MDI+: A Flexible Random Forest-Based Feature Importance Framework

Abhineet Agarwal*
Department of Statistics, University of California, Berkeley
Ana M. Kenney*
Department of Statistics, University of California, Berkeley
Yan Shuo Tan*
Department of Statistics and Data Science, National University of Singapore
Tiffany M. Tang*
Departments of Statistics, University of California, Berkeley
Bin Yu
Department of Statistics and EECS, University of California, Berkeley

July 6, 2023

Abstract

Mean decrease in impurity (MDI) is a popular feature importance measure for random forests (RFs). We show that the MDI for a feature X_k in each tree in an RF is equivalent to the unnormalized R^2 value in a linear regression of the response on the collection of decision stumps that split on X_k. We use this interpretation to propose a flexible feature importance framework called MDI+. Specifically, MDI+ generalizes MDI by allowing the analyst to replace the linear regression model and R^2 metric with regularized generalized linear models (GLMs) and metrics better suited for the given data structure. Moreover, MDI+ incorporates additional features to mitigate known biases of decision trees against additive or smooth models. We further provide guidance on how practitioners can choose an appropriate GLM and metric based upon the Predictability, Computability, Stability framework for veridical data science. Extensive data-inspired simulations show that MDI+ significantly outperforms popular feature importance measures in identifying signal features. We also apply MDI+ to two real-world case studies on drug response prediction and breast cancer subtype classification. We show that MDI+ extracts well-established predictive genes with significantly greater stability compared to existing feature importance measures. All code and models are released in a full-fledged python package on Github.[1]

Keywords: Interpretable machine learning, Explainable AI, Decision trees, Ensembles, Non-parametrics

*Denotes equal contribution.

[1]MDI+ is integrated into the imodels package github.com/csinva/imodels [67]

1 Introduction

Random forests (RFs) [10] are among the most popular supervised learning algorithms. They achieve state-of-the-art prediction performance over a wide class of learning problems [13, 20, 55], often outperforming deep learning methods on small or moderately-sized tabular datasets [66]. These types of datasets frequently arise in high-stakes applications such as biology, medicine, and the social sciences due to high costs of data collection.

Given the strong predictive performance of RFs in these settings, practitioners are also keen on using them to extract new scientific insights [5, 17]. To illustrate the utility of RFs in real-world scientific problems, we focus on the two following case studies: (i) predicting the efficacy of cancer drugs given genetic information in the Cancer Cell Line Encyclopedia (CCLE) [2] and (ii) classifying breast cancer subtypes using gene expression data cataloged in The Cancer Genome Atlas (TCGA) [58]. In both problems, RFs have not only shown strong predictive performance [16, 77, 43], but have also been used to understand the genetic risk factors that drive drug response or are predictive of a particular breast cancer subtype.

Determining these genetic risk factors via RFs is done via feature importance measures, which summarize each feature's contribution to a model's predictions. Mean decrease in impurity (MDI) [11] is arguably the most popular feature importance method for RFs, serving as the default measure in `scikit-learn` [59]. It measures the importance for a given feature X_k by tallying up the decrease in variance associated to each split on X_k in the ensemble. Scientists have used MDI in both case studies to pinpoint genes predictive of a drug's response on cancer cell lines [77, 42], and to identify chemical pathways that differentiate between breast cancer subtypes [43]. Using feature importance measures to identify these biological factors allows machine learning (ML) techniques such as RFs to be used for improved diagnosis, early prevention, and development of novel therapeutics.

Despite these successes, MDI suffers from notable "biases" when used to identify important features. It favors features with higher entropy (e.g., continuous variables) or lower correlation with other features, and does so in a way that is independent of their relationship with the response function [69, 53, 68, 54, 51]. This is unfortunate, because causal genes with low entropy or whose expression is strongly correlated with other nearby genes are commonly observed in datasets such as CCLE and TCGA. Consequently, MDI may produce inaccurate and misleading feature importance rankings, often leading to large downstream costs when used in high-stakes decisions

such as deciding which genes to prioritize for future investigation.

Separately, Tan et al. [71] established that trees are statistically inefficient at fitting regression functions with additive components. Such structure is ubiquitous in many scientific problems [21], as seen by the prevalent use of linear models in both case studies [35, 58]. Since MDI is based on the fitted model, its fidelity to the true feature importances degrades when there is poor model fit [49, 80].

To address these challenges, we propose a new framework for feature importance measures known as MDI+. The starting point of our framework is a recently discovered connection between decision trees and linear models [38, 1]. That is, a decision tree is the best fit linear model on a collection of engineered features associated with splits (i.e., local decision stumps) from the tree. We use this connection to *establish a new interpretation of MDI as an R^2 value in this linear regression*. Further, we use this interpretation to explain the drawbacks listed above as well as to reveal several new ones. MDI+ overcomes these shortcomings and generalizes MDI by:

1. Using more flexible models (e.g., regularized generalized linear models (GLMs)) and similarity metrics in place of linear regression and R^2.
2. Appending smooth, linear features (e.g., the raw features) and possibly other features (e.g., engineered from prior knowledge) to the nonlinear local decision stump representation.
3. Introducing computationally-efficient sample splitting for evaluating similarity metrics such as R^2.

As will be explained in subsequent sections, regularization and sample splitting help to overcome the known biases of MDI, while appending linear features helps to overcome the model mismatch between trees and additive regression functions. The increased flexibility provided by the MDI+ framework also offers practitioners the ability to tailor the feature importance computation to the problem structure and include any prior information via the choice of GLM, similarity metric, and expanded feature representation. To further assist practitioners with these choices, we provide a data- and stability-driven procedure to guide model selection and aggregation (i.e., combining importance rankings from multiple models) inspired by the Predictability, Computability, and Stability (PCS) framework for veridical data science [80]. The PCS framework builds upon the three core principles in its name to bridge, unify, and expand on the best ideas from machine learning and statistics for the entire data science life cycle. Moreover, as a by-product of our MDI+ framework, the flexible use of GLMs and expanded feature representations yields a new

class of powerful prediction models, RF+, which often improves upon the prediction performance of RFs and provides a bridge between classical statistical models (i.e., GLMs) with modern ML.

To demonstrate the effectiveness of MDI+, we conduct an extensive data-inspired simulation study, showing that MDI+ significantly outperforms other popular feature importance measures (e.g., TreeSHAP [47]) in its ability to identify signal features. These data-inspired simulations were carefully designed to reflect a diverse range of problem structures (e.g., regression/classification, linear/non-linear) and challenges (e.g., low signal-to-noise, outliers, omitted variables, small sample sizes) commonly encountered in the real world. Through these simulations, we highlight the flexibility of MDI+ by showing how tailoring the GLM (e.g., using Huber regression in the presence of outliers) and metric to the problem structure at hand can lead to better feature ranking performance. We also establish that MDI+ overcomes the biases of MDI through simulations with highly-correlated features and features with varying levels of entropy.

Finally, we return to the two motivating case studies and show that top-ranked features from MDI+ are highly predictive and concur with established domain knowledge with significantly greater stability than other feature importance measures. As advocated by the PCS framework [80], interpretation methods can only be trusted if they are both predictive and stable with respect to reasonable perturbations. Additionally, RF+ increases prediction accuracy over RF in these case studies by approximately 5%, further enhancing the credibility of the feature ranking from MDI+. The strong predictive performance of RF+ and stability of MDI+ in both case studies demonstrate its utility as a powerful and practical tool to extract reliable scientific insights in real-world problems.

Organization. In Section 2, we review other popular feature importance measures. In Section 3, we establish our connection between MDI and R^2 values, and discuss how this interpretation explains known drawbacks, and reveals new ones. In Section 4, we introduce MDI+ and RF+. In Section 5, we conduct an extensive data-inspired simulation study, demonstrating the efficacy of MDI+ in identifying signal features. In Section 6, we show that MDI+ overcomes the biases of MDI against highly-correlated and low-entropy features. In Section 7, we apply MDI+ to both of our motivating case studies, and investigate the stability and accuracy of its feature rankings in comparison with other feature importance measures. Finally, we conclude with a discussion in Section 8.

2 Related Work

In this section, we briefly review existing approaches to combat the known biases of MDI as well as other RF feature importance measures that are commonly used in practice.

MDI. Previous work [69, 53] has established that MDI is biased towards high entropy features (e.g., continuous features). Many strategies have been proposed to mitigate this bias. Sandri and Zuccolotto [64] proposed creating artificial uninformative features to evaluate bias, and Nembrini et al. [52] created a fast implementation of this computationally intensive procedure. Zhou and Hooker [83] and Li et al. [40] proposed UFI and MDI-oob respectively, which both use out-of-bag samples to evaluate MDI in slightly different ways. Additionally, a penalized framework which combines in-bag and out-of-bag (OOB) samples has been shown to debias MDI [44, 45]. Researchers have also tried to understand MDI theoretically by analyzing the population MDI values under some distributional assumptions [46, 65] as well as obtaining consistency guarantees [65]. In particular, Scornet [65] shows that the normalized sum of the MDI values of a single tree is the global R^2 (total variance explained) of the model, but does not connect individual MDI values to linear regression R^2 values. Klusowski and Tian [39] show a connection between MDI and linear correlation for decision stump models and prove nonparametric variable selection consistency.

MDA. In addition to MDI, Breiman [10] proposed a permutation-based feature importance measure called Mean Decrease in Accuracy (MDA). To measure the importance of a feature X_k, MDA permutes the values of X_k marginally for OOB samples, and calculates the excess prediction loss incurred on these samples. Recent work has studied MDA and variants thereof both empirically [68, 24, 27, 25, 31] and theoretically [26, 61, 8].

TreeSHAP. Unlike MDI and MDA, SHAP (SHapley Additive exPlanations) [48] is a model-agnostic feature attribution procedure that provides local (i.e., sample-specific) feature importance scores. SHAP uses Shapley values from economic game theory to compute the contribution of each feature to a prediction for any given sample. These values can be summarized into a global feature importance measure by taking a mean over the samples. TreeSHAP [47] is a computationally-efficient implementation of SHAP values for tree-based methods.

Other feature importance methods and comparisons. To the best of our knowledge, MDI, MDA, and TreeSHAP are the most popular feature importance measures for RFs, although both

global and local scores can be defined in other ways [33, 34, 36, 19, 63, 70, 39]. Also related are model-agnostic conditional dependence scores [3, 82], model-agnostic confidence intervals for feature importances [23], and importance measures for interactions, such as via iterative random forests that have been widely adopted in the genomics community [5, 6].

Dealing with correlated features. Empirical studies show that neither MDI nor MDA work well when features are highly correlated [68, 54, 53, 51]. In particular, permutation-based measures such as MDA have been criticized because permutations break dependencies between features in the dataset [31]. Hence, they result in evaluations of the model out-of-distribution, i.e., on regions of the covariate space with little or no data. These scores may have little connection to the underlying data-generating process because RFs are known to extrapolate to such regions in unreliable ways. This problem of out-of-distribution evaluation also affects TreeSHAP. To overcome this, variants of permutation scores have been proposed [68, 31] while other works have investigated altering the RF algorithm altogether [32].

3 Connecting MDI to R^2 Values from Linear Regression

Assume we are given a dataset $\mathcal{D}_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, with covariates $\mathbf{x}_i \in \mathbb{R}^p$ and responses $y_i \in \mathbb{R}$. An RF is an ensemble of classification or regression trees (CART) [11, 10] that are fitted independently of one another on bootstrapped versions $\mathcal{D}_n^*$ of $\mathcal{D}_n$. Each CART model is fit by performing recursive axis-aligned splits according to a minimum impurity decrease criterion.

In more detail, a *potential split* s of a node $\mathfrak{t}$ partitions it into two children nodes $\mathfrak{t}_L = \{\mathbf{x}_0 \in \mathfrak{t} \colon x_{0,k} \leq \tau\}$ and $\mathfrak{t}_R = \{\mathbf{x}_0 \in \mathfrak{t} \colon x_{0,k} > \tau\}$ for some feature index k and threshold τ. The *impurity decrease* of s is defined as

$$\hat{\Delta}(s, \mathcal{D}_n^*) := N(\mathfrak{t})^{-1} \left(\sum_{\mathbf{x}_i \in \mathfrak{t}} (y_i - \bar{y}_\mathfrak{t})^2 - \sum_{\mathbf{x}_i \in \mathfrak{t}_L} (y_i - \bar{y}_{\mathfrak{t}_L})^2 - \sum_{\mathbf{x}_i \in \mathfrak{t}_R} (y_i - \bar{y}_{\mathfrak{t}_R})^2 \right), \tag{1}$$

where all summations are over samples in $\mathcal{D}_n^*$, $N(\mathfrak{t})$ represents the number of bootstrap samples in node $\mathfrak{t}$, and $\bar{y}_\mathfrak{t}, \bar{y}_{\mathfrak{t}_L}, \bar{y}_{\mathfrak{t}_R}$ are the mean responses in each node. Starting with the entire covariate space as the root node and until it reaches a stopping condition, CART recursively splits each node by picking and actualizing the potential split with the largest impurity decrease. When CART is used as part of RF, a random subset of features is chosen at each node and the best split

is chosen from within this subset. To predict, CART identifies the unique leaf node containing the query point and predicts the mean response over that node.

To define the MDI values for a CART model, first let $\mathcal{S}=\{s_1,...,s_m\}$ denote all the splits it contains and call this its *tree structure*. For each $k=1,...,p$, the MDI of feature X_k is defined as

$$\mathrm{MDI}_k(\mathcal{S},\mathcal{D}_n^*) \coloneqq \sum_{s\in\mathcal{S}^{(k)}} n^{-1} N(\mathbf{t}(s))\hat{\Delta}(s,\mathcal{D}_n^*), \tag{2}$$

where $\mathcal{S}^{(k)}$ is the subset of splits in $\mathcal{S}$ that split on the feature X_k and $\mathbf{t}(s)$ is the node being split by a split s. For RFs, the MDI of feature X_k is the mean across all trees in the ensemble.

Now to build the connection between MDI and R^2 values given a tree structure $\mathcal{S}$ and data $\mathcal{D}_n$, we associate to each split $s\in\mathcal{S}$ the local decision stump function

$$\psi(\mathbf{x};s,\mathcal{D}_n) = \frac{N(\mathbf{t}_R)\mathbf{1}\{\mathbf{x}\in\mathbf{t}_L\} - N(\mathbf{t}_L)\mathbf{1}\{\mathbf{x}\in\mathbf{t}_R\}}{\sqrt{N(\mathbf{t}_L)N(\mathbf{t}_R)}}. \tag{3}$$

Intuitively, ψ is a tri-valued function that indicates whether the sample $\mathbf{x}$ lies to the left of the threshold, lies to the right of the threshold, or is not contained in node $\mathbf{t}$ at all. If $\mathcal{S}$ has m splits, concatenating these m functions yields the learned feature map $\Psi(\mathbf{x};\mathcal{S},\mathcal{D}_n) \coloneqq (\psi(\mathbf{x};s_1,\mathcal{D}_n),...,\psi(\mathbf{x};s_m,\mathcal{D}_n))$ and its corresponding transformed dataset $\Psi(\mathbf{X};\mathcal{S},\mathcal{D}_n)\in\mathbb{R}^{n\times m}$, where $\mathbf{X}\in\mathbb{R}^{n\times p}$ is the original data matrix. Klusowski [38] showed that the CART model is equivalent to the best fit linear model of the responses $\mathbf{y}$ on the transformed dataset $\Psi(\mathbf{X};\mathcal{S},\mathcal{D}_n)$. We further note that there is a natural partition of decision stumps according to the (original) feature that they split on. That is, we have $\mathcal{S}=\mathcal{S}^{(1)}\sqcup\cdots\sqcup\mathcal{S}^{(p)}$, and can write $\Psi(\mathbf{x};\mathcal{S},\mathcal{D}_n) = \big(\Psi(\mathbf{x};\mathcal{S}^{(1)},\mathcal{D}_n),...,\Psi(\mathbf{x};\mathcal{S}^{(p)},\mathcal{D}_n)\big)$. One can check that the corresponding blocks of $\Psi(\mathbf{X};\mathcal{S},\mathcal{D}_n)$ are orthogonal, enabling us to extend Klusowski [38]'s result and derive the following new connection between MDI and R^2 values.

Theorem 1 *Assume we have a tree structure $\mathcal{S}$ and a dataset $\mathcal{D}_n=(\mathbf{X},\mathbf{y})$, with $\mathbf{X}$ denoting the matrix of covariates and $\mathbf{y}$ denoting the response vector. For any feature X_k, we have the following identity:*

$$\frac{\mathrm{MDI}_k(\mathcal{S},\mathcal{D}_n)}{n^{-1}\sum_{i=1}^n (y_i-\bar{y})^2} = 1 - \frac{\sum_{i=1}^n \left(y_i-\hat{y}_i^{(k)}\right)^2}{\sum_{i=1}^n (y_i-\bar{y})^2} =: R^2(\mathbf{y},\hat{\mathbf{y}}^{(k)}), \tag{4}$$

where $\hat{\mathbf{y}}^{(k)} = \left(\hat{y}_1^{(k)},\hat{y}_2^{(k)},...,\hat{y}_n^{(k)}\right)$ is the vector of fitted response values when regressing $\mathbf{y}\sim\Psi(\mathbf{X};\mathcal{S}^{(k)},\mathcal{D}_n)$.[2]

[2]If $\mathcal{S}^{(k)}=\emptyset$, the fitted model is the constant intercept model, and both the MDI and the R^2 value of this regression are equal to 0.

Here, the fitted values $\hat{\mathbf{y}}^{(k)}$, or *partial model predictions*, are the resulting model predictions using only those decision stumps that split on X_k. Theorem 1 thus formalizes the intuition that a more important feature X_k leads to more accurate predictions based solely on X_k and hence a larger MDI.

We also note that the partial model predictions $\hat{\mathbf{y}}^{(k)}$ are precisely the Saabas [63] local feature importance scores for X_k. As such, Theorem 1 is implied by but also helps to clarify Proposition 1 in Li et al. [40], which equates the MDI of X_k to the sample covariance between the Saabas scores and the responses but does not derive this in a linear regression setting.

3.1 Reinterpreting MDI via OLS

Theorem 1 enables us to reinterpret the computation of MDI for a single tree in an RF via the following procedure (Figure 1, left). Given a fitted tree, which was trained on a bootstrapped dataset $\mathcal{D}_n^* = (\mathbf{X}^*, \mathbf{y}^*)$ and has tree structure $\mathcal{S} = \mathcal{S}(\mathcal{D}_n^*)$, MDI can be computed as follows.

Step 1: Obtain transformed dataset on in-bag samples. Construct the feature map $\Psi(\mathbf{x}; \mathcal{S}, \mathcal{D}_n^*)$ and use it to obtain the transformed in-bag dataset: $\Psi(\mathbf{X}^*; \mathcal{S}, \mathcal{D}_n^*) = [\Psi(\mathbf{X}^*; \mathcal{S}^{(1)}, \mathcal{D}_n^*), ..., \Psi(\mathbf{X}^*; \mathcal{S}^{(p)}, \mathcal{D}_n^*)]$. For ease of notation, we will henceforth suppress the dependence on $\mathcal{S}$ and $\mathcal{D}_n^*$ when describing the MDI procedure, and denote $\Psi^k(-) = \Psi(-; \mathcal{S}^{(k)}, \mathcal{D}_n^*)$.

Step 2: Fit linear model. Fit an ordinary least squares (OLS) model for $\mathbf{y}$ on $\Psi(\mathbf{X}^*)$, obtaining the estimated regression coefficient $\hat{\boldsymbol{\beta}}$.

Step 3: Make partial model predictions. For each $k = 1, ..., p$, obtain the partial model predictions $\hat{\mathbf{y}}^{(k)}$. Due to the orthogonality and centering of $\Psi(\mathbf{X}^*)$, $\hat{\mathbf{y}}^{(k)}$ as defined in Theorem 1 is equivalent to imputing the mean value for all stump features in $\Psi(\mathbf{X}^*)$ that do not split on X_k and then multiplying this modified matrix by $\hat{\boldsymbol{\beta}}$. Formally,

$$\hat{\mathbf{y}}^{(k)} = [\bar{\Psi}^1(\mathbf{X}^*), ..., \bar{\Psi}^{k-1}(\mathbf{X}^*), \Psi^k(\mathbf{X}^*), \bar{\Psi}^{k+1}(\mathbf{X}^*), ..., \bar{\Psi}^p(\mathbf{X}^*)]\hat{\boldsymbol{\beta}},$$

where $\bar{\Psi}^j(\mathbf{X}^*) = \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T\Psi^j(\mathbf{X}^*)$ and $\mathbf{1}_n$ is an $n \times 1$ vector of all ones.

Step 4: Evaluate predictions via R^2. For each $k = 1, ..., p$, the MDI for feature k is precisely the unnormalized R^2 value between the observed responses $\mathbf{y}$ and the partial model predictions $\hat{\mathbf{y}}^{(k)}$.

3.2 Drawbacks of MDI

Viewing MDI via this new linear regression lens allows us to explain known drawbacks as well as reveal new challenges. We consolidate these into two main issues, differential optimism bias

and model mismatch, and then discuss possible ways to overcome them.

3.2.1 Differential Optimism Bias.

For a fixed design with homoskedastic noise, it is known that the average difference between training and held-out test mean-squared error (i.e., the *optimism bias*) for linear models is twice the number of degrees of freedom scaled by the noise variance [29]. Combining Theorem 1 with a similar set of calculations leads to the following optimism bias of MDI.

Proposition 1 *Suppose $\mathcal{D}_n$ is generated according to a fixed design with fixed covariate vectors $\mathbf{x}_1,...,\mathbf{x}_n$ and responses $y_i = f(\mathbf{x}_i)+\epsilon_i$ with $\epsilon_1,...,\epsilon_n$ independent and satisfying $Var(\epsilon_i)=\sigma^2$ for $i=1,...,n$. Let $\mathcal{D}_n^0=\{(\mathbf{x}_i,f(\mathbf{x}_i))\}$ denote the noiseless dataset. Then for any fixed tree structure $\mathcal{S}$, we have*

$$\mathbb{E}\{\mathrm{MDI}_k(\mathcal{S},\mathcal{D}_n)\}=\mathrm{MDI}_k(\mathcal{S},\mathcal{D}_n^0)+\frac{\sigma^2\left|\mathcal{S}^{(k)}\right|}{n}.$$

The noiseless MDI value, $\mathrm{MDI}_k(\mathcal{S},\mathcal{D}_n^0)$, is the proportion of variance in $f(\mathbf{x})$ explained by $\Psi(\mathbf{x};\mathcal{S}^{(k)},\mathcal{D}_n)$. Proposition 1 reveals that $\mathrm{MDI}(k;\mathcal{S},\mathcal{D}_n)$ is optimistic for $\mathrm{MDI}(k;\mathcal{S},\mathcal{D}_n^0)$ and further that the optimism bias is larger for features X_k that have more splits, leading to a *differential optimism bias*. This differential optimism is problematic because the split frequency of a feature X_k reflects not only the feature's inherent influence on the response, but also some biases due to the tree-growing process of CART. In particular, CART's splitting rule results in highly-correlated and low-entropy features receiving fewer splits (see Section 6). As a result, signal features with these characteristics often have lower MDI than non-signal features without these characteristics.

Though Proposition 1 requires a fixed tree structure, when $\mathcal{S}$ is random, the result above can be applied conditionally on $\mathcal{S}$, as long as it is independent of $\mathcal{D}_n$. This independence can be achieved if OOB samples are used to compute MDI, as is sometimes done in practice. However, Proposition 1 shows that this does not resolve the differential optimism bias. More commonly, in-bag samples are used to both generate $\mathcal{S}$ and compute MDI. In this case, Proposition 1 does not apply directly, but because the splits $\mathcal{S}$ tend to be correlated with the noise, the amount of overfitting (i.e., differential optimism bias) should be worse.

Mitigating Differential Optimism Bias. To remedy the differential optimism bias, we can (1) replace OLS (i.e., step 2 of MDI procedure) with a regularized regression algorithm such as ridge

[30] or LASSO [74] to reduce the effective degrees of freedom (i.e., number of splits) and stabilize the feature importance calculation. (2) Traditionally, MDI is evaluated on the same data (i.e., in-bag) used to train the tree, thereby amplifying biases in the learned model. To overcome this issue, Li et al. [40] proposed a sample-splitting technique, known as MDI-oob, which computes the covariance between OOB partial model predictions $\hat{\mathbf{y}}^{(k)}$ and OOB responses $\mathbf{y}$. This is equivalent to computing the R^2 value for the fitted linear model (trained on in-bag samples) when evaluated on OOB samples. That is, sample-splitting approaches mitigate differential optimism bias by using one portion of the data (e.g., in-bag data) to learn the linear model, and the other (OOB data) to evaluate its generalization error. However, this comes at the cost of using fewer samples to *both* fit OLS and estimate its out-of-sample error. To more efficiently use samples, one can instead use a leave-one-out (LOO) scheme that leverages the entire dataset (in-bag and OOB samples) to both learn the linear model and estimate its out-of-sample error.

3.2.2 Model Mismatch

While RF is a nonparametric method that can in principle approximate any functional structure, it comes with a set of inductive biases that allow it to adapt better to some types of structure rather than others. First, trees are statistically inefficient at estimating smoothly-varying functions [76] and additive generative models [71], which are ubiquitous in real-world datasets [28]. This inefficiency results from the local decision stump features being poorly adapted to fit smooth or additive functions. Specifically, their piecewise constant nature makes them inefficient at representing smooth relationships, while their locality makes them inefficient at representing additive structure. Second, our reinterpretation shows MDI implicitly relies on OLS and R^2 to measure feature importance. However, OLS and R^2 may not be well-suited for all problem structures (e.g., in situations with a categorical response or gross outliers).

Mitigating Model Mismatch. To alleviate the inductive biases of RFs against additive or smooth structures, we can perform more flexible feature engineering by augmenting the k^{th} block of local decision stump features $\Psi^{(k)}(\mathbf{X})$ with other engineered features based on X_k. In particular, simply augmenting X_k itself can help bridge the gap between RFs and linear models. Secondly, we can replace OLS and the R^2 metric with prediction models and metrics that may be better suited for the given problem and data structure. For example, logistic regression with negative log-loss or Huber regression with negative Huber loss might be more appropriate for classification

problems or problems with outliers, respectively.

4 Introducing MDI+

In this section, we build upon the ideas introduced in the previous section to develop our feature importance framework, MDI+.

4.1 The MDI+ Framework

MDI+ improves upon MDI by directly addressing its drawbacks described in the previous section and by offering practitioners a flexible and adaptable framework for computing feature importances. In particular, within this framework, practitioners are allowed several choices (i.e., the choice of feature augmentation, GLM, and similarity metric) that can be tailored to the data or problem structure. We discuss how these choices can be made in a principled manner in more detail in Section 4.2 and proceed now to introduce the overarching MDI+ framework for a given choice of feature augmentation, GLM, and similarity metric.

As a reminder of relevant notation, the original dataset is denoted by $\mathcal{D}_n = (\mathbf{X}, \mathbf{y})$, and each tree is fitted on a bootstrapped dataset $\mathcal{D}_n^* = (\mathbf{X}^*, \mathbf{y}^*)$. Also, recall we denote the tree structure as $\mathcal{S} = \mathcal{S}(\mathcal{D}_n^*)$, and the feature map as $\Psi(\mathbf{x}; \mathcal{S}, \mathcal{D}_n^*)$. We next introduce MDI+ for a single tree in an RF (Figure 1, right) using an analogous scaffolding as MDI (see Section 3.1) but with the main differences highlighted in bold.

Step 1: Obtain ***augmented*** *transformed dataset on* ***in- and out-of-bag*** *samples.* For features $k = 1,...,p$, construct an augmented feature map $\tilde{\Psi}(\mathbf{x}; \mathcal{S}^{(k)}, \mathcal{D}_n^*)$ by appending the raw feature X_k to the feature map $\Psi(\mathbf{x}; \mathcal{S}^{(k)}, \mathcal{D}_n^*)$.[3] That is, let $\tilde{\Psi}(\mathbf{x}; \mathcal{S}^{(k)}, \mathcal{D}_n^*) = [\Psi(\mathbf{x}; \mathcal{S}^{(k)}, \mathcal{D}_n^*), x_k])$ if $\mathcal{S}^{(k)} \neq \emptyset$. Denote $\tilde{\Psi}(\mathbf{x}; \mathcal{S}, \mathcal{D}_n^*) = \Big(\tilde{\Psi}(\mathbf{x}; \mathcal{S}^{(1)}, \mathcal{D}_n^*), ..., \tilde{\Psi}(\mathbf{x}; \mathcal{S}^{(p)}, \mathcal{D}_n^*)\Big)$, and apply $\tilde{\Psi}(\mathbf{x}; \mathcal{S}, \mathcal{D}_n^*)$ to the entire dataset (in- and out-of-bag samples) to obtain $\tilde{\Psi}(\mathbf{X}; \mathcal{S}, \mathcal{D}_n^*)$. Henceforth, we denote $\tilde{\Psi}(\mathbf{X}) = \tilde{\Psi}(\mathbf{X}; \mathcal{S}, \mathcal{D}_n^*)$ and $\tilde{\Psi}^{(k)}(\mathbf{X}) = \tilde{\Psi}(\mathbf{X}; \mathcal{S}^{(k)}; \mathcal{D}_n^*)$.

Step 2: Fit ***regularized generalized*** *linear model (GLM).* Fit a regularized GLM $\mathcal{M}$ with link function g and penalty parameter λ for $\mathbf{y}$ using $\tilde{\Psi}(\mathbf{X})$ to obtain the estimated regression coefficients $\hat{\boldsymbol{\beta}}_\lambda$ and intercept $\hat{\alpha}_\lambda$. We tune λ via the approximate leave-one-out (LOO) method described in Rad

[3] We append the raw feature X_k as a default choice. More generally, we can augment the feature map with any engineered feature derived from X_k.

and Maleki [60], which does not require re-fitting the GLM n times. Note that we use the full dataset $(\mathbf{X}, \mathbf{y})$ comprising both in- and out-of-bag samples to perform this as well as subsequent steps.

Step 3: Make partial model prediction via ***LOO****.* For a feature X_k, impute the mean value for all features in $\tilde{\Psi}(\mathbf{X})$ not derived from X_k. Then, for each feature $k=1,...,p$, and each sample $i=1,...,n$, obtain the LOO partial model predictions:

$$\hat{y}_i^k = g^{-1}\Big(\Big[\bar{\tilde{\Psi}}^1(\mathbf{X}),...,\bar{\tilde{\Psi}}^{k-1}(\mathbf{X}),\tilde{\Psi}^k(\mathbf{X}),\bar{\tilde{\Psi}}^{k+1}(\mathbf{X}),...,\bar{\tilde{\Psi}}^p(\mathbf{X})\Big]\hat{\boldsymbol{\beta}}_{-i,\lambda}+\alpha_\lambda\Big), \tag{5}$$

where $\bar{\tilde{\Psi}}^j(\mathbf{X}) = \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T\tilde{\Psi}^j(\mathbf{X})$, $\mathbf{1}_n$ is an $n\times 1$ vector of all ones, and $\hat{\boldsymbol{\beta}}_{-i,\lambda}$ is the LOO coefficient vector learned without using sample $\mathbf{x}_i$. Again, we use the approximate LOO method of Rad and Maleki [60] to obtain $\hat{\boldsymbol{\beta}}_{-i,\lambda}$ without refitting the GLM. Denote the LOO partial model predictions for feature k by $\hat{\mathbf{y}}^{(k)} = \big(\hat{y}_1^k,..., \hat{y}_n^k\big)$.

Step 4: Evaluate predictions via ***similarity metric****.* Pick a similarity metric m and use it to evaluate the similarity[4] between the true responses and the partial model predictions. That is, for each $k=1,...,p$, we define the MDI+ for feature k as

$$\text{MDI}_k^+(\mathcal{S},\mathcal{D}_n^*,\tilde{\Psi},\mathcal{M},m) \coloneqq m\big(\mathbf{y},\hat{\mathbf{y}}^{(k)}\big), \tag{6}$$

where $\tilde{\Psi}$, $\mathcal{M}$ and m represent the choices of augmented feature representation, GLM, and similarity metric. For an RF, the MDI+ of a feature k can be computed by taking the average across the trees.[5]

RF+. As a by-product of MDI+, the GLM $\mathcal{M}$ fitted on the augmented transformed dataset $\tilde{\Psi}(\mathbf{X})$ serves as a new class of prediction models, which we refer to as RF+. RF+ is a generalization of RFs that provides a natural link between classical statistical models (i.e., GLMs) and modern ML (i.e., RFs). In Section 7 and Appendix G, we show that RF+ improves upon the prediction performance of RFs by approximately 5% across a variety of real-world datasets in the regression and classification settings. Though the main focus of this work is feature importance, the strong prediction performance of RF+ suggests that it better captures the underlying data-generating process, giving additional credence to MDI+ [49, 80].

4.2 PCS-Informed Model Recommendation

One rarely has sufficient prior knowledge of the problem structure in order to make definitive choices for $\tilde{\Psi}$, $\mathcal{M}$ and m. Further, even if there is prior information, a practitioner might be

[4]The metric should attain larger values on closer arguments.

[5]If a feature is never split on amongst all trees in the RF, we set its MDI+ to be $-\infty$.

Figure 1: Overview of MDI+ for a single tree. For each tree $\mathcal{S}$ in the random forest, **Step 1:** Obtain the transformed dataset on the in- and out-of-bag samples using stumps from the tree and append the raw and/or any additional (possibly engineered) features. **Step 2:** Fit a regularized GLM. **Step 3:** Using the fitted GLM, make partial model predictions $\hat{\mathbf{y}}^{(k)}$ for each feature $k = 1,...,p$ (stacked boxes) using a leave-one-out (LOO) data splitting scheme. **Step 4:** For each $k = 1,...,p$, evaluate partial model predictions via any user-defined similarity metric to obtain the MDI+ for feature k in tree $\mathcal{S}$.

left with multiple choices (e.g., using lasso or ridge for $\mathcal{M}$). To address this challenge, we briefly discuss two approaches inspired by the PCS framework [80] to help practitioners make these modeling choices in a data-driven manner, and defer details to Appendix F.

Stability-based selection for $\tilde{\Psi}$, $\mathcal{M}$, m. Let $h = \left(\tilde{\Psi}(\mathbf{X}), \mathcal{M}, m\right)$ denote an MDI+ model defined by the choices of augmented feature representation, GLM, and similarity metric, respectively. Accordingly, let $\mathcal{H} = \{h_1,...,h_N\}$ denote the set of possible MDI+ models under consideration.

Step 1: Prediction Screening. For each $h \in \mathcal{H}$, evaluate the prediction performance of $\mathcal{M}$ on a

held-out test set, and filter out all h whose prediction performance is worse than that of RF. This prediction check ensures that the interpreted model is a reasonably faithful approximation of the underlying data generating process [49, 80].

Step 2: Stability Selection. Generate B bootstrapped samples of the fitted trees in the ensemble. For each h that passed the prediction screening, evaluate MDI+ for each of the B bootstrapped samples, and choose the h which yields the most similar (or stable) feature rankings across the B bootstrapped samples. We measure the similarity between different bootstrap samples via Rank-based Overlap [79]. While we measure stability via bootstrap sampling, one can also measure stability over different algorithmic perturbations (e.g., random seeds used to train the RF).

Model aggregation via PCS principles. Instead of choosing a single MDI+ model, it may be sensible to compare and ensemble multiple models that have similar predictive performance (e.g., those that passed the prediction screening described above). This is because it can be argued that these models all provide competing descriptions of reality since they have approximately equal statistical evidence [49, 62]. Further, investigating and comparing the feature rankings of different predictive-screened MDI+ models might also be of independent scientific interest.

In Appendix F, we perform a preliminary simulation study to establish the efficacy of both approaches. In particular, we show that the GLM and metric selected by the stability-based procedure leads to the best feature ranking performance across different candidate MDI+ models. While the model aggregation approach does not lead to the best feature ranking performance, it performs competitively, and we present it here since both approaches might be useful in practice. However, we leave a more thorough investigation of these model recommendation techniques to future work.

5 Data-Inspired Feature Ranking Simulations

In this section, we describe a number of simulations that illustrate the effectiveness of MDI+ to identify signal features in three common statistical settings: regression, classification, and robust regression (i.e., presence of outliers). In all settings, we highlight the flexibility of MDI+ for choosing an appropriate GLM and similarity metric for the problem at hand. In addition, for all simulations, we use covariate matrices coming from real-world datasets to capture naturally occurring structures. Using these covariate matrices, we simulate responses with analytical functions that depend on a sparse

set of features. These response functions were chosen to reflect both canonical forms and domain knowledge. We then measure the ability of MDI+ and other feature importance methods to recover these signal features, with the specific metric for recovery discussed below. Additional simulations under varying sparsity levels, number of features, and misspecified model settings with omitted variables are provided in Appendix C, further supporting the strong empirical performance of MDI+.

5.1 Simulation Setup

Real-world datasets used. For our covariate matrices, we use the following datasets: (i) Juvenile dataset ($n = 3640$, $p = 277$) [56]; (ii) a subset of the Cancer Cell Line Encyclopedia (CCLE) RNASeq gene expression dataset ($n = 472$, $p = 1000$) [4]; (iii) Enhancer dataset ($n = 7809$, $p = 41$) [5]; and (iv) Splicing dataset ($n = 23823$, $p = 264$) [5].

Simulated responses. Using each dataset above as the covariate matrix $\mathbf{X}$, we consider the following response functions:

1. Linear model: $\mathbb{E}[Y \mid X] = \sum_{j=1}^{5} X_j$;
2. Locally-spiky-sparse (LSS) model [7]: $\mathbb{E}[Y \mid X] = \sum_{m=1}^{3} \mathbf{1}(X_{2m-1} > 0)\mathbf{1}(X_{2m} > 0)$;
3. Polynomial interaction model: $\mathbb{E}[Y \mid X] = \sum_{m=1}^{3} X_{2m-1} + \sum_{m=1}^{3} X_{2m-1}X_{2m}$;
4. Linear + LSS model: $\mathbb{E}[Y \mid X] = \sum_{m=1}^{3} X_{2m-1} + \sum_{m=1}^{3} \mathbf{1}(X_{2m-1} > 0)\mathbf{1}(X_{2m} > 0)$.

These regression functions were chosen to reflect several archetypes of real DGPs. (1) and (3) reflect well-studied models in the statistics literature. (2) exhibits the discontinuous interactions observed in biological processes [50, 7]. (4) reflects a combination of linear and discontinuous interactions also thought to be prevalent in genomics. For our classification simulations, we pass the mean response function (i.e., $\mathbb{E}[Y \mid X]$) through a logistic link function to generate binary responses.

Feature importance methods under consideration. We compare MDI+ to a number of popular feature importance methods for RFs: MDI [11], MDI-oob [40], MDA [10], and TreeSHAP [48].

RF settings. For the regression (Section 5.2) and robust regression (Section 5.4) settings, we train an RF regressor using `scikit-learn` [59] with *n_estimators*=100 (i.e., number of trees), *max_features*=$p/3$ (i.e., proportion of features subsampled at each node), and *min_samples_leaf*=5

alongside other default parameters. In the classification setting (Section 5.4), we use `scikit-learn`'s RF classifier with *n_estimators*=100, *max_features*=$\sqrt{p}$, and *min_samples_leaf*=1 alongside other default parameters.

Feature Ranking Performance Metric. As in previous work [40, 80], we evaluate the performance of each feature importance method by how well it can be used to classify features used in the regression function (signal) vs. those that are not (non-signal). Each set of feature importance scores induces a ranking of the features, which can then be evaluated for this classification problem using AUROC. A high AUROC scores indicates that the signal features are ranked higher (i.e., more important) than the non-signal features. Performance results are averaged across 50 simulation replicates. In each simulation replicate, we choose the signal features randomly from $\mathbf{X}$.

5.2 Regression Simulations

We simulate responses as discussed above, but introduce additive Gaussian noise: $Y = \mathbb{E}[Y \mid X] + \epsilon$, where $\epsilon \sim N(0, \sigma^2)$. We examine performance across various signal-to-noise ratios as measured by the proportion of variance explained,[6] defined as $PVE = \text{Var}\{\mathbb{E}\{Y|X\}\}/\text{Var}\{Y\}$. Specifically, we vary across PVE in $\{0.1, 0.2, 0.4, 0.8\}$ (or equivalently, signal-to-noise ratio in $\{0.11, 0.25, 0.66, 4\}$) and across a range of sample sizes n (i.e., $n \in \{100, 250, 472\}$ for CCLE and $n \in \{100, 250, 500, 1000, 1500\}$ for the other three datasets). For MDI+, we use l_2-regularized regression (i.e., ridge regression) as the GLM, and R^2 as our similarity metric of choice.

We display results for the Splicing dataset for the LSS and polynomial interaction model in Figure 2. Results for other datasets and regression functions are deferred to Appendix B.2, but are similar to those displayed in Figure 2. Across all simulation scenarios, MDI+ produces more accurate feature rankings, often enjoying more than a 10% improvement in AUROC over its closest competitor (typically MDI-oob or MDA). In particular, MDI+ produces the most significant improvement for low PVEs, which is especially important since real-world data in fields such as biology, medicine, and social sciences have low SNRs. The strong performance of MDI+ relies on a number of choices made to mitigate the aforementioned biases such as: using l_2 regularization, including the raw feature, and evaluating predictions via LOO. In Appendix D, we provide simulations that

[6] PVE is a monotone transformation of the signal-to-noise ratio such that its range is bounded between 0 and 1 and thus more interpretable. It is also a standard measurement of noise used in many fields (e.g., it is often called *heritability* in genomics). PVE in genomics is estimated to range from between 0.05 to 0.4 [78].

show how each of these choices individually leads to an increase in the ranking accuracy of MDI+.

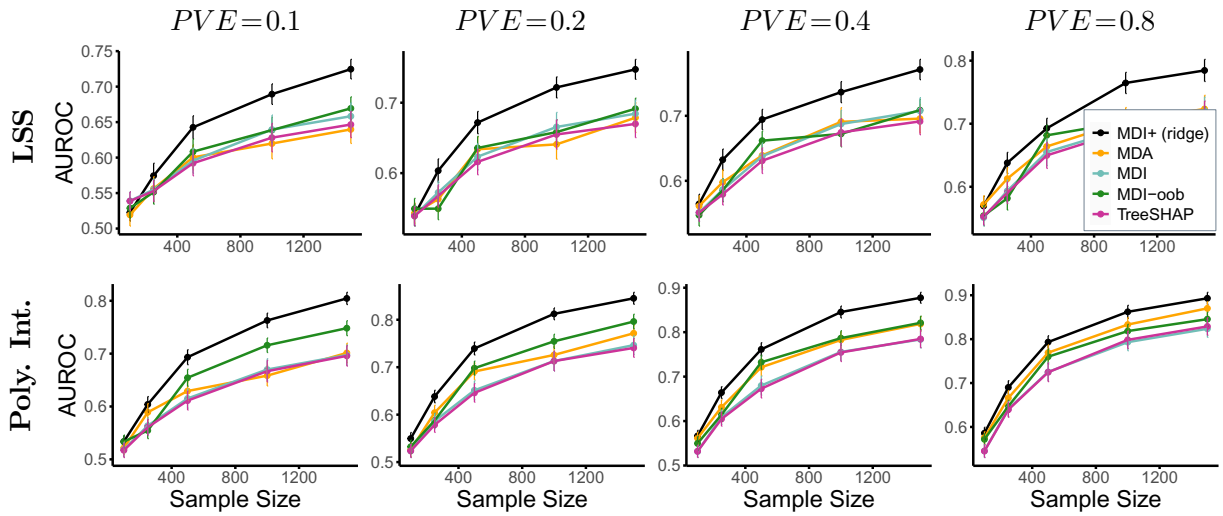


Figure 2: MDI+ outperforms all other feature importance methods for the data-inspired regression simulations described in Section 5.2 using the Splicing dataset. This pattern is evident across various regression functions (specified by row), proportions of variance explained (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

5.3 Classification Simulations

We simulate responses according to the response functions defined in Section 5.1, and introduce noise by randomly flipping a percentage of the binary response labels to the opposite class. We measure AUROC as we vary the percentage of corrupted labels in {0%,5%,15%,25%} and the number of samples as before. To tailor MDI+ to the classification setting, we use l_2-regularized logistic regression and negative log-loss as our choice of GLM and similarity metric, respectively. We compare these choices to those used in the regression setting (i.e., ridge regression and R^2). Henceforth, we shall refer to these particular settings as MDI+ (logistic) and MDI+ (ridge), respectively.

We display results for the CCLE dataset for the linear and linear + LSS model in Figure 3. We defer results for other datasets and regression functions to Appendix B.3, but are similar to those displayed in Figure 2. Both MDI+ (logistic) and MDI+ (ridge) outperform competitors by over 10% across most simulation scenarios. Further, we see that MDI+ (logistic) often improves upon MDI+ (ridge), indicating the benefit of tailoring GLMs and metric to the statistical problem at hand. Exploring other GLMs and similarity metrics may further improve performance.

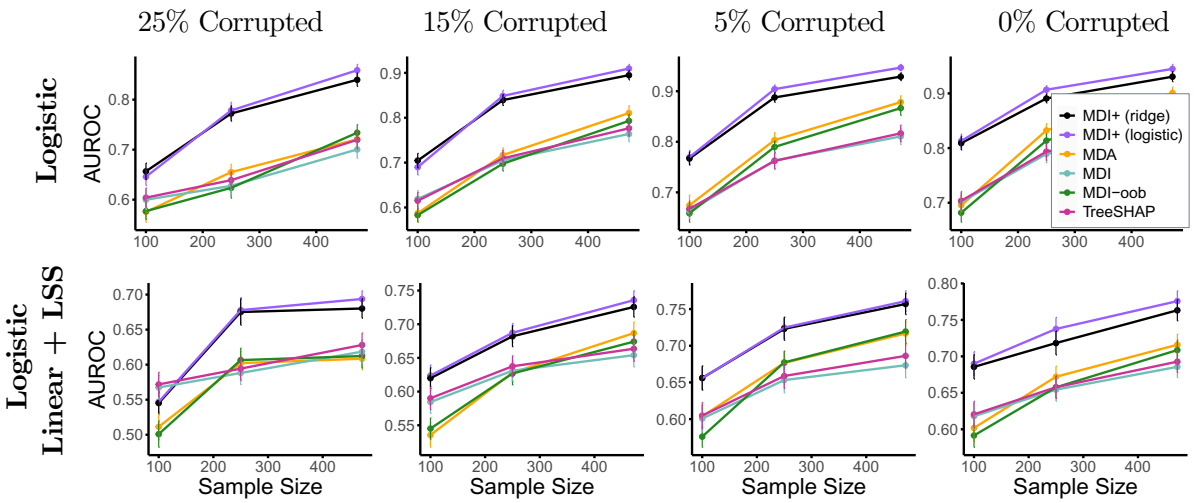


Figure 3: Both MDI+ (ridge) and MDI+ (logistic) outperform all other feature importance methods for the data-inspired classification simulations described in Section 5.3 using the CCLE RNASeq dataset. Furthermore, MDI+ (logistic) slightly outperforms MDI+ (ridge) in the majority of settings, indicating the benefit of tailoring the choices of MDI+ to the data at hand. This pattern is evident across various regression functions (specified by row), proportions of corrupted labels (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

5.4 Robust Regression Simulations

We illustrate another use-case of our framework by showing how MDI+ can be tailored to account for the presence of outliers. We simulate responses as done in the regression setting, and introduce outliers as follows. We first select samples in the top and bottom $q/2\%$ quantiles for a randomly chosen *non-signal feature*. For these selected samples, we corrupt their responses by drawing them from $N(\mu_{corrupt},1)$ and $N(-\mu_{corrupt},1)$ for the bottom and top quantile samples, respectively. In our simulations, we vary across q in $\{0,1,2.5,5\}$ and $\mu_{corrupt}$ in $\{10,25\}$ for a variety of sample sizes n. We tailor MDI+ to this setting by using a robust version of ridge regression [57] as our choice of regularized GLM, and negative Huber loss as our similarity metric. We compare these choices to those used in the regression setting (ridge regression and R^2). We shall refer to these particular settings as MDI+ (Huber) and MDI+ (ridge).

Simulation results for the Enhancer dataset with responses simulated via the LSS function are shown in Figure 4. Results for other datasets and regression functions are shown in Appendix B.4. We observe that MDI+ (Huber)'s performance is more robust than competing methods including MDI+ (ridge) and often remains accurate as both $\mu_{corrupt}$ and the percentage of outliers increase.

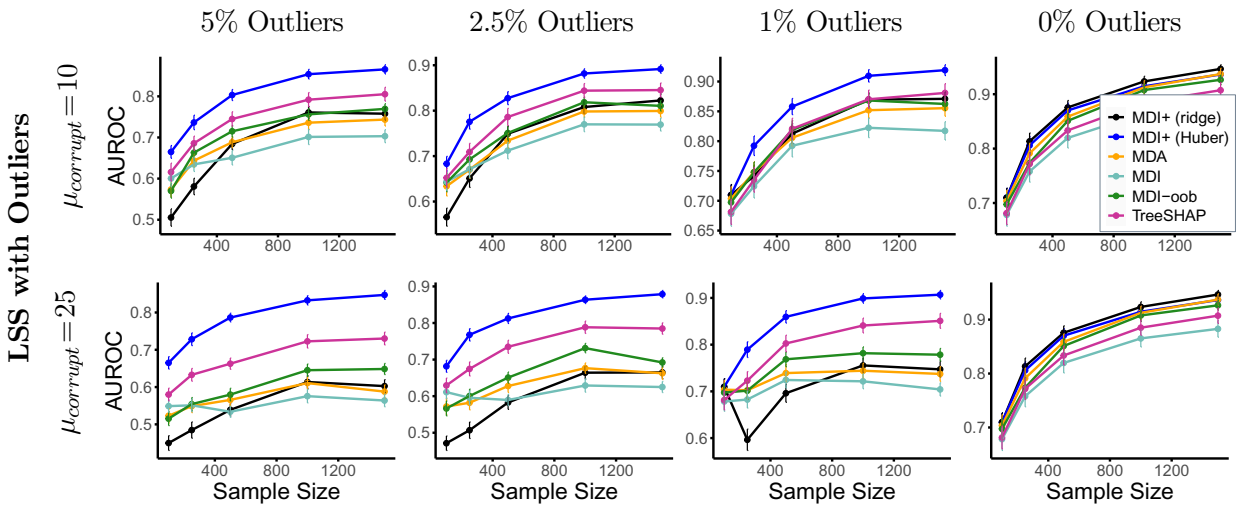


Figure 4: Under the LSS with outliers regression setting using the Enhancer dataset (described in Section 5.4), MDI+ (Huber)'s performance remains suffers far less than other methods including MDI+ (Ridge) as the level of corruption $\mu_{corrupt}$ (specified by row) and the proportion of outliers (specified by column) grow. This pattern also holds across sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

6 MDI+ Overcomes Biases of MDI

In this section, we run simulations to show that MDI+ overcomes the biases of MDI against highly-correlated, and low-entropy features.

6.1 Correlated Feature Bias

Experimental details. We draw $\mathbf{x}_i \sim N(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{x}_i \in \mathbb{R}^{100}$ and $\mathbf{\Sigma} \in \mathbb{R}^{100 \times 100}$ has the following block-covariance structure: Features $X_1,...,X_{50}$ have pairwise correlation ρ, and features $X_{51},...,X_{100}$ are independent of all other features. We then simulate responses from the linear+LSS model (see Section 5.1) with PVE in $\{0.1,0.4\}$. Denote the group of signal features (i.e., $X_1,...,X_6$) as Sig, the non-signal features that have non-zero correlation with the signal features (i.e., $X_7,...,X_{50}$) as C-NSig, and the non-signal uncorrelated features (i.e., $X_{51},...,X_{100}$) as NSig. We generate $n = 250$ samples, and vary the correlation across ρ in $\{0.5,0.6,0.7,0.8,0.9,0.99\}$. For all feature importance methods used in our simulation study in Section 5, we compute the average rank of features within each group (Sig, C-NSig, and NSig).

Results. The results for the average ranks of features per group are shown in Figure 5. Across all levels of correlation and PVEs, MDI+, MDI-oob, and MDA rank the true signal features (Sig, dark

green) as the most important feature group and the uncorrelated non-signal group (NSig, red) as the least important feature group by a sizeable margin. MDI's behavior can be explained by Proposition 1 and Figure 30, which displays the average percentage of RF splits per feature in each group. As ρ increases, Figure 30 shows the percentage of splits on NSig features increases, while that of Sig and C-NSig decreases. Intuitively, this occurs because Sig and C-NSig features, being correlated with each other, result in similar decision stump functions and must compete over splits. Since MDI grows with the number of splits as established in Proposition 1, this leads to an overestimate of MDI for the NSig features. Moreover, MDI and TreeSHAP are measured on in-bag (training) samples, thereby amplifying biases that are learned during the tree construction. In contrast, MDI+, MDI-oob, and MDA use sample-splitting, which helps to mitigate this overfitting issue. A direct comparison between MDI+ with and without LOO sample splitting helps to confirm this intuition (see Figure 29).

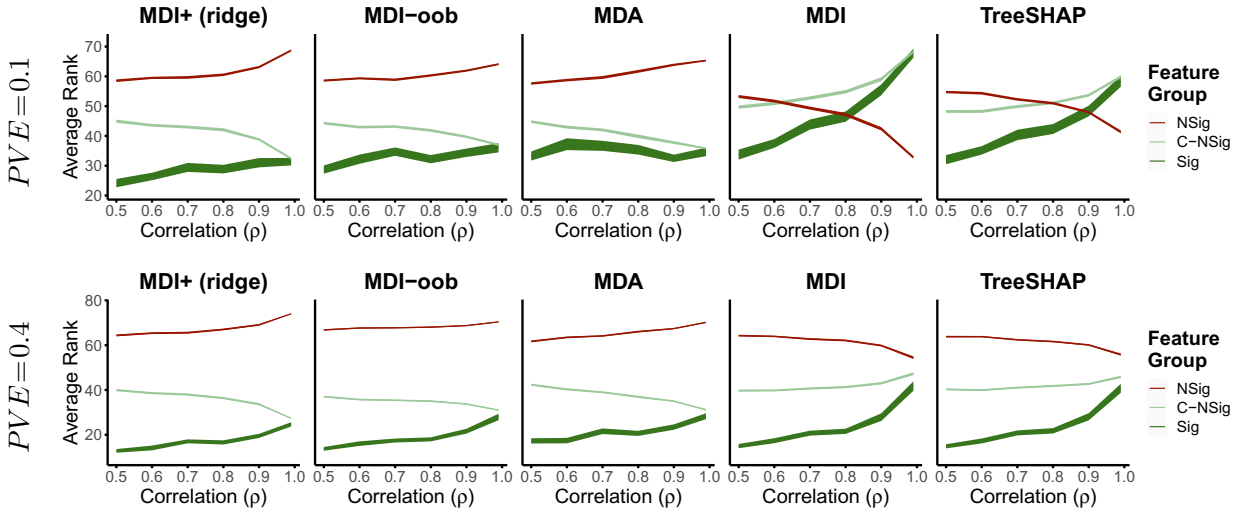


Figure 5: MDI+ improves upon the bias that MDI has against selecting features that are highly correlated. We show the average ranks ($\pm 1SE$) within feature groups (Sig, C-NSig, NSig) for various correlation (ρ) levels over 50 replicates. When the signal is moderate-to-high ($PVE = 0.4$, bottom), all methods rank the true signal features (Sig, dark green) as more important than the non-signal features (NSig, red) for all ρ; however, the gap is small for MDI and TreeSHAP. When the signal is low ($PVE = 0.1$, top), MDI+ , MDI-oob, and MDA are still able to identify the true signal features (Sig, dark green) as most important. In contrast, MDI and TreeSHAP rank the non-signal features (NSig, red) as more important than the true signal (Sig, dark green) and the correlated, non-signal (C-NSig, light green) feature groups when the ρ is large.

6.2 Entropy Bias

Experimental details. Following the simulation setup proposed in Li et al. [40], we sample 5 features: X_1 from a Bernoulli distribution with $p = 0.5$, X_2 from a standard Gaussian distribution,

and X_3, X_4, and X_5 from a uniform discrete distribution with 4, 10, and 20 categories, respectively. To investigate entropy bias, we simulate the response as a function of only the lowest entropy feature, X_1, under (1) the regression setting via $Y = X_1 + N(0, \sigma^2)$ where σ^2 is chosen to achieve $PVE = 0.1$ and (2) the binary classification setting via $\mathbb{P}(Y = 1 \mid X) = \frac{1+X_1}{3}$. We vary the number of samples $n \in \{50, 100, 250, 500, 1000\}$ and measure the rank for each feature across 50 replicates.

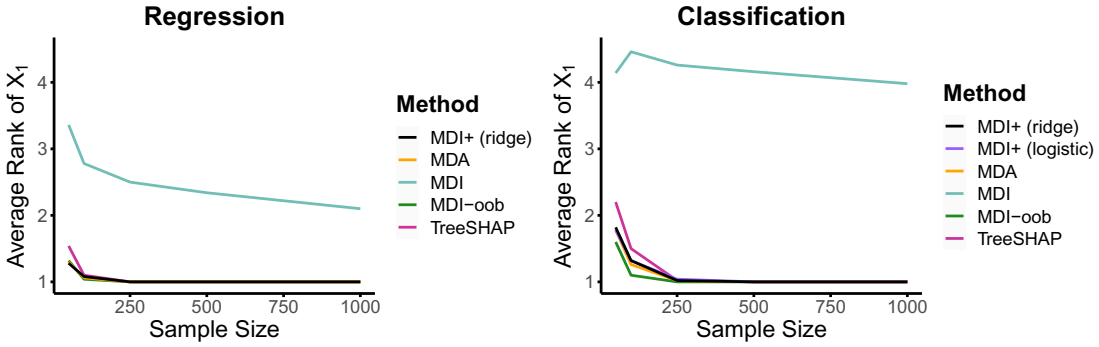


Figure 6: MDI+ improves upon the bias that MDI has against selecting features with lower entropy in both the regression (left) and classification (right) simulation settings described in Section 6.2. The feature ranking of the solo signal feature, X_1, averaged across 50 replicates, is shown on the y-axis. Here, a lower value indicates greater importance.

Results. Figure 6 displays the average rank of X_1, the solo signal feature, for each feature importance method. Here, only MDI is unable to identify X_1 as the signal feature. This occurs because X_1 has the lowest entropy relative to other features, and is hence split upon the least (see Figure 30). As a result of Proposition 1, this leads to an overestimate of MDI for high-entropy features $X_2, \ldots, X_5$ that are split upon more often. To combat this, MDI+ employs regularization to control the effective degrees of freedom and sample-splitting to mitigate biases learned during the tree construction. A direct comparison of MDI+ with and without ridge regularization and LOO sample-splitting in Figure 31 helps confirm this.

7 Case Studies

In this section, we revisit our two driving case studies and use MDI+ as well as other feature importance measures to identify important features in predicting drug responses and breast cancer subtypes. In both case studies, we show that RF+ increases the prediction accuracy over RF and that the feature rankings from MDI+ agree with established domain knowledge with significantly greater stability than other feature importance measures across different perturbations (i.e., across

different train-test splits and random seeds used to train the RF). Given that both predictability and stability are important prerequisites for interpretability [49, 80], these findings showcase the effectiveness of MDI+ for extracting reliable interpretations in real-world scientific problems.

7.1 Case Study I: Drug Response Prediction

Accurately predicting a cancer drug's efficacy for a patient before prescribing it can tremendously improve both the patient's health and financial well-being, given the exorbitant costs of many cancer drugs. Moreover, identifying the important biological predictors, such as genes, that are influential of the drug response can provide valuable insights into potential targets and novel candidates for future preclinical research. Towards this end, we leverage data from the Cancer Cell Line Encyclopedia (CCLE) [4] to build accurate drug response models and identify genes whose expression levels are highly predictive of a drug's response.

Data and Methods. We use gene expression data $\mathbf{X} \in \mathbb{R}^{472 \times 5000}$, measured via RNASeq, from $n = 472$ cell lines and $p = 5000$ genes after filtering. For each cell line, the response of 24 different drugs was measured, yielding a multivariate response matrix $\mathbf{Y} \in \mathbb{R}^{472 \times 24}$. We split the data into 80% training and 20% test. Details regarding the data and preprocessing can be found in Appendix H.1. Using the training data, we fit 24 separate RFs, one to predict the response for each drug using the gene expression data $\mathbf{X}$, and investigate the gene importance rankings for each drug. We repeat this for 32 different train-test splits. Here, the RF settings and feature importance methods used are the same as those in Section 5.1 for regression.

Prediction accuracy. For 23 out of the 24 drugs, the test R^2, averaged across the 32 train-test splits, was higher for RF+ than RF. Furthermore, RF+ improved the test R^2 by an average of 4% across the 18 drugs with RF test $R^2 > 0.1$. We use this threshold of $R^2 > 0.1$ to focus on models that have non-trivial predictive power; however, the improvement from RF+ is even higher without this filter. Details and the full prediction results can be found in Appendix G.

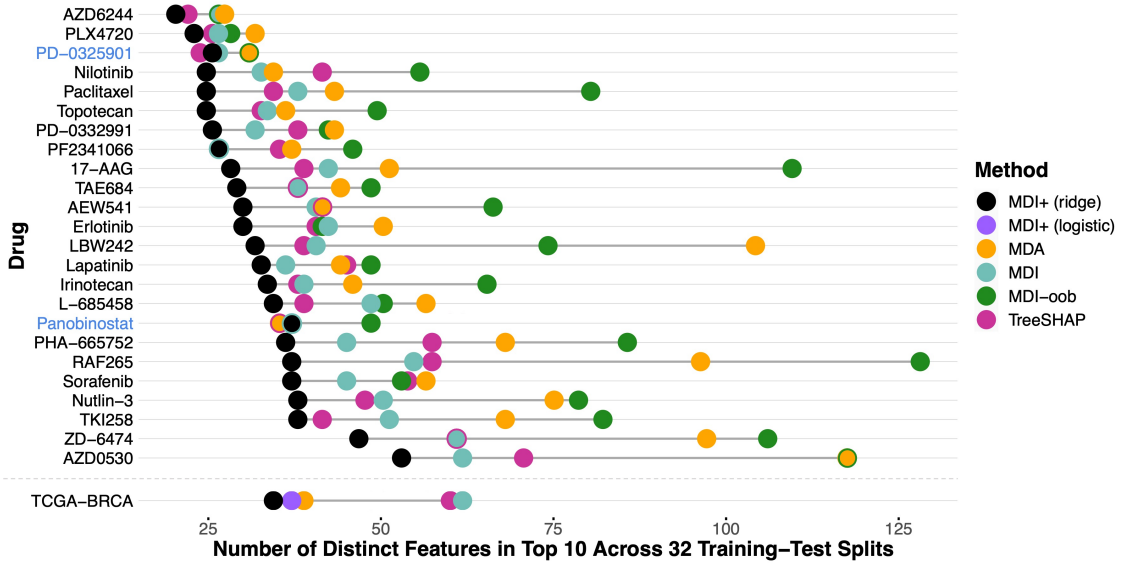
Accuracy of gene importance rankings. MDI+ was able to identify several well-known genes that have been previously identified as drug targets in the previous literature. In particular, every known gene expression predictor of drug response identified in the original work on CCLE [4], except for one, was ranked in the top 5 by MDI+ for their respective drugs. The one gene outside the top 5 was *MDM2*, a known predictor for the drug Nutlin-3 [4]. Still, MDI+ ranked *MDM2*

17^{th}, which is higher than the rankings given by its competitors (MDI-oob: 22^{nd}, TreeSHAP: 30^{th}, MDI: 35^{th}, MDA: 53^{rd}). A full list of the top 5 genes for each drug using various feature importance methods is provided in Appendix H.1.

To provide another assessment of the gene importances beyond evidence from the scientific literature, we also evaluated the predictive power of the top 10 genes from each feature importance method. We found that the top 10 genes from MDI+ were often more predictive than that from other feature importance methods. Per the predictability principle of the PCS framework, strong prediction performance suggests that the model (and in this case, the top-ranked features) may better capture the underlying data-generating process. For brevity, we defer details and additional discussion regarding this prediction analysis to Appendix H.1.

Stability of gene importance rankings. Lastly, for each feature importance method, we investigated the stability (or similarity) of gene importance rankings across the 32 train-test splits. Methods that exhibit greater stability are highly advantageous in practice, since it is undesirable for interpretations to change due to arbitrary choices like train-test splits and random seeds. In Figure 7a, we display one measure of stability, namely, the number of distinct genes ranked in the top 10 across the 32 train-test splits. For 22 out of the 24 drugs (the exceptions being PD-0325901 and Panobinostat), MDI+ had the fewest number of distinct genes that were ranked in the top 10 across the 32 train-test splits. Moreover, for all 24 drugs, we found that MDI+ had the highest number of genes that were always ranked in the top 10 across all train-test splits. Interestingly, sample-splitting techniques such as MDA and MDI-oob were significantly less stable than other methods, highlighting the downside of using fewer samples to measure feature importances. On the other hand, MDI+, which leverages LOO and regularization, overcomes this drawback. To further establish the stability of MDI+, we also examined the distribution of feature rankings of each method for the five most important genes across the 32 splits. As visualized in Figure 7b (results for all drugs in Figures 35 and 36), feature rankings of MDI+ tend to have smaller variance as compared to other methods. This remains true for PD-0325901 and Panobinostat, for which MDI+ did not perform optimally according to the aforementioned top 10 stability metric. A similar improvement in stability is also seen when fixing the training data and only altering the random seed used for fitting the RF (see Appendix H.1). This improved stability of MDI+, in addition to the increase in prediction performance from RF+, demonstrate the practical advantages of MDI+ in this real-data case study on drug response prediction.

(a) Stability of Features Ranked in Top 10



(b) Stability of Top 5 Ranked Features

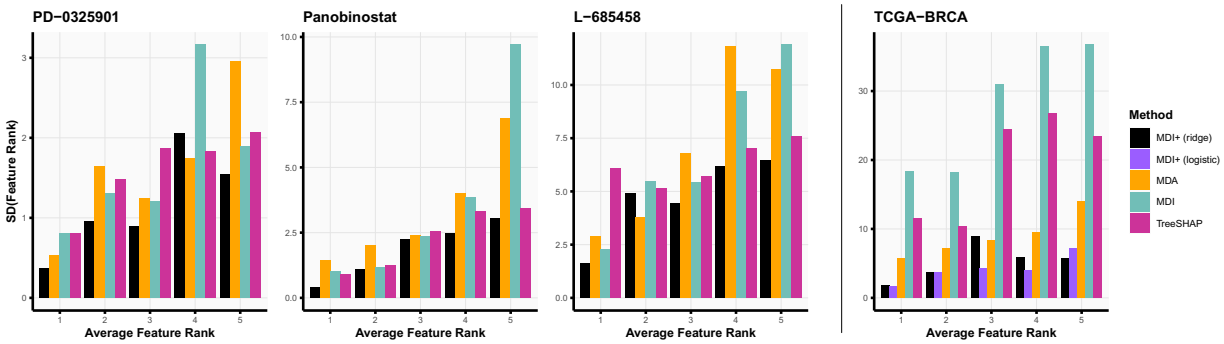


Figure 7: In the CCLE drug response and TCGA breast cancer (TCGA-BRCA) subtype case studies, MDI+ provided the most stable feature importance rankings across 32 train-test splits. Specifically, (a) for 22 out of the 24 drugs in the CCLE case study (the exceptions being PD-0325901 and Panobinostat, highlighted in blue) and in the TCGA-BRCA case study, MDI+ had the fewest number of distinct features that were ranked in the top 10 across the 32 train-test splits. Note that the outlined points denote ties and that MDI+ (logistic) is only used in the classification setting (i.e., for TCGA-BRCA). Moreover, as shown in (b), the standard deviation of the top 5 features smaller for MDI+ as compared to other methods. We provide the results for three CCLE drugs and the TCGA-BRCA subtypes here. MDI-oob is also omitted due to its high instability. Full results can be found in Appendix H.

7.2 Case Study II: Breast Cancer Subtype Prediction

Breast cancer is known to be a heterogeneous disease [2] and is thus often classified into various molecular subtypes. PAM50 is one prominent breast cancer subtyping method, which is frequently used to inform treatment plans [58]. PAM50 divides breast cancers into five intrinsic subtypes: Luminal A, Luminal B, human epidermal growth factor receptor 2 (HER2)-enriched, Basal-like, and Normal-like. Each subtype differs based upon their biological properties and generally corresponds to a different prognosis. Luminal A typically has the best prognosis, while HER2-enriched and Basal-like are more aggressive. In this study, we aim to predict the PAM50-defined breast cancer subtypes using gene expression data from The Cancer Genome Atlas (TCGA) and more importantly, identify the genes that heavily determine the breast cancer subtype.

Data and Methods. Using data from the TCGA, we have RNASeq gene expression data from $n = 1083$ individuals and $p = 5000$ genes after filtering (details in Appendix H.2). We also have the PAM50-defined breast cancer subtype for each of these individuals, giving rise to a multi-class classification problem with five classes. As in Section 7.1, we split the data into 80% training and 20% test and evaluate results across 32 train-test splits. The RF settings and feature importance methods used are the same as those in Section 5.1 for classification. Note however that MDI-oob has not been implemented for multi-class classification and is thus omitted from this analysis.

Prediction accuracy. Comparing RF and RF+ across the 32 train-test splits, we found that RF+ (logistic) yielded the best prediction performance over a variety of metrics (see Table 3). In particular, RF+ (logistic) yielded an average classification accuracy of 88.4%, compared to 86.1% from RF. RF+ (ridge) also improved the prediction performance over RF (accuracy: 87.3%), but was slightly worse than RF+ (logistic). The improvement of RF+ (logistic) over RF+ (ridge) provides further evidence of the benefit of tailoring the GLM to the problem structure.

Accuracy of gene importance rankings. Both MDI+ (ridge) and MDI+ (logistic) produced the same set of top 10 genes, albeit in a different order. For each of these 10 genes (Table 4), there is significant literature, supporting the gene's involvement in the development and progression of breast cancer. For example, *ESR1*, the top-ranked gene across all feature importance methods, has been widely studied over the last decade due to its prominent role in the pathogenesis of breast cancer [75] as well as in novel promising therapies [12]. *ESR1* encodes the estrogen receptor-α protein and is known to cause increased resistance to standard-of-care endocrine therapy [12].

Several other genes in the top 10, namely, *FOXA1*, *FOXM1*, and *MLPH*, are also associated with estrogen receptor-positive breast cancer and increased resistance to endocrine therapy [22, 9, 73]. Beyond this, *GATA3* (ranked 2$^{\text{nd}}$ in MDI+ (logistic) and 4$^{\text{th}}$ in MDI+ (ridge)) was shown to control luminal breast cancer predominantly through differential regulation of other genes including *THSD4* [15] while *FOXC1* can counteract *GATA3* binding and impact endocrine resistance [81].

In addition to this supporting literature evidence, we show in Figure 40 that the top 25 genes from MDI+ (logistic) and MDI+ (ridge) are more predictive of the PAM50 breast cancer subtypes than the top 25 genes from other feature importance measures (details in Appendix H). This top 25 from MDI+ includes genes such as *RRM2* and *SFRP1* that are not in the top 25 from any of the competing feature importance methods. However, these genes are known to be associated with breast cancer. For instance, up-regulation of *RRM2* was shown to enhance breast cancer cell proliferation [41], while loss of *SFRP1* expression is associated with breast tumor stage and poor prognosis when present in early stage breast tumors [37].

Stability of gene importance rankings. As in the CCLE case study, we further evaluated the stability of the gene importance rankings across 32 train-test splits and found that MDI+ gave the most stable rankings. In Figure 7a, MDI+ (ridge), followed by MDI+ (logistic), yielded the smallest number of distinct features in the top 10 across the 32 train-test splits. Moreover, for the top five features, the variability of the feature importance rankings across the 32 train-test splits is much smaller for MDI+ (ridge) and MDI+ (logistic) compared to its competitors (Figure 7b).

8 Discussion

Feature importance measures are used not only to describe how complicated ML models make predictions, but also to identify features that are most relevant to the underlying data-generating process. MDI+ provides a novel feature importance framework that generalizes the popular MDI importance measure. Specifically, MDI+ builds upon a recent interpretation of decision trees as a representation learning algorithm and connects MDI to the R^2 value from a linear regression on this learned representation. Leveraging this interpretation, MDI+ allows one to (1) replace the regression model and/or R^2 metric with other models and/or metrics that are tailored to a given data structure and (2) incorporate additional features or knowledge to mitigate the known biases of decision tree-based techniques. Furthermore, MDI+ adds regularization and efficient sample-splitting to

overcome known biases of MDI against correlated features and features with low entropy.

We demonstrate the utility of MDI+ via a wide array of data-inspired simulations designed to reflect a number of real-world scenarios as advocated by the PCS framework [80, 18], and via two real-world case studies on drug response prediction and breast cancer subtype prediction. Our simulation studies show that tailoring MDI+ to different problem structures results in more accurate feature rankings, compared to other popular feature importance measures. Moreover, applying MDI+ to both case studies shows the robustness of our method to different data perturbations (e.g., different train-test splits and random seeds) as well as its ability to identify predictive features. However, given the potentially high-stakes and downstream impacts of using feature importance measures in real-world scientific problems, our data-inspired simulations and real-world case studies only represent a first step in establishing the trustworthiness of MDI+, or any feature importance measure. Towards this end, developing a formal protocol and benchmarks to evaluate the effectiveness of feature importance measures is needed and will have a significant impact on the applicability of these methods in practice. One possible pathway to constructing this protocol is via the PCS framework. In particular, the PCS framework advocates for holistic evaluation of feature importances including its predictability, and stability, as we have begun to explore in our case studies. We leave it as important future work to formalize and expand these ideas.

Beyond this, MDI+ opens the door to many other natural directions for future work. First, MDI+ need not be limited to RFs and can be defined for any tree ensemble, and in particular, XGBoost [14] and FIGS [72]. Second, as we illustrated in Section 5, exploiting the flexibility of MDI+ can substantially improve performance. Though we demonstrated this in the classification and robust regression settings, there are many other contexts to consider such as multitarget prediction, survival analysis, longitudinal data analysis, etc. In addition, while we have provided practitioners with a PCS-based approach to assist practitioners with modeling choices in the MDI+ framework, further investigation is needed (e.g., different approaches to combining scores from different MDI+ models). Thirdly, the connection to linear models, and now GLMs, can also be further utilized to adapt traditional hypothesis testing tools to convert our importance measures into significance scores. Finally, the flexibility of RF+ as a general prediction method may be of significant independent interest. Beyond fitting GLMs on the augmented transformed dataset, RF+ allows practitioners to fit any ML algorithm on the augmented transformed dataset. This provides a novel avenue to incorporate the advantages of tree-based methodology and other ML tools for prediction.

9 Acknowledgements

We gratefully acknowledge partial support from NSF TRIPODS Grant 1740855, DMS-2209975, 1613002, 1953191, 2015341, IIS 1741340, ONR grant N00014-17-1-2176, the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-0939370, NSF grant 2023505 on Collaborative Research: Foundations of Data Science Institute (FODSI), the NSF and the Simons Foundation for the Collaboration on the Theoretical Foundations of Deep Learning through awards DMS-2031883 and 814639, a grant from the Weill Neurohub, and a Chan Zuckerberg Biohub Intercampus Research Award. TMT acknowledges support from the NSF Graduate Research Fellowship Program DGE-2146752. YT was partially supported by NUS Start-up Grant A-8000448-00-00.

References

[1] Agarwal, A., Tan, Y. S., Ronen, O., Singh, C. and Yu, B. [2022], Hierarchical shrinkage: Improving the accuracy and interpretability of tree-based models., *in* 'International Conference on Machine Learning', PMLR, pp. 111–135.

[2] Atlas, T. C. G. [2012], 'Comprehensive molecular portraits of human breast tumours', *Nature* **490**(7418), 61–70.

[3] Azadkia, M. and Chatterjee, S. [2021], 'A simple measure of conditional dependence', *The Annals of Statistics* **49**(6), 3070–3102.

[4] Barretina, J., Caponigro, G., Stransky, N., Venkatesan, K., Margolin, A. A., Kim, S., Wilson, C. J., Lehár, J., Kryukov, G. V., Sonkin, D. et al. [2012], 'The cancer cell line encyclopedia enables predictive modelling of anticancer drug sensitivity', *Nature* **483**(7391), 603–607.

[5] Basu, S., Kumbier, K., Brown, J. B. and Yu, B. [2018], 'Iterative random forests to discover predictive and stable high-order interactions', *Proceedings of the National Academy of Sciences* **115**(8), 1943–1948.

[6] Behr, M., Kumbier, K., Cordova-Palomera, A., Aguirre, M., Ashley, E., Butte, A. J., Arnaout, R., Brown, B., Priest, J. and Yu, B. [2020], 'Learning epistatic polygenic phenotypes with boolean interactions', *bioRxiv* .

[7] Behr, M., Wang, Y., Li, X. and Yu, B. [2021], 'Provable boolean interaction recovery from tree ensemble obtained via random forests', *arXiv preprint arXiv:2102.11800* .

[8] Bénard, C., Da Veiga, S. and Scornet, E. [2021], 'Mda for random forests: inconsistency, and a practical solution via the sobol-mda', *arXiv preprint arXiv:2102.13347* .

[9] Bergamaschi, A., Madak-Erdogan, Z., Kim, Y. J., Choi, Y.-L., Lu, H. and Katzenellenbogen, B. S. [2014], 'The forkhead transcription factor foxm1 promotes endocrine resistance and invasiveness in estrogen receptor-positive breast cancer by expansion of stem-like cancer cells', *Breast Cancer Research* **16**, 1–18.

[10] Breiman, L. [2001], 'Random forests', *Machine Learning* **45**(1), 5–32.

[11] Breiman, L., Friedman, J., Stone, C. J. and Olshen, R. A. [1984], *Classification and Regression Trees*, CRC press.

[12] Brett, J. O., Spring, L. M., Bardia, A. and Wander, S. A. [2021], 'Esr1 mutation as an emerging clinical biomarker in metastatic hormone receptor-positive breast cancer', *Breast Cancer Research* **23**, 1–15.

[13] Caruana, R., Karampatziakis, N. and Yessenalina, A. [2008], An empirical evaluation of supervised learning in high dimensions, *in* 'Proceedings of the 25th International Conference on Machine learning', pp. 96–103.

[14] Chen, T. and Guestrin, C. [2016], XGBoost: A scalable tree boosting system, *in* 'Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', pp. 785–794.

[15] Cohen, H., Ben-Hamo, R., Gidoni, M., Yitzhaki, I., Kozol, R., Zilberberg, A. and Efroni, S. [2014], 'Shift in gata3 functions, and gata3 mutations, control progression and clinical presentation in breast cancer', *Breast Cancer Research* **16**, 1–16.

[16] Costello, J. C., Heiser, L. M., Georgii, E., Gönen, M., Menden, M. P., Wang, N. J., Bansal, M., Ammad-Ud-Din, M., Hintsanen, P., Khan, S. A. et al. [2014], 'A community effort to assess and improve drug sensitivity prediction algorithms', *Nature biotechnology* **32**(12), 1202–1212.

[17] De Rosa, A., Fontana, A., Nuzzo, T., Garofalo, M., Di Maio, A., Punzo, D., Copetti, M., Bertolino, A., Errico, F., Rampino, A. et al. [2022], 'Machine learning algorithm unveils glutamatergic alterations in the post-mortem schizophrenia brain', *Schizophrenia* **8**(1), 8.

[18] Duncan, J. and Tang, T. [2022], 'simchef v0.0.2'.
URL: *https://doi.org/10.5281/zenodo.6565209*

[19] Epifanio, I. [2017], 'Intervention in prediction measure: a new approach to assessing variable importance for random forests', *BMC bioinformatics* **18**(1), 1–16.

[20] Fernández-Delgado, M., Cernadas, E., Barro, S. and Amorim, D. [2014], 'Do we need hundreds of classifiers to solve real world classification problems?', *The Journal of Machine Learning Research* **15**(1), 3133–3181.

[21] Friedman, J. H. and Popescu, B. E. [2008], 'Predictive learning via rule ensembles', *The Annals of Applied Statistics* **2**(3), 916–954.

[22] Fu, X., Pereira, R., De Angelis, C., Veeraraghavan, J., Nanda, S., Qin, L., Cataldo, M. L., Sethunath, V., Mehravaran, S., Gutierrez, C. et al. [2019], 'Foxa1 upregulation promotes enhancer and transcriptional reprogramming in endocrine-resistant breast cancer', *Proceedings of the National Academy of Sciences* **116**(52), 26823–26834.

[23] Gan, L., Zheng, L. and Allen, G. I. [2022], 'Inference for interpretable machine learning: Fast, model-agnostic confidence intervals for feature importance', *arXiv preprint arXiv:2206.02088* .

[24] Genuer, R., Poggi, J.-M. and Tuleau, C. [2008], 'Random forests: some methodological insights', *arXiv preprint arXiv:0811.3619* .

[25] Genuer, R., Poggi, J.-M. and Tuleau-Malot, C. [2010], 'Variable selection using random forests', *Pattern recognition letters* **31**(14), 2225–2236.

[26] Gregorutti, B., Michel, B. and Saint-Pierre, P. [2017], 'Correlation and variable importance in random forests', *Statistics and Computing* **27**(3), 659–678.

[27] Grömping, U. [2009], 'Variable importance assessment in regression: linear regression versus random forest', *The American Statistician* **63**(4), 308–319.

[28] Hastie, T. and Tibshirani, R. [1986], 'Generalized additive models', *Statistical Science* pp. 297–310.

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

[30] Hoerl, A. E. and Kennard, R. W. [1970], 'Ridge regression: Biased estimation for nonorthogonal problems', *Technometrics* **12**(1), 55–67.

[31] Hooker, G. and Mentch, L. [2019], 'Please stop permuting features: An explanation and alternatives', *arXiv e-prints* pp. arXiv–1905.

[32] Hothorn, T., Hornik, K. and Zeileis, A. [2006], 'Unbiased recursive partitioning: A conditional inference framework', *Journal of Computational and Graphical statistics* **15**(3), 651–674.

[33] Ishwaran, H. [2007], 'Variable importance in binary regression trees and forests', *Electronic Journal of Statistics* **1**, 519–537.

[34] Ishwaran, H., Kogalur, U. B., Gorodeski, E. Z., Minn, A. J. and Lauer, M. S. [2010], 'High-dimensional variable selection for survival data', *Journal of the American Statistical Association* **105**(489), 205–217.

[35] Jang, I. S., Neto, E. C., Guinney, J., Friend, S. H. and Margolin, A. A. [2014], Systematic assessment of analytical methods for drug sensitivity prediction from cancer cell line data, *in* 'Biocomputing 2014', World Scientific, pp. 63–74.

[36] Kazemitabar, J., Amini, A., Bloniarz, A. and Talwalkar, A. S. [2017], 'Variable importance using decision trees', *Advances in neural information processing systems* **30**.

[37] Klopocki, E., Kristiansen, G., Wild, P. J., Klaman, I., Castanos-Velez, E., Singer, G., Stöhr, R., Simon, R., Sauter, G., Leibiger, H. et al. [2004], 'Loss of sfrp1 is associated with breast cancer progression and poor prognosis in early stage tumors', *International journal of oncology* **25**(3), 641–649.

[38] Klusowski, J. M. [2021], 'Universal consistency of decision trees in high dimensions', *arXiv preprint arXiv:2104.13881* .

[39] Klusowski, J. and Tian, P. [2021], Nonparametric variable screening with optimal decision stumps, *in* 'International Conference on Artificial Intelligence and Statistics', PMLR, pp. 748–756.

[40] Li, X., Wang, Y., Basu, S., Kumbier, K. and Yu, B. [2019], 'A debiased mdi feature importance measure for random forests', *Advances in Neural Information Processing Systems* **32**.

[41] Liang, W.-H., Li, N., Yuan, Z.-Q., Qian, X.-L. and Wang, Z.-H. [2019], 'Dscam-as1 promotes tumor growth of breast cancer by reducing mir-204-5p and up-regulating rrm2', *Molecular carcinogenesis* **58**(4), 461–473.

[42] Lind, A. P. and Anderson, P. C. [2019], 'Predicting drug activity against cancer cells by random forest models based on minimal genomic information and chemical properties', *PloS one* **14**(7), e0219774.

[43] List, M., Hauschild, A.-C., Tan, Q., Kruse, T. A., Baumbach, J. and Batra, R. [2014], 'Classification of breast cancer subtypes by combining gene expression and dna methylation data', *Journal of integrative bioinformatics* **11**(2), 1–14.

[44] Loecher, M. [2022*a*], Debiasing mdi feature importance and shap values in tree ensembles, *in* 'Machine Learning and Knowledge Extraction: 6th IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 International Cross-Domain Conference, CD-MAKE 2022, Vienna, Austria, August 23–26, 2022, Proceedings', Springer, pp. 114–129.

[45] Loecher, M. [2022*b*], 'Unbiased variable importance for random forests', *Communications in Statistics-Theory and Methods* **51**(5), 1413–1425.

[46] Louppe, G., Wehenkel, L., Sutera, A. and Geurts, P. [2013], 'Understanding variable importances in forests of randomized trees', *Advances in neural information processing systems* **26**.

[47] Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N. and Lee, S.-I. [2020], 'From local explanations to global understanding with explainable ai for trees', *Nature machine intelligence* **2**(1), 56–67.

[48] Lundberg, S. M. and Lee, S.-I. [2017], 'A unified approach to interpreting model predictions', *Advances in neural information processing systems* **30**.

[49] Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R. and Yu, B. [2019], 'Definitions, methods, and applications in interpretable machine learning', *Proceedings of the National Academy of Sciences* **116**(44), 22071–22080.

[50] Nelson, D. L., Lehninger, A. L. and Cox, M. M. [2008], *Lehninger principles of biochemistry*, Macmillan.

[51] Nembrini, S. [2019], 'Bias in the intervention in prediction measure in random forests: illustrations and recommendations', *Bioinformatics* .

[52] Nembrini, S., König, I. R. and Wright, M. N. [2018], 'The revival of the gini importance?', *Bioinformatics* **34**(21), 3711–3718.

[53] Nicodemus, K. K. [2011], 'On the stability and ranking of predictors from random forest variable importance measures', *Briefings in bioinformatics* **12**(4), 369–373.

[54] Nicodemus, K. K. and Malley, J. D. [2009], 'Predictor correlation impacts machine learning algorithms: implications for genomic studies', *Bioinformatics* **25**(15), 1884–1890.

[55] Olson, R. S., Cava, W. L., Mustahsan, Z., Varik, A. and Moore, J. H. [2018], Data-driven advice for applying machine learning to bioinformatics problems, *in* 'Biocomputing 2018: Proceedings of the Pacific Symposium', World Scientific, pp. 192–203.

[56] Osofsky, J. D. [1997], 'The effects of exposure to violence on young children (1995).', *Carnegie Corporation of New York Task Force on the Needs of Young Children; An earlier version of this article was presented as a position paper for the aforementioned corporation.* .

[57] Owen, A. B. [2007], 'A robust hybrid of lasso and ridge regression', *Contemporary Mathematics* **443**(7), 59–72.

[58] Parker, J. S., Mullins, M., Cheang, M. C., Leung, S., Voduc, D., Vickery, T., Davies, S., Fauron, C., He, X., Hu, Z. et al. [2009], 'Supervised risk predictor of breast cancer based on intrinsic subtypes', *Journal of clinical oncology* **27**(8), 1160.

[59] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. et al. [2011], 'Scikit-learn: Machine learning in python', *The Journal of Machine Learning Research* **12**, 2825–2830.

[60] Rad, K. R. and Maleki, A. [2020], 'A scalable estimate of the out-of-sample prediction error via approximate leave-one-out cross-validation', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **82**(4), 965–996.

[61] Ramosaj, B. and Pauly, M. [2019], 'Asymptotic unbiasedness of the permutation importance measure in random forest models', *arXiv preprint arXiv:1912.03306* .

[62] Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L. and Zhong, C. [2021], 'Interpretable machine learning: Fundamental principles and 10 grand challenges', *arXiv preprint arXiv:2103.11251* .

[63] Saabas, A. [2022], 'reeinterpreter python package', https://github.com/andosa/treeinterpreter.

[64] Sandri, M. and Zuccolotto, P. [2008], 'A bias correction algorithm for the gini variable importance measure in classification trees', *Journal of Computational and Graphical Statistics* **17**(3), 611–628.

[65] Scornet, E. [2020], 'Trees, forests, and impurity-based variable importance', *arXiv preprint arXiv:2001.04295* .

[66] Shwartz-Ziv, R. and Armon, A. [2022], 'Tabular data: Deep learning is not all you need', *Information Fusion* **81**, 84–90.
URL: *https://www.sciencedirect.com/science/article/pii/S1566253521002360*

[67] Singh, C., Nasseri, K., Tan, Y. S., Tang, T. and Yu, B. [2021], 'imodels: a python package for fitting interpretable models', *Journal of Open Source Software* **6**(61), 3192.

[68] Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T. and Zeileis, A. [2008], 'Conditional variable importance for random forests', *BMC bioinformatics* **9**(1), 1–11.

[69] Strobl, C., Boulesteix, A.-L., Zeileis, A. and Hothorn, T. [2007], 'Bias in random forest variable importance measures: Illustrations, sources and a solution', *BMC bioinformatics* **8**(1), 1–21.

[70] Sutera, A., Louppe, G., Huynh-Thu, V. A., Wehenkel, L. and Geurts, P. [2021], 'From global to local mdi variable importances for random forests and when they are shapley values', *Advances in Neural Information Processing Systems* **34**.

[71] Tan, Y. S., Agarwal, A. and Yu, B. [2021], 'A cautionary tale on fitting decision trees to data from additive models: generalization lower bounds'.

[72] Tan, Y. S., Singh, C., Nasseri, K., Agarwal, A. and Yu, B. [2022], 'Fast interpretable greedy-tree sums (figs)', *arXiv preprint arXiv:2201.11931* .

[73] Thakkar, A. D., Raj, H., Chakrabarti, D., Ravishankar, Saravanan, N., Muthuvelan, B., Balakrishnan, A. and Padigaru, M. [2010], 'Identification of gene expression signature in estrogen receptor positive breast carcinoma', *Biomarkers in cancer* **2**, BIC–S3793.

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

[75] Toy, W., Shen, Y., Won, H., Green, B., Sakr, R. A., Will, M., Li, Z., Gala, K., Fanning, S., King, T. A. et al. [2013], 'Esr1 ligand-binding domain mutations in hormone-resistant breast cancer', *Nature genetics* **45**(12), 1439–1445.

[76] Tsybakov, A. B. [2004], 'Introduction to nonparametric estimation, 2009', *URL https://doi.org/10.1007/b13794. Revised and extended from the* **9**(10).

[77] Wan, Q. and Pal, R. [2014], 'An ensemble based top performing approach for nci-dream drug sensitivity prediction challenge', *PloS one* **9**(6), e101183.

[78] Wang, G., Sarkar, A., Carbonetto, P. and Stephens, M. [2020], 'A simple new approach to variable selection in regression, with application to genetic fine mapping', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **82**(5), 1273–1300.

[79] Webber, W., Moffat, A. and Zobel, J. [2010], 'A similarity measure for indefinite rankings', *ACM Transactions on Information Systems (TOIS)* **28**(4), 1–38.

[80] Yu, B. and Kumbier, K. [2020], 'Veridical data science', *Proceedings of the National Academy of Sciences* **117**(8), 3920–3929.

[81] Yu-Rice, Y., Jin, Y., Han, B., Qu, Y., Johnson, J., Watanabe, T., Cheng, L., Deng, N., Tanaka, H., Gao, B. et al. [2016], 'Foxc1 is involved in erα silencing by counteracting gata3 binding and is implicated in endocrine resistance', *Oncogene* **35**(41), 5400–5411.

[82] Zhang, L. and Janson, L. [2020], 'Floodgate: inference for model-free variable importance', *arXiv preprint arXiv:2007.01283* .

[83] Zhou, Z. and Hooker, G. [2021], 'Unbiased measurement of feature importance in tree-based methods', *ACM Transactions on Knowledge Discovery from Data (TKDD)* **15**(2), 1–21.

A Proofs

A.1 Proof of Theorem 1

Using the law of total variance, we may rewrite the formula for impurity decrease for a node $\mathfrak{t}$ via

$$\begin{aligned}\hat{\Delta}(s,\mathcal{D}_n^*) &= N(\mathfrak{t})^{-1}\left(\sum_{\mathbf{x}_i\in\mathfrak{t}}(y_i-\bar{y}_{\mathfrak{t}})^2-\sum_{\mathbf{x}_i\in\mathfrak{t}_L}(y_i-\bar{y}_{\mathfrak{t}_L})^2-\sum_{\mathbf{x}_i\in\mathfrak{t}_R}(y_i-\bar{y}_{\mathfrak{t}_R})^2\right)\\ &= \frac{N(\mathfrak{t}_L)N(\mathfrak{t}_R)}{N(\mathfrak{t})^2}(\bar{y}_{\mathfrak{t}_L}-\bar{y}_{\mathfrak{t}_R})^2.\end{aligned}$$

Note that we use a binary indicator of a sample lying in the right child node as the conditioning variable. Next, using equation (3) and using $\psi(\mathfrak{t})$ to denote the resulting feature vector on the training set, we compute

$$\begin{aligned}\psi(\mathfrak{t})^T\mathbf{y} &= \frac{N(\mathfrak{t}_R)\sum_{\mathbf{x}_i\in\mathfrak{t}_L}y_i-N(\mathfrak{t}_L)\sum_{\mathbf{x}_i\in\mathfrak{t}_R}y_i}{\sqrt{N(\mathfrak{t}_L)N(\mathfrak{t}_R)}}\\ &= \sqrt{N(\mathfrak{t}_L)N(\mathfrak{t}_R)}(\bar{y}_{\mathfrak{t}_L}-\bar{y}_{\mathfrak{t}_R}).\end{aligned}$$

By combining the above two equations, we obtain the formula

$$\hat{\Delta}(s,\mathcal{D}_n^*) = \frac{\left(\psi(\mathfrak{t})^T\mathbf{y}\right)^2}{N(\mathfrak{t})^2}. \tag{7}$$

Now, let $\mathfrak{t}_1,\ldots,\mathfrak{t}_m$ denote the nodes splitting on feature X_k, which means that we have $\Psi^k = [\psi(\mathfrak{t}_1),\cdots,\psi(\mathfrak{t}_m)]$. We may then compute

$$\begin{aligned}\left\|\Psi^k(\mathbf{X})\left(\Psi^k(\mathbf{X})^T\Psi^k(\mathbf{X})\right)^{-1}\Psi^k(\mathbf{X})^T\mathbf{y}\right\|_2^2 &= \mathbf{y}^T\Psi^k(\mathbf{X})\left(\Psi^k(\mathbf{X})^T\Psi^k(\mathbf{X})\right)^{-1}\Psi^k(\mathbf{X})^T\mathbf{y}\\ &= \sum_{i=1}^m\frac{\left(\psi(\mathfrak{t}_i)^T\mathbf{y}\right)^2}{N(\mathfrak{t}_i)}\\ &= \sum_{i=1}^m N(\mathfrak{t}_i)\hat{\Delta}(s_i,\mathcal{D}_n^*),\end{aligned}$$

where the second equality comes from recognizing that $\left(\Psi^k(\mathbf{X})^T\Psi^k(\mathbf{X})\right)^{-1}$ is a diagonal matrix with i-th diagonal entry equal to $N(\mathbf{t}_i)^{-1}$, and the third equality comes from plugging in (7). Recognizing that the right-hand side is simply $n\text{MDI}_k(\mathcal{S},\mathcal{D}_n^*)$ completes the proof of the left equality in (4). The right equality follows from the definition of training R^2 as measuring the fraction of variance in the responses explained by the fitted values.

A.2 Proof of Proposition 1

Denote $\mathbf{P} := \Psi^k(\mathbf{X})\left(\Psi^k(\mathbf{X})^T\Psi^k(\mathbf{X})\right)^{-1}\Psi^k(\mathbf{X})^T$. We have shown in the proof of Theorem 1 that

$$\text{MDI}_k(\mathcal{S},\mathcal{D}_n) = \frac{1}{n}\|\mathbf{P}\mathbf{y}\|_2^2. \tag{8}$$

Now write $\mathbf{y} = \mathbf{f} + \boldsymbol{\epsilon}$ where $\mathbf{f} = (f(\mathbf{x}_1),...,f(\mathbf{x}_n))^T$ and $\boldsymbol{\epsilon} = (\epsilon_1,...,\epsilon_n)^T$. Under a fixed design, $\mathbf{f}$ is deterministic while $\boldsymbol{\epsilon}$ is random. Taking expectations therefore gives

$$\begin{aligned}\mathbb{E}\{\text{MDI}_k(\mathcal{S},\mathcal{D}_n)\} &= \mathbb{E}\left\{\frac{1}{n}\|\mathbf{P}(\mathbf{f}+\boldsymbol{\epsilon})\|_2^2\right\} \\ &= \frac{1}{n}\|\mathbf{P}\mathbf{f}\|_2^2 + \frac{1}{n}\mathbb{E}\left\{\|\mathbf{P}\boldsymbol{\epsilon}\|_2^2\right\} \\ &= \frac{1}{n}\|\mathbf{P}\mathbf{f}\|_2^2 + \frac{\sigma^2\text{Trace}(\mathbf{P})}{n}. \end{aligned} \tag{9}$$

By orthogonality of local decision stumps, $\Psi^k(\mathbf{X})$ has full rank, and $\text{Trace}(\mathbf{P})$ is just the number of columns, which is equal to $\left|\mathcal{S}^{(k)}\right|$. Finally, notice that

$$\text{MDI}_k(\mathcal{S},\mathcal{D}_n^0) = \frac{1}{n}\|\mathbf{P}\mathbf{f}\|_2^2. \tag{10}$$

B Feature Ranking Performance Simulations

In this section, we provide additional simulation experiments to supplement those already provided in Section 5. Unless specified otherwise, we follow the simulation protocol described in Section 5, and all plots show the mean evaluation metric, averaged across 50 experimental replicates, with error bars denoting $\pm$ 1SE.

B.1 Data and Code Availability

The Juvenile dataset can be downloaded using the `imodels` python package [67]. The Enhancer and Splicing datasets were taken from [5]. The CCLE dataset can be downloaded from DepMap

Public 18Q3 (https://depmap.org/portal/download/).

Data Preprocessing. We performed basic data cleaning on these datasets before using them as covariate matrices in the simulation study. Specifically, for all four datasets, we removed constant and duplicated columns. We also applied a $\log(x+1)$ transformation to all values in the Enhancer and CCLE datasets as these are count-valued and highly right-skewed. For the Enhancer dataset, the raw data contained repeated measurements (i.e., multiple feature columns) for the same transcription factor; we thus removed all but one measurement (i.e., column) for each transcription factor. Note finally that the CCLE dataset contains 50114 features, but we chose a random subset of 1000 features to use in the covariate matrix in each simulation replicate.

The preprocessed datasets can be found on Zenodo at https://zenodo.org/record/8111870. Code to run all simulations can be found on GitHub at https://github.com/Yu-Group/imodels-experiments.

B.2 Regression Simulations

Following the simulation setup discussed in Sections 5.1 and 5.2, we provide the results for all datasets and regression functions in the regression setting. We show the results for all datasets under the linear response in Figure 8, the LSS response in Figure 9, the polynomial interaction response in Figure 10, and the linear + LSS response in Figure 11.

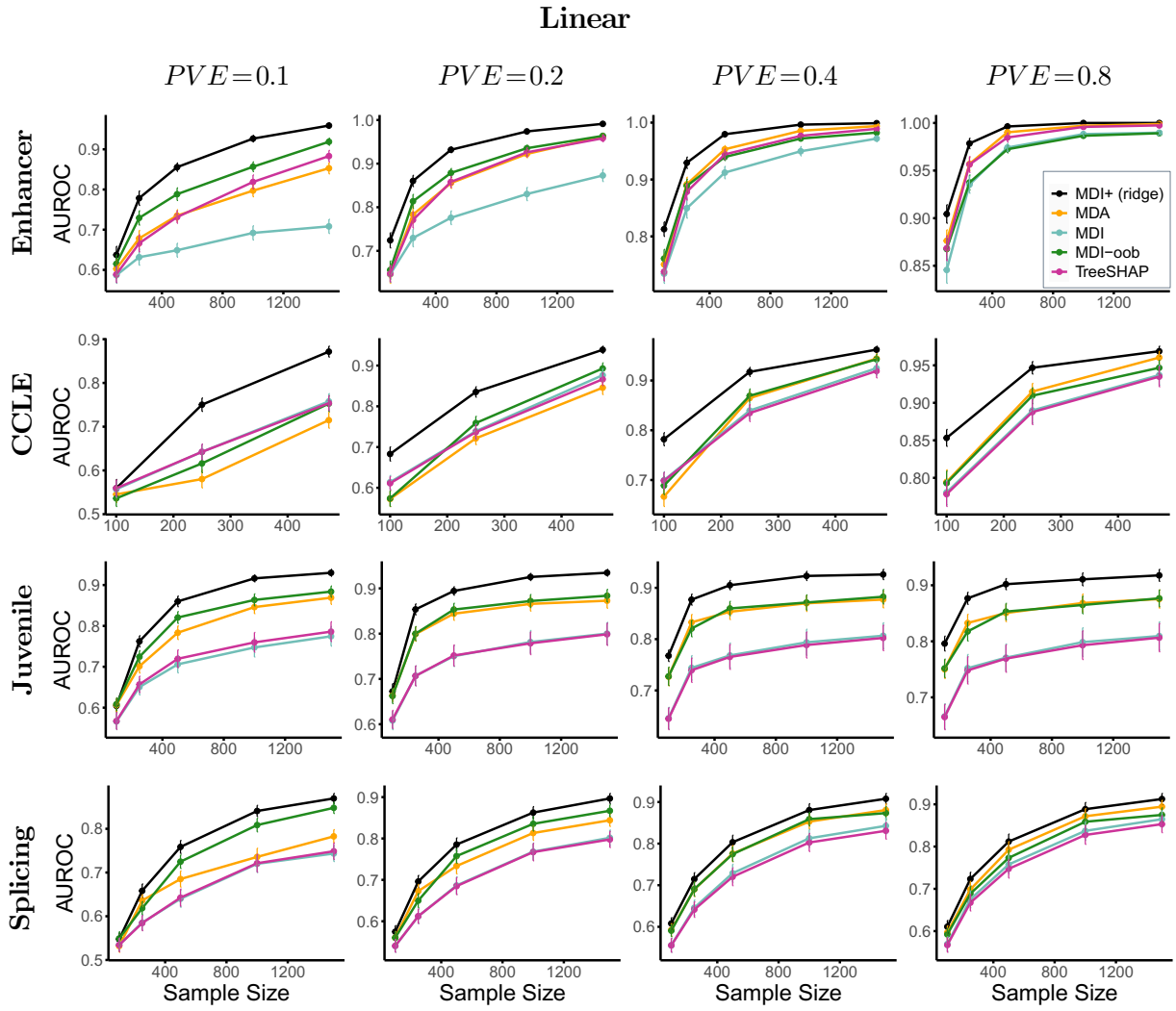


Figure 8: MDI+ (ridge) outperforms other feature importance methods for the linear regression function described in Section 5.1. This pattern is evident across various datasets (specified by row), proportions of variance explained (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE

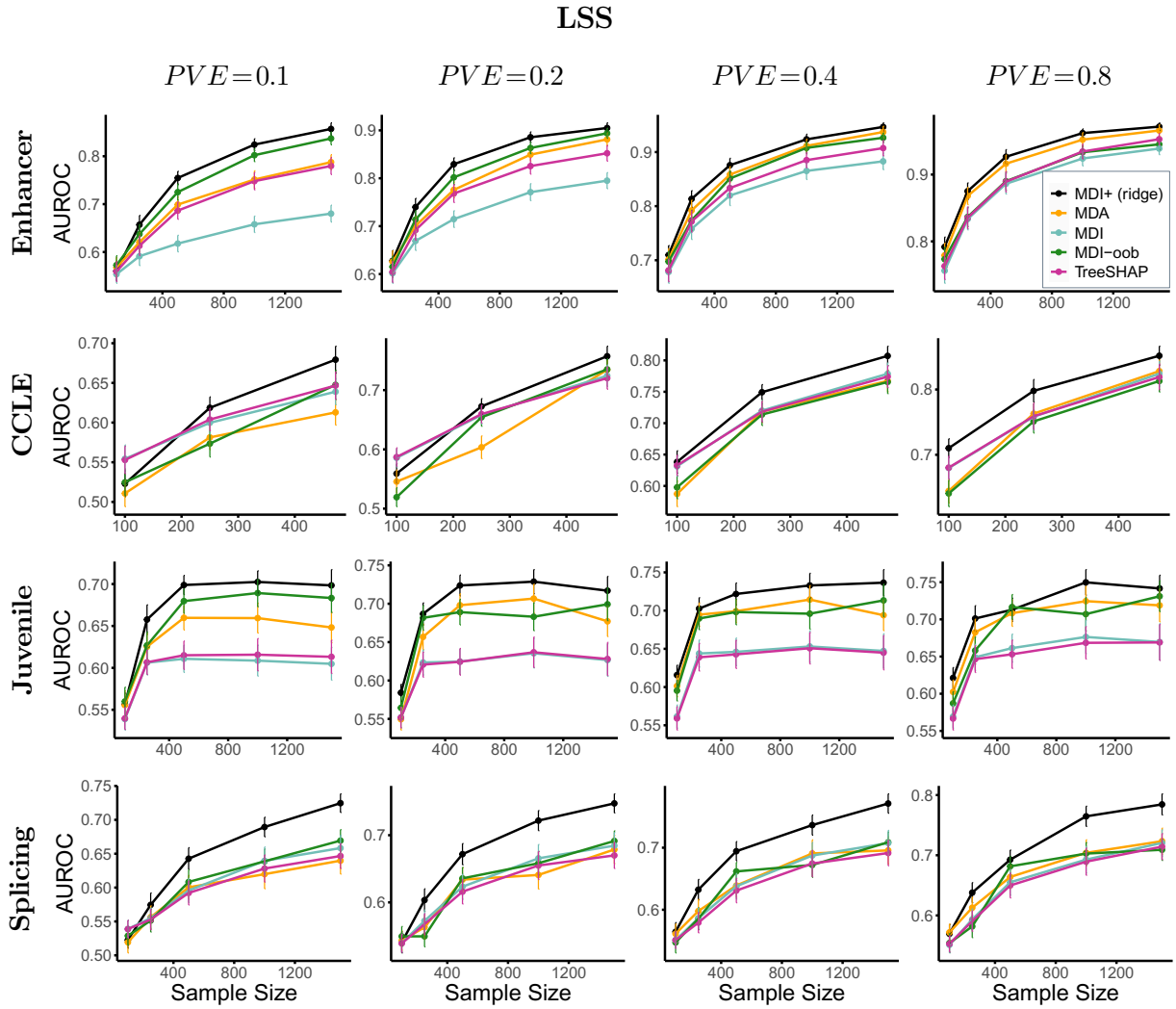


Figure 9: MDI+ (ridge) outperforms other feature importance methods for the LSS function described in Section 5.1. This pattern is evident across various datasets (specified by row), proportions of variance explained (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE

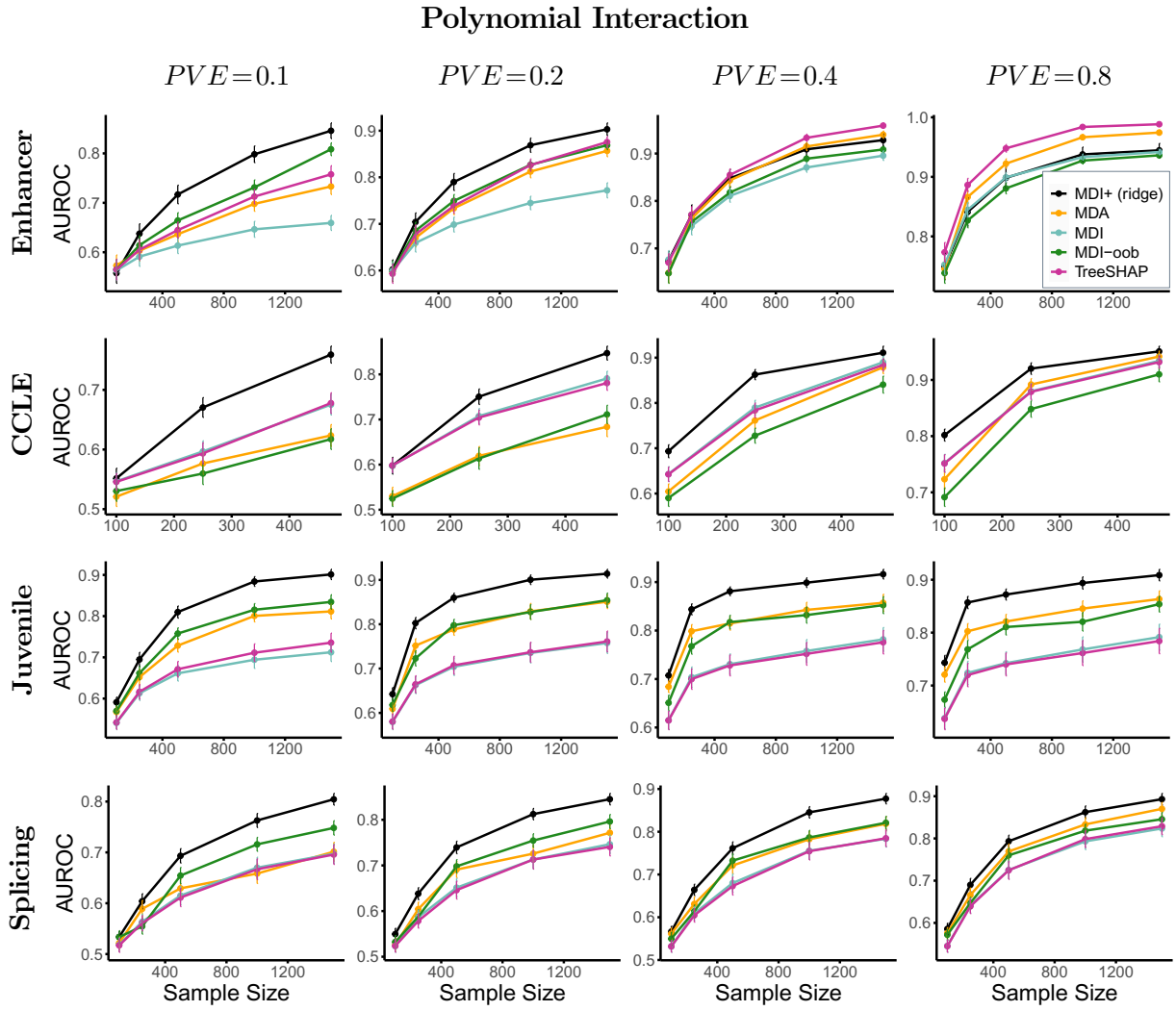


Figure 10: MDI+ (ridge) outperforms other feature importance methods for the polynomial interaction regression function described in Section 5.1. This pattern is evident across various datasets (specified by row), proportions of variance explained (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE

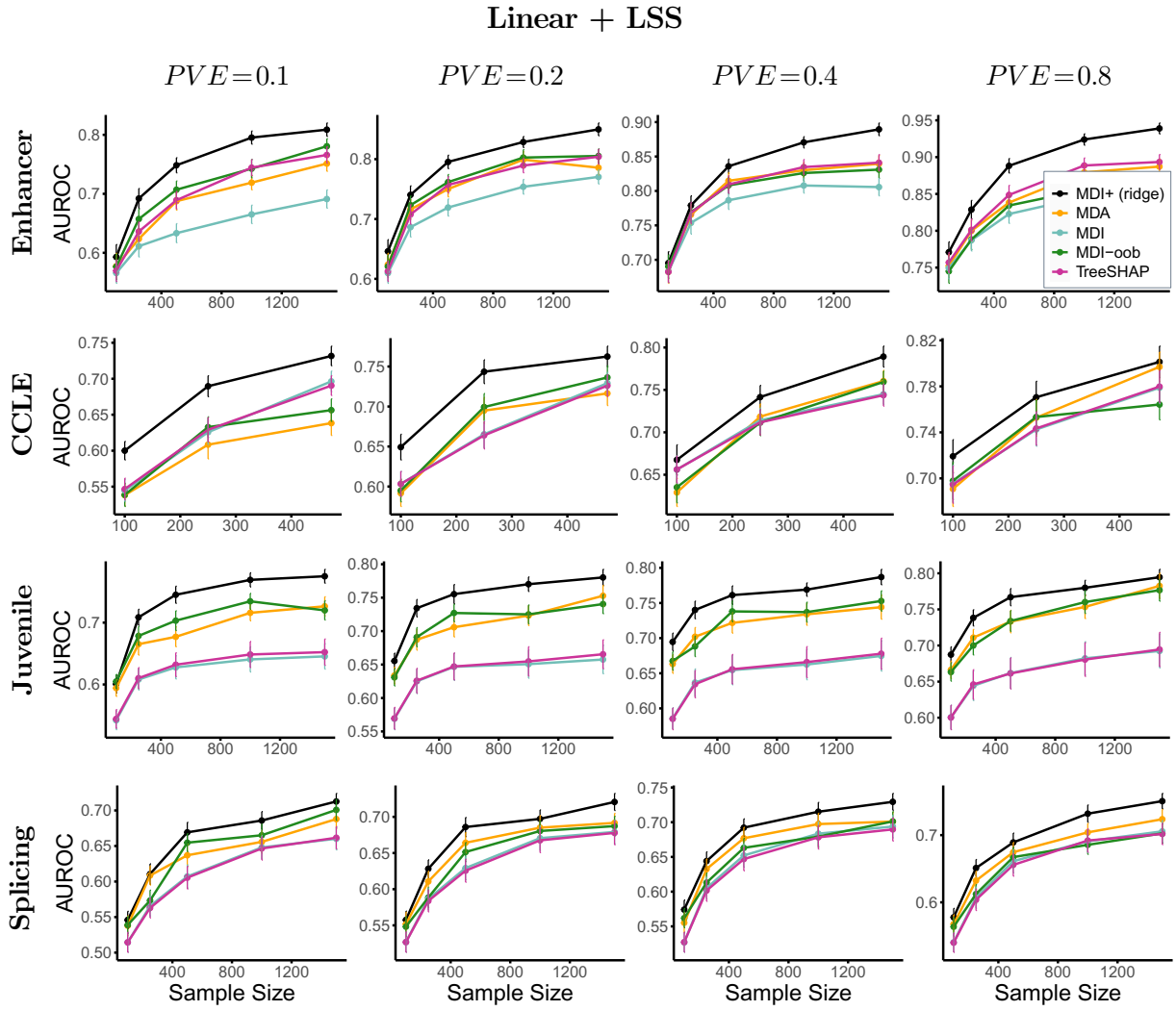


Figure 11: MDI+ (ridge) outperforms other feature importance methods for the linear + LSS regression function described in Section 5.1. This pattern is evident across various datasets (specified by row), proportions of variance explained (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE

B.3 Classification Simulations

Following the simulation setup discussed in Sections 5.1 and 5.3, we provide the results for all datasets and regression functions in the classification setting. We show the results for all datasets under the linear response in Figure 8, the LSS response in Figure 9, the polynomial interaction response in Figure 10, and the linear + LSS response in Figure 11. All regression functions in the classification setting were passed through the logistic link function to ensures that the responses are binary.

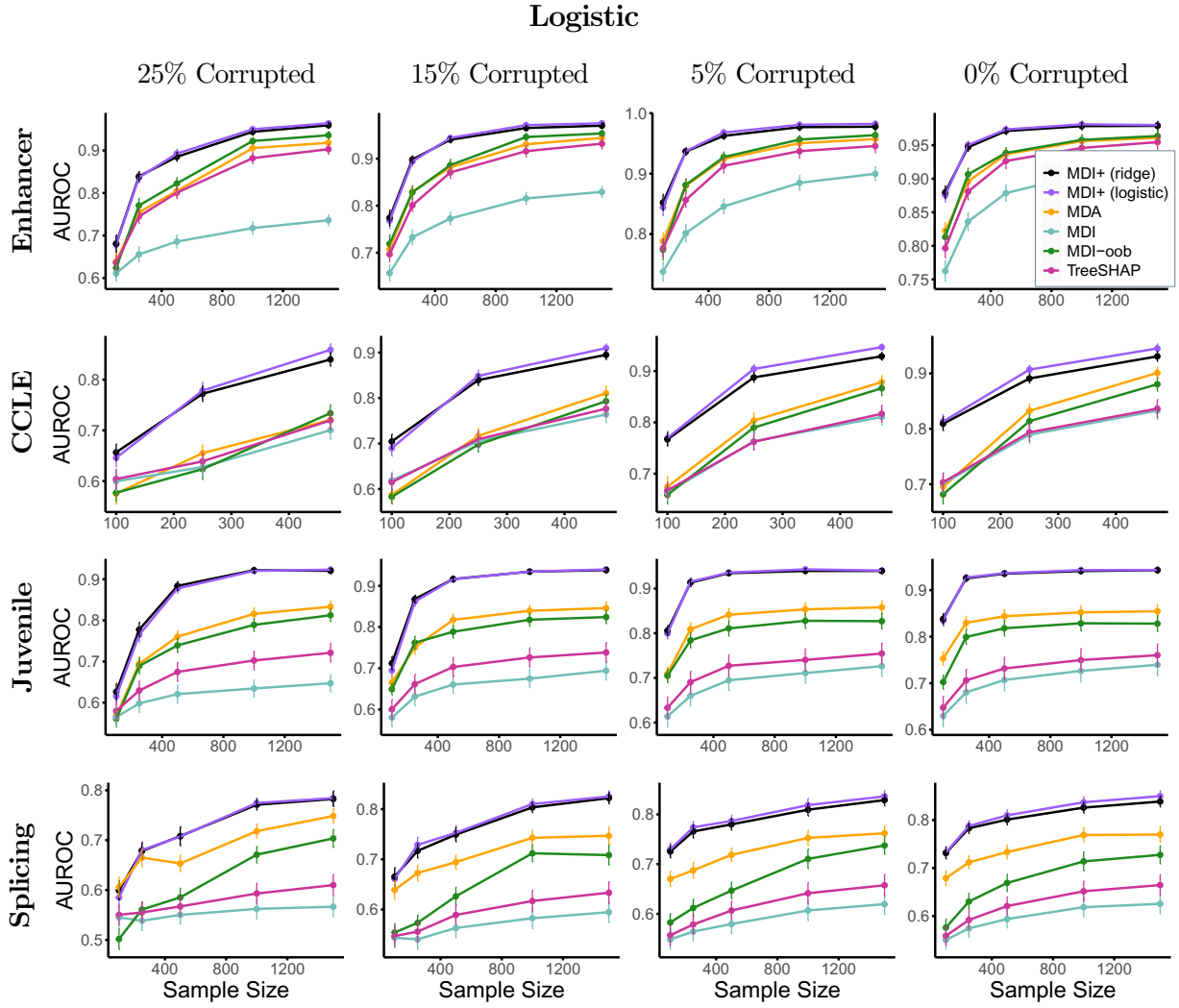


Figure 12: Both MDI+ (ridge) and MDI+ (logistic) outperform all other feature importance methods for the logistic linear regression function described in Section 5.1. This pattern is evident across datasets with different covariate structures (specified by row), proportions of corrupted labels (specified by column), and sample sizes (on the x-axis). Furthermore, MDI+ (logistic) often slightly outperforms MDI+ (ridge), indicating the benefit of tailoring the choices of MDI+ to the data at hand. In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

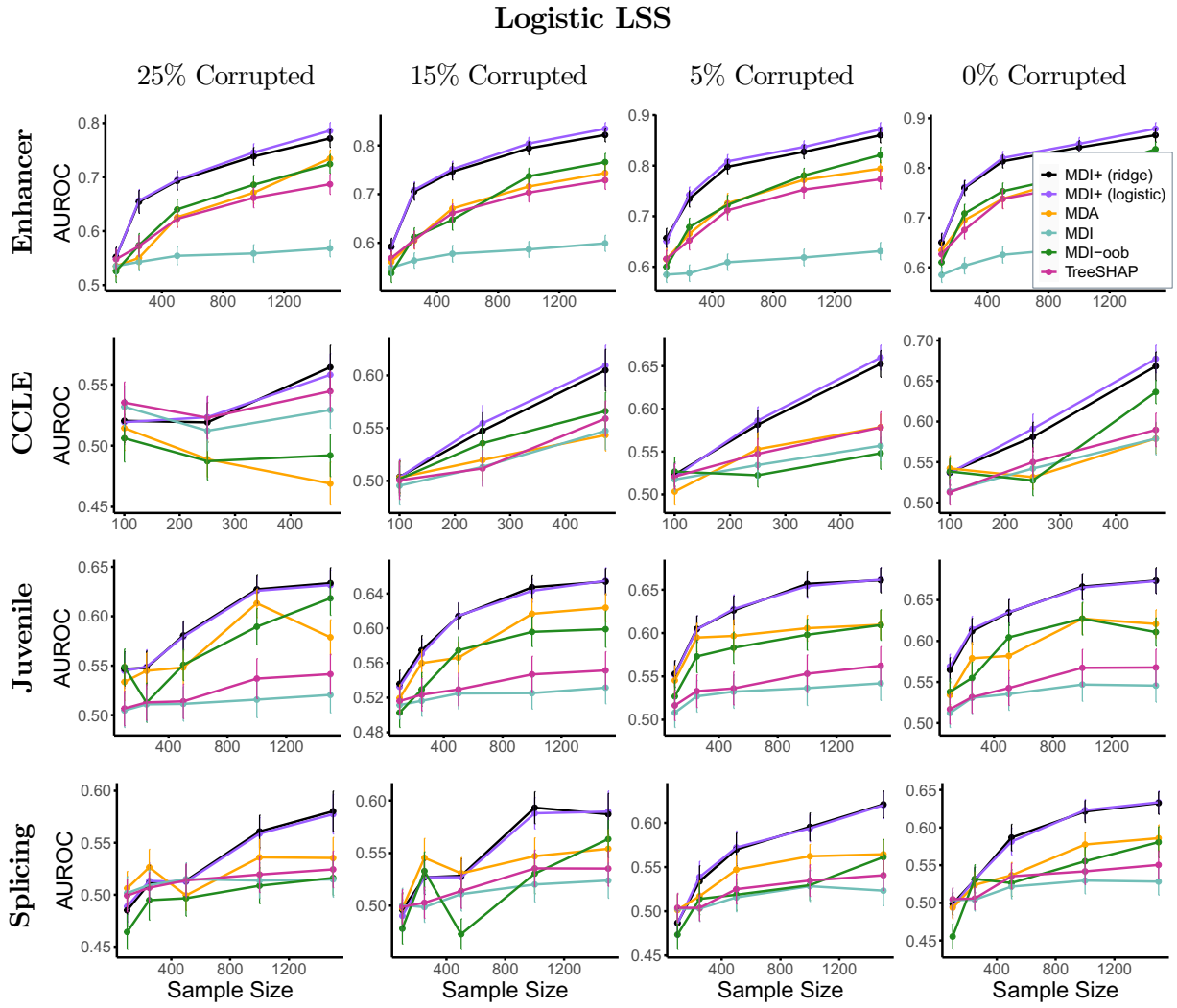


Figure 13: Both MDI+ (ridge) and MDI+ (logistic) outperform all other feature importance methods for the logistic LSS regression function described in Section 5.1. This pattern is evident across datasets with different covariate structures (specified by row), proportions of corrupted labels (specified by column), and sample sizes (on the x-axis). Furthermore, MDI+ (logistic) often slightly outperforms MDI+ (ridge), indicating the benefit of tailoring the choices of MDI+ to the data at hand. In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

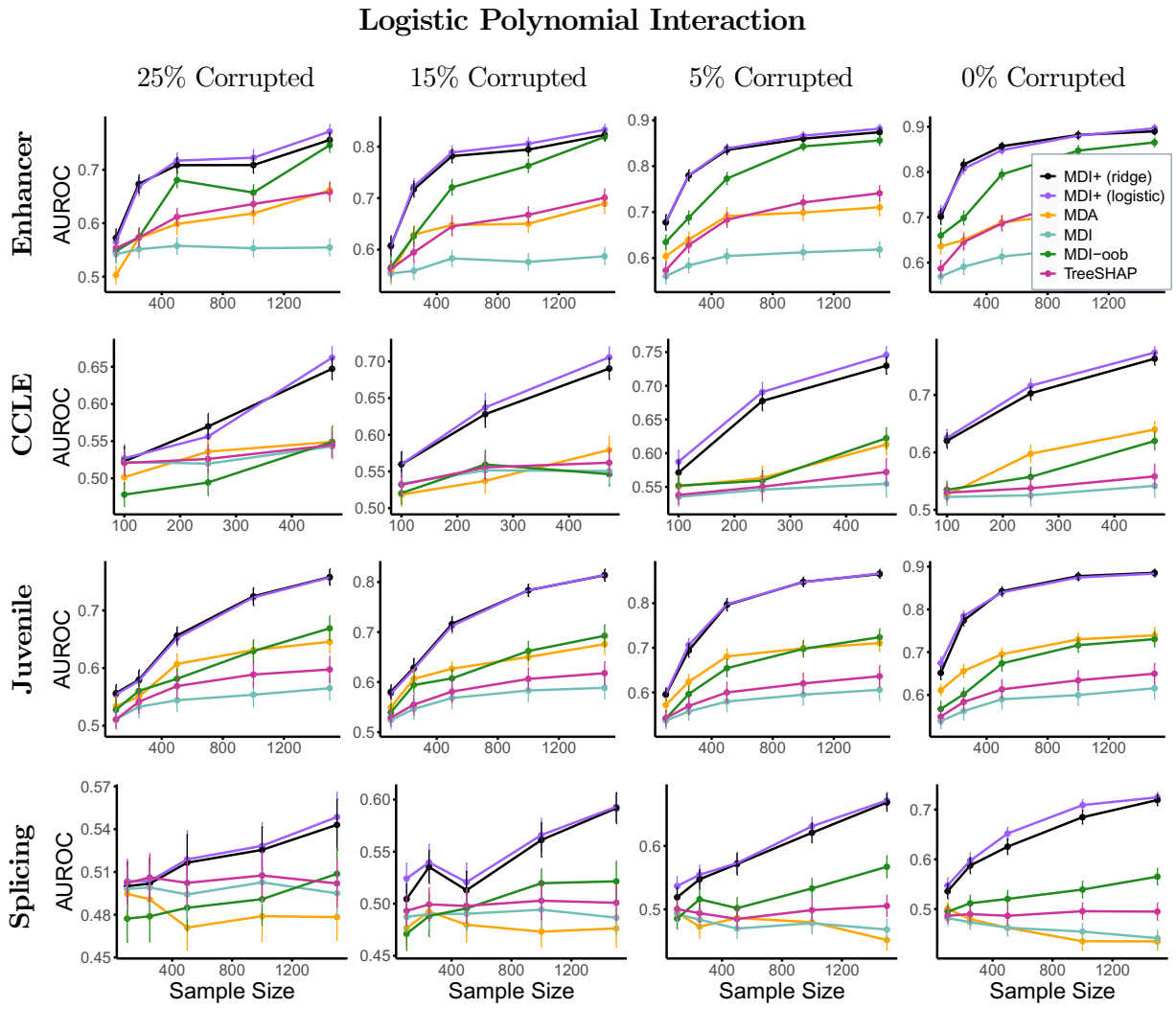


Figure 14: Both MDI+ (ridge) and MDI+ (logistic) outperform all other feature importance methods for the logistic polynomial interaction regression function described in Section 5.1. This pattern is evident across datasets with different covariate structures (specified by row), proportions of corrupted labels (specified by column), and sample sizes (on the x-axis). Furthermore, MDI+ (logistic) often slightly outperforms MDI+ (ridge), indicating the benefit of tailoring the choices of MDI+ to the data at hand. In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

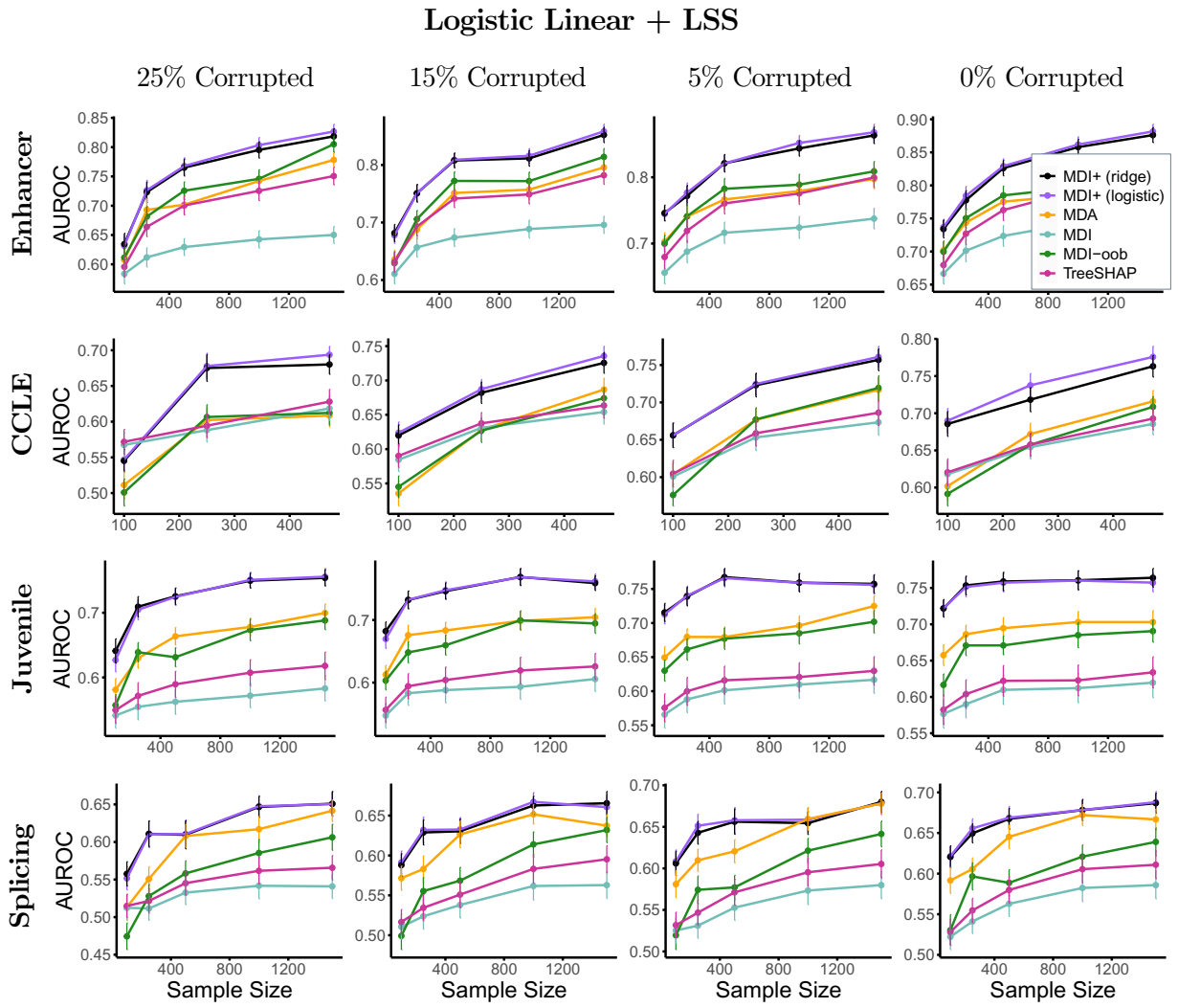


Figure 15: Both MDI+ (ridge) and MDI+ (logistic) outperform all other feature importance methods for the logistic linear + LSS regression function described in Section 5.1. This pattern is evident across datasets with different covariate structures (specified by row), proportions of corrupted labels (specified by column), and sample sizes (on the x-axis). Furthermore, MDI+ (logistic) often slightly outperforms MDI+ (ridge), indicating the benefit of tailoring the choices of MDI+ to the data at hand. In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

B.4 Robust Regression Simulations

Following the simulation setup discussed in Sections 5.1 and 5.4, we provide the results for the Enhancer and CCLE datasets under the linear response function in Figure 16, the LSS response function in Figure 17, the polynomial interaction response function in Figure 18, and the linear + LSS response function in Figure 19.

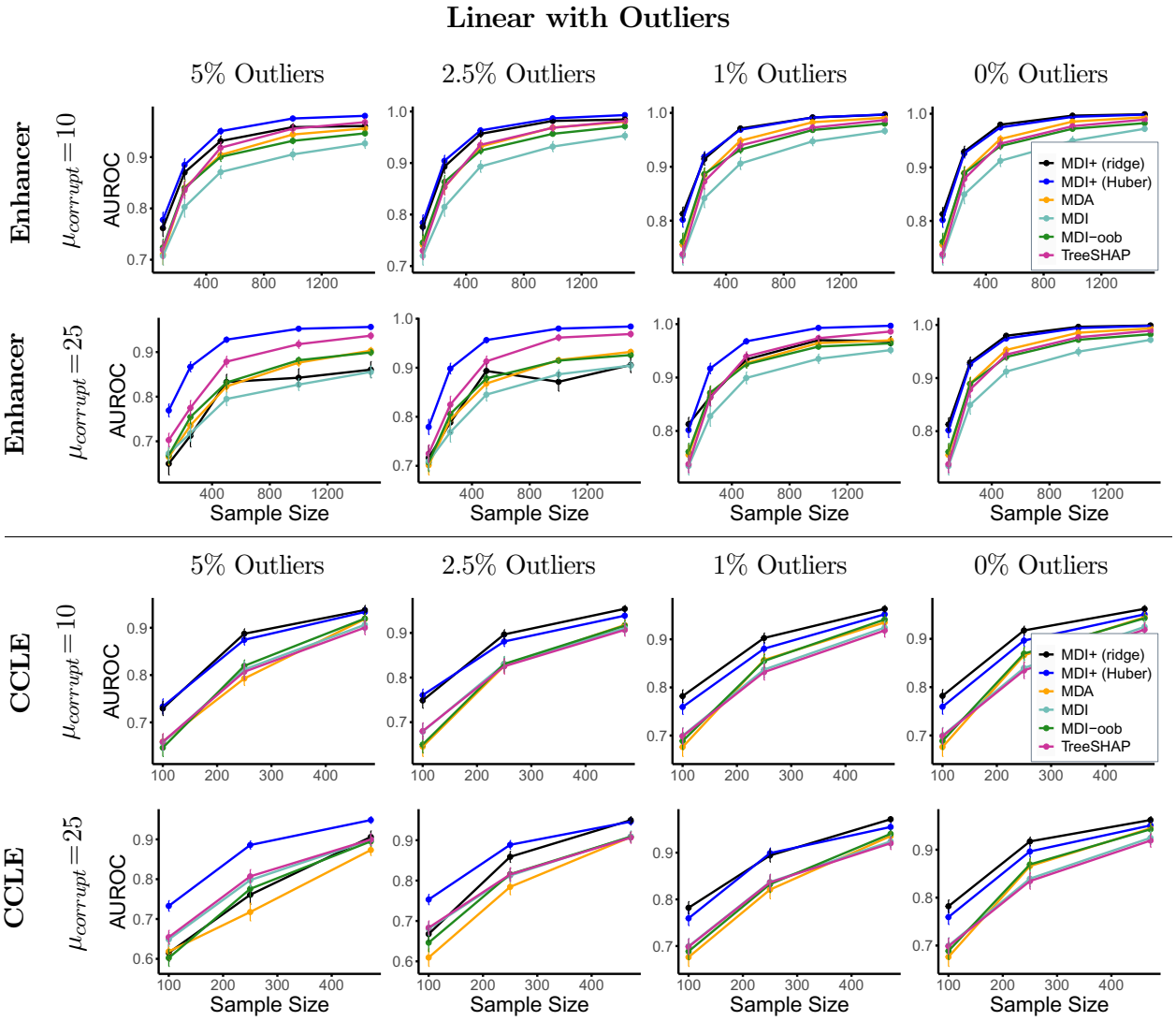


Figure 16: Under the linear with outliers regression setting (see Sections 5.1 and 5.4 for details), MDI+ (Huber)'s performance remains suffers far less than other methods including MDI+ (Ridge) as the level of corruption $\mu_{corrupt}$ (specified by row) and the proportion of outliers (specified by column) grow. This pattern also holds across datasets (specified by panel) and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

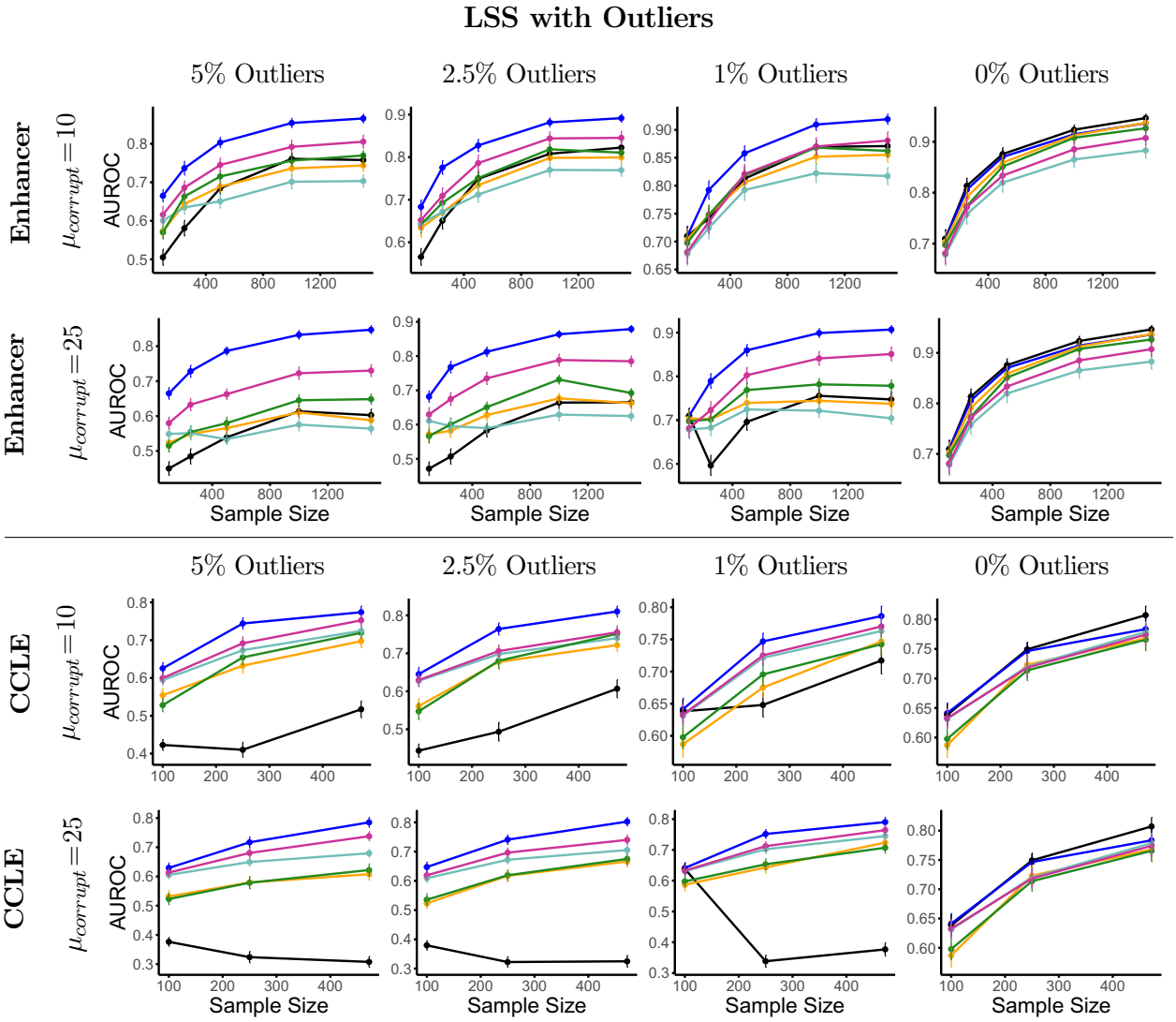


Figure 17: Under the LSS with outliers regression setting (see Sections 5.1 and 5.4 for details), MDI+ (Huber)'s performance remains suffers far less than other methods including MDI+ (Ridge) as the level of corruption $\mu_{corrupt}$ (specified by row) and the proportion of outliers (specified by column) grow. This pattern also holds across datasets (specified by panel) and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

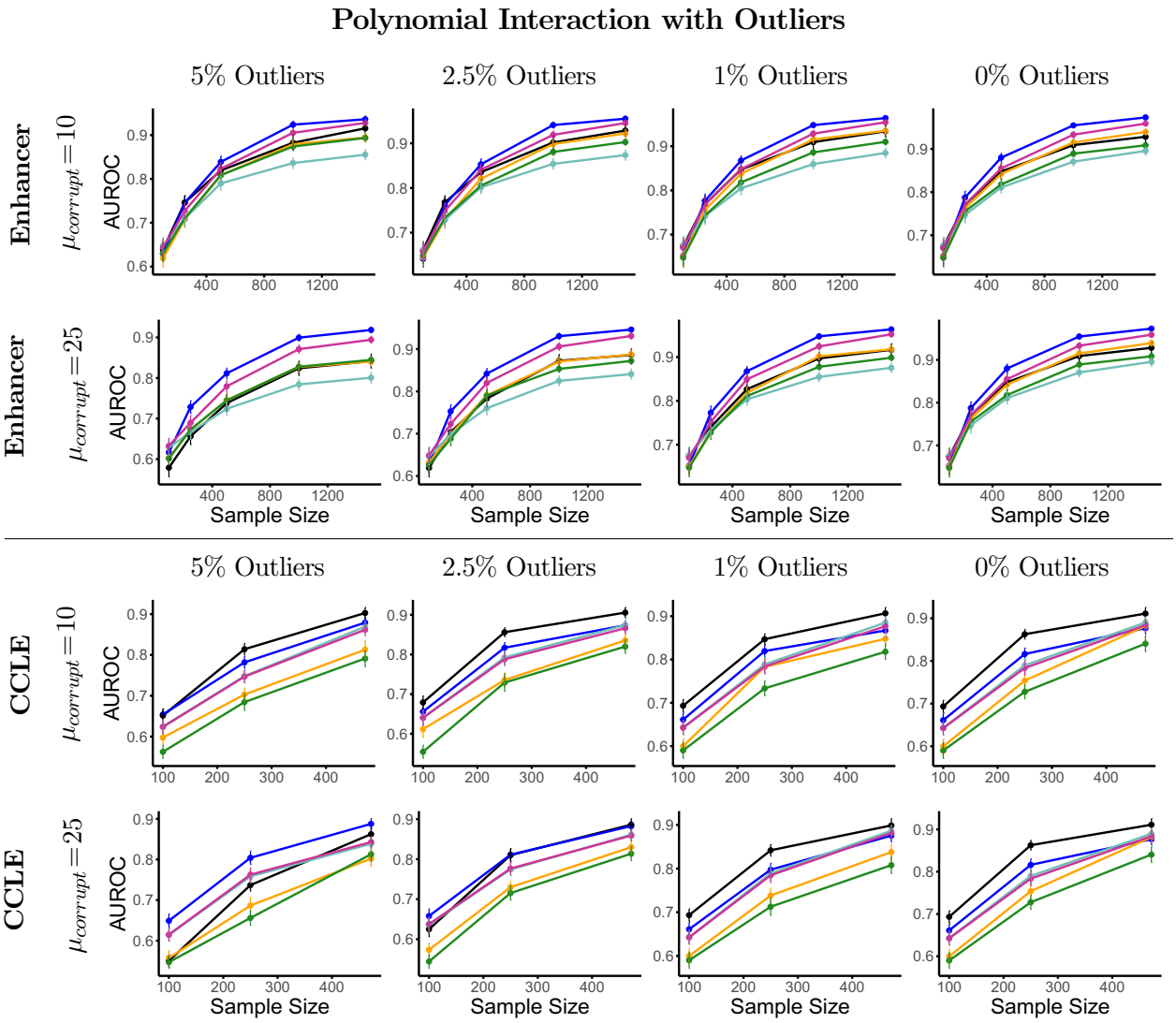


Figure 18: Under the polynomial interaction with outliers regression setting (see Sections 5.1 and 5.4 for details), MDI+ (Huber)'s performance remains suffers far less than other methods including MDI+ (Ridge) as the level of corruption $\mu_{corrupt}$ (specified by row) and the proportion of outliers (specified by column) grow. This pattern also holds across datasets (specified by panel) and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

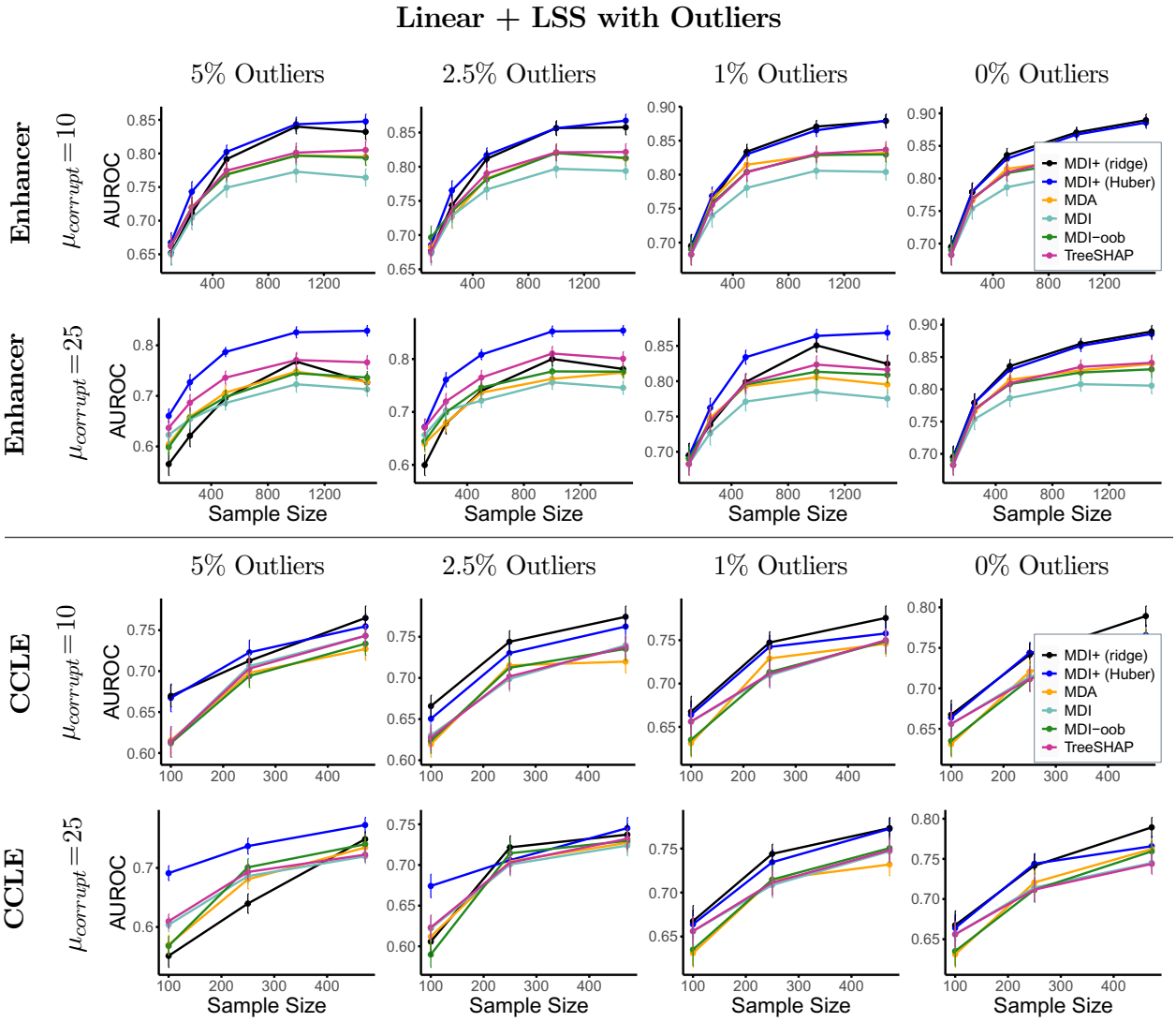


Figure 19: Under the linear + LSS with outliers regression setting (see Sections 5.1 and 5.4 for details), MDI+ (Huber)'s performance remains suffers far less than other methods including MDI+ (Ridge) as the level of corruption $\mu_{corrupt}$ (specified by row) and the proportion of outliers (specified by column) grow. This pattern also holds across datasets (specified by panel) and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE.

C Additional Data-Inspired Feature Ranking Simulations

In this section, we describe other data-inspired simulations we perform to establish the efficacy of MDI+ in a number of settings. Specifically, we perform simulations under a misspecificied setting (e.g., presence of unobserved features), under varying levels of sparsity of the generating function, and under varying number of features in the covariate matrix.

C.1 Misspecified Regression Simulations

In practice, we are often unable to observe all covariates that are relevant for the response [80]. To investigate the performance of MDI+ under this type of misspecified model scenario, we consider the following simulation setup.

Experimental details. We simulate data according to the four regression functions described in Section 5.1 (i.e., linear, LSS, polynomial interaction, and linear+LSS) but omit the first two signal features (i.e., X_1,X_2) from the covariate matrix $\mathbf{X}$ before fitting the RF and feature importance method under study. For example, in the linear regression simulation, we simulate the response y via $Y = \sum_{j=1}^{5} X_j + \varepsilon$, where $\varepsilon \overset{iid}{\sim} N(0,\sigma^2)$; however, we fit the RF and compute the feature importance measure using only y and X_3,...,X_p. The rest of the experimental details are identical to those described previously.

Results. For each of the four regression functions, the AUROC results under the misspecified model regime are summarized in Figures 20-23. In terms of the AUROC, MDI+ improves the ranking performance compared to the existing methods under this misspecified model scenario across a variety of regression functions, datasets with different covariate structures, proportions of variance explained, and sample sizes. Note here that the AUROC is computed with respect to only the observed covariates and ignore the omitted variables X_1 and X_2.

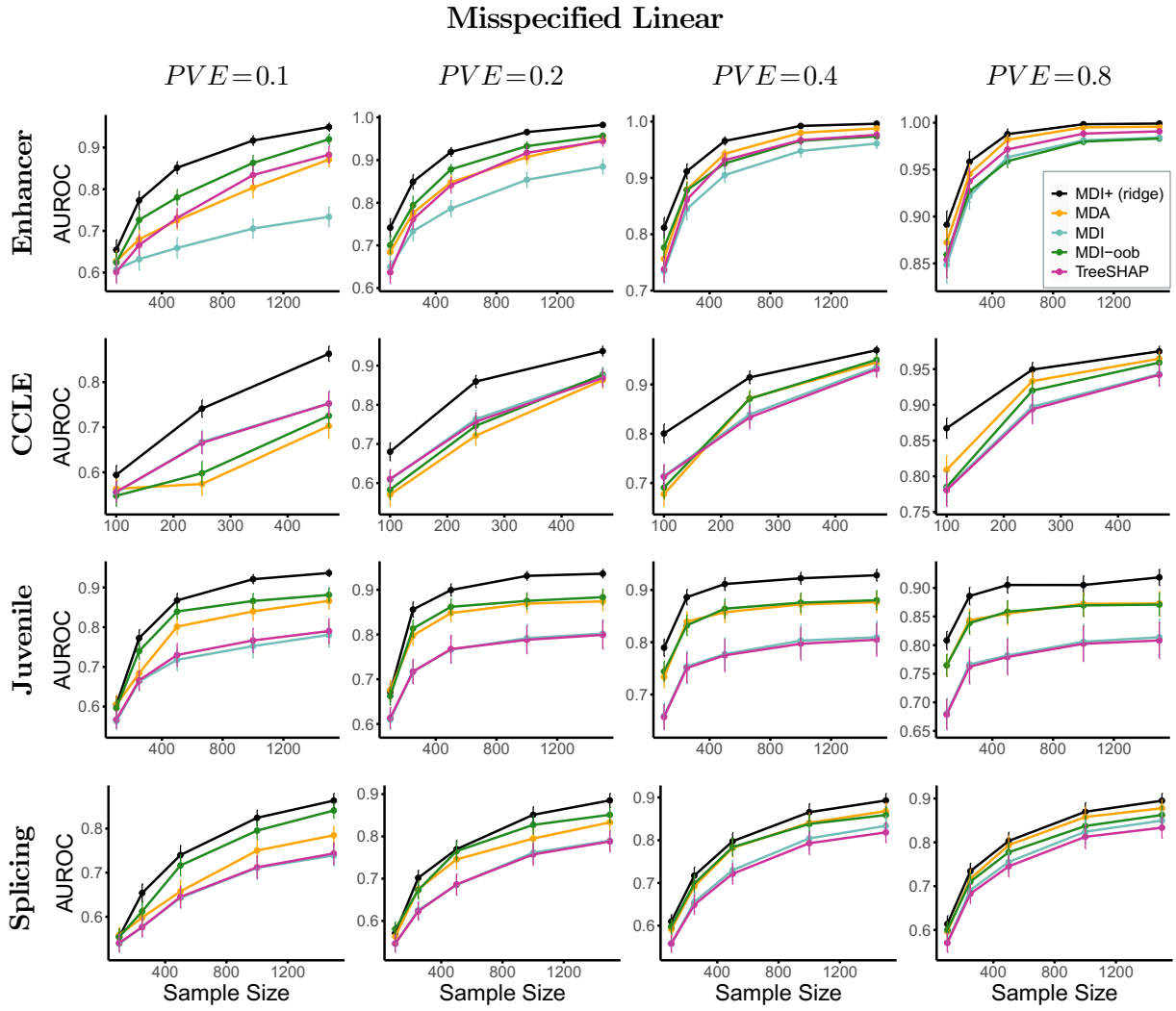


Figure 20: MDI+ (ridge) outperforms other feature importance methods across almost all misspecified linear regression simulations described in Appendix C.1. This pattern is evident across various datasets with different covariate structures (specified by row), proportions of variance explained (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE

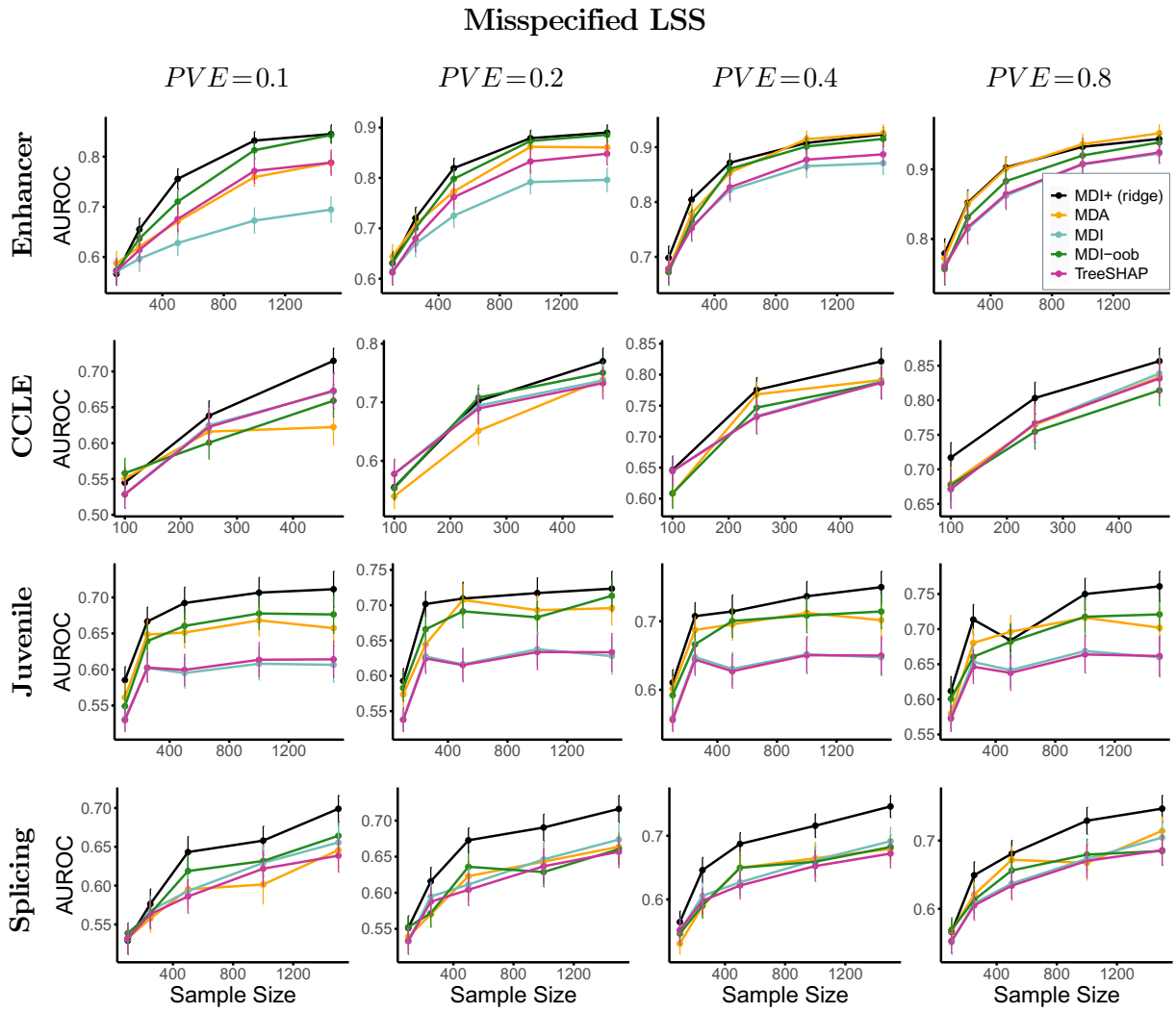


Figure 21: MDI+ (ridge) outperforms other feature importance methods across almost all misspecified LSS regression simulations described in Appendix C.1. This pattern is evident across various datasets with different covariate structures (specified by row), proportions of variance explained (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE

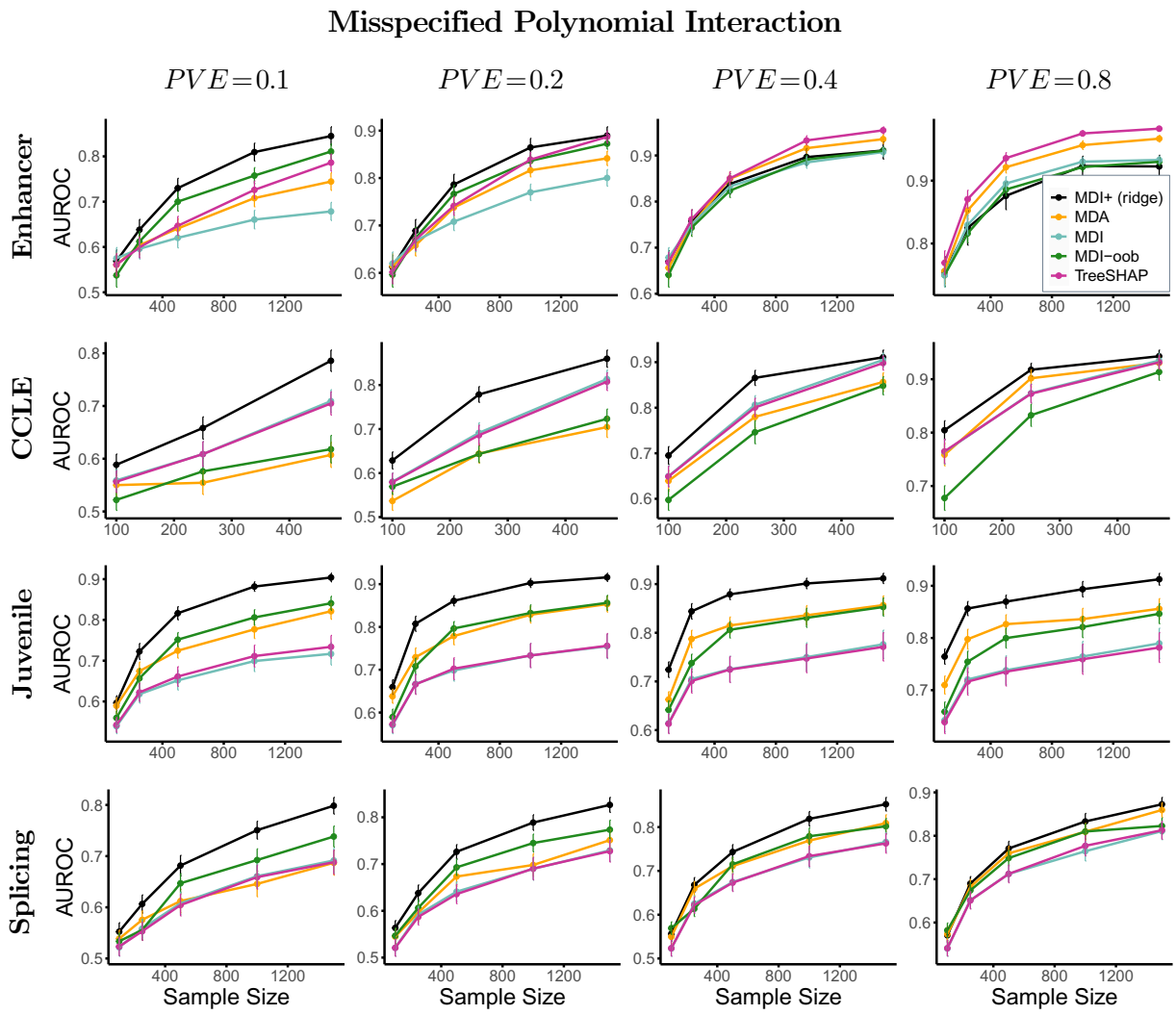


Figure 22: MDI+ (ridge) outperforms other feature importance methods across almost all misspecified polynomial interaction regression simulations described in Appendix C.1. This pattern is evident across various datasets with different covariate structures (specified by row), proportions of variance explained (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE

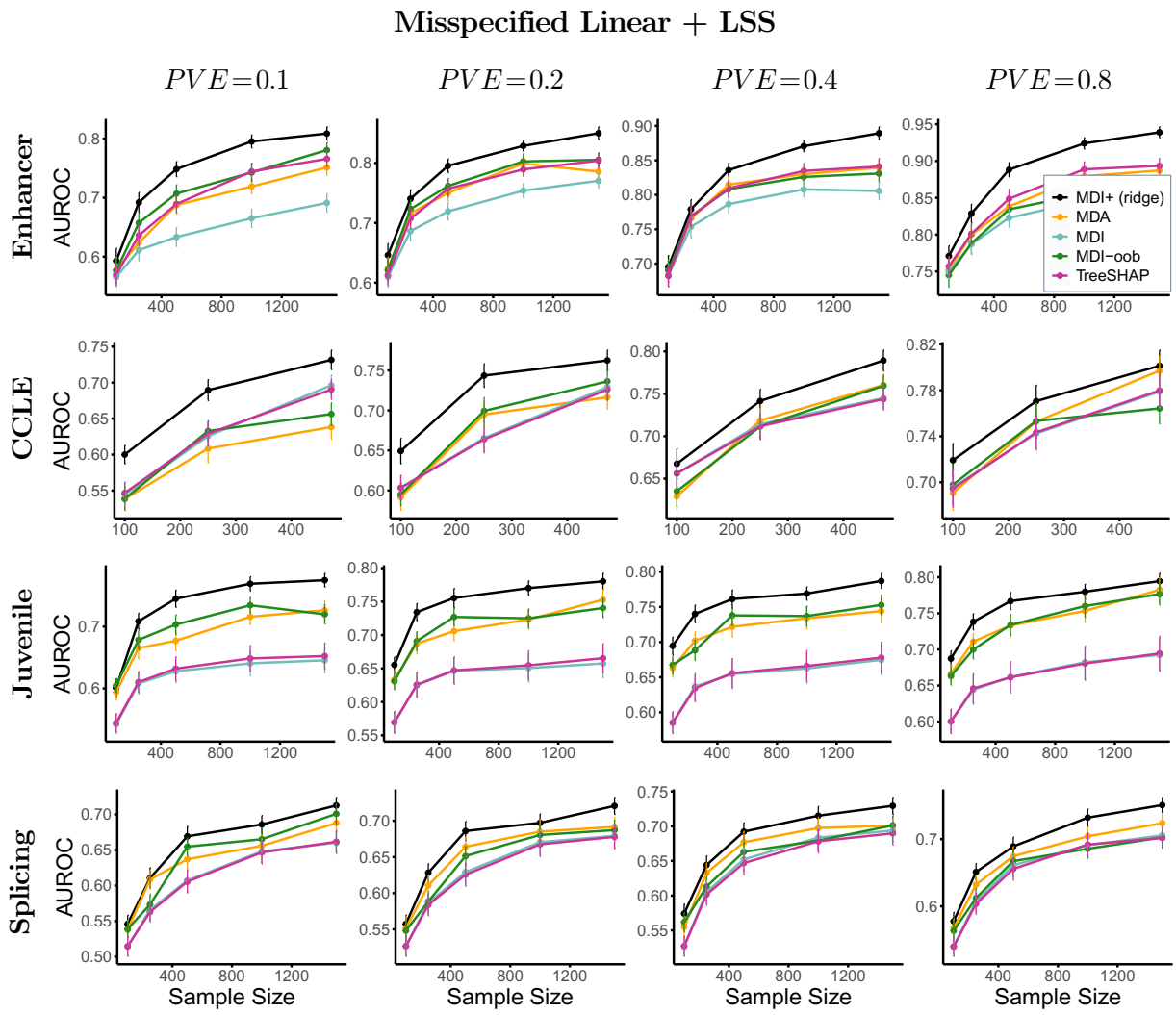


Figure 23: MDI+ outperforms other feature importance methods for the linear + LSS regression function described in Section 5.1. This pattern is evident across various datasets (specified by row), proportions of variance explained (specified by column), and sample sizes (on the x-axis). In all subplots, the AUROC has been averaged across 50 experimental replicates, and error bars represent $\pm$ 1SE

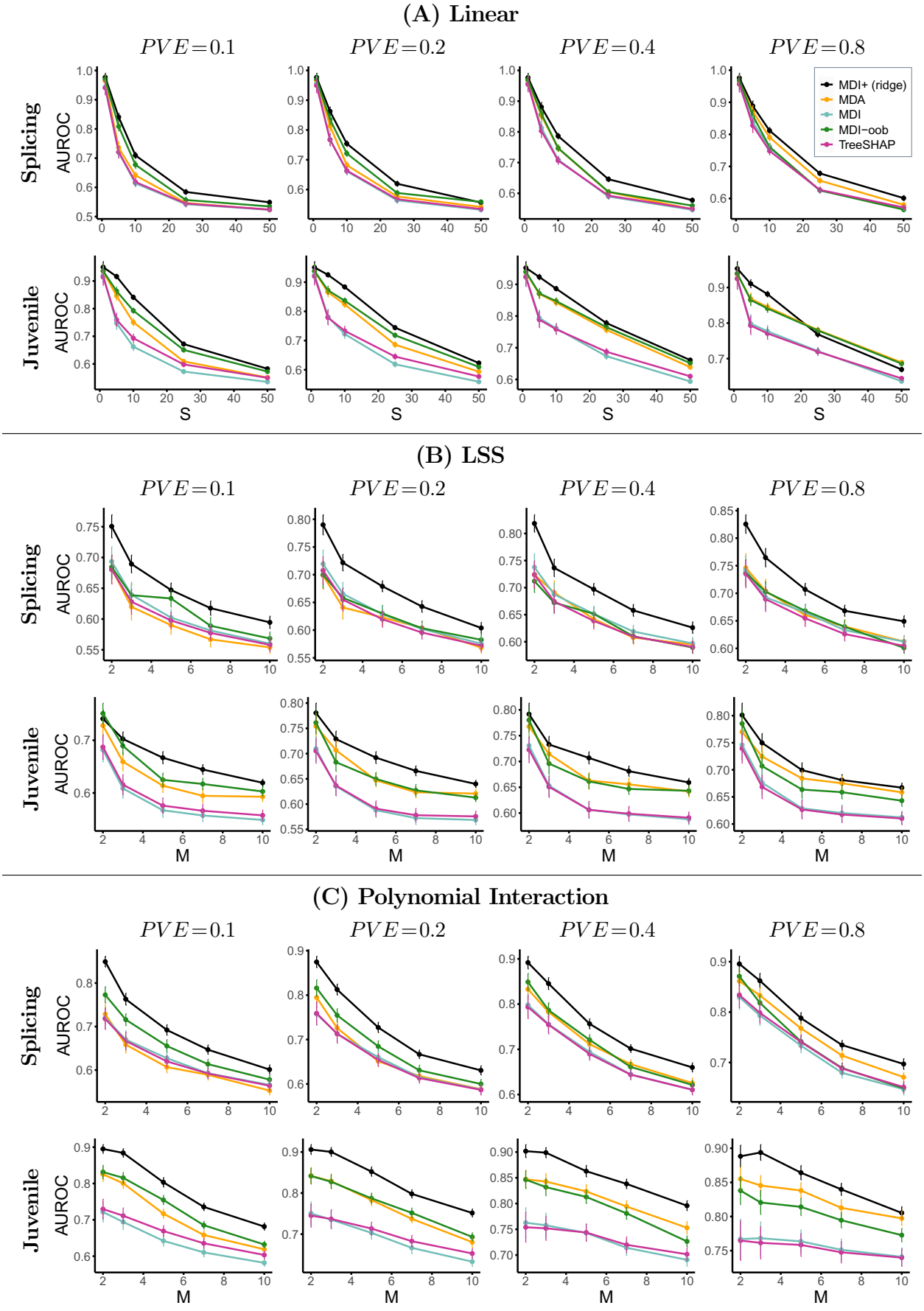
C.2 Varying Sparsity Simulations

In this section, we compare the performance of MDI+ against its competitors as we vary the sparsity level of the regression function.

Experimental details. Using the Juvenile and Splicing datasets as the covariate matrices $\mathbf{X}$, we simulate the responses $\mathbf{y}$ according to the four regression functions described in Section 5.1 (i.e., linear, LSS, polynomial interaction, and linear+LSS), and we vary the sparsity level of the regression function. Specifically, for the linear regression function, we vary S, which denotes the number of signal features used in the regression function. For the LSS, polynomial interaction,

and linear+LSS models, we vary M, which represents the number of interaction terms. For the simulations described in this section, we take $n = 1000$ samples and evaluate the performance across varying proportions of variance explained ($PVE = 0.1, 0.2, 0.4, 0.8$).

Results. Our results are summarized in Figure 24, which shows that MDI+ significantly outperforms competitors across various sparsity levels, as measured by AUROC.


(A) Linear
PVE=0.1
PVE=0.2
PVE=0.4
PVE=0.8
Splicing
Juvenile
AUROC
S
MDI+ (ridge)
MDA
MDI
MDI−oob
TreeSHAP
(B) LSS
M
(C) Polynomial Interaction

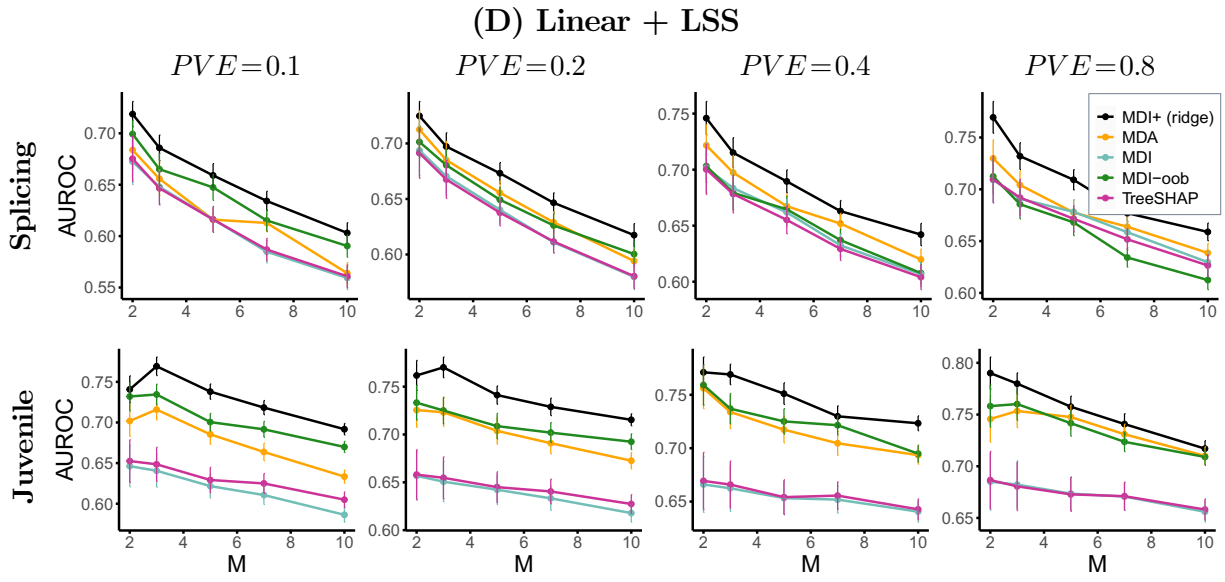


Figure 24: In the regression simulations described in Appendix C.2, MDI+ outperforms other feature importance methods across a variety of sparsity levels (specified on the x-axis). This pattern is evident across various regression functions (specified by panel), datasets with different covariate structures (specified by row), and proportions of variance explained (specified by column).

C.3 Varying Number of Features Simulations

In this section, we compare the performance of MDI+ against its competitors under a variety of real data-inspired simulations as we vary the number of features in the covariate matrix $\mathbf{X}$.

Experimental details. Using the CCLE gene expression dataset as the covariate matrix, we simulate the responses $\mathbf{y}$ according to the four regression functions described in Section 5.1 (i.e., linear, LSS, polynomial interaction, and linear+LSS), and we vary the number of features p in the covariate matrix $\mathbf{X}$. Specifically, for each choice of $p = 10,25,50,100,250,500,1000,2000$, we subsample the desired number of columns from the full CCLE gene expression dataset, which originally consists of 50,114 genes (or features). For the simulations described in this section, we take the max number of samples in the CCLE dataset ($n = 472$) and evaluate the performance across varying proportions of variance explained ($PVE = 0.1,0.2,0.4,0.8$).

Results. Our results are summarized in Figure 25, which shows that MDI+ significantly outperforms competitors in terms of AUROC, regardless of the number of features in the covariate matrix $\mathbf{X}$.

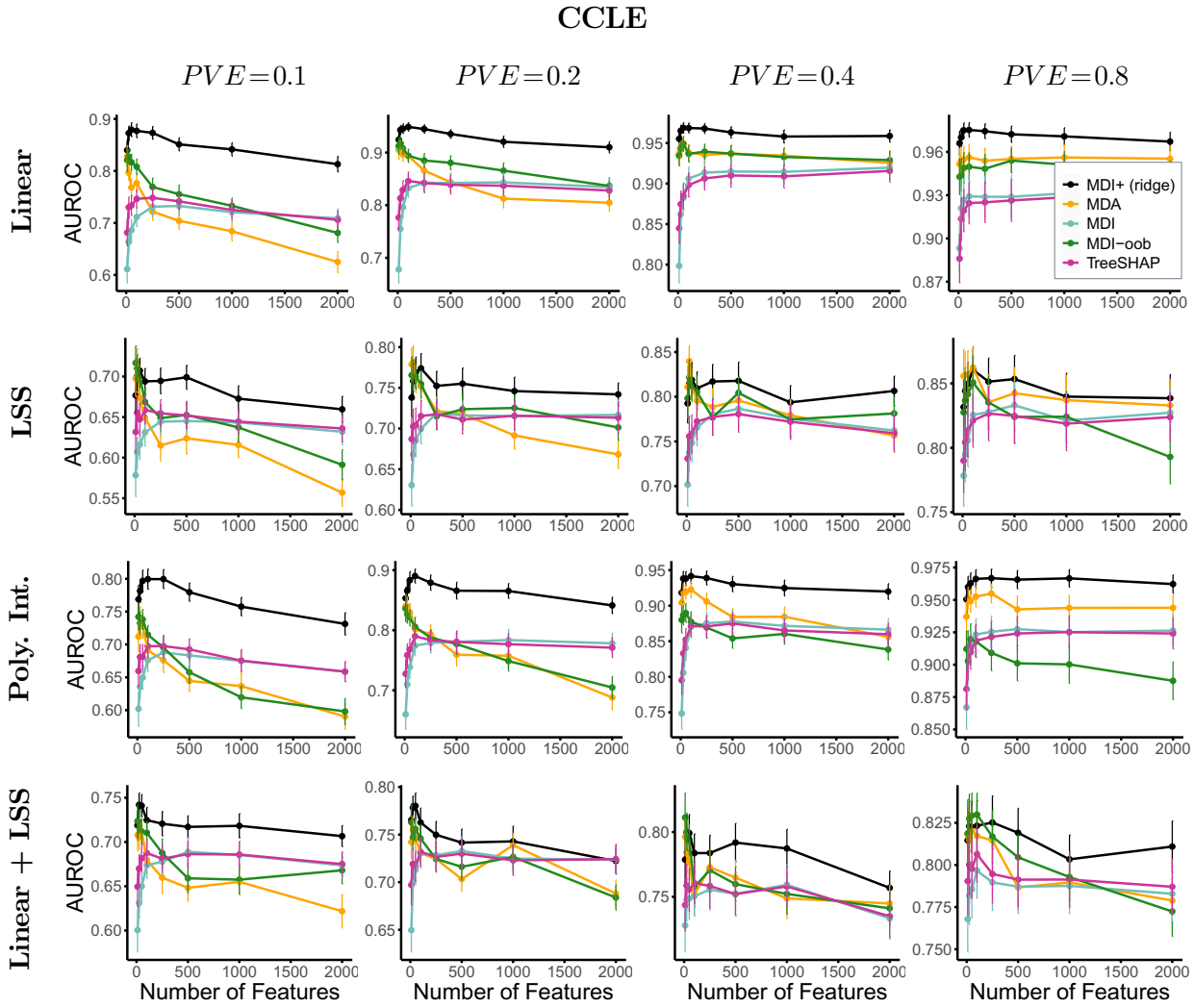


Figure 25: In the regression simulations described in Appendix C.3, MDI+ (ridge) outperforms other feature importance methods regardless of the number of features in the covariate matrix $\mathbf{X}$ (specified on the x-axis). This pattern is evident across various regression functions (specified by row) and proportions of variance explained (specified by column).

D Justifying MDI+ Framework Construction

In this section, we provide justification for using regularization, including the raw feature, and evaluating predictions via LOO within the MDI+ framework. That is, we provide simulations that show how the inclusion of each of these components in the MDI+ framework leads to an increase in feature ranking AUROC. Our experimental details and results are as follows.

Experimental details. We use the Enhancer and CCLE gene expression datasets as our covariate matrix $\mathbf{X}$, and simulate responses $\mathbf{y}$ according to the linear and polynomial interaction

function described in Section 5.1. As in Section 5.2, we vary the number of samples n across {100,250,500,1000,1500} for the Enhancer dataset and {100,250,472} for the CCLE gene expression dataset. We also vary PVE in {0.1,0.2,0.4,0.8}. In order to illustrate the impact of the choices in constructing the MDI+ framework, we consider the following sequence of models:

1. **MDI**: equivalent to MDI+ using OLS as the GLM without the raw feature and evaluating R^2 on the in-bag samples.

2. **MDI-oob**: equivalent to MDI+ using OLS as the GLM without the raw feature and evaluating R^2 on the out-of-bag samples.

3. **MDI+ (raw only)**: MDI+ using OLS as the GLM with the raw feature and in-bag evaluation.

4. **MDI+ (loo only)**: MDI+ using OLS as the GLM without the raw feature but using LOO evaluation on the entire dataset.

5. **MDI+ (raw+loo only)**: MDI+ using OLS as the GLM with the raw feature and LOO evaluation on the entire dataset.

6. **MDI+ (ridge+raw+loo)**: MDI+ using ridge as the GLM with the raw feature and LOO evaluation (i.e., the default MDI+ settings described in Section 5.2).

We perform the simulation with an RF regressor using *min_samples_leaf* = {1,5} alongside other default parameters and average the performance of each feature importance method across 50 simulation replicates. The results for *min_samples_leaf* = {1,5} are displayed in Figures 26 and 27 respectively.

Results. From Figures 26 and 27, MDI+ with the default regression settings, labeled in black as MDI+ (ridge+raw+loo), most consistently outperforms MDI, MDI-oob, as well as other MDI+ configurations in terms of the AUROC across the various regression functions, datasets, proportions of variance explained, and choices of *min_samples_per_leaf*. Moreover, when using an RF regressor with *min_samples_per_leaf*=5 in Figure 27, we see the added benefits of including the raw feature and/or using LOO evaluation within the MDI+ framework, compared to MDI which does not include the raw feature and uses in-bag evaluation. This point becomes more obscure when using an RF regressor trained to purity with *min_samples_per_leaf*=1. Here, MDI

outperforms MDI-oob, MDI+ (loo only), and MDI+ (raw+loo only) on the CCLE gene expression dataset, that is, in a small n, large p setting. We hypothesize that this is due to overfitting and thus high instability, given that the trees are being grown to purity. By incorporating shrinkage using ridge regression [1] as opposed to OLS as the GLM within the MDI+ framework, we are able to mitigate this instability and regain the added benefits of including the raw feature and LOO evaluation, as illustrated by the strong performance of MDI+ (ridge+raw+loo).

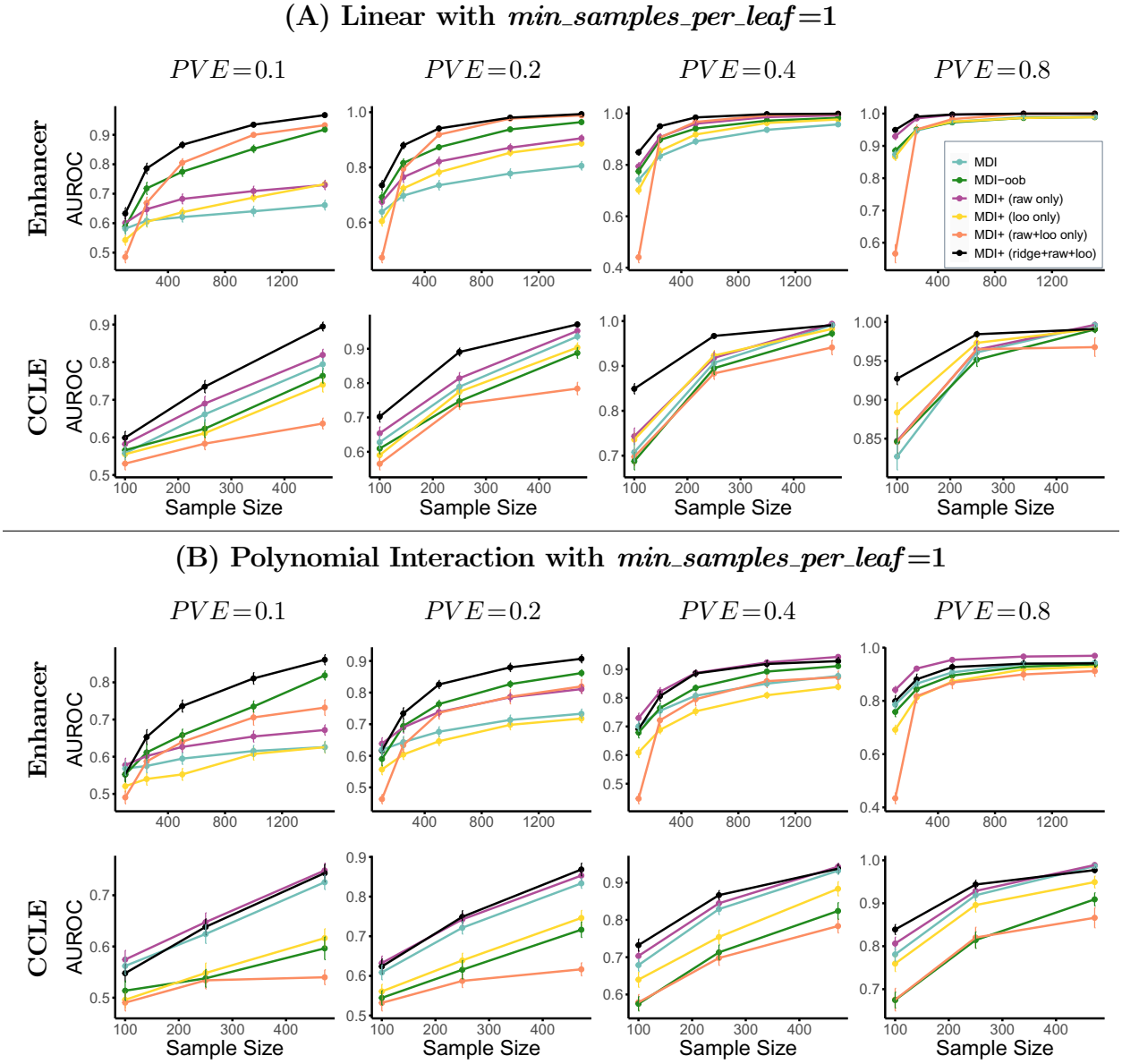


Figure 26: We illustrate the impact of various construction choices in the MDI+ framework, namely, regularization (via ridge regression), including the raw feature, and evaluating predictions via LOO, applied to an RF regressor with *min_samples_per_leaf*=1. MDI+ with ridge (i.e., shrinkage), including the raw feature, and LOO evaluation (black) consistently outperforms or is on par with other MDI+ construction choices across various regression functions (specified by panel), datasets with different covariate structures (specified by row), and proportions of variance explained (specified by column).

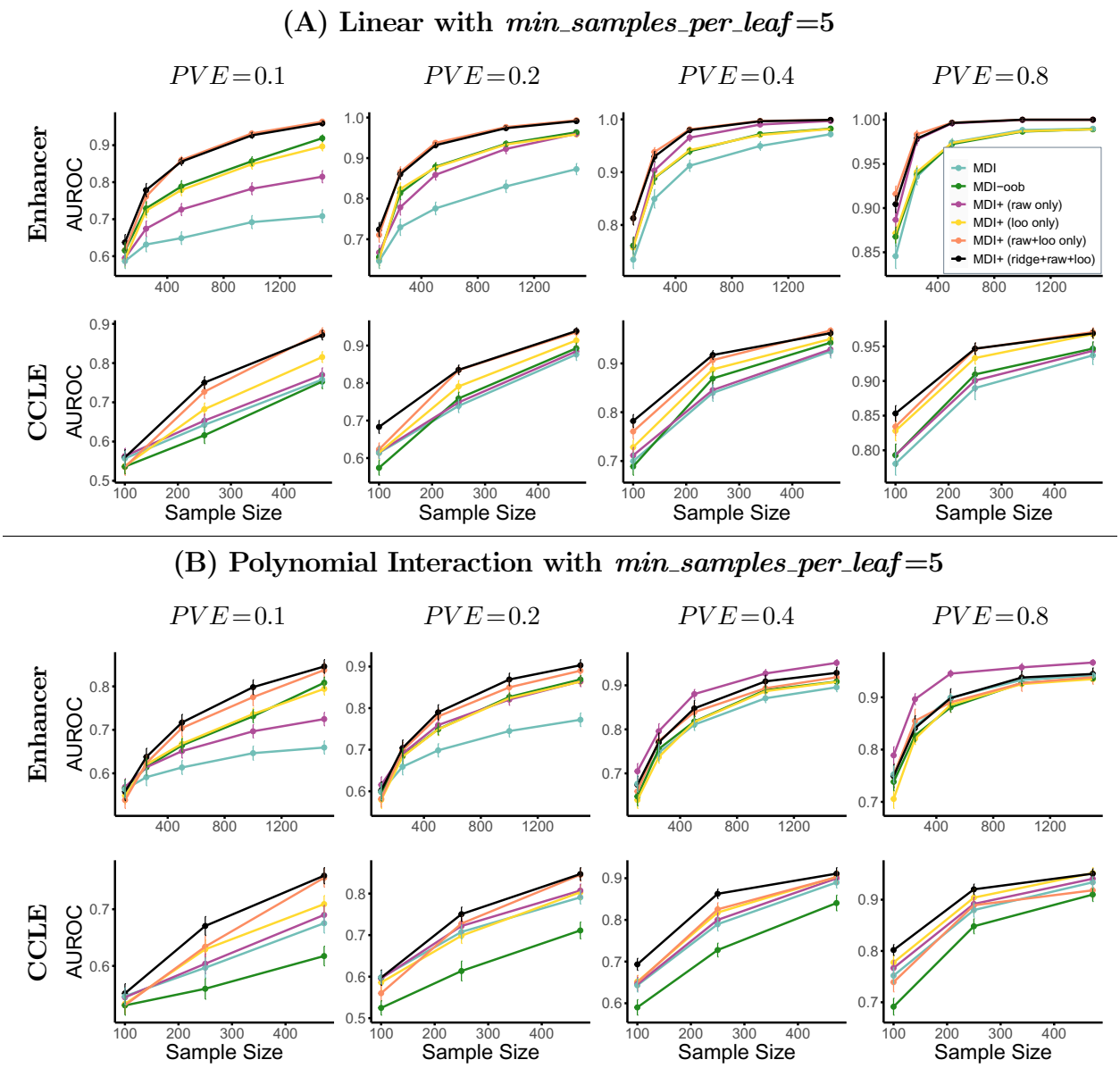


Figure 27: We illustrate the impact of various construction choices in the MDI+ framework, namely, regularization (via ridge regression), including the raw feature, and evaluating predictions via LOO, applied to an RF regressor with *min_samples_per_leaf*=5. MDI+ with ridge (i.e., shrinkage), including the raw feature, and LOO evaluation (black) consistently outperforms or is on par with other MDI+ construction choices across various regression functions (specified by panel), datasets with different covariate structures (specified by row), and proportions of variance explained (specified by column).

E MDI Biases Simulations

In this section, we provide additional simulation results regarding the biases of MDI against highly-correlated and low-entropy features, as well as the ability of MDI+ to overcome these biases. Throughout this section, we follow the simulation setup discussed in Section 6.

E.1 Correlation Bias

In Figure 28, we display the average percentage of RF splits per feature in each group (i.e., Sig, CSig, and NSig) under the correlation bias simulations described in Section 6.1. Further, in Figure 29, we examine the performance of MDI+ with and without the LOO data splitting scheme, denoted MDI+ (LOO) and MDI+ (in-bag), respectively. As seen in Figure 29, for both $PVE = 0.1, 0.4$, LOO sample-splitting overcomes the correlation bias that using in-bag samples induces.

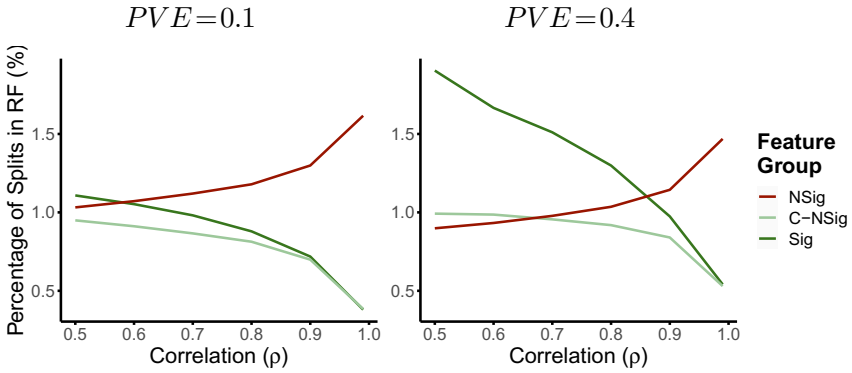


Figure 28: As the correlation increases, the percentage of splits in the RF that are made using features from the correlated group (Sig or C-NSig) decreases. This pattern is true for both $PVE = 0.1$ (left) and $PVE = 0.4$ (right) under the correlation simulation setup described in Section 6.1.

E.2 Entropy Bias

In Figure 30, we display the average percentage of RF splits per feature under the entropy bias simulations described in Section 6.2. Further, in Figure 31, we examine the performance of MDI+ with and without ridge regularization and the LOO data splitting scheme in both the regression and classification setting. As seen in Figure 31, LOO sample-splitting overcomes the entropy bias in both settings. In the regression setting, l_2 regularization also helps to mitigate the entropy bias of MDI (demonstrated by the difference between MDI and MDI+ (ridge, in-bag)).

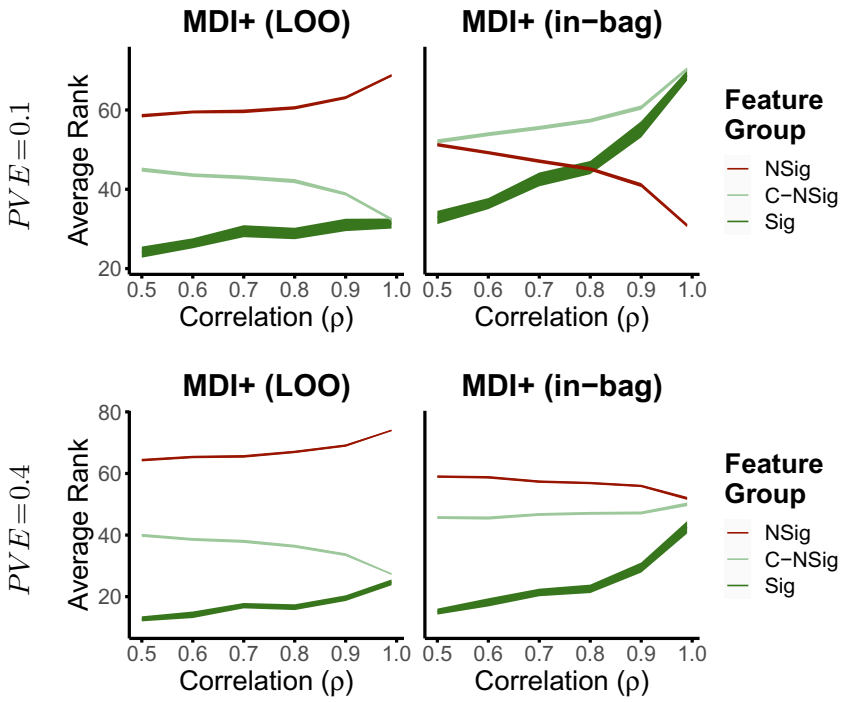


Figure 29: While MDI+ using the LOO partial model predictions is able to mitigate the correlated feature bias, MDI+ without the LOO scheme suffers from the correlated feature bias like MDI. This pattern holds for both the $PVE=0.1$ (top) and the $PVE=0.4$ (bottom) simulation settings described in Section 6.1.

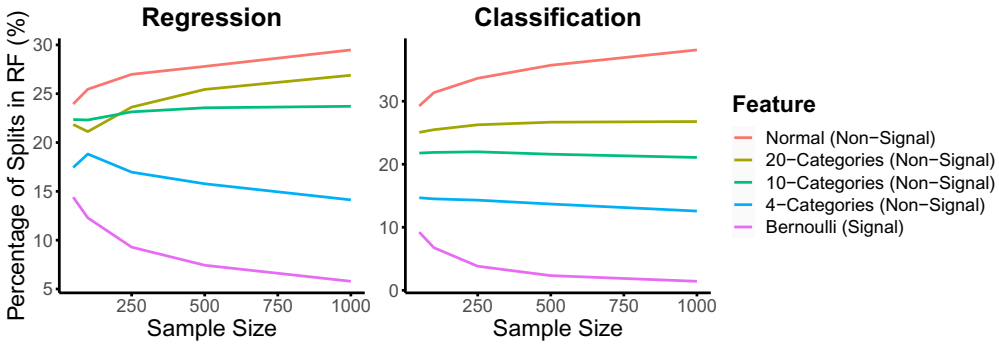


Figure 30: Noisy features with higher entropy are inherently split on more frequently in the RF. This pattern is true across various sample sizes under both the regression and classification simulation settings described in Section 6.2.

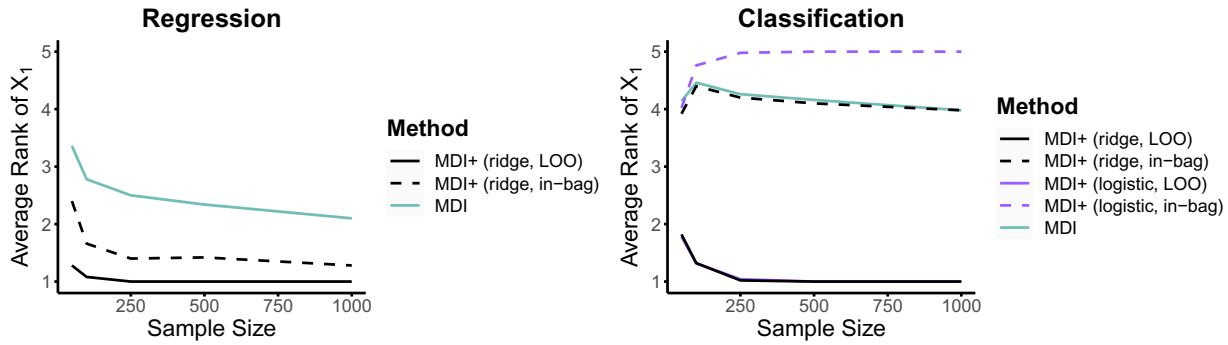


Figure 31: While MDI+ using the LOO partial model predictions is able to mitigate the entropy bias, MDI+ without the LOO scheme suffers from the entropy bias like MDI. This pattern holds for both the regression (left) and the classification (bottom) simulation settings described in Section 6.2.

F PCS-Informed Model Recommendation

In this section, we provide details on the stability-driven selection and model aggregation procedure discussed in Section 4.2. We also establish the efficacy of both approaches through a data-inspired simulation study.

For both approaches, the first step is to evaluate the prediction performance of each $h \in \mathcal{H}$ on an independent test set (obtained from a train-test split). Filter out all h whose predictive performance is worse than that of RF. Given this set of screened models $\{h_1,...,h_M\}$, we propose the following two approaches to compute feature rankings.

Stability-based selection for $\tilde{\Psi}$, $\mathcal{M}$, m. Let $\mathcal{T} = \{\mathcal{S}_1,...,\mathcal{S}_T\}$ denote the fitted trees of the RF trained on the data $\mathcal{D}_n$. Generate bootstrap samples of the fitted trees $\mathcal{T}_b$, $b = 1,...,B$. For each bootstrap sample, $\mathcal{T}_b$, compute feature rankings $R_b(h_i)$ for the MDI+ model h_i. Denote the set of feature rankings generated by h_i as $\mathcal{R}_i = \{R_1(h_i),...,R_B(h_i)\}$. To evaluate the stability of h_i, we measure the similarity between rankings from all pairs of the B bootstraps, and take the average across these $\binom{B}{2}$ similarity scores. To measure this similarity between two ranked lists, we use Rank-based Overlap (RBO) [79]. RBO measures how frequently two ranked lists agree on ordering of items, weighing agreement between higher ranks more heavily than lower ranks. This makes it appropriate for feature importance rankings where we are often concerned with the most important features (i.e., the highest ranking features). Finally, we choose h^* to be the $h \in \{h_1,...,h_M\}$ with the highest RBO score averaged across all $\binom{B}{2}$ pairs of bootstraps.

Model aggregation. While the approach above results in feature importance rankings that correspond to a single MDI+ model, it is sometimes desirable to obtain rankings that do not depend on a single choice of MDI+ model. Hence, a proposed alternative is an ensemble-based approach where we rank features based upon the median ranking of each feature X_k across all prediction-screened MDI+ models $h \in \{h_1,...,h_M\}$. We refer to this as MDI+ (ensemble). Other ways of aggregating feature importances can also be performed; we leave this to future work.

Simulation Study. We evaluate the effectiveness of these two PCS-informed model recommendation techniques for feature importance ranking via the following simulation study. We follow the simulation setup described in Section 5.2. We consider ridge and LASSO as the two (regularized) GLMs, and R^2 and mean absolute error (MAE) as the two metrics, producing a total of four candidate MDI+ models. We also aggregate these four candidate MDI+ models into an ensemble, denoted MDI+ (ensemble), by taking the median feature rankings as described above. We present here the results using the CCLE dataset with responses generated by a linear and polynomial interaction model as described in Section 5.1. We note that all four MDI+ models passed the predictive check. In Figure 32, we thus plot the RBO for all four MDI+ models, where a higher RBO indicates greater ranking stability. Additionally, we display the feature ranking performance (as measured by AUROC) for these four MDI+ models, MDI+ (ensemble), as well as MDI and MDI-oob. As seen in Figure 32, MDI+ (ridge, R^2) has the highest stability score, which often translates to the best feature ranking performance across various sample sizes and PVEs. MDI+ (ensemble) also performs reasonably well, typically yielding an AUROC for feature ranking accuracy in between the different MDI+ versions and better than the baselines of MDI and MDI-oob. Despite the stability-based selection providing the best feature ranking accuracy in these simulations, these simulations may not capture all subtleties of applying these methods in practice, and MDI+ (ensemble) may prove useful in practice.

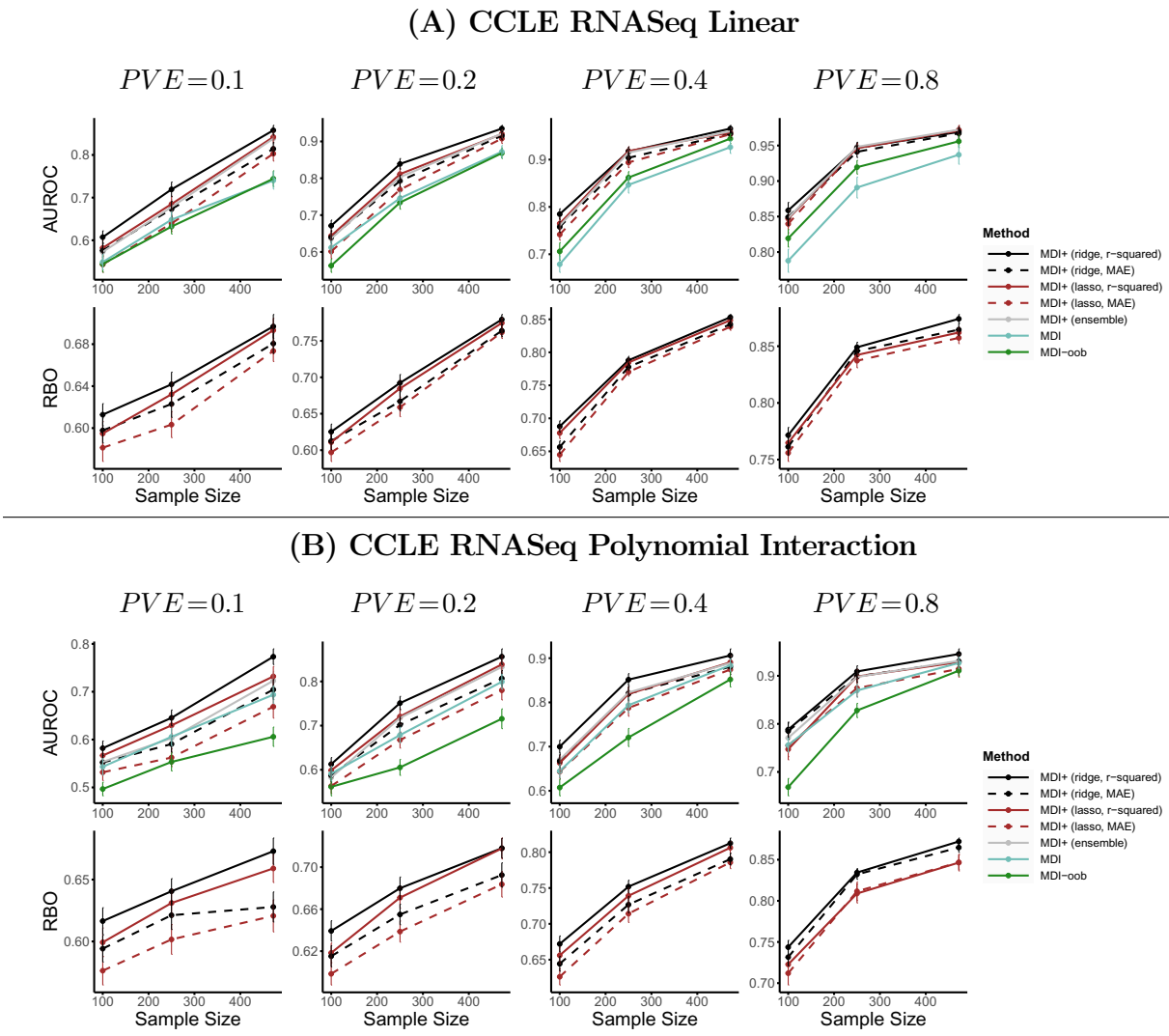


Figure 32: In each panel, the top row shows the accuracy of the feature importance rankings using various feature importance methods, and the bottom row shows the stability scores, as measured by RBO, for each choice of GLM and similarity metric used in MDI+. The GLM and similarity metric yielding the most stable feature rankings generally gives the most accurate feature importance rankings. MDI+ (ensemble) typically falls in between the different MDI+ versions and outperforms existing methods (MDI and MDI-oob) in terms of feature importance ranking accuracy. These patterns hold across various choices of sample sizes (specified on the x-axis) and regression functions (specified by panel).

G RF+ Improves Prediction Performance

In this section, we show that RF+ increases prediction performance over RF for various real-world datasets in the regression and (binary and multi-class) classification settings. Alongside stability, predictability is a crucial prerequisite for interpretation [49]. The improved predictive performance of RF+ suggests it better fits the underlying DGP, giving additional credence to MDI+.

Regression. We use the CCLE RNASeq gene expression data described in Section 7 to predict the response for 24 drugs, each independently in separate regression problems. We split the data into 80% training and 20% testing. The prediction models under study are the default `scikit-learn` RF regressor and RF+ using ridge regression as the GLM and the R^2 metric. We evaluate the relative difference in prediction performance as measured by the R^2 between RF+ (ridge) and RF for each drug averaged across 32 different random train-test splits. We present the results for drugs, where the vanilla RF yielded an average test set $R^2 > 0.1$ in Figure 33(A). We use this prediction screening threshold to focus on models that have non-trivial predictive power. Here, RF+ (ridge) increases R^2 performance for 17 out of 18 drugs. Averaged across these 18 drugs, RF+ (ridge) increases performance by an average of 4.4% relative to RF. The full results including all drugs can be found in Figure 34.

Classification. We use the Enhancer, Splicing and Juvenile datasets described in Section 5.1. Additionally, we use gene expression data from The Cancer Genome Atlas (TCGA) (described in Section 7.2) to predict the PAM50 breast cancer subtypes (a multiclass classification problem with five different subtype labels) [2]. We split the data into 80% training and 20% testing. We use the default `scikit-learn` RF classifier and RF+ using l_2-regularized logistic regression as the GLM with the log-loss metric. We evaluate the relative difference in prediction performance as measured by the $F1$ score and AUPRC between RF+ (logistic) and RF, averaged across 32 different random train-test splits. The results are displayed in Figure 33(B). RF+ (logistic) increases the $F1$ score for three of the four datasets and has approximately the same $F1$ score for the splicing dataset. On average, RF+ (logistic) increases the $F1$ score by an average of 3.6% relative to RF. RF+ (logistic) also increases the AUPRC score for all four datasets, and on average by 2.3% relative to RF.

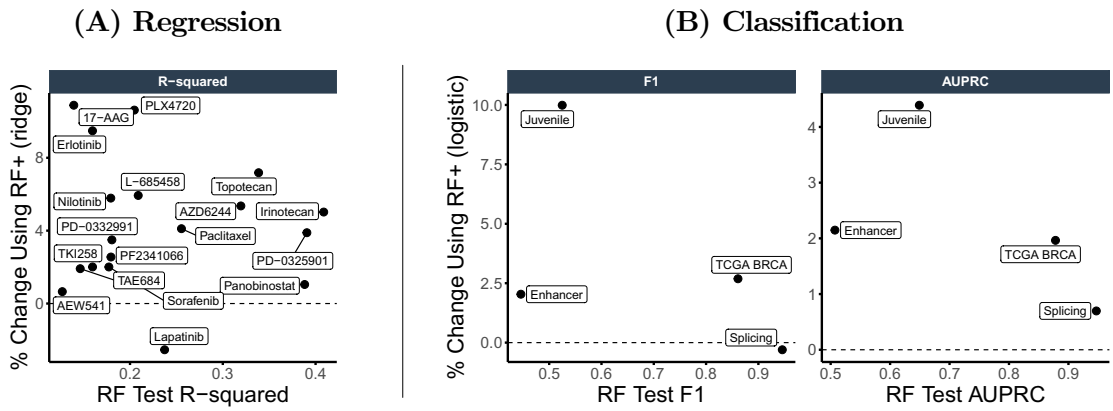


Figure 33: Relative performance of RF+ (ridge) as compared to RF in both the (A) regression and (B) classification settings. In the regression setting, RF+ (ridge) increases performance by an average of 4.4% averaged across all 18 drugs that have test set $R^2 > 0.1$. In the classification setting, RF+ (logistic) increases performance according to the $F1$ score for three of the four datasets, and on average by 3.6%. RF+ (logistic) increases AUPRC for all datasets, and on average by 2.3%.

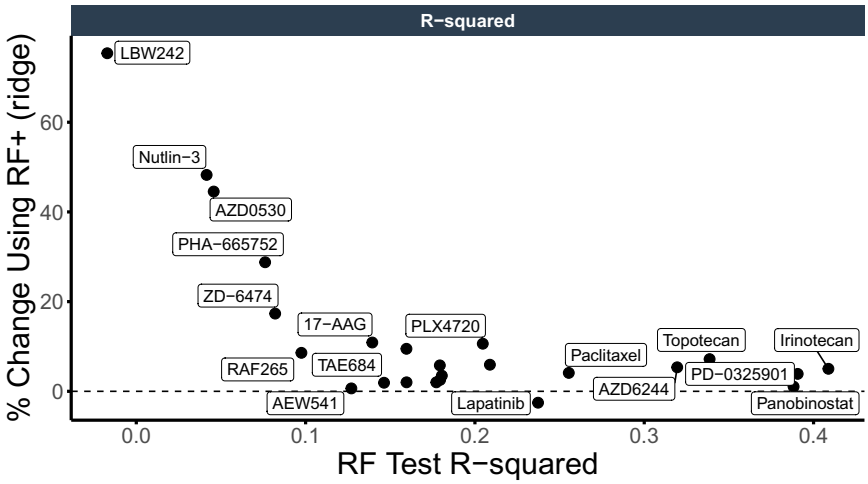


Figure 34: Relative performance of RF+ (ridge) as compared to RF for all 24 CCLE drugs in the regression setting. Here, we measure prediction performance using the test set R^2. Averaged across 32 train-test splits, RF+ (ridge) yields higher test prediction performance than RF for 23 out of 24 drugs.

H Case Studies

In this section, we provide additional background and results regarding the two case studies on drug response prediction and breast cancer subtype prediction.

H.1 Drug Response Prediction

Data preprocessing. Originally, the CCLE RNASeq gene expression data set consisted of 50114 genes. To reduce the number of features to a more manageable size for our analysis, we took the top 5000 genes with the highest variance. Screening based on these fast, marginal measurements (e.g., variance, correlation) is both common and often beneficial prior to a more in-depth analysis, especially in this ultra-high-dimensional regime. The unprocessed CCLE data can be downloaded from DepMap Public 18Q3 (https://depmap.org/portal/download/). The processed CCLE data used in our case study can be found on Zenodo at https://zenodo.org/record/8111870.

Details of drug response outcome variable. In this case study, the outcome of interest is the efficacy, or response, of a drug. To obtain a measure of the drug response for each cell line, the CCLE project measured the pharmacological sensitivity of each drug in vitro across eight different dosages and quantified this using the *activity area*, or area under the dose-response curve. This activity area is measured on an 8-point scale, with 0 corresponding to an inactive compound (i.e., a drug that did not inhibit the growth of the cancer cells across all 8 dosages) and 8 corresponding to a compound with 100% inhibition of cancer cell growth at all 8 dosages. Further details on the CCLE data can be found in Barretina et al. [4].

Results. To supplement the discussion in Section 7.1, we provide additional tables and figures below.

In Table 1, we list the top 5 gene expression predictors for each drug according to various feature importance methods. Since we evaluated the feature importance methods across 32 train-test splits (details in Section 7.1), the genes are ranked according to their average feature importance ranking across the 32 splits.

We also provide the stability distribution plots for all 24 drugs in Figure 35. Specifically, for the top 5 genes using each feature importance method under study (i.e., those listed in Table 1), we plot the standard deviation of the gene's feature importance rankings across the 32 train-test splits. A small standard deviation indicates a more stable feature importance measure. For many drugs, these stability distribution plots are skewed by the high instability of MDI-oob and MDA. We thus show the same plot in Figure 36, excluding MDI-oob and MDA. Across all drugs, we see that MDI+ generally yields the most stable rankings for the top 5 features across the 32 train-test splits.

To further evaluate the stability of MDI+ across only the randomness in the RF training, we

performed an analogous stability analysis in Figures 37 and 38, where we varied the random seed used to train the RF but kept the training data fixed. Specifically, we trained 32 RFs on the full CCLE data, each RF using a different random seed. We then evaluate the stability of the feature rankings across these 32 RF fits. We see in Figures 37 and 38 that MDI+ remains the most stable across the different RF fits. This highlights another practical advantage of MDI+, as it is highly undesirable for the feature importance rankings to change due to an arbitrary choice such as the random seed used in training the RF.

Lastly, in Figure 39, we evaluated the predictive power of the top k genes from each feature importance method across the 32 train-test (80-20%) splits. Specifically, for each k and each feature importance method, we took the top k ranked genes from the given feature importance method and trained an RF using the training data, restricted to only these k features. We then evaluated the prediction accuracy of the fitted RF on the test set and average these results across the 32 train-test splits. While evidence from the existing scientific literature remains the main source of validation, supporting the top-ranked genes from MDI+, the prediction accuracy of the top-ranked genes (as shown in Figure 39) can provide another check. In Table 2, we summarize the prediction results when $k=10$ by counting the number of drugs, for which each feature importance method gave the best test R^2, second-best test R^2, etc. We see that for 12 out of the 24 drugs, the top 10 genes from MDI+ (ridge) generally had the highest prediction power compared to other methods. In accordance with the predictability principle of the PCS framework, strong prediction performance suggests that the model (and in this case, the top-ranked features) may better capture the underlying data-generating process.

It is important to note, however, that correlated variables can pose subtle issues when interpreting these top k prediction results. For concreteness, consider the scenario where the top k ranked features from Method A tend to be independent while the top-ranked features from Method B tend to be highly-correlated. It is likely that the predictive power from the top k features from Method A is higher than that from Method B simply because the k independent features from Method A inherently contain more information than the k highly-correlated features from Method B. That is, the prediction accuracies from these two sets of k features are not directly comparable and requires additional investigation into the correlation structure of $\mathbf{X}$. From preliminary explorations, this complication is typically most apparent for small values of k. Intuitively, when k is small, correlation structures between two sets of features can be very

different. As k grows, the two correlation structures tend to become most similar to each other. This motivated our choice of $k=10$ in Table 2. However, further investigation is warranted, and we leave a deeper investigation of this phenomenon to ongoing and future work.

Table 1: Top 5 most important genes for each drug's response according to various feature importance methods. Genes are ranked by their average feature importance ranking across 32 train-test splits (shown in parentheses).

	MDI+	MDI	TreeSHAP	MDI-oob	MDA
17-AAG					
1	PCSK1N (1.47)	PCSK1N (2.19)	PCSK1N (2.19)	PCSK1N (2.5)	PCSK1N (9.16)
2	MMP24 (3.41)	MMP24 (4.97)	MMP24 (4.06)	NQO1 (9.03)	MMP24 (194.5)
3	RP11-109D20.2 (4.59)	ZSCAN18 (6.94)	ZSCAN18 (8.56)	ZNF667-AS1 (26.59)	ZNF667-AS1 (241.38)
4	ZSCAN18 (8.09)	RP11-109D20.2 (7.41)	RP11-109D20.2 (10.09)	ZSCAN18 (44.03)	RP11-109D20.2 (525.22)
5	NQO1 (8.84)	NQO1 (9.53)	NQO1 (11.81)	TST (49.38)	SH3BP1 (587.94)
AEW541					
1	TXNDC5 (1.41)	TCEAL4 (1.62)	TXNDC5 (1.75)	TXNDC5 (1.5)	TCEAL4 (5.59)
2	ATP8B2 (4.34)	TXNDC5 (3.53)	ATP8B2 (3.84)	ATP8B2 (4.69)	IQGAP2 (238.8)
3	VAV2 (6.03)	ATP8B2 (8.38)	VAV2 (5.41)	VAV2 (5.84)	RP11-343H19.2 (303.25)
4	TNFRSF17 (8.53)	VAV2 (10.47)	TCEAL4 (9.44)	TCEAL4 (6.5)	TXNDC5 (312.62)
5	TCEAL4 (9.03)	PLEKHF1 (19.25)	TNFRSF17 (9.69)	TNFRSF17 (13.56)	ATP8B2 (318.59)
AZD0530					
1	PRSS57 (5.16)	SYTL1 (17.62)	PRSS57 (7.69)	PRSS57 (12.09)	VTN (105.69)
2	SYTL1 (12.31)	PRSS57 (32.5)	SYTL1 (42.94)	DDAH2 (440.16)	SYTL1 (216.62)
3	STXBP2 (15.38)	SFTA1P (36.5)	NFE2 (43.06)	SLC16A9 (484.34)	STXBP2 (245.31)
4	NFE2 (23.12)	STXBP2 (62.59)	STXBP2 (61.62)	STXBP2 (486.09)	ZBED2 (466.48)
5	THEM4 (34.41)	CLDN16 (67.28)	SLC16A9 (61.81)	RAPGEF3 (514.2)	DDAH2 (472.06)
AZD6244					
1	LYZ (1.66)	TOR4A (2.31)	LYZ (1.72)	LYZ (2.53)	LYZ (2.09)
2	SPRY2 (2.34)	SPRY2 (3.31)	RP11-1143G9.4 (3.59)	SPRY2 (2.59)	RP11-1143G9.4 (3.59)
3	RP11-1143G9.4 (2.84)	LYZ (3.69)	SPRY2 (3.91)	TOR4A (3.25)	TOR4A (3.66)
4	ETV4 (5.22)	ETV4 (5.19)	TOR4A (6.41)	RP11-1143G9.4 (4.75)	SPRY2 (4.5)
5	TOR4A (6.41)	RP11-1143G9.4 (6.84)	RNF125 (6.66)	ETV4 (6.91)	ETV4 (6.34)
Erlotinib					
1	CDH3 (1.47)	CDH3 (1.84)	CDH3 (2.03)	CDH3 (1.97)	CDH3 (1.88)
2	RP11-615I2.2 (2.28)	RP11-615I2.2 (3.28)	RP11-615I2.2 (2.88)	RP11-615I2.2 (3.53)	RP11-615I2.2 (3.16)
3	EGFR (4.34)	SPRR1A (3.97)	EGFR (3.97)	SPRR1A (6.25)	SPRR1A (3.84)
4	SPRR1A (4.44)	SYTL1 (7.84)	SPRR1A (4.19)	GJB3 (8.78)	EGFR (8.31)
5	GJB3 (7.44)	EGFR (8.69)	KRT16 (11.41)	EGFR (9.31)	SYTL1 (8.72)
Irinotecan					
1	SLFN11 (1)	SLFN11 (1)	SLFN11 (1)	SLFN11 (1)	SLFN11 (1)
2	S100A16 (4.12)	S100A16 (3.75)	S100A16 (3.84)	S100A16 (3.25)	WWTR1 (6.03)
3	IFITM10 (4.19)	IFITM10 (4.09)	WWTR1 (4.28)	WWTR1 (4.12)	TRIM16L (150.38)
4	WWTR1 (4.94)	WWTR1 (4.78)	IFITM10 (8.03)	RP11-359P5.1 (8.22)	IFITM10 (163.44)
5	PPIC (7.81)	PPIC (10.22)	RP11-359P5.1 (8.47)	IFITM10 (8.41)	S100A16 (182.19)
L-685458					

Table 1: *(continued)*

	MDI+	MDI	TreeSHAP	MDI-oob	MDA
1	PXK (2.03)	DEF6 (4.62)	PXK (2.03)	PXK (3)	PXK (2.94)
2	DEF6 (4.28)	PXK (4.94)	DEF6 (4.38)	IKZF1 (3.44)	CXorf21 (4.62)
3	CXorf21 (4.84)	CXorf21 (5.44)	CXorf21 (5.62)	CXorf21 (5.75)	DEF6 (4.66)
4	IKZF1 (6.03)	IKZF1 (5.75)	IKZF1 (7.91)	DEF6 (10.31)	IKZF1 (5.47)
5	RP11-359P5.1 (9.09)	RP11-359P5.1 (9.94)	RP11-359P5.1 (12.81)	CTNNA1 (13.88)	RP11-359P5.1 (10.06)
LBW242					
1	SERPINB6 (1.12)	SERPINB6 (1)	SERPINB6 (1.66)	SERPINB6 (1.31)	SERPINB6 (1.56)
2	RGS14 (5.12)	RGS14 (6.66)	RGS14 (3.66)	GPT2 (45.62)	HERC5 (54.31)
3	HERC5 (5.41)	MAGEC1 (7.5)	MAGEC1 (5.53)	GBP1 (179.28)	ITGA1 (73.34)
4	MAGEC1 (7.62)	ITGA1 (10.53)	GBP1 (5.78)	ZNF32 (222.03)	PTGS1 (296.62)
5	GBP1 (8.22)	HERC5 (12.41)	CCL2 (13.5)	IGSF3 (257.88)	GPT2 (316.12)
Lapatinib					
1	ERBB2 (1.06)	ERBB2 (1.5)	ERBB2 (1.41)	ERBB2 (1.69)	ERBB2 (1.47)
2	PGAP3 (3.09)	NA (6.81)	PGAP3 (3.44)	PGAP3 (8.03)	NA (4.31)
3	NA (5.03)	PGAP3 (12.41)	IKBIP (6.19)	C2orf54 (13.09)	PGAP3 (14.03)
4	C2orf54 (6.91)	DPYSL2 (16.16)	NA (6.22)	DPYSL2 (15.41)	PKP3 (20.31)
5	IKBIP (8.28)	PKP3 (16.47)	C2orf54 (7.41)	EMP3 (20.47)	EMP3 (22.38)
Nilotinib					
1	SPN (1.25)	SPN (1.81)	SPN (1.62)	SPN (1.47)	SPN (3.59)
2	GPC1 (3.5)	GPC1 (4.38)	GPC1 (3.5)	GPC1 (3.44)	SELPLG (9.03)
3	TRDC (6.62)	SELPLG (7.5)	TRDC (10.16)	SELPLG (9.97)	KLF13 (26.22)
4	SELPLG (6.78)	KLF13 (16.97)	LMO2 (10.44)	TRDC (10.22)	BCL2 (51.09)
5	LMO2 (9.44)	TRDC (20.22)	CISH (11.03)	LMO2 (10.75)	GPC1 (166.19)
Nutlin-3					
1	RP11-148O21.4 (1.41)	MET (2.53)	RP11-148O21.4 (1.59)	RP11-148O21.4 (1.72)	RAPGEF5 (37)
2	MET (2.53)	RP11-148O21.4 (4.75)	MET (4.06)	LRRC16A (9.56)	G6PD (63.53)
3	BLK (5.12)	LAYN (6.41)	BLK (5.25)	BLK (10.97)	MET (147.78)
4	LRRC16A (5.16)	RPS27L (12.94)	LRRC16A (5.97)	MET (26.09)	BLK (160.06)
5	LAT2 (7.34)	ADD3 (21.03)	LAT2 (7.78)	LAYN (138.84)	RP11-148O21.4 (164.94)
PD-0325901					
1	SPRY2 (1.16)	SPRY2 (1.53)	SPRY2 (1.75)	SPRY2 (1.19)	SPRY2 (1.75)
2	LYZ (2.72)	ETV4 (2.88)	LYZ (2.09)	LYZ (2.88)	LYZ (2.62)
3	ETV4 (2.72)	LYZ (3.59)	ETV4 (3.66)	ETV4 (3.38)	ETV4 (3.47)
4	RP11-1143G9.4 (4.34)	TOR4A (4.56)	RP11-1143G9.4 (4.53)	TOR4A (4.72)	TOR4A (4.88)
5	PLEKHG4B (5.62)	PLEKHG4B (4.66)	PLEKHG4B (5.31)	RP11-1143G9.4 (5.38)	PLEKHG4B (5.5)
PD-0332991					
1	SH2D3C (4.31)	SH2D3C (6.81)	SH2D3C (4.56)	SH2D3C (8)	KRT15 (10.56)
2	FMNL1 (6.56)	FMNL1 (8.38)	HSD3B7 (6.75)	AL162151.3 (9.62)	HSD3B7 (14.5)
3	HSD3B7 (6.59)	AL162151.3 (11.59)	FMNL1 (7.09)	HSD3B7 (11.03)	SEPT6 (16.44)
4	KRT15 (7.19)	TWF1 (12.03)	KRT15 (7.78)	KRT15 (11.97)	PPIC (17.03)
5	AL162151.3 (8.84)	KRT15 (12.56)	TWF1 (8.97)	FMNL1 (16.34)	AL162151.3 (18.34)
PF2341066					
1	ENAH (1.03)	ENAH (1)	ENAH (1.31)	ENAH (1.12)	ENAH (1.06)

Table 1: *(continued)*

	MDI+	MDI	TreeSHAP	MDI-oob	MDA
2	SELPLG (2.09)	SELPLG (2.81)	SELPLG (2.06)	SELPLG (2.28)	SELPLG (2.47)
3	HGF (3.62)	HGF (3.94)	MET (5.44)	HGF (7.03)	CTD-2020K17.3 (10.31)
4	CTD-2020K17.3 (9.72)	CTD-2020K17.3 (9.41)	HGF (6)	MET (10.69)	HGF (14.06)
5	MET (10.53)	MET (11.5)	MLKL (12)	CTD-2020K17.3 (11.88)	DOK2 (14.41)
PHA-665752					
1	ARHGAP4 (1)	ARHGAP4 (1.06)	ARHGAP4 (1.06)	ARHGAP4 (1.09)	ARHGAP4 (1)
2	CTD-2020K17.3 (2.88)	CTD-2020K17.3 (2.66)	CTD-2020K17.3 (4.25)	CTD-2020K17.3 (2.56)	FMNL1 (4.22)
3	FMNL1 (4.88)	FMNL1 (7.44)	PFN2 (10.84)	PFN2 (24.31)	CTD-2020K17.3 (5.44)
4	PFN2 (8.06)	PGPEP1 (13.28)	FMNL1 (11.78)	FMNL1 (33.12)	INHBB (216.5)
5	PGPEP1 (10.12)	INHBB (18.88)	FDFT1 (18.66)	MICB (59.69)	PGPEP1 (335.97)
PLX4720					
1	RXRG (1)	RXRG (1.16)	RXRG (1)	RXRG (1)	RXRG (1.03)
2	MMP8 (5.19)	MMP8 (3.97)	MMP8 (5.16)	MMP8 (5.56)	MMP8 (4.88)
3	RP11-164J13.1 (6.28)	RP11-599J14.2 (7.22)	MYO5A (7.09)	MYO5A (6.81)	MYO5A (8.75)
4	RP11-599J14.2 (7)	AP1S2 (8.47)	LYST (7.97)	LYST (11.31)	AP1S2 (10.53)
5	RP4-718J7.4 (7.84)	LYST (9.34)	RP11-599J14.2 (8.41)	RP4-718J7.4 (13.22)	RP11-599J14.2 (12.69)
Paclitaxel					
1	MMP24 (1.09)	MMP24 (1.28)	MMP24 (1.22)	MMP24 (2.38)	SH3BP1 (10.38)
2	AGAP2 (3.16)	SH3BP1 (2.75)	AGAP2 (3.44)	SH3BP1 (2.88)	PRODH (60.41)
3	SH3BP1 (3.5)	AGAP2 (4.06)	SH3BP1 (3.78)	SLC38A5 (3.84)	AGAP2 (157.41)
4	SLC38A5 (4.34)	SLC38A5 (4.22)	PTTG1IP (3.91)	AGAP2 (5.88)	SLC38A5 (179.34)
5	PTTG1IP (4.72)	PTTG1IP (4.34)	SLC38A5 (4.31)	PTTG1IP (6.97)	MMP24 (308.66)
Panobinostat					
1	AGAP2 (1.12)	AGAP2 (1.56)	AGAP2 (1.81)	AGAP2 (2.16)	CYR61 (2.56)
2	CYR61 (2.44)	CYR61 (2.09)	CYR61 (2.16)	CYR61 (2.78)	AGAP2 (3.25)
3	RPL39P5 (4.19)	RPL39P5 (4.41)	RPL39P5 (3.75)	RPL39P5 (3.88)	RPL39P5 (152.44)
4	WWTR1 (5.16)	WWTR1 (5.78)	WWTR1 (5.94)	WWTR1 (6.53)	S100A2 (316.66)
5	MYOF (6.56)	MYOF (6.16)	IKZF1 (12.41)	IKZF1 (9.72)	MYOF (366.28)
RAF265					
1	CMTM3 (1.34)	CMTM3 (1.5)	CMTM3 (1.69)	SH2B3 (34)	CMTM3 (7.06)
2	SYT17 (5.69)	SYT17 (5.66)	SYT17 (7.69)	CMTM3 (155.25)	SH2B3 (470.66)
3	SH2B3 (6.03)	SH2B3 (8.91)	SH2B3 (17.5)	SLC29A3 (159)	SYT17 (652.84)
4	EMILIN2 (11.94)	SLC29A3 (11.84)	STAT5A (19.66)	PRKCQ (235)	RGS16 (713.47)
5	STAT5A (12.47)	NA (19.91)	SLC29A3 (22.22)	LCP2 (259.7)	AC007620.3 (1087.11)
Sorafenib					
1	PXK (4.47)	TP63 (6.72)	PXK (4.12)	FAM212A (6.41)	PXK (9.34)
2	P2RX1 (4.62)	P2RX1 (7.78)	FAM212A (5.75)	P2RX1 (8)	ARHGAP9 (34.25)
3	FAM212A (4.69)	PXK (8.88)	STAC3 (7.16)	SEC31B (41.69)	P2RX1 (156.91)
4	STAC3 (5.16)	FAM212A (16.97)	P2RX1 (7.25)	ARHGAP9 (43.19)	FAM212A (187.31)
5	ARHGAP9 (7.91)	STAC3 (20.16)	TP63 (40.97)	CXCL8 (57.72)	SEC31B (222.41)
TAE684					
1	SELPLG (1.09)	SELPLG (1.06)	SELPLG (1.12)	SELPLG (1.03)	SELPLG (1.34)
2	IL6R (3.19)	ARID3A (8.12)	IL6R (3.34)	IL6R (6.41)	ARID3A (18.31)

Table 1: *(continued)*

	MDI+	MDI	TreeSHAP	MDI-oob	MDA
3	NFIL3 (6.34)	GALNT18 (8.62)	NFIL3 (6.25)	NFIL3 (8.06)	FMNL1 (25.25)
4	ARID3A (7)	IL6R (10.16)	RRAS2 (10.19)	ARID3A (16.38)	RP11-334A14.2 (144.34)
5	RRAS2 (8.66)	PPP2R3A (15.69)	FMNL1 (10.19)	RRAS2 (17.97)	PPP2R3A (165.84)
TKI258					
1	TWF1 (2.44)	TWF1 (3.31)	TWF1 (2.41)	TWF1 (2.28)	LAPTM5 (20.84)
2	SLC43A1 (2.88)	GPR162 (3.75)	PRTN3 (3.31)	LAPTM5 (5.34)	SLC43A1 (156.12)
3	PRTN3 (5.25)	SLC43A1 (6.84)	SLC43A1 (6.06)	PRTN3 (6.12)	TWF1 (163.78)
4	LAPTM5 (5.78)	TTC28 (8.94)	LAT2 (6.62)	SLC43A1 (14.91)	GPR162 (169.66)
5	LAT2 (5.94)	LAPTM5 (11.53)	LAPTM5 (8.19)	LAT2 (17.41)	LYL1 (387.97)
Topotecan					
1	SLFN11 (1)	SLFN11 (1)	SLFN11 (1)	SLFN11 (1)	SLFN11 (1)
2	HSPB8 (2.16)	HSPB8 (2.28)	HSPB8 (2.84)	HSPB8 (2.06)	HSPB8 (2.62)
3	PPIC (5.69)	OSGIN1 (5.28)	OSGIN1 (5.31)	PPIC (7.81)	OSGIN1 (7.19)
4	OSGIN1 (5.88)	AGAP2 (8.81)	PPIC (6.81)	RP11-359P5.1 (10.75)	AGAP2 (15.25)
5	AGAP2 (6.53)	PPIC (8.91)	RP11-359P5.1 (8.25)	CORO1A (12.06)	HMGB2 (21.53)
ZD-6474					
1	MAP3K12 (1)	MAP3K12 (1)	MAP3K12 (1)	MAP3K12 (1.09)	MAP3K12 (1)
2	PIM1 (5.47)	CTSH (21.88)	PIM1 (10.28)	PIM1 (31.44)	SCD5 (339.58)
3	PRKCQ (9.03)	TIMP1 (26.31)	PRKCQ (15.28)	DYNLT3 (192.12)	ITGA10 (527.41)
4	CTSH (12.91)	PRKCQ (26.47)	CTSH (18.16)	TIMP1 (211.25)	TIMP1 (569.81)
5	ITGA10 (20.81)	PIM1 (33.81)	ANXA5 (78.78)	EPHA1 (320.22)	CTSH (631.5)

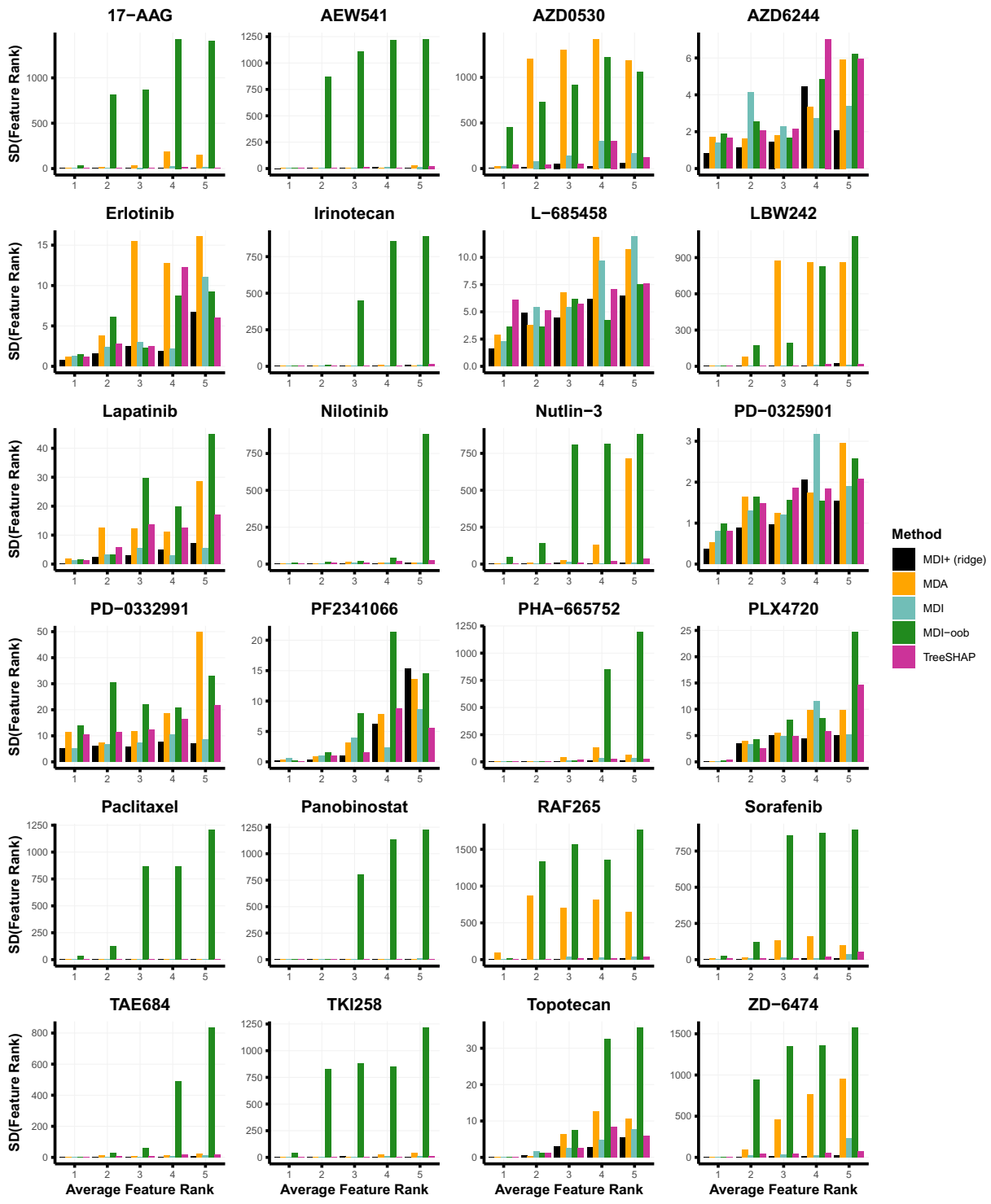


Figure 35: Stability of top 5 genes for each drug response prediction model across 32 train-test splits. The x-axis corresponds to the top 5 features for each method, ranked according to their average feature ranking across 32 train-test splits. On the y-axis, we provide one measure of stability – namely, the standard deviation of the feature rankings across the 32 train-test splits. MDI+ generally provides the most stable feature importance rankings for these top 5 genes. Results from all feature importance methods under study are shown here.

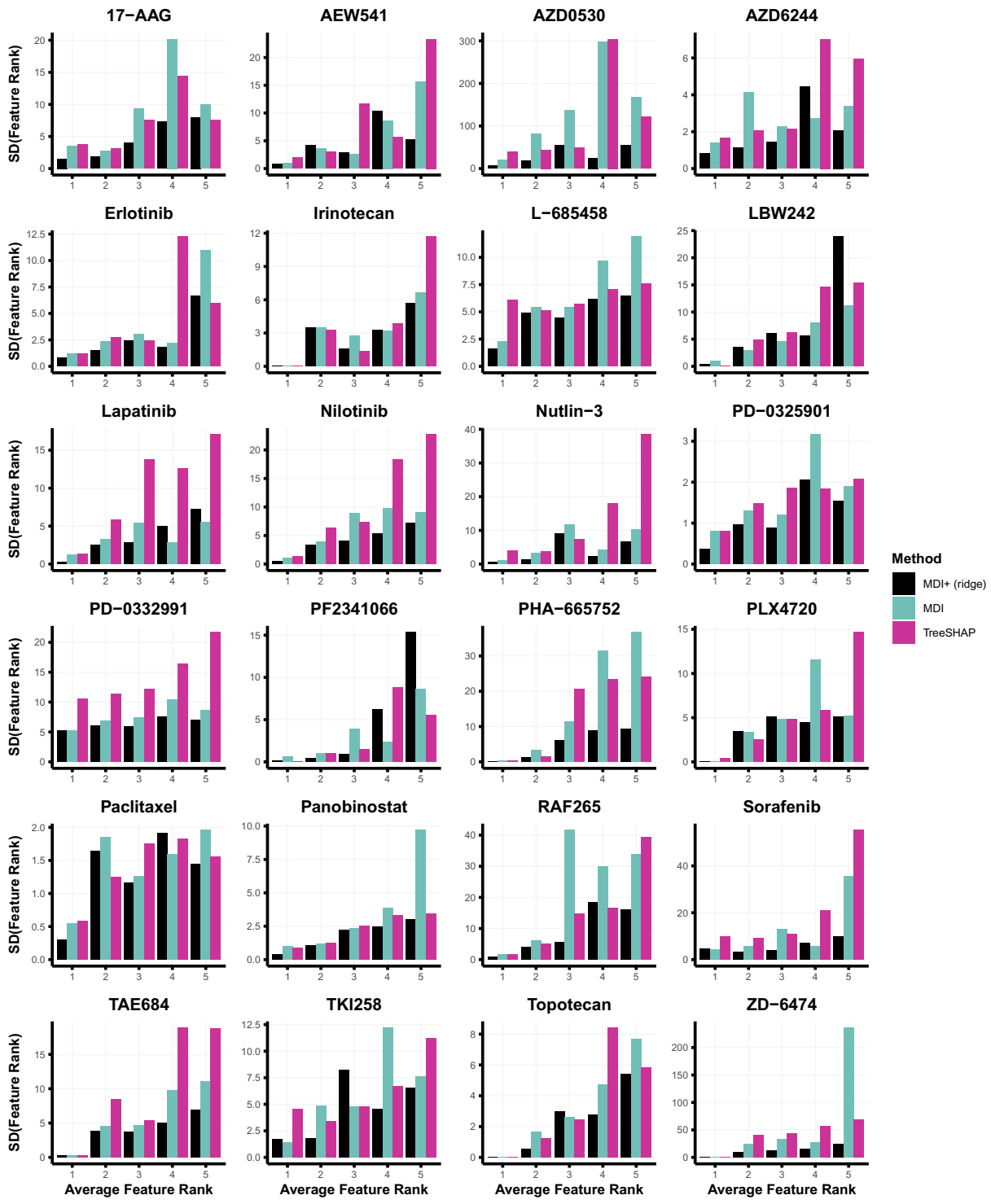


Figure 36: Stability of top 5 genes for each drug response prediction model across 32 train-test splits. The x-axis corresponds to the top 5 features for each method, ranked according to their average feature ranking across 32 train-test splits. On the y-axis, we provide one measure of stability – namely, the standard deviation of the feature rankings across the 32 train-test splits. MDI+ generally provides the most stable feature importance rankings for these top 5 genes. Results from the feature importance methods under study, excluding MDI-oob and MDA, are shown here.

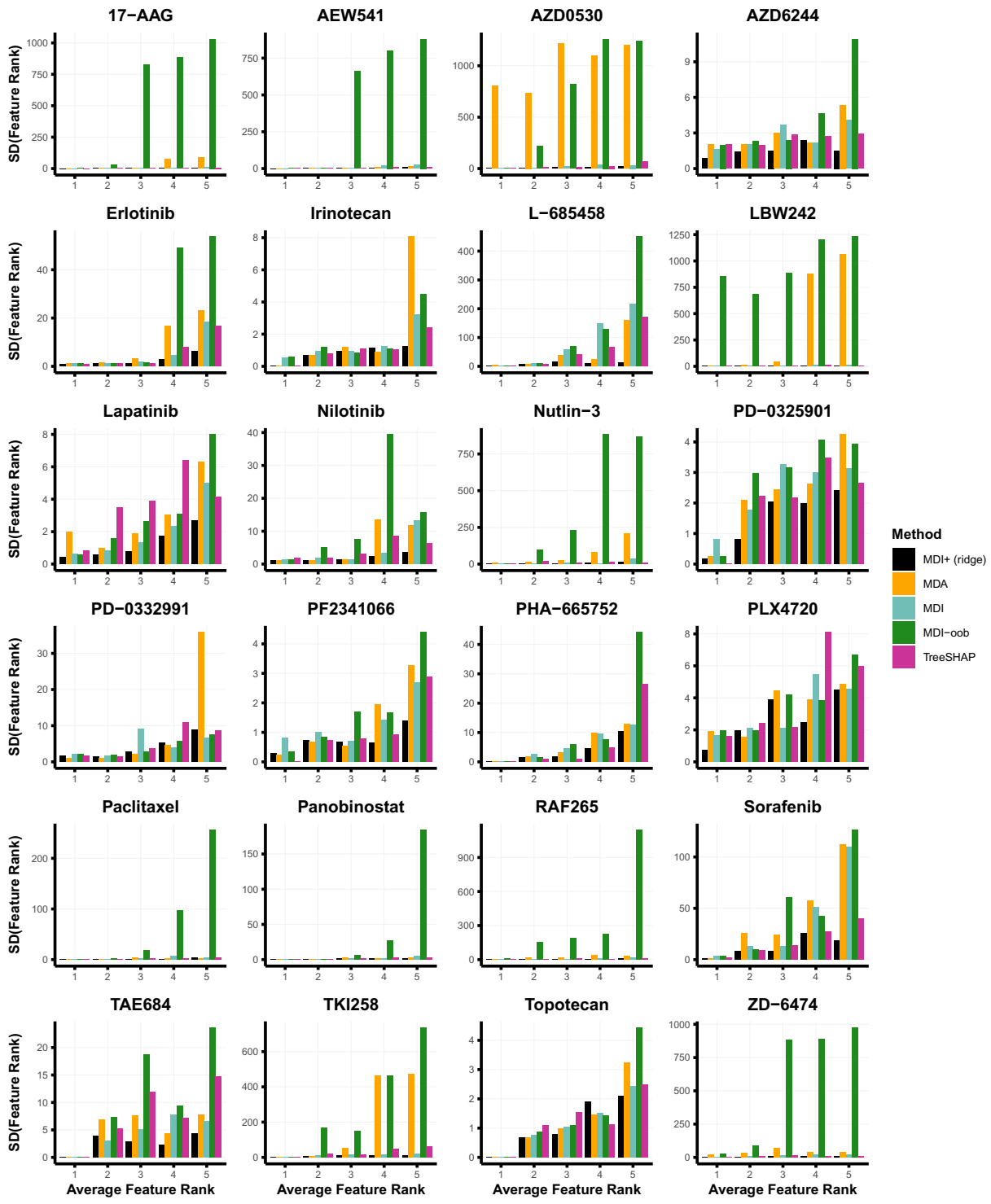


Figure 37: Stability of top 5 genes for each drug response prediction model across 32 RF fits, trained using different random seeds. The x-axis corresponds to the top 5 features for each method, ranked according to their average feature ranking across the 32 RF fits. On the y-axis, we provide one measure of stability – namely, the standard deviation of the feature rankings across the 32 RF fits. MDI+ generally provides the most stable feature importance rankings for these top 5 genes. Results from all feature importance methods under study are shown here.

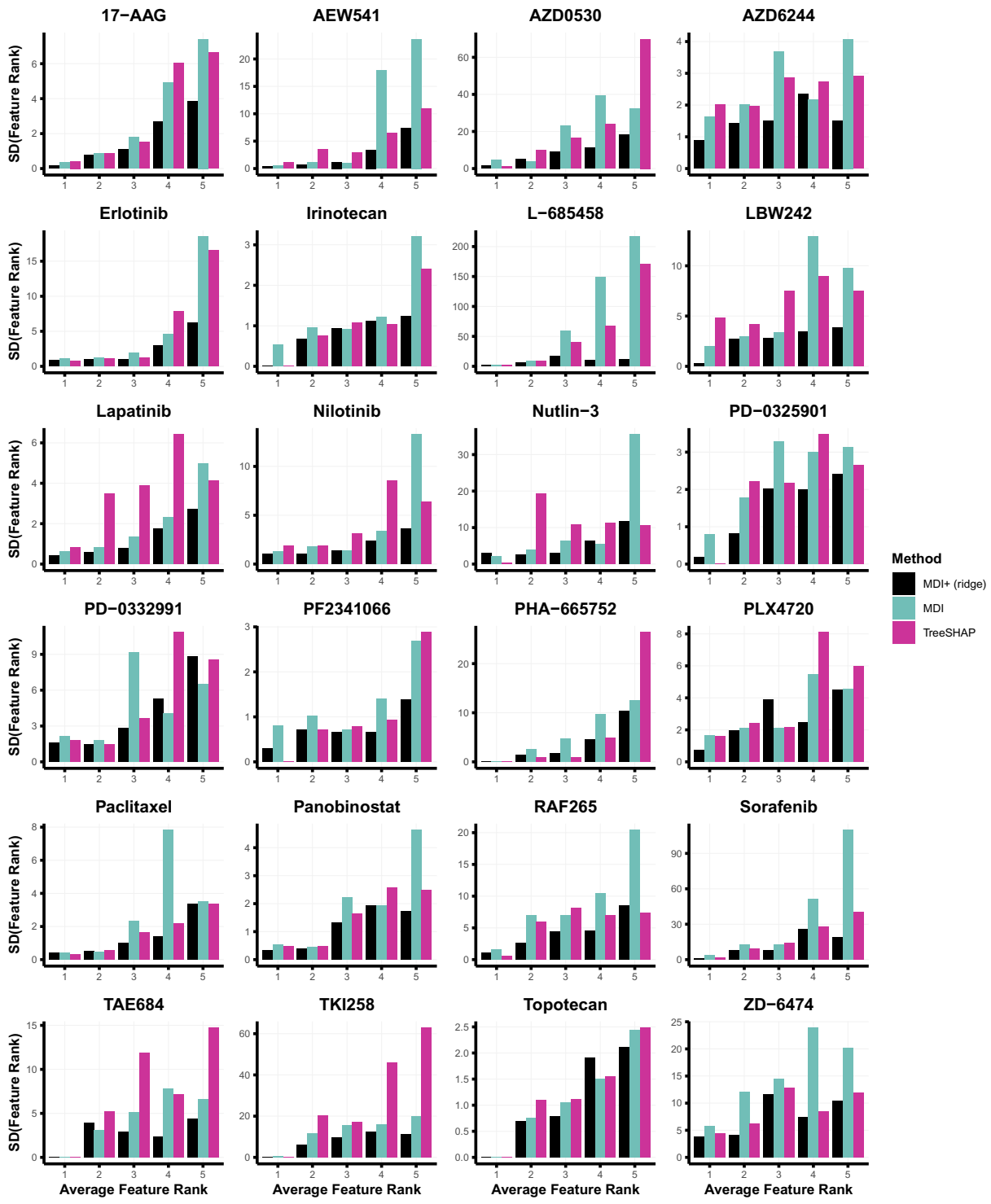


Figure 38: Stability of top 5 genes for each drug response prediction model across 32 RF fits, trained using different random seeds. The x-axis corresponds to the top 5 features for each method, ranked according to their average feature ranking across the 32 RF fits. On the y-axis, we provide one measure of stability – namely, the standard deviation of the feature rankings across the 32 RF fits. MDI+ generally provides the most stable feature importance rankings for these top 5 genes. Results from the feature importance methods under study, excluding MDI-oob and MDA, are shown here.

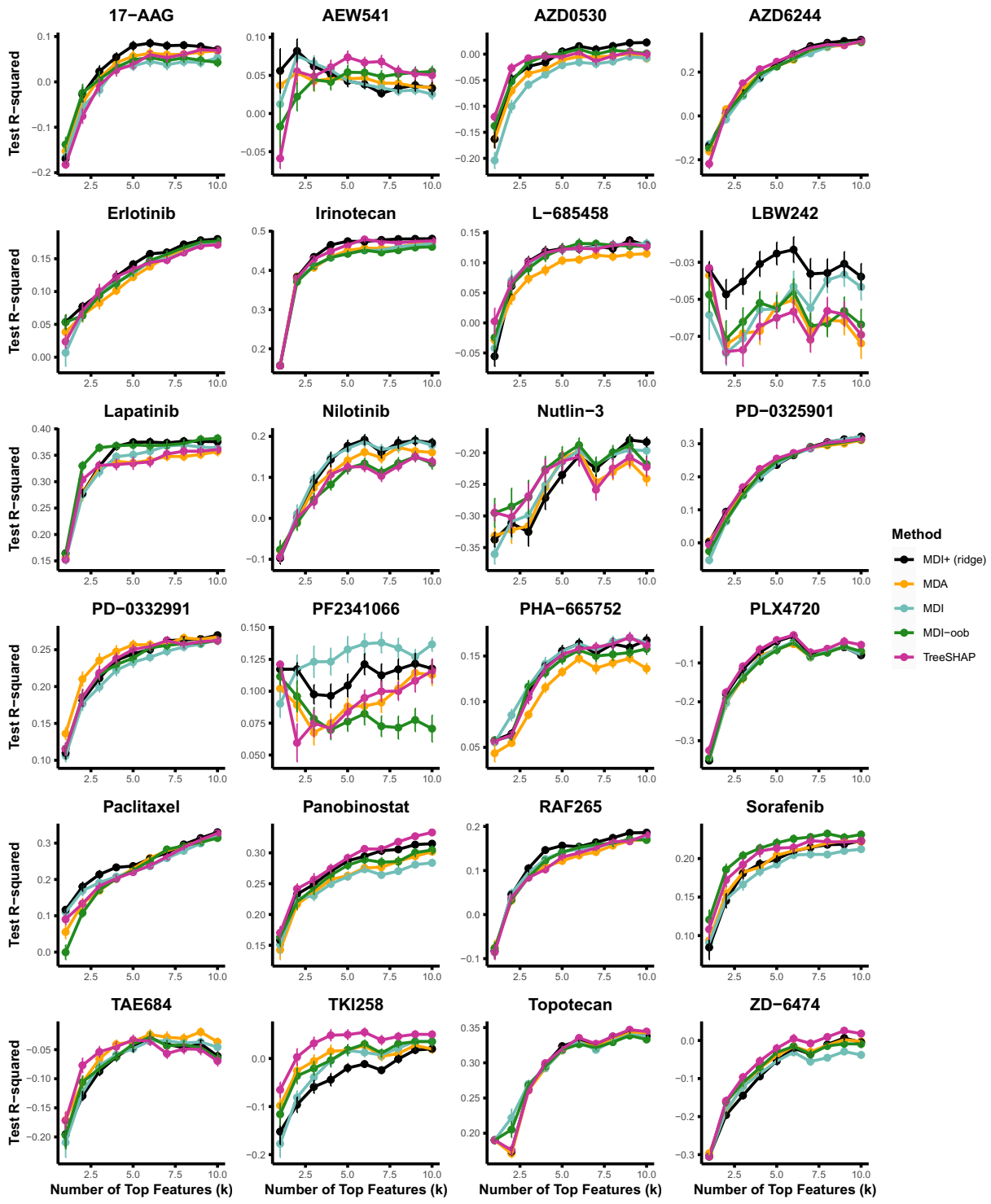


Figure 39: RF prediction performance, measured via test R^2, using the top k features from each feature importance methods across the 24 drugs in the CCLE case study. Results are averaged across 32 train-test data splits.

Table 2: Summary of prediction power using the top 10 features from each feature importance method in the CCLE drug response case study. For each drug, we evaluated the average test R^2 (averaged across 32 train-test splits) from an RF trained using only the top 10 features from each feature importance method. We then rank the feature importance methods by this average test R^2 (1 = best test R^2, 5 = worst test R^2) and display the number of drugs, for which that rank was achieved. In particular, taking the top 10 genes from MDI+ (ridge) gave the best prediction performance for 12 out of the 24 drugs.

Rank	MDI+ (ridge)	TreeSHAP	MDI	MDI-oob	MDA
1	12	5	3	3	1
2	6	6	6	3	3
3	2	6	6	4	6
4	3	5	5	5	6
5	1	2	4	9	8

H.2 Breast Cancer Subtype Prediction

Data preprocessing. We pulled data from the TCGA breast cancer project using the `TCGAbiolinks` R package. This gene expression dataset originally consisted of 19,947 genes. As before, we reduced the number of features under consideration by taking only the top 5000 features with the highest variance. The processed TCGA data used in our case study can be found on Zenodo at https://zenodo.org/record/8111870.

Results. In Table 3, we summarize the test prediction performance for RF, RF+ (ridge), and RF+ (logistic), applied to the TCGA case study. We show the average test classification accuracy, AUROC, and area under the precision recall curve (AUPRC), averaged across 32 train-test splits. In Table 4, we list the top 25 gene expression predictors according to each feature importance method. These genes are ranked according to their average feature importance ranking across the 32 train-test splits. We also show the RF predictive power of the top k genes from each feature importance method in Figure 39. Details and discussion on this procedure were provided in the previous section. As discussed, correlated features can make it challenging to directly compare the prediction accuracy using the top k genes from various feature importance methods. Still, it is worth noting that the top 10 features from MDI+ (ridge) and MDI+ (logistic) yield a higher test prediction performance (in terms of both AUROC and classification accuracy) than other competing methods. Moreover, the top 15 features from MDI+ (ridge) and MDI+ (logistic) yield

a higher test prediction performance (in terms of both AUROC and classification accuracy) as the top 25 features from the other competing feature importance methods. This improvement in predictive power further supports the practical utility of MDI+ for feature ranking [80].

Table 3: Test prediction performance for various methods, averaged across 32 train-test splits, on the TCGA case study. Standard errors are shown in parentheses.

Model	Classification Accuracy	AUROC	AUPRC
RF+ (logistic)	**0.884 (0.003)**	**0.981 (0.001)**	**0.895 (0.004)**
RF+ (ridge)	0.873 (0.003)	0.981 (0.001)	0.892 (0.004)
RF	0.861 (0.003)	0.978 (0.001)	0.878 (0.005)

Table 4: Top 25 most important genes for predicting breast cancer subtype according to various feature importance methods. Genes are ranked by their average feature importance ranking across 32 train-test splits (shown in parentheses).

Rank	MDI+ (ridge)	MDI+ (logistic)	MDA	TreeSHAP	MDI
1	ESR1 (1.91)	ESR1 (1.91)	ESR1 (4.5)	ESR1 (7.62)	ESR1 (13.91)
2	FOXA1 (4.25)	GATA3 (4.5)	GATA3 (6.38)	TPX2 (10.41)	TPX2 (15.34)
3	FOXC1 (6.12)	FOXA1 (5.09)	FOXA1 (8.11)	GATA3 (19.62)	FOXM1 (22.84)
4	GATA3 (6.97)	TPX2 (6.81)	TPX2 (10.12)	FOXM1 (20.06)	MLPH (24.97)
5	AGR3 (7.94)	AGR3 (10.22)	MLPH (10.16)	FOXA1 (20.72)	FOXA1 (25.66)
6	MLPH (8.16)	FOXC1 (12.94)	AGR3 (12.94)	CDK1 (22.38)	GATA3 (30.44)
7	TPX2 (11.03)	MLPH (15.69)	TBC1D9 (14.22)	MLPH (22.53)	CDK1 (31.41)
8	TBC1D9 (14.44)	FOXM1 (18.12)	FOXC1 (15.09)	AGR3 (25.88)	THSD4 (34.69)
9	FOXM1 (18.66)	TBC1D9 (21.03)	FOXM1 (19.88)	PLK1 (28.47)	FOXC1 (35)
10	THSD4 (21.78)	THSD4 (23.66)	THSD4 (21.28)	TBC1D9 (29.84)	TBC1D9 (35.44)
11	SPDEF (25.81)	CDK1 (24.44)	CDK1 (21.77)	FOXC1 (30.44)	AGR3 (36.44)
12	CA12 (29.44)	MYBL2 (25.34)	XBP1 (24.88)	MYBL2 (33.06)	PLK1 (36.81)
13	CDK1 (36.09)	RACGAP1 (26.81)	KIF2C (27.95)	THSD4 (33.53)	MYBL2 (45.22)
14	GABRP (36.72)	ASPM (27.56)	PLK1 (28.58)	KIF2C (35.56)	KIF2C (46.22)
15	PLK1 (37.97)	PLK1 (28.84)	MYBL2 (30.97)	ASPM (40.59)	ASPM (49.72)
16	FAM171A1 (38.59)	UBE2C (30.41)	GABRP (31.94)	GMPS (41.41)	GMPS (52.03)
17	ASPM (39.5)	SPAG5 (31.81)	ASPM (35.97)	XBP1 (50.44)	SPDEF (58.88)
18	SFRP1 (40.25)	GMPS (35.34)	FAM171A1 (38.55)	CENPF (56.69)	FAM171A1 (68.56)
19	XBP1 (40.44)	KIF2C (36.03)	CA12 (40.16)	MKI67 (59.69)	CA12 (69.59)
20	MYBL2 (43.12)	CA12 (37.31)	CDC20 (48.45)	CA12 (59.91)	UBE2C (69.78)
21	TFF3 (43.12)	RRM2 (46.09)	SPDEF (53.05)	RACGAP1 (60.53)	TFF3 (70)
22	KIF2C (44.62)	CENPF (46.31)	UBE2C (54.27)	SPAG5 (61.94)	CENPF (70.31)
23	PRR15 (44.88)	GABRP (47.59)	ANXA9 (55.41)	FAM171A1 (63)	RACGAP1 (70.94)
24	AGR2 (45)	SFRP1 (49.34)	C1orf64 (57.22)	KIF11 (63.56)	SPAG5 (71.81)
25	MIA (45.31)	XBP1 (49.78)	RACGAP1 (57.97)	ANLN (64.28)	KIF11 (73.12)

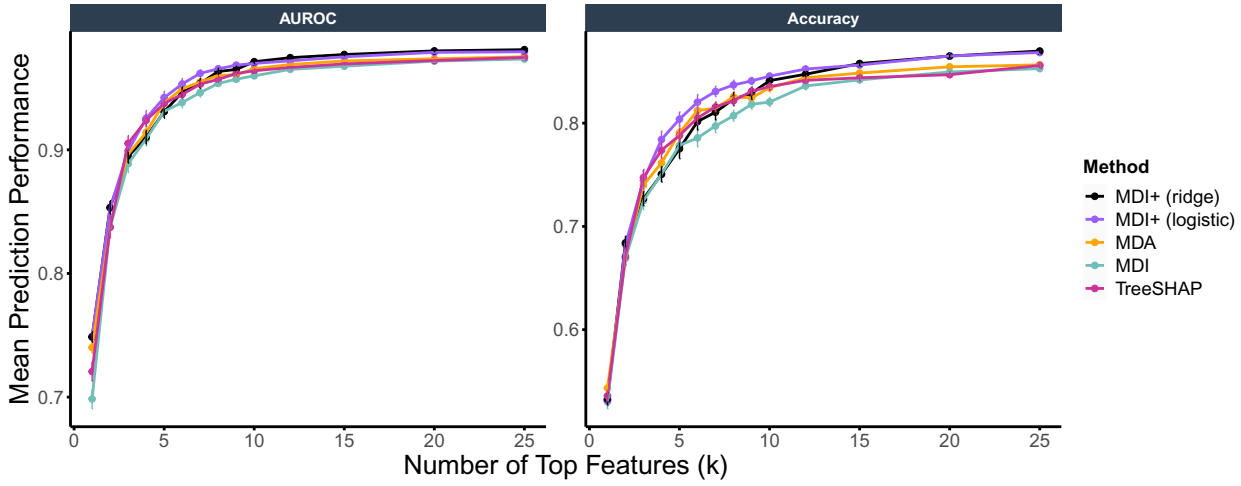


Figure 40: RF prediction performance, measured via test AUROC and classification accuracy, using the top k features from each feature importance methods in the TCGA case study. Results are averaged across 32 train-test data splits.