen as replacing or superseding more efficient procedures like random forests. As detailed in the introduction, random forests have a long documented history of off-the-shelf success and depending on the size of the data at hand, may be much more computationally feasible to implement in practice. Indeed, while random forests reduce the number of features considered at each node, AugBagg, by construction, explicitly increases this computational burden. Furthermore, while recent work by Mentch and Zhou (2020) and Zhou and Mentch (2021) suggests tuning a random forests can sometimes improve performance, at least moderate success can often be found at default values. In contrast, a generic implementation of AugBagg involves tuning both the number of additional features and their correlation with the original features and we are not able to offer default values of these likely to be successful across a broad range of data settings.

Finally, we end by noting that all of the work above was considered within the context of regression. In tree-based contexts, this simply means that predictions at both the tree and ensemble level are formed by averaging. If one were to consider, for example, a classical 0-1 binary response setting in which these kinds of regression trees were still employed (so as to produce estimates generally interpreted as probabilities), we expect the same kinds of potential benefits of random noise features to be present. If, however, those probabilities are then used to perform classification or one employs a majority vote rather than an average, it is unclear to what extent those noise features may remain beneficial. We suspect that such benefits may depend heavily upon the class imbalance in the original data as well as the decision threshold(s) employed. We leave an in-depth study of these issues in classification settings as an open area for potential future work.

Acknowledgments

This research was supported in part by the University of Pittsburgh Center for Research Computing through the resources provided. LM was partially supported by NSF DMS-2015400. We are also grateful to one anonymous reviewer for encouraging the extended discussion on intrinsic vs extrinsic testing.

Appendix A. Intrinsic Testing and Extrinsic Analogues

In this appendix, we review the testing procedure developed in Williamson et al. (2021). For readability, we begin by looking at the extrinsic analogue to the authors' intrinsic test and show that it produces test results similar to those seen in Section 5.1. We then go on to discuss the updates necessary in order to consider that test intrinsic.

To begin, following the sample-splitting scheme introduced in Sec 3.4 of Williamson et al. (2021), suppose that we have 3 independent datasets: a training set $\mathcal{D}_n$ of size n, a test set $\mathcal{D}_{Test,1}$ of size n_1 and another test set $\mathcal{D}_{Test,2}$ of size n_2, containing i.i.d. observations as a generic random vector $\boldsymbol{Z} = (Y, \boldsymbol{X})$ where $\boldsymbol{X} = (X_1, \ldots, X_{p+q})$ and the response $Y \in \mathbb{R}$. For simplicity, we further assume that $n_1 = n_2 = n'$. Let $\boldsymbol{X}^*$ denote the modified random vector where the last q features $(X_{p+1}, \ldots, X_{p+q})$ in $\boldsymbol{X}$ are either dropped or replaced with a random substitutes. Thus, as in the main text, for "drop tests", $\boldsymbol{X}^*$ will be of length p whereas for "replacement tests", $\boldsymbol{X}^*$ will be of length $p + q$.

Let $\hat{f}$ and $\hat{f}^*$ be the ensemble (bagged) estimates on the original data $(Y, \boldsymbol{X})$ and on the modified data $(Y, \boldsymbol{X}^*)$, respectively. Let $f = \mathbb{E}(\hat{f})$ and $f^* = \mathbb{E}(\hat{f}^*)$ where the expectation is over the training set $\mathcal{D}_n$. Define

$$\begin{aligned} MSE(\hat{f}) &= \frac{1}{n'} \sum_{i \in \mathcal{D}_{Test,1}} (Y_i - \hat{f}(\boldsymbol{X}_i))^2, \\ MSE(\hat{f}^*) &= \frac{1}{n'} \sum_{i \in \mathcal{D}_{Test,2}} (Y_i - \hat{f}^*(\boldsymbol{X}_i^*))^2, \\ T &= MSE(\hat{f}) - MSE(\hat{f}^*) \end{aligned}$$

where $MSE(\hat{f})$ and $MSE(\hat{f}^*)$ are independent since $\mathcal{D}_{Test,1}$ and $\mathcal{D}_{Test,2}$ are independent.

By the central limit theorem,

$$\begin{aligned} \sqrt{n'} \left(MSE(\hat{f}) - \mathbb{E}\left[\left(Y - \hat{f}(\boldsymbol{X}) \right)^2 \right] \right) &\to_d N\left(0, \mathrm{Var}\left(\left(Y - \hat{f}(\boldsymbol{X}) \right)^2 \right) \right), \\ \sqrt{n'} \left(MSE(\hat{f}^*) - \mathbb{E}\left[\left(Y - \hat{f}^*(\boldsymbol{X}^*) \right)^2 \right] \right) &\to_d N\left(0, \mathrm{Var}\left(\left(Y - \hat{f}^*(\boldsymbol{X}^*) \right)^2 \right) \right) \end{aligned}$$

where the expectations are with respect to the corresponding test set. Let

$$\Delta = \mathbb{E}\left[(Y - \hat{f}(\boldsymbol{X}))^2 \right] - \mathbb{E}\left[(Y - \hat{f}^*(\boldsymbol{X}^*))^2 \right], \tag{8}$$

$$\sigma^2 = \mathrm{Var}\left(\left(Y - \hat{f}(\boldsymbol{X}) \right)^2 \right) + \mathrm{Var}\left(\left(Y - \hat{f}^*(\boldsymbol{X}^*) \right)^2 \right). \tag{9}$$

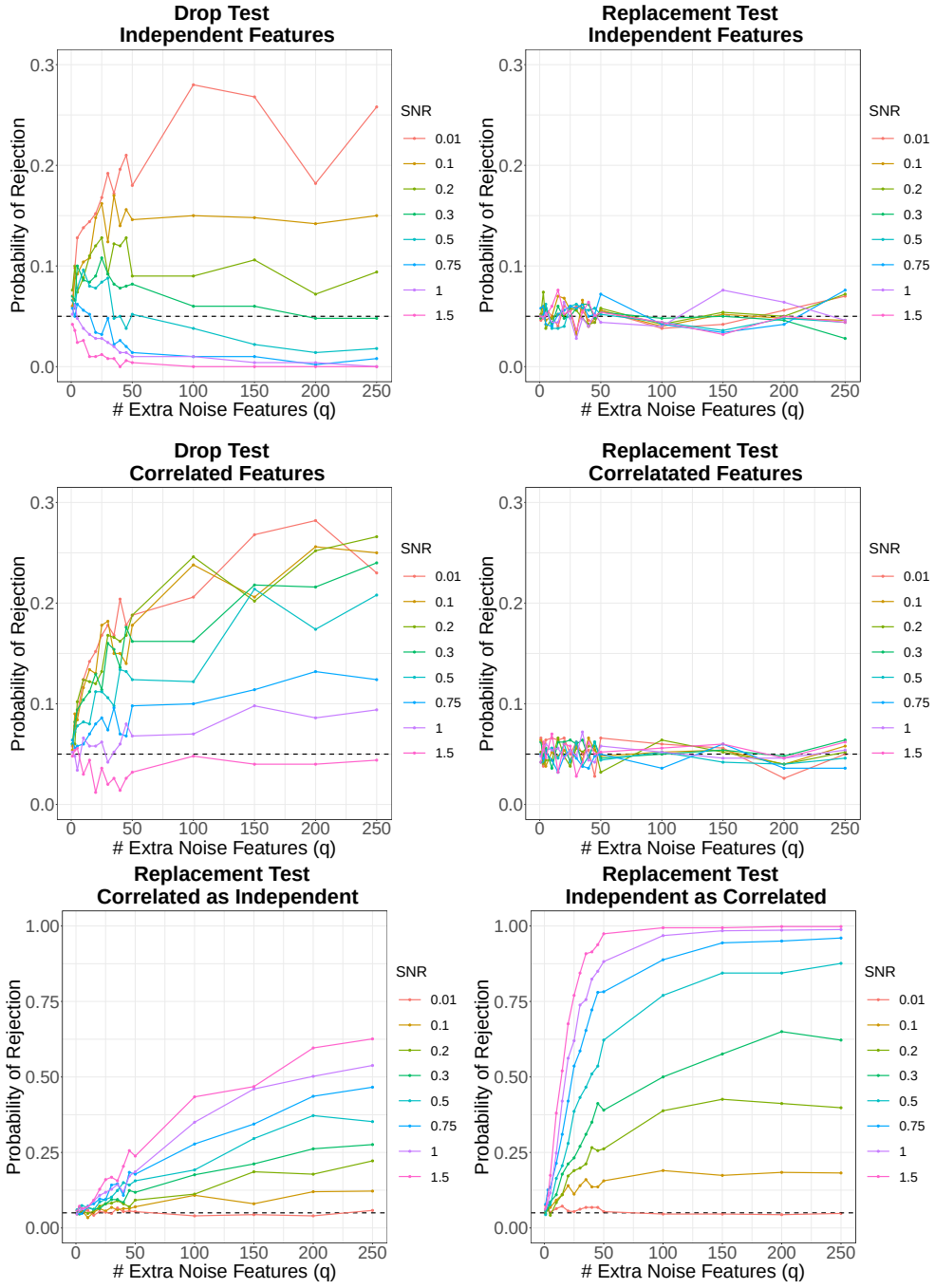

Figure 7: Probability of rejecting the null hypothesis and concluding an additional independent set of noise features are important when dropping the features in question (left column, top two rows) vs replacing the features in question (right column, top two rows) when those features are independent (top row) vs correlated (middle row) and replacement features are sampled from the original underlying distribution (top two rows) In the bottom row, features are replaced by samples from a different distribution.

Then by independence of $MSE(\hat{f})$ and $MSE(\hat{f}^*)$,

$$\sqrt{n'}\,(T-\Delta) \to_d N\left(0, \sigma^2\right)$$

and σ^2 can be estimated with

$$\hat{\sigma}^2 = s_1^2 + s_2^2 \tag{10}$$

where s_1^2 and s_2^2 are the empirical variances of $(Y_1 - \hat{f}(\boldsymbol{X}_i))^2$, $i \in \mathcal{D}_{Test,1}$ and $(Y_1 - \hat{f}^*(\boldsymbol{X}_i^*))^2$, $i \in \mathcal{D}_{Test,2}$, respectively. Here we want to classify the features of interest as important whenever the test MSE is larger in the second ensemble where those features are either dropped or replaced, and thus the null and alternative hypotheses of interest are $H_0 : \Delta = 0$ and $H_1 : \Delta < 0$, respectively.

Figure 7 shows the results of applying this extrinsic tests under identical setups to those described in Section 5.1. Note that the top two rows in Figure 7 correspond to the plots in Figure 5 and the two plots in the bottom row of Figure 7 correspond to those in Figure 6. As noted above, the patterns here are qualitatively quite similar. In Figure 7, we see that these rejection rates are not quite as large in the correlated case with the drop test (middle row, left) compared with those in Figure 5 (bottom left). On the other hand, the rates here are larger across a wider range of SNRs in the independent feature setting (top row, left) as compared with those in Figure 5 (top left). Rejection rates in the replacement tests do not appear to be inflated above the nominal level of 0.05 (Figure 7 top and middle row, right), and as before, rejection rates are far above the nominal level when features are replaced with those from a different distribution (Figure 7 bottom row).

We stress again that the test applied here is extrinsic, and thus it should come as no surprise that these results are much in keeping with those seen in Section 5.1. To turn this into an intrinsic test, we need to find the influence function to estimate the asymptotic variance.

Let P_0 denote the population distribution of $\boldsymbol{Z} = (Y, \boldsymbol{X})$ and for simplicity, denote $\mathbb{E}_{P_0} = \mathbb{E}_0$. Suppose our measure of predictiveness is negative MSE so that we can define $V(f, P_0) = -\mathbb{E}_0(Y - f(\boldsymbol{X}))^2$ and we have population maximizers $f_0(\boldsymbol{X}) = \mathbb{E}_0(Y|\boldsymbol{X}) := \mu_0(\boldsymbol{X})$ and $f_{0,s}(\boldsymbol{X}) = \mathbb{E}_0(Y|\boldsymbol{X}_{-s}) := \mu_{0,s}(\boldsymbol{X}_{-s})$.

Let $\delta_{\boldsymbol{z}}$ denote the degenerate distribution on $\{\boldsymbol{z}\}$. Denote by $\dot{V}(f, P_0; h)$ the Gâteaux derivative of $P \mapsto V(f, P)$ at P_0 in the direction of h. By definition,

$$\dot{V}(f, P_0; h) = \lim_{\tau \to 0} \frac{1}{\tau} \left[V(f, P_0 + \tau h) - V(f, P_0)\right].$$

The influence function corresponding to f_0 is defined to be $\phi_0 : \boldsymbol{z} \mapsto \dot{V}(f_0, P_0; \delta_{\boldsymbol{z}} - P_0)$ and $\phi_{0,s}(\boldsymbol{z})$ can be defined similarly for $f_{0,s}$. Following the discussion in page 6 in Williamson et al. (2021), with equal-size splitting, the asymptotic variance will be of the form of $\eta_0^2 + \eta_{0,s}^2$ where $\eta_0^2 := \mathbb{E}_0(\phi_0(\boldsymbol{Z}))^2$ and $\eta_{0,s}^2 := \mathbb{E}_0(\phi_{0,s}(\boldsymbol{Z}))^2$ and these terms can be estimated separately on the two test sets $\mathcal{D}_{Test,1}$ and $\mathcal{D}_{Test,2}$. Let $P_\tau(\boldsymbol{z}) := P_0 + \tau(\delta_{\boldsymbol{z}} - P_0) = \tau\delta_{\boldsymbol{z}} + (1-\tau)P_0$. Then we have

$$\begin{aligned} V(f_0, P_\tau(\boldsymbol{z})) &= -\,\mathbb{E}_{P_\tau(\boldsymbol{z})}(Y - f_0(\boldsymbol{X}))^2 = -\left[\tau\,\mathbb{E}_{\delta_{\boldsymbol{z}}}(Y - f_0(\boldsymbol{X}))^2 + (1-\tau)\,\mathbb{E}_0(Y - f_0(\boldsymbol{X}))^2\right] \\ &= -\left[\tau(y - f_0(\boldsymbol{x}))^2 + (1-\tau)\,\mathbb{E}_0(Y - f_0(\boldsymbol{X}))^2\right] \end{aligned}$$

and so

$$\begin{aligned}\phi_0(\boldsymbol{z}) &= \lim_{\tau\to 0}\frac{1}{\tau}\left[V(f_0, P_\tau(\boldsymbol{z})) - V(f_0, P_0)\right] \\ &= \lim_{\tau\to 0}\frac{1}{\tau}\left[-\left[\tau(y-f_0(\boldsymbol{x}))^2 + (1-\tau)\,\mathbb{E}_0(Y-f_0(\boldsymbol{X}))^2\right] + \mathbb{E}_0(Y-f_0(\boldsymbol{X}))^2\right] \\ &= \mathbb{E}_0(Y-f_0(\boldsymbol{X}))^2 - (y-f_0(\boldsymbol{x}))^2.\end{aligned}$$

Thus, $\eta_0^2 = \mathbb{E}_0(\phi_0(\boldsymbol{Z}))^2 = \mathrm{Var}((Y-f_0(\boldsymbol{X}))^2)$ can be estimated by the empirical variance s_1^2 on the test set $\mathcal{D}_{Test,1}$. Similarly, $\phi_{0,s}(\boldsymbol{z}) = \mathbb{E}_{0,s}(Y-f_{0,s}(\boldsymbol{X}))^2 - (y-f_{0,s}(\boldsymbol{x}))^2$, and $\eta_{0,s}^2 = \mathbb{E}_0(\phi_{0,s}(\boldsymbol{Z}))^2 = \mathrm{Var}((Y-f_{0,s}(\boldsymbol{X}))^2)$ can be estimated by the empirical variance s_2^2 on the test set $\mathcal{D}_{Test,2}$. Thus, estimated variance of the difference in test MSE based on the influence function is the same as equation (10) in the extrinsic version, thus showing the direct correspondence between the intrinsic and extrinsic versions of these tests.

References

Mehreen Ahmed, Maham Jahangir, Hammad Afzal, Awais Majeed, and Imran Siddiqi. Using crowd-source based features from social media and conventional features to predict the movies popularity. In *2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, pages 273–278. IEEE, 2015.

Rina Foygel Barber, Emmanuel J Candès, et al. Controlling the false discovery rate via knockoffs. *The Annals of Statistics*, 43(5):2055–2085, 2015.

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

Christel AS Bergström, Ulf Norinder, Kristina Luthman, and Per Artursson. Molecular descriptors influencing melting point and their role in classification of solid drugs. *Journal of chemical information and computer sciences*, 43(4):1177–1185, 2003.

Simon Bernard, Sébastien Adam, and Laurent Heutte. Using random forests for handwritten digit recognition. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 2, pages 1043–1047. IEEE, 2007.

Gérard Biau and Luc Devroye. On the layered nearest neighbour estimate, the bagged nearest neighbour estimate and the random forest method in regression and classification. *Journal of Multivariate Analysis*, 101(10):2499–2518, 2010.

Chris M Bishop. Training with noise is equivalent to tikhonov regularization. *Neural computation*, 7(1):108–116, 1995.

Leo Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

Leo Breiman. Random Forests. *Machine Learning*, 45:5–32, 2001.

Leo Breiman, Jerome Friedman, Charles J. Stone, and R.A. Olshen. *Classification and Regression Trees*. Wadsworth, Belmont, CA, 1st edition, 1984.

Arthur Cammarata. Interrelation of the regression models used for structure-activity analyses. *Journal of medicinal chemistry*, 15(6):573–577, 1972.

Emmanuel Candes, Yingying Fan, Lucas Janson, and Jinchi Lv. Panning for gold:'model-x'knockoffs for high dimensional controlled variable selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(3):551–577, 2018.

Tim Coleman, Lucas Mentch, Daniel Fink, Frank A La Sorte, David W Winkler, Giles Hooker, and Wesley M Hochachka. Statistical inference on tree swallow migrations with random forests. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 69 (4):973–989, 2020.

Tim Coleman, Wei Peng, and Lucas Mentch. Scalable and efficient hypothesis testing with random forests. *The Journal of Machine Learning Research*, 23(170):1–35, 2022.

D Richard Cutler, Thomas C Edwards Jr, Karen H Beard, Adele Cutler, Kyle T Hess, Jacob Gibson, and Joshua J Lawler. Random forests for classification in ecology. *Ecology*, 88 (11):2783–2792, 2007.

Ramón Díaz-Uriarte and Sara Alvarez De Andres. Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, 7(1):3, 2006.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Stéphane d'Ascoli, Maria Refinetti, Giulio Biroli, and Florent Krzakala. Double trouble in double descent: Bias and variance (s) in the lazy regime. In *International Conference on Machine Learning*, pages 2280–2290. PMLR, 2020.

Phillip Ein-Dor and Jacob Feldmesser. Attributes of the performance of central processing units: A relative performance prediction model. *Communications of the ACM*, 30(4): 308–318, 1987.

Hadi Fanaee-T and Joao Gama. Event labeling combining ensemble detectors and background knowledge. *Progress in Artificial Intelligence*, 2(2-3):113–127, 2014.

Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinani Amorim. Do we need hundreds of classifiers to solve real world classification problems? *The Journal of Machine Learning Research*, 15(1):3133–3181, 2014.

Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *J. Mach. Learn. Res.*, 20(177):1–81, 2019.

Rajarshi Guha and Peter C Jurs. Development of linear, ensemble, and nonlinear models for the prediction and interpretation of the biological activity of a set of pdgfr inhibitors. *Journal of Chemical Information and Computer Sciences*, 44(6):2179–2189, 2004.

David Harrison Jr and Daniel L Rubinfeld. Hedonic housing prices and the demand for clean air. *Journal of environmental economics and management*, 5(1):81–102, 1978.

Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *arXiv preprint arXiv:1903.08560*, 2019.

Trevor Hastie, Robert Tibshirani, and Ryan Tibshirani. Best subset, forward stepwise or lasso? analysis and recommendations based on extensive comparisons. *Statistical Science*, 35(4):579–592, 2020.

Linnan He and Peter C Jurs. Assessing the reliability of a qsar model's predictions. *Journal of Molecular Graphics and Modelling*, 23(6):503–523, 2005.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Giles Hooker, Lucas Mentch, and Siyu Zhou. Unrestricted permutation forces extrapolation: variable importance requires at least one more model, or there is no free variable importance. *Statistics and Computing*, 31(6):1–16, 2021.

Arthur Jacot, Berfin Simsek, Francesco Spadaro, Clément Hongler, and Franck Gabriel. Implicit regularization of random feature models. In *International Conference on Machine Learning*, pages 4631–4640. PMLR, 2020.

Jason Klusowski. Sharp analysis of a simple model for random forests. In *International Conference on Artificial Intelligence and Statistics*, pages 757–765. PMLR, 2021.

Dmitry Kobak, Jonathan Lomond, and Benoit Sanchez. The optimal ridge penalty for real-world high-dimensional data can be zero or negative due to the implicit ridge regularization. *Journal of Machine Learning Research*, 21(169):1–16, 2020.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.

Daniel LeJeune, Hamid Javadi, and Richard Baraniuk. The implicit regularization of ordinary least squares ensembles. In *International Conference on Artificial Intelligence and Statistics*, pages 3525–3535. PMLR, 2020.

Jinyan Li, Guozhu Dong, and Kotagiri Ramamohanarao. Instance-based classification by emerging patterns. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 191–200. Springer, 2000.

Mahya Mehrmohamadi, Lucas K Mentch, Andrew G Clark, and Jason W Locasale. Integrative modelling of tumour dna methylation quantifies the contribution of metabolism. *Nature communications*, 7:13666, 2016.

Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*, 2019.

Lucas Mentch and Giles Hooker. Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. *The Journal of Machine Learning Research*, 17(1):841–881, 2016.

Lucas Mentch and Giles Hooker. Formal hypothesis tests for additive structure in random forests. *Journal of Computational and Graphical Statistics*, 26(3):589–597, 2017.

Lucas Mentch and Siyu Zhou. Randomization as regularization: A degrees of freedom explanation for random forest success. *Journal of Machine Learning Research*, 21(171): 1–36, 2020.

Sérgio Moro, Paulo Rita, and Bernardo Vala. Predicting social media performance metrics and evaluation of the impact on brand building: A data mining approach. *Journal of Business Research*, 69(9):3341–3351, 2016.

Kristin K Nicodemus, James D Malley, Carolin Strobl, and Andreas Ziegler. The behaviour of random forest permutation-based variable importance measures under predictor correlation. *BMC bioinformatics*, 11(1):110, 2010.

Wei Peng, Tim Coleman, and Lucas Mentch. Asymptotic distributions and rates of convergence for random forests and other resampled ensemble learners. *arXiv preprint arXiv:1905.10651*, 2019.

J Ross Quinlan. Combining instance-based and model-based learning. In *Proceedings of the tenth international conference on machine learning*, pages 236–243, 1993.

Ali Rahimi, Benjamin Recht, et al. Random features for large-scale kernel machines. In *NIPS*, volume 3, page 5. Citeseer, 2007.

Alessandro Rudi and Lorenzo Rosasco. Generalization properties of learning with random features. In *NIPS*, pages 3215–3225, 2017.

Erwan Scornet, Gérard Biau, Jean-Philippe Vert, et al. Consistency of random forests. *The Annals of Statistics*, 43(4):1716–1741, 2015.

Joseph Sexton and Petter Laake. Standard errors for bagged and random forest estimators. *Computational Statistics & Data Analysis*, 53(3):801–811, 2009.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Carolin Strobl, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC bioinformatics*, 8(1):25, 2007.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

R Todeschini, P Gramatica, R Provenzani, and E Marengo. Weighted holistic invariant molecular descriptors. part 2. theory development and applications on modeling physicochemical properties of polyaromatic hydrocarbons. *Chemometrics and Intelligent Laboratory Systems*, 27(2):221–229, 1995.

Laura Toloşi and Thomas Lengauer. Classification with correlated features: unreliability of feature ranking and solutions. *Bioinformatics*, 27(14):1986–1994, 2011.

Stefan Wager and Susan Athey. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, 113(523):1228–1242, 2018.

Stefan Wager, Trevor Hastie, and Bradley Efron. Confidence intervals for random forests: The jackknife and the infinitesimal jackknife. *The Journal of Machine Learning Research*, 15(1):1625–1651, 2014.

Samuel George Waugh. *Extending and benchmarking Cascade-Correlation: extensions to the Cascade-Correlation architecture and benchmarking of feed-forward supervised artificial neural networks*. PhD thesis, University of Tasmania, 1995.

Brian D Williamson, Peter B Gilbert, Noah R Simon, and Marco Carone. A general framework for inference on algorithm-agnostic variable importance. *Journal of the American Statistical Association*, pages 1–14, 2021.

I-C Yeh. Modeling of strength of high-performance concrete using artificial neural networks. *Cement and Concrete research*, 28(12):1797–1808, 1998.

Siyu Zhou and Lucas Mentch. Trees, forests, chickens, and eggs: when and why to prune trees in a random forest. *arXiv preprint arXiv:2103.16700*, 2021.