A generalized random forest framework

for improved prediction and interpretations

Tiffany Tang
Department of Applied and Computational Mathematics and Statistics
University of Notre Dame

Joint with:

Abhineet Agarwal
UC Berkeley

Ana Kenney
UC Irvine

Yan Shuo Tan
National Univ. of Singapore

Bin Yu
UC Berkeley

Ji Zhu
Univ. of Michigan

Elizaveta (Liza) Levina
Univ. of Michigan

Why Random Forests?

+ A **powerful, nonparametric prediction algorithm**, which often outperforms deep learning on moderate-sized tabular datasets

> “ ... the method that performs consistently well across all dimensions is **random forests**, followed by neural nets, boosted trees, and SVMs. [11 datasets] ”
>
> - Caruana, Karampatziakis, Yessenalina (2008)

> “ The classifiers most likely to be the bests are the **random forest** versions. [121 data sets, 179 models] ”
>
> - Fernandez-Delgado, Cernadas, Barro, Amorim (2014)

> “ *Why do tree-based models still outperform deep learning on tabular data?*
> ... tree-based models [i.e., **random forests**, XGBoost] remain state-of-the-art on medium-sized data (~10K samples) even without accounting for their superior speed. [45 data sets] ”
>
> - Grinsztajn, Oyallon, Varoquaux (2022)

Why Random Forests?

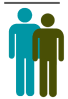
- A **powerful, nonparametric prediction algorithm**, which often outperforms deep learning on moderate-sized tabular datasets
- Numerous feature importance measures exist to enable **interpretability** [Breiman 2001, Ishwaran 2007, Epifanio 2017, Kazemitabar et al. 2017, Li et al. 2019, Lundberg et al. 2020, Klusowski and Tian 2021, Saabas 2022, and more...]
  - **Mean Decrease in Impurity (MDI):** most popular in practice (and default feature importance in sklearn) [Breiman et al. 1984]

Random Forest (RF) [Breiman 2001]

A **collection of decision trees**, where

[redacted]

training data

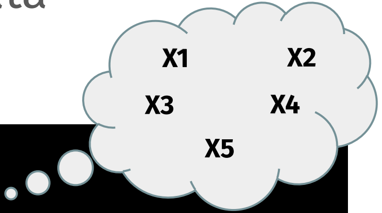
$\hat{y}_i = ($ + … +  $)/$ # trees

Leo Breiman. "Random forests." *Machine learning* (2001)

Random Forest (RF) [Breiman 2001]

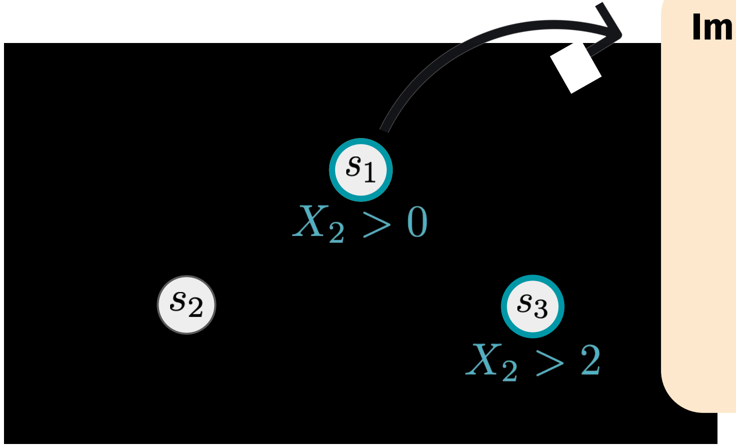
A **collection of decision trees**, where

- each tree is fitted on a different **bootstrap** version of the data
- **features are subsampled** at each node



Leo Breiman. "Random forests." *Machine learning* (2001)

Mean Decrease in Impurity (MDI)



Impurity decrease at s_1

"Measures decrease in variance from making the split"

$$\hat{\Delta}(s_1) = \underbrace{\sum_{\mathbf{x}\in s_1} (y_i - \bar{y}_{s_1})^2}_{\text{Var(node of interest)}} - \underbrace{\sum_{\mathbf{x}\in s_2} (y_i - \bar{y}_{s_2})^2}_{\text{Var(left child node)}} - \underbrace{\sum_{\mathbf{x}\in s_3} (y_i - \bar{y}_{s_3})^2}_{\text{Var(right child node)}}$$

For each feature k, MDI(k) is the weighted sum of impurity decreases across nodes that split on X_k, e.g.,

$$MDI(X_2) = \frac{n_1}{n}\hat{\Delta}(s_1) + \frac{n_3}{n}\hat{\Delta}(s_3)$$

Mean Decrease in Impurity (MDI)

Advantages of MDI:

Conceptually simple

Fast to compute

Well-known drawbacks of MDI:

Unstable in **low-signal** problems

Biased against features are highly **correlated** or have low **entropy**

Inefficient measure if **additive structure** is present **(Limitation of RF)**

Nicodemus, K. K. and Malley, J. D. “Predictor correlation impacts machine learning algorithms: implications for genomic studies.” *Bioinformatics* (2009)
Nicodemus, K. K. “On the stability and ranking of predictors from random forest variable importance measures.” *Briefings in Bioinformatics* (2011)
Tan, Y. S., Agarwal, A., and Yu, B. “A cautionary tale on fitting decision trees to data from additive models: generalization lower bounds.” AISTATS (2022)

Talk outline

0 We exploit a recent connection between **decision trees** and **linear regression**

1 We develop **RF+**, a generalization of RFs,
which improves upon the **prediction** accuracy of RFs,
especially when there is smooth additive structure

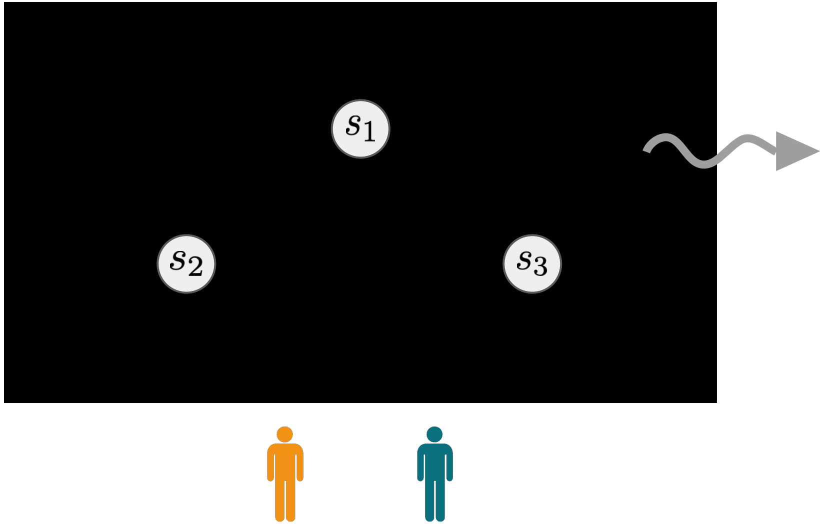
+ Extensions of RF+, including to the network (or spatial) data setting

2 We develop **MDI+**, a generalization of MDI,
which provides a general framework for improved **interpretations** using RF/RF+

Reinterpreting decision trees via linear regression

Connecting decision trees to linear regression

Step 1: Obtain engineered “stump” features $\psi(\cdot\,;s_k)$ from decision tree



$$\psi(\underset{\text{Input data } \mathbf{x} \in \mathbb{R}^p}{\mathbf{x}}; \overset{\text{node}}{s_k}) = \begin{cases} 0 & \text{if } \mathbf{x} \notin s_k \\ \frac{-N_R}{\sqrt{N_L N_R}} & \text{if } \mathbf{x} \in \text{left child of } s_k \\ \frac{N_L}{\sqrt{N_L N_R}} & \text{if } \mathbf{x} \in \text{right child of } s_k \end{cases}$$

$$\text{where } N_R = \text{number of samples in right child of } s_k$$
$$N_L = \text{number of samples in left child of } s_k$$

$$\Psi(\mathbf{X};\mathcal{S}) := \begin{array}{c|ccc} & s_1 & s_2 & s_3 \\ \hline & - & + & 0 \\ & + & 0 & - \\ & \vdots & \vdots & \vdots \end{array}$$

A new basis using supervised tree features

Connecting decision trees to linear regression

Step 2: Fit OLS on stump features

$$\mathbf{y} \sim \Psi(\mathbf{X}, \mathcal{S})$$

Key Connection: OLS predictions = original tree predictions [Klusowski 2021]

assuming tree prediction = mean response per leaf node (e.g., in CART)

Upshot #1: Provides a natural framework for developing a new class of prediction models → RF+

Upshot #2: Reinterpret MDI via linear regression → MDI+

Jason M. Klusowski. "Universal consistency of decision trees in high dimensions." (2021)

RF+:
A generalization of random forests

RF+: A generalization of random forests

A decision tree in **RF**:

$$\underset{\boldsymbol{\tau}\in\mathbb{R}^{\#\text{ stumps}}}{\arg\min} \|\mathbf{y} - \Psi(\mathbf{X})\boldsymbol{\tau}\|_2^2$$

A decision tree in **RF+**:

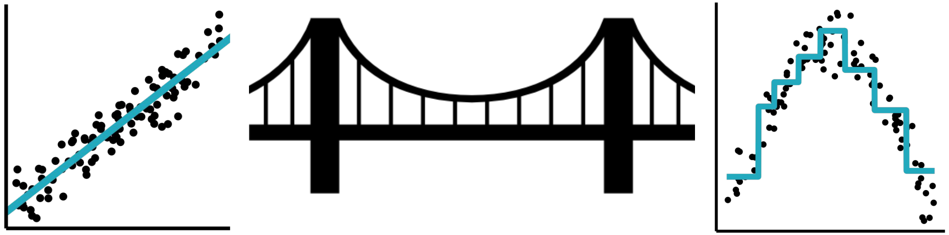
$$\underset{\substack{\boldsymbol{\beta}\in\mathbb{R}^p\\ \boldsymbol{\tau}\in\mathbb{R}^{\#\text{ stumps}}}}{\arg\min} \|\mathbf{y} - \underbrace{\mathbf{X}\beta}_{\textbf{Linear}} - \underbrace{\boldsymbol{\Psi}(\mathbf{X})\boldsymbol{\tau}}_{\textbf{Nonlinear}}\|_2^2 + P_{\boldsymbol{\beta}}(\beta) + P_{\boldsymbol{\tau}}(\boldsymbol{\tau})$$

- Why restrict ourselves to only stump features?
  This is the source of RF's implicit bias against smooth data structures
- Why not add regularization?
- Why restrict ourselves to L_2 loss?

RF+: A generalization of random forests

RF+: a new class of prediction algorithms, which generalizes RFs

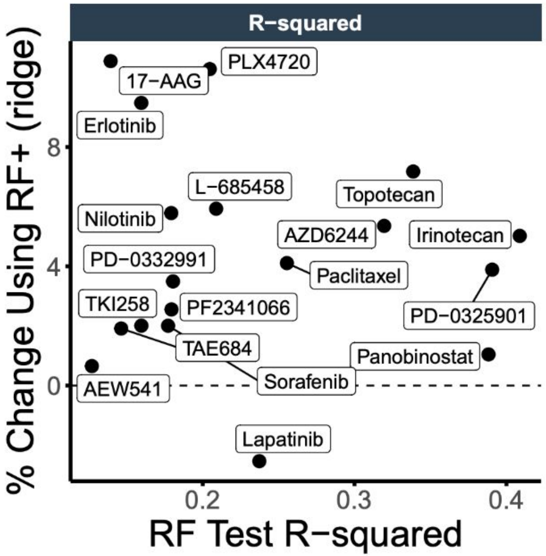
$$\underset{\substack{\boldsymbol{\beta}\in\mathbb{R}^p\\ \boldsymbol{\tau}\in\mathbb{R}^{\#\text{ stumps}}}}{\arg\min} \quad \|\mathbf{y} - \underbrace{\mathbf{X}\boldsymbol{\beta}}_{\text{Linear}} - \underbrace{\boldsymbol{\Psi}(\mathbf{X})\boldsymbol{\tau}}_{\text{Nonlinear}}\|_2^2 + P_{\boldsymbol{\beta}}(\boldsymbol{\beta}) + P_{\boldsymbol{\tau}}(\boldsymbol{\tau})$$



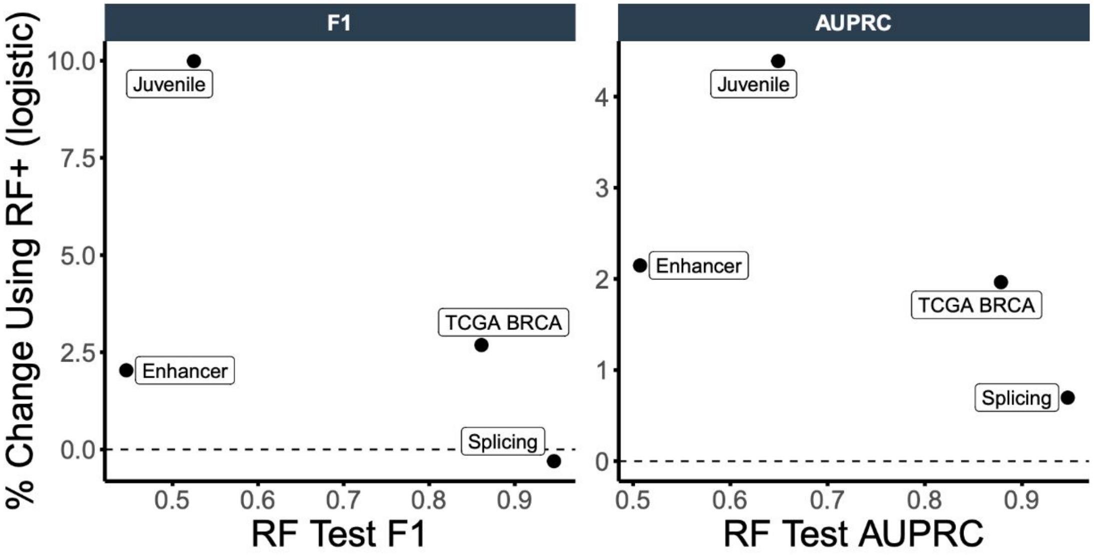
- Fitted per tree using bootstrappped samples and averaged across trees
- Ridge penalty generally works well
- Can apply general loss functions
  (e.g., logistic for classification, robust regression when outliers are present)

RF+ improves **prediction accuracy** over RF

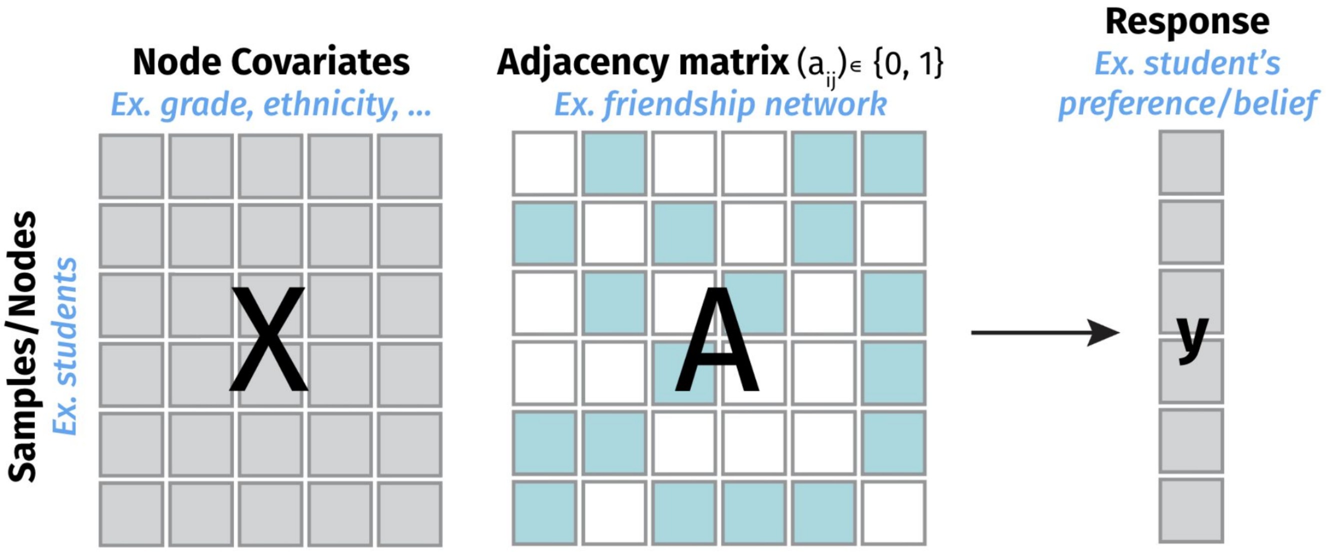
(A) Regression



(B) Classification



Extending RF+ to network-assisted regression setting



Node Covariates
Ex. grade, ethnicity, ...

Adjacency matrix $(a_{ij}) \in \{0, 1\}$
Ex. friendship network

Response
Ex. student's preference/belief

Samples/Nodes
Ex. students

Network cohesion assumption: neighboring nodes have similar responses to each other

(unnormalized) Laplacian
L = **D** - **A** where **D** = degree matrix

NeRF+: Network-assisted RF+

In the linear regression setting, network effects can be incorporated through a **network cohesion penalty** [Li et al. (2019)]:

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^n,\ \boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min} \ \|\mathbf{y} - \underbrace{\boldsymbol{\alpha}}_{\text{Network Effects}} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \underbrace{\lambda\boldsymbol{\alpha}^\top L\boldsymbol{\alpha}}_{\text{Network Cohesion Penalty}}$$

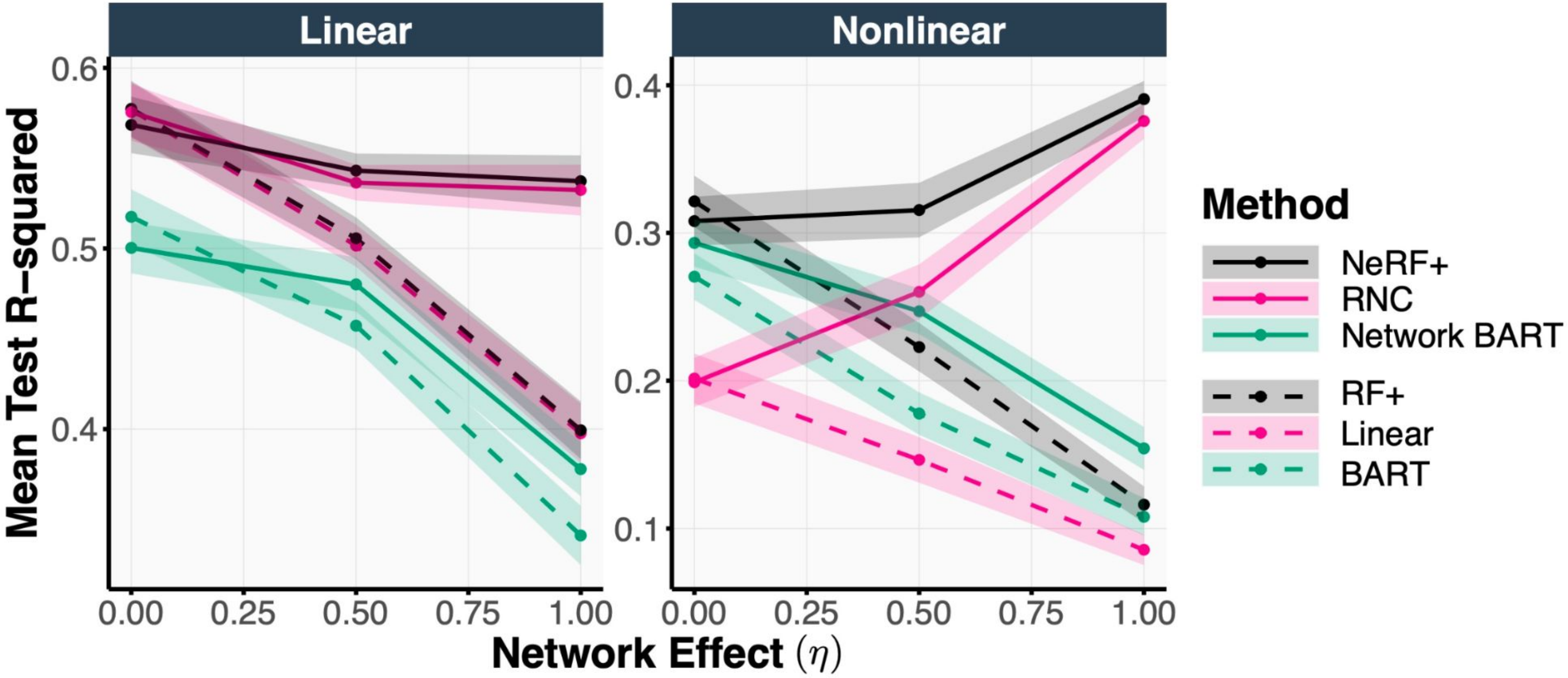
where $$\boldsymbol{\alpha}^T L\boldsymbol{\alpha} = \sum_{(i,j)\in E} (\alpha_i - \alpha_j)^2 = \sum_{i,j} A_{ij}(\alpha_i - \alpha_j)^2$$

NeRF+: an extension of RF+ to exploit cohesion between samples in a network

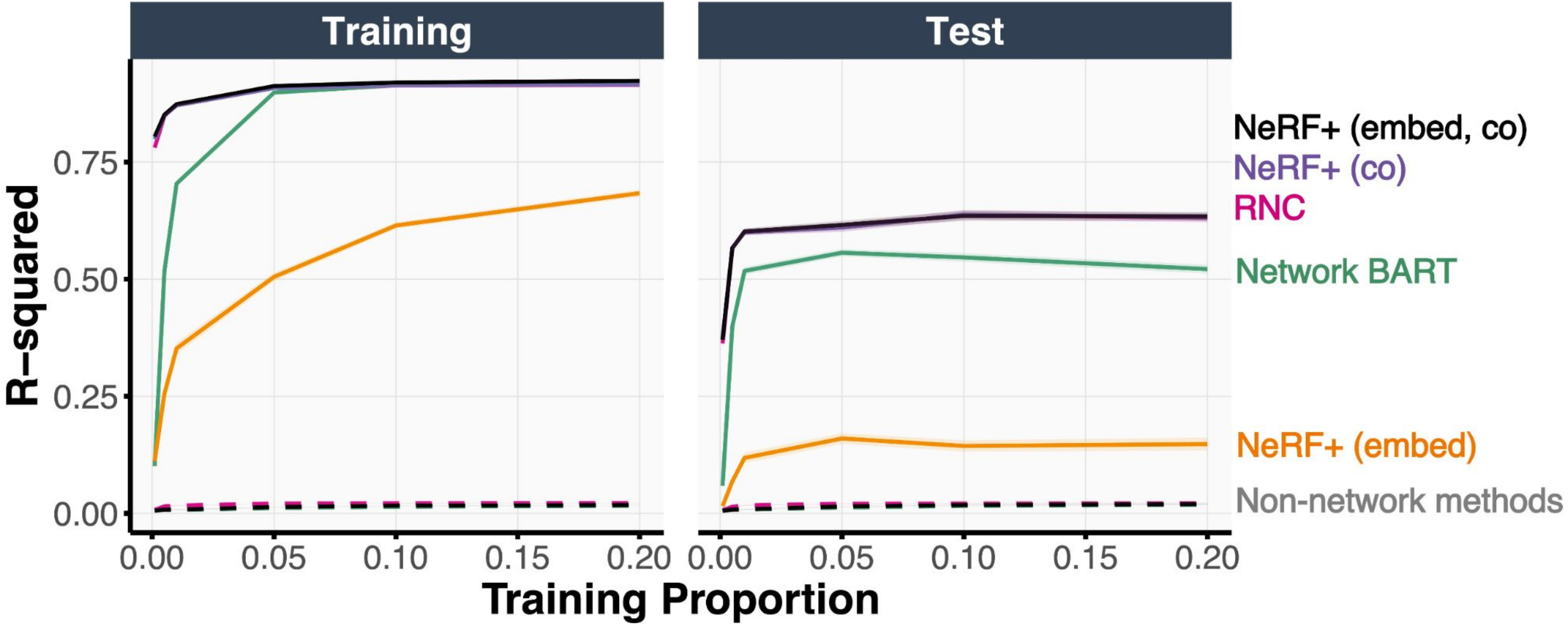
$$\underset{\substack{\boldsymbol{\alpha}\in\mathbb{R}^n\\ \boldsymbol{\beta}\in\mathbb{R}^p\\ \boldsymbol{\tau}\in\mathbb{R}^{\#\text{ stumps}}}}{\arg\min} \ \|\mathbf{y} - \underbrace{\boldsymbol{\alpha}}_{\text{Network Effects}} - \underbrace{\mathbf{X}\boldsymbol{\beta}}_{\text{Linear}} - \underbrace{\boldsymbol{\Psi}(\mathbf{X})\boldsymbol{\tau}}_{\text{Nonlinear}}\|_2^2 + P_{\boldsymbol{\beta}}(\boldsymbol{\beta}) + P_{\boldsymbol{\tau}}(\boldsymbol{\tau}) + \underbrace{\lambda\boldsymbol{\alpha}^\top L\boldsymbol{\alpha}}_{\text{Network Cohesion Penalty}}$$

Li et al. "Prediction models for network-linked data." *Annals of Applied Statistics* (2019)

NeRF+ improves prediction performance



NeRF+ improves prediction on Philadelphia crime dataset



MDI+:
A generalization of mean decrease in impurity

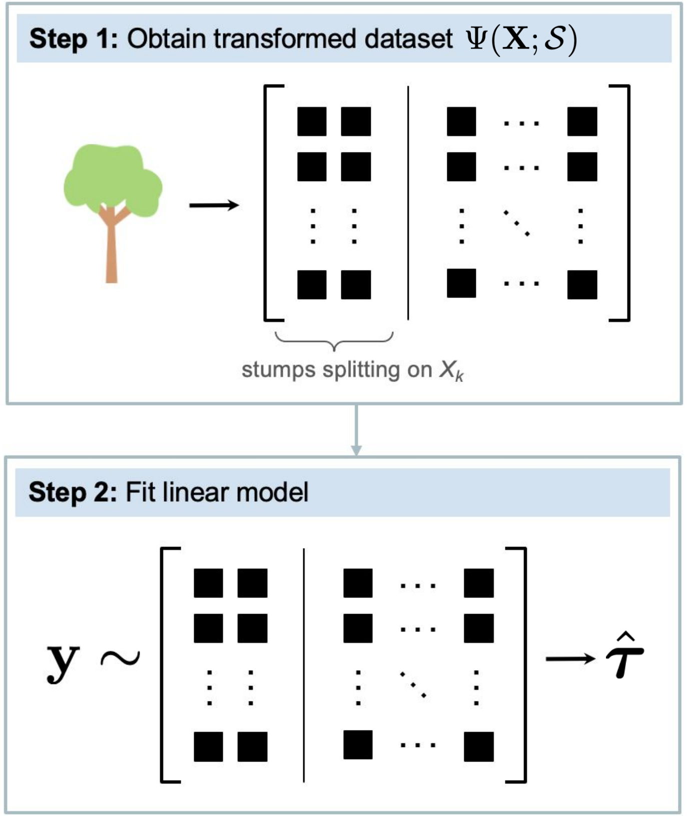
Overview of MDI+

MDI+: a flexible framework for computing feature importances using RF/RF+

- Avoids aforementioned drawbacks of MDI
- Allows the analyst to tailor the feature importance computation to the data/problem structure (e.g., handle outliers, classification vs. regression)

Key idea: MDI can be viewed as an R^2 value from a linear regression model

Reinterpreting MDI as an R^2

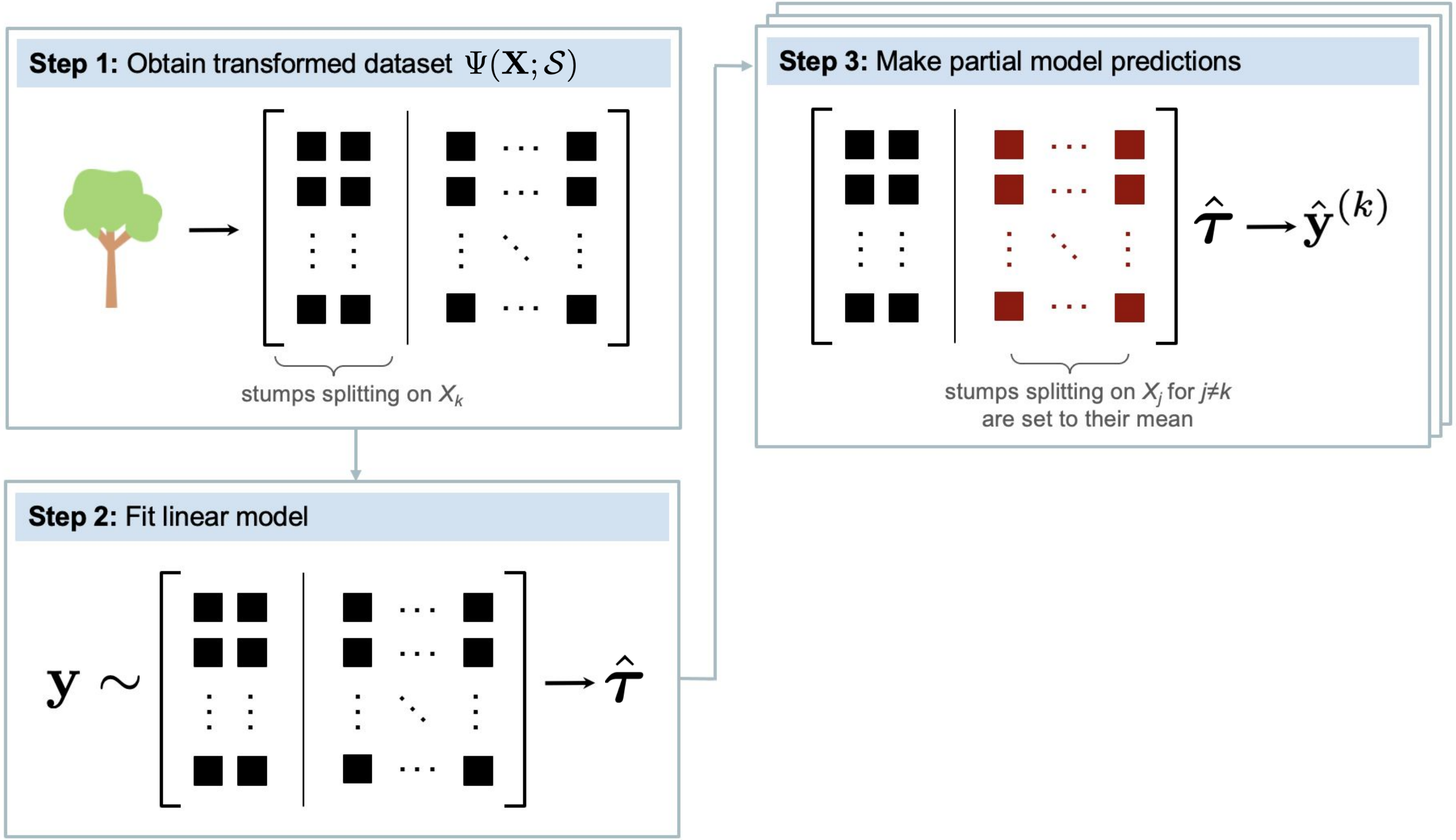


Step 1: Obtain transformed dataset $\Psi(\mathbf{X}; \mathcal{S})$

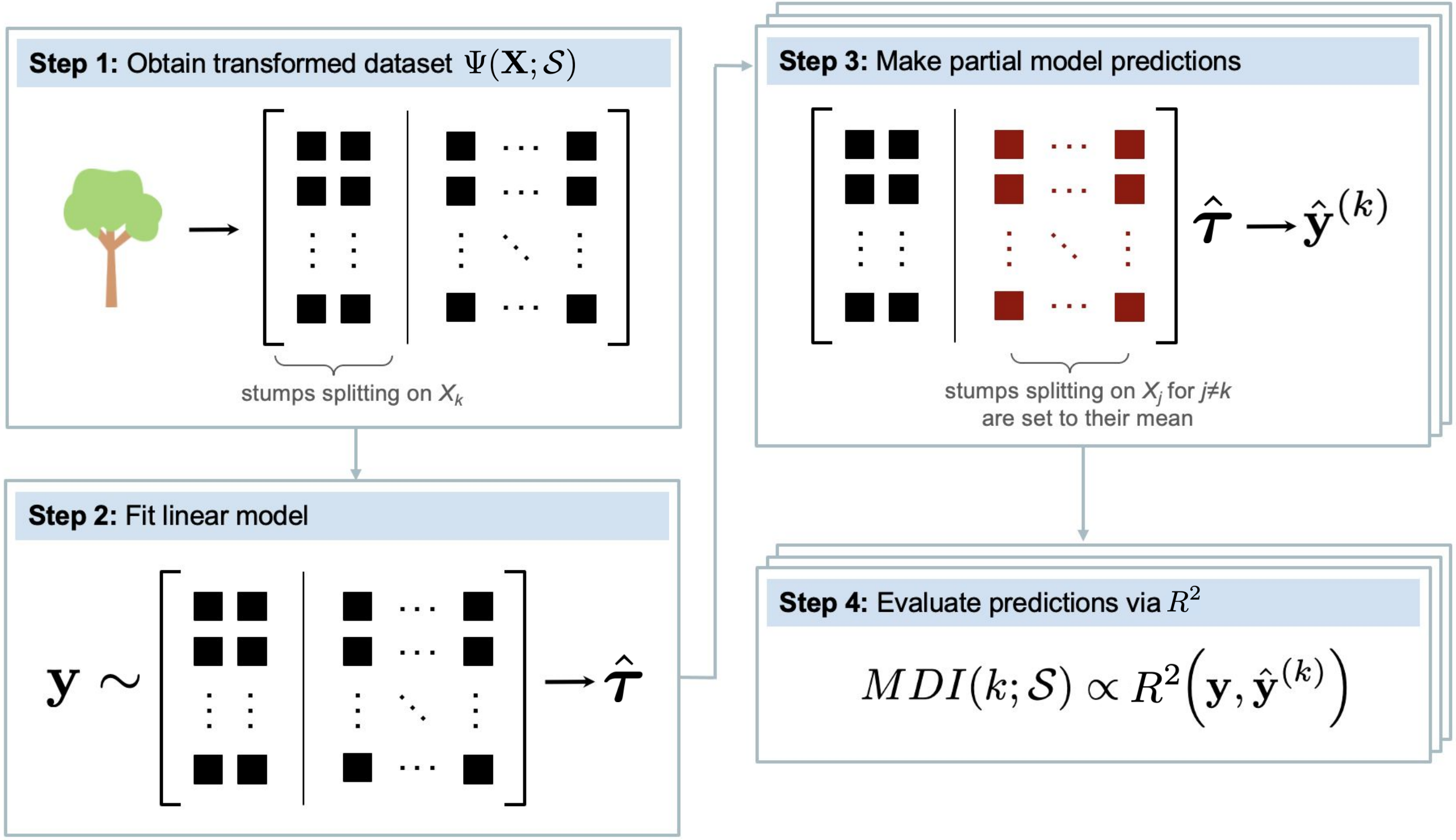
stumps splitting on X_k

Step 2: Fit linear model

Reinterpreting MDI as an R^2


Step 1: Obtain transformed dataset $\Psi(\mathbf{X}; \mathcal{S})$
stumps splitting on X_k
Step 2: Fit linear model
$\mathbf{y} \sim$
$\rightarrow \hat{\boldsymbol{\tau}}$
Step 3: Make partial model predictions
$\hat{\boldsymbol{\tau}} \rightarrow \hat{\mathbf{y}}^{(k)}$
stumps splitting on X_j for $j \neq k$
are set to their mean

Reinterpreting MDI as an R^2



Step 1: Obtain transformed dataset $\Psi(\mathbf{X};\mathcal{S})$

stumps splitting on X_k

Step 2: Fit linear model

$\mathbf{y} \sim \longrightarrow \hat{\boldsymbol{\tau}}$

Step 3: Make partial model predictions

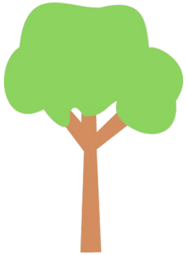
$\hat{\boldsymbol{\tau}} \longrightarrow \hat{\mathbf{y}}^{(k)}$

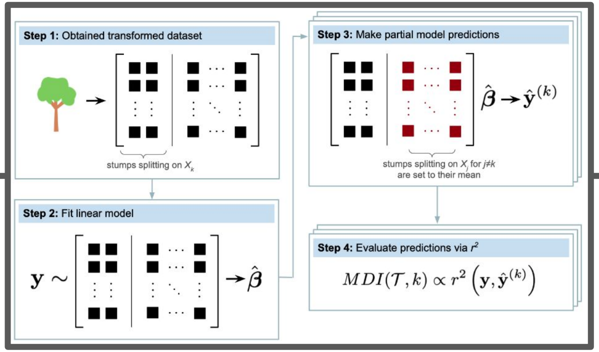
stumps splitting on X_j for $j \neq k$ are set to their mean

Step 4: Evaluate predictions via R^2

$$MDI(k;\mathcal{S}) \propto R^2\left(\mathbf{y},\hat{\mathbf{y}}^{(k)}\right)$$

Reinterpreting MDI as an R^2





Step 1: Obtained transformed dataset

stumps splitting on X_k

Step 2: Fit linear model

$\mathbf{y} \sim \rightarrow \hat{\boldsymbol{\beta}}$

Step 3: Make partial model predictions

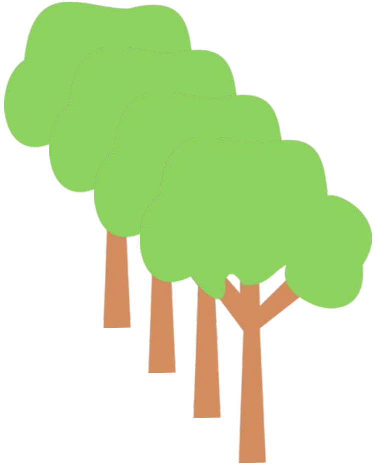
$\hat{\boldsymbol{\beta}} \rightarrow \hat{\mathbf{y}}^{(k)}$

stumps splitting on X_j for $j \neq k$ are set to their mean

Step 4: Evaluate predictions via r^2

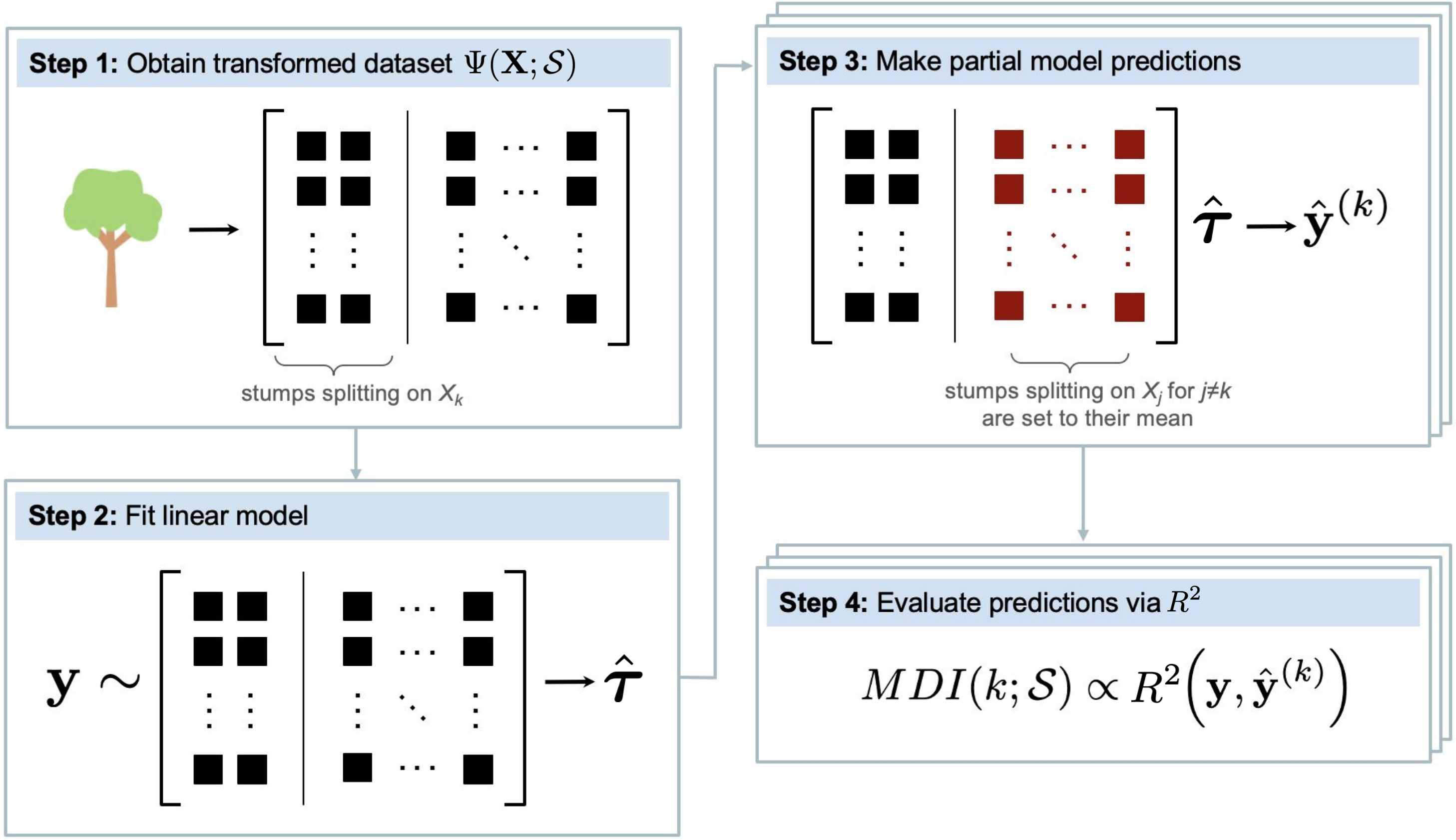
$MDI(\mathcal{T}, k) \propto r^2\left(\mathbf{y}, \hat{\mathbf{y}}^{(k)}\right)$

$$MDI(k;\mathcal{S})$$



$$MDI(k) = \frac{1}{n_{\text{trees}}} \sum_{i=1}^{n_{\text{trees}}} MDI(k;\mathcal{S}_i)$$

Reinterpreting MDI as an R^2



Step 1: Obtain transformed dataset $\Psi(\mathbf{X}; \mathcal{S})$

stumps splitting on X_k

Step 2: Fit linear model

$\mathbf{y} \sim [\;\cdots\;] \rightarrow \hat{\boldsymbol{\tau}}$

Step 3: Make partial model predictions

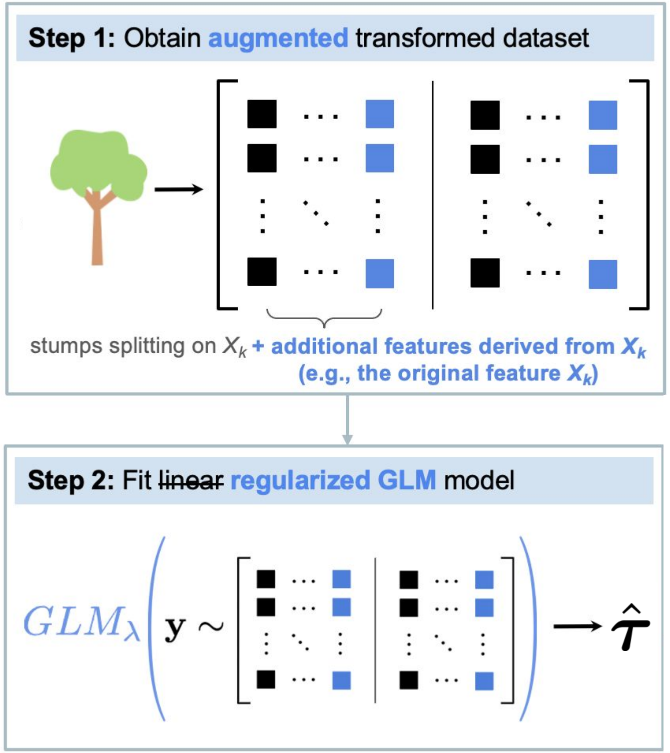
$\hat{\boldsymbol{\tau}} \rightarrow \hat{\mathbf{y}}^{(k)}$

stumps splitting on X_j for $j \neq k$ are set to their mean

Step 4: Evaluate predictions via R^2

$$MDI(k; \mathcal{S}) \propto R^2\left(\mathbf{y}, \hat{\mathbf{y}}^{(k)}\right)$$

MDI+: A Generalized Mean Decrease in Impurity

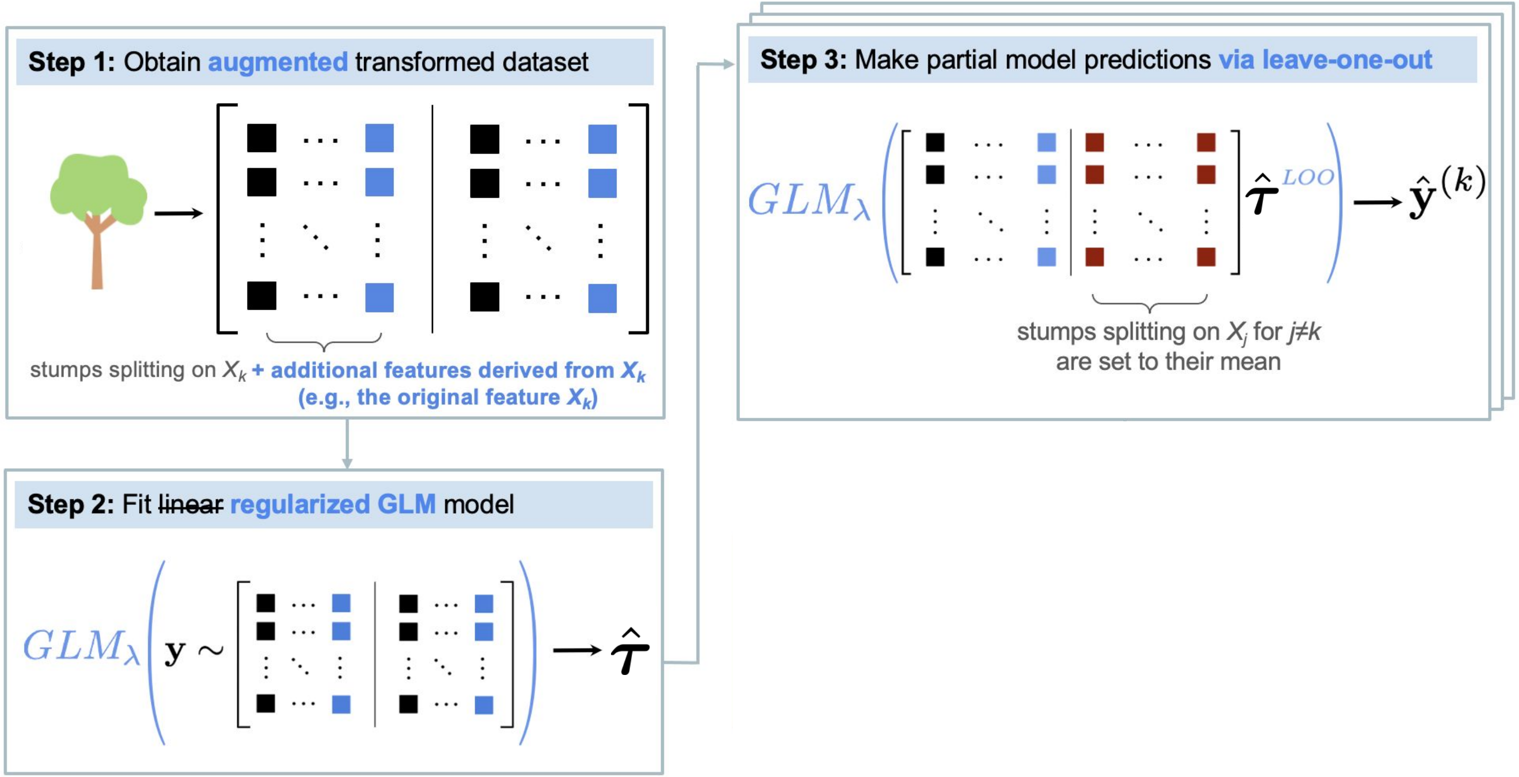


Step 1: Obtain **augmented** transformed dataset

stumps splitting on X_k **+ additional features derived from X_k (e.g., the original feature X_k)**

Step 2: Fit ~~linear~~ **regularized GLM** model

MDI+: A Generalized Mean Decrease in Impurity



Step 1: Obtain augmented transformed dataset

stumps splitting on X_k + additional features derived from X_k (e.g., the original feature X_k)

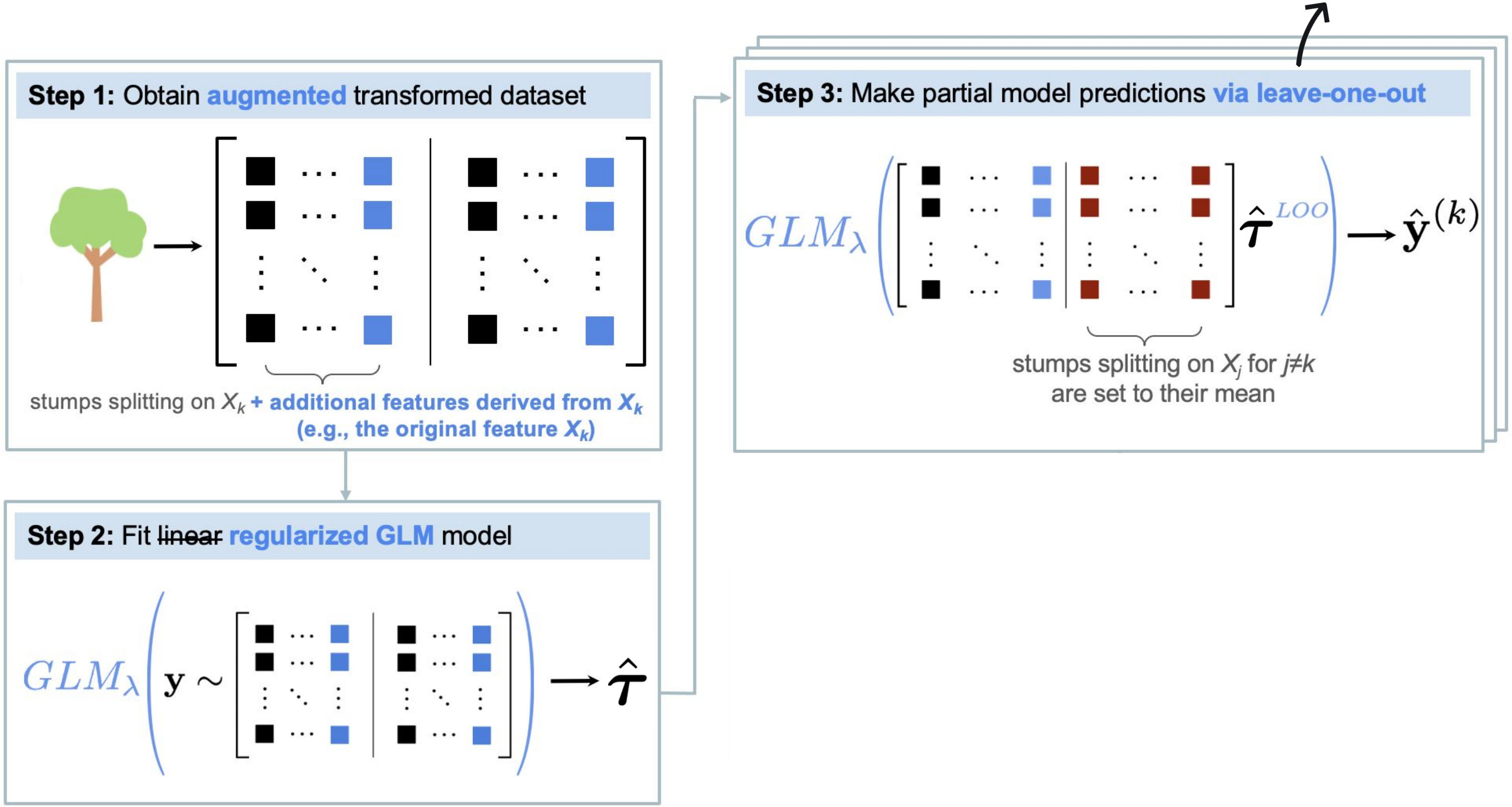
Step 2: Fit ~~linear~~ regularized GLM model

Step 3: Make partial model predictions via leave-one-out

stumps splitting on X_j for $j≠k$ are set to their mean

MDI+: A Generalized Mean Decrease in Impurity

Approximate leave-one-out predictions can be computed without refitting the RF



Step 1: Obtain **augmented** transformed dataset

stumps splitting on X_k **+ additional features derived from X_k (e.g., the original feature X_k)**

Step 2: Fit ~~linear~~ **regularized GLM** model

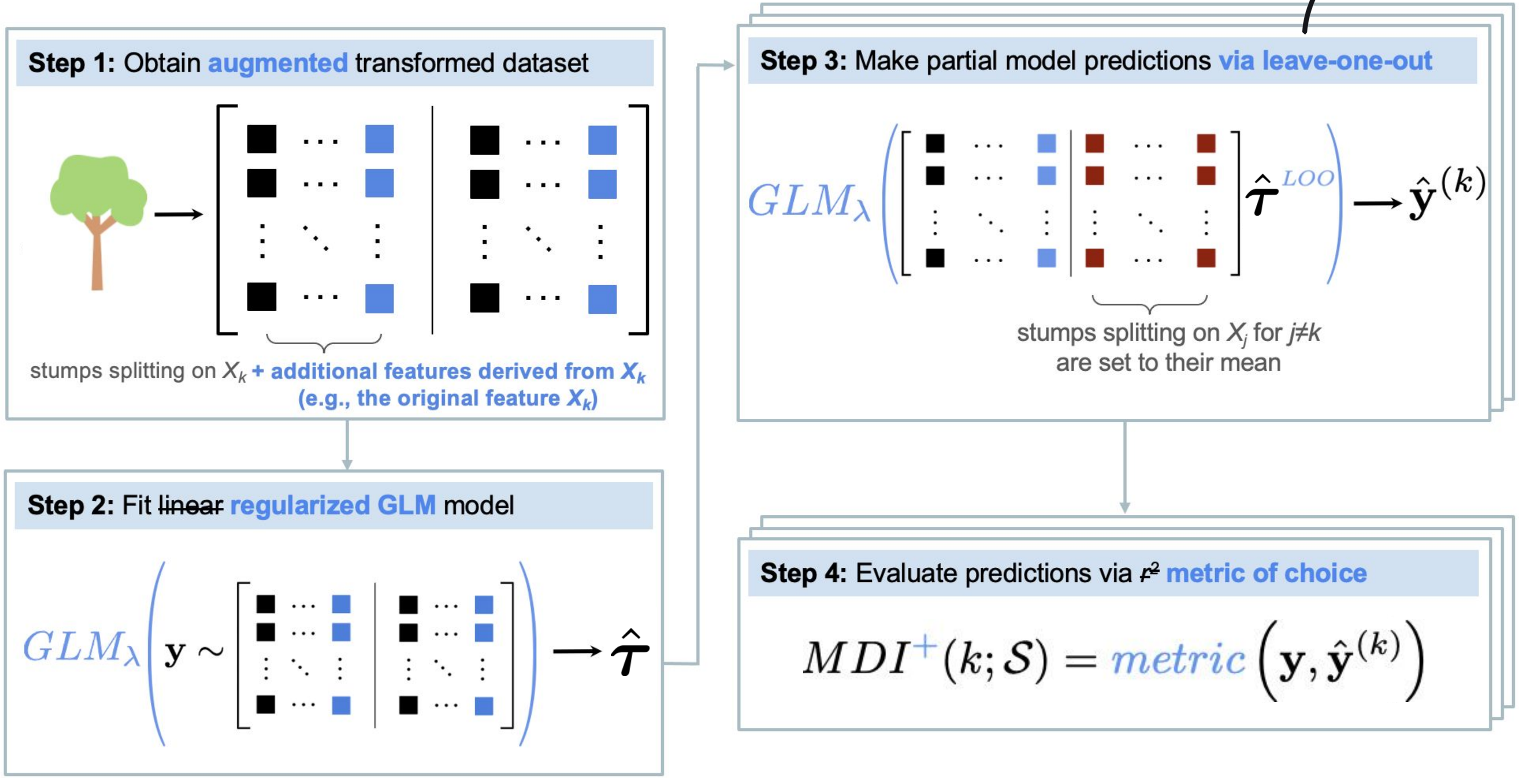
Step 3: Make partial model predictions **via leave-one-out**

stumps splitting on X_j for $j \neq k$ are set to their mean

GLM_λ
$\mathbf{y} \sim$
$\rightarrow \hat{\boldsymbol{\tau}}$
GLM_λ
$\hat{\boldsymbol{\tau}}^{LOO}$
$\rightarrow \hat{\mathbf{y}}^{(k)}$

MDI+: A Generalized Mean Decrease in Impurity

Approximate leave-one-out predictions can be computed without refitting the RF



Step 1: Obtain **augmented** transformed dataset

stumps splitting on X_k **+ additional features derived from X_k (e.g., the original feature X_k)**

Step 2: Fit ~~linear~~ **regularized GLM** model

$GLM_\lambda\left(\mathbf{y} \sim [\,\cdot\,]\right) \longrightarrow \hat{\boldsymbol{\tau}}$

Step 3: Make partial model predictions **via leave-one-out**

$GLM_\lambda\left([\,\cdot\,]\,\hat{\boldsymbol{\tau}}^{LOO}\right) \longrightarrow \hat{\mathbf{y}}^{(k)}$

stumps splitting on X_j for $j{\neq}k$ are set to their mean

Step 4: Evaluate predictions via ~~r^2~~ **metric of choice**

$$MDI^{+}(k;\mathcal{S}) = metric\left(\mathbf{y},\hat{\mathbf{y}}^{(k)}\right)$$

Roadmap of Empirical Results

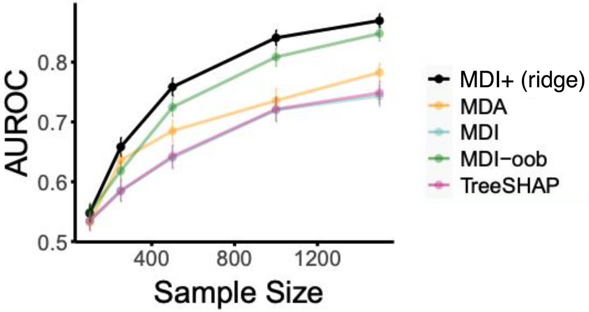
- **Correlation/entropy bias:** MDI+ overcomes correlation and entropy bias using out-of-sample prediction
- **Real data-inspired simulations:** MDI+ improves feature rankings in various regression, classification, and robust regression scenarios
    - Regression: MDI+ with ridge regression as GLM + r^2 metric
    - Classification: MDI+ with l_2-regularized logistic regression as GLM + log-loss metric
    - Robust regression: MDI+ with regularized Huber regression as GLM + Huber loss metric
- **Two real data case studies:** MDI+ identifies well-known gene predictors with greater stability than competing methods (for drug response prediction and breast cancer subtyping)

Roadmap of Empirical Results

- **Correlation/entropy bias:** MDI+ overcomes correlation and entropy bias using out-of-sample prediction
- **Real data-inspired simulations:** MDI+ improves feature rankings in various regression, classification, and robust regression scenarios
  - Regression: MDI+ with ridge regression as GLM + r^2 metric
  - Classification: MDI+ with l_2-regularized logistic regression as GLM + log-loss metric
  - Robust regression: MDI+ with regularized Huber regression as GLM + Huber loss metric
- **Two real data case studies:** MDI+ identifies well-known gene predictors with greater stability than competing methods (for drug response prediction and breast cancer subtyping)

Regression simulation results

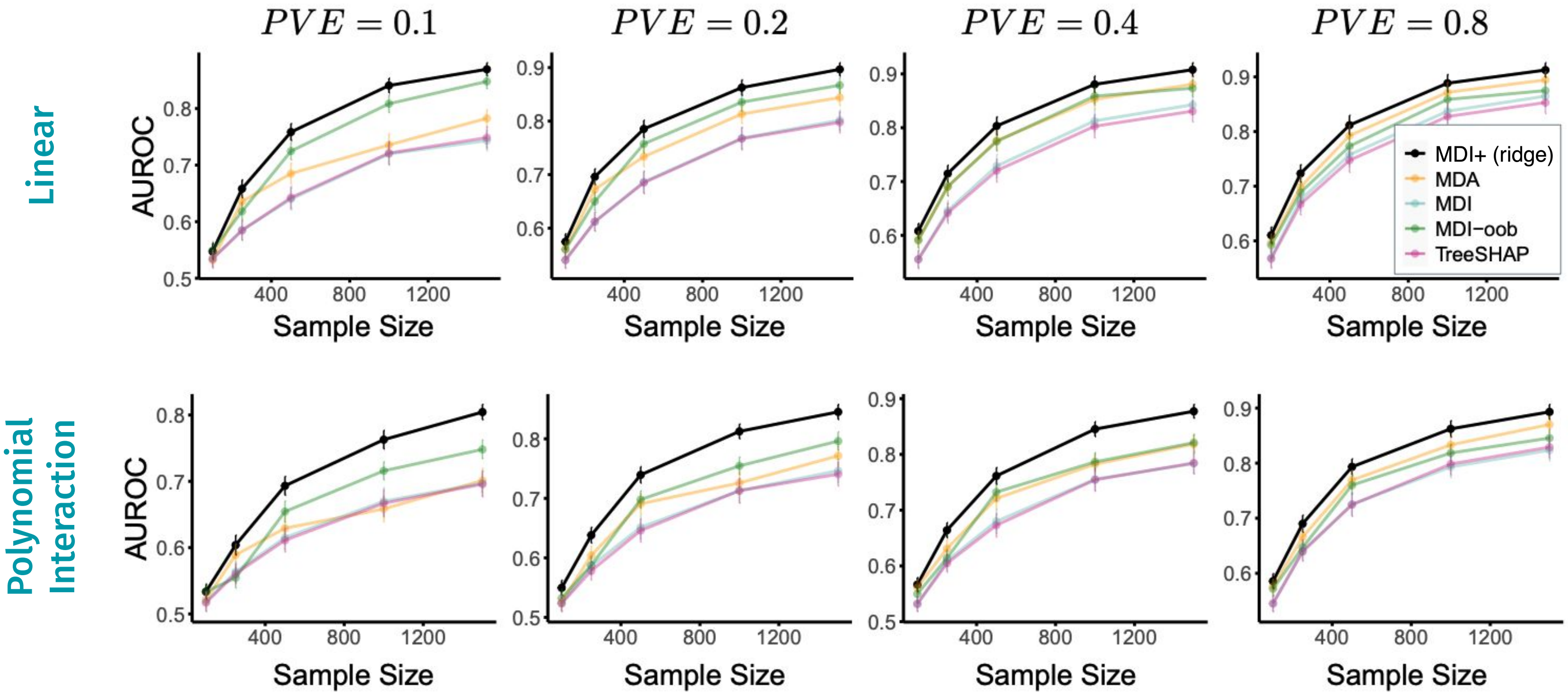
Linear



* X = Splicing dataset

Regression simulation results

Increasing Proportion of Variance Explained (PVE) [i.e., signal] →

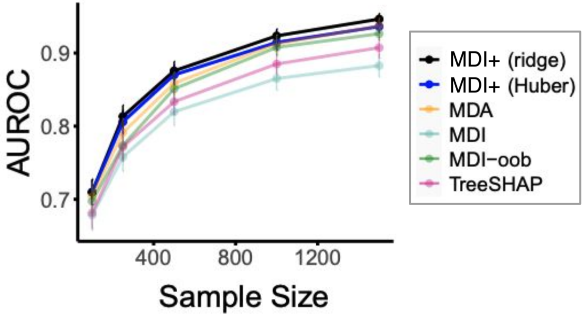


* X = Splicing dataset

In the presence of outliers

Boolean Interaction

0% outliers

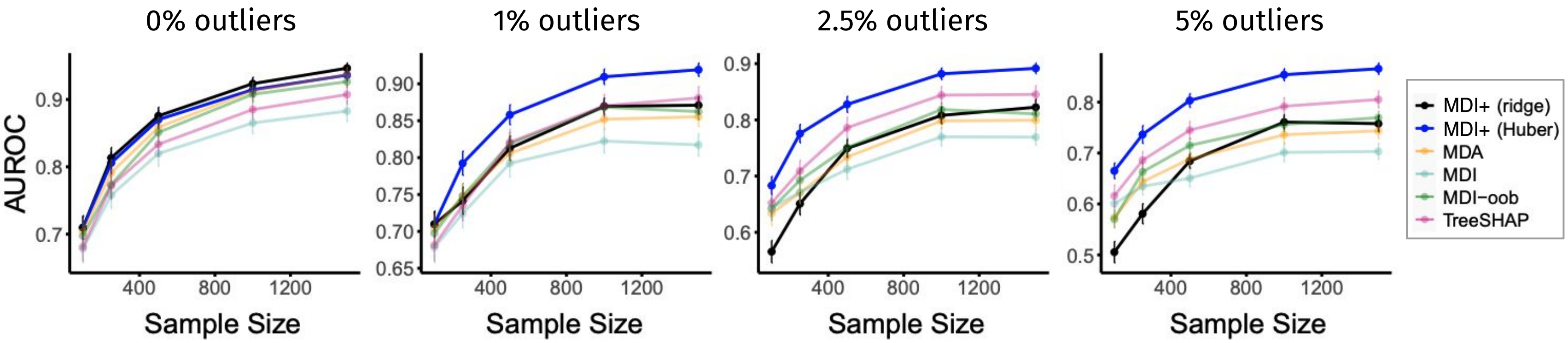


* X = Enhancer activity dataset

In the presence of outliers

Tailoring MDI+ to the problem setting improves feature ranking accuracy

Boolean Interaction

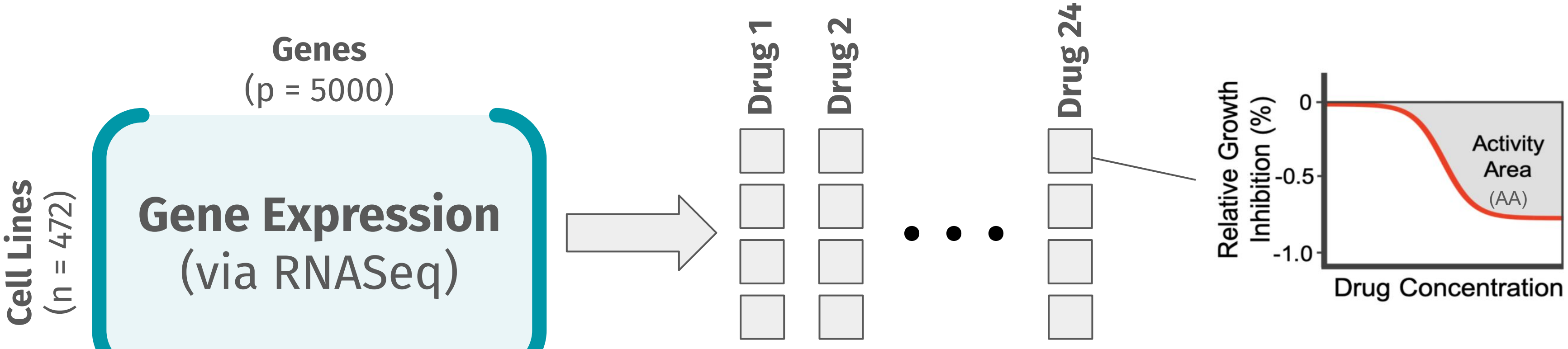


* X = Enhancer activity dataset

Real Data Case Studies

Predicting cancer drug responses (regression)

Dataset: Cancer Cell Line Encyclopedia [Barretina et al. (2012)]



Genes
(p = 5000)

Cell Lines
(n = 472)

Gene Expression
(via RNASeq)

Drug 1

Drug 2

Drug 24

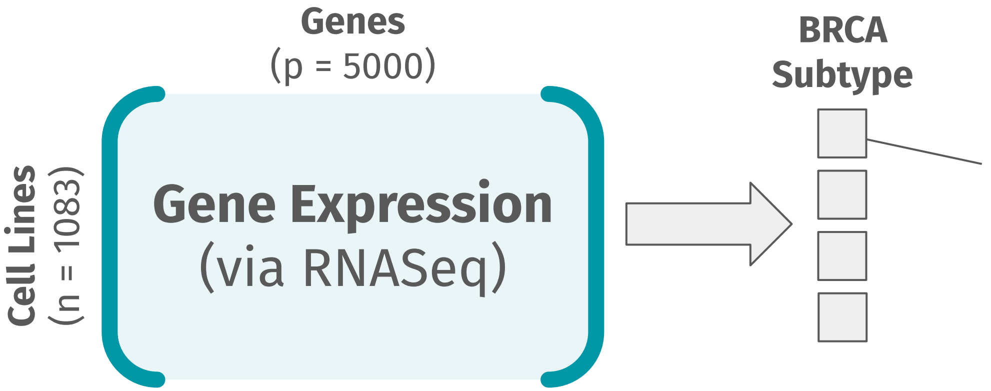
* 24 independent regression problems

Barretina et al. "The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity." *Nature* (2012)

Real Data Case Studies

Predicting breast cancer subtypes (classification)

Dataset: The Cancer Genome Atlas (TCGA) [Parker et al. (2009)]



One of five PAM50 subtypes:

1) Luminal A
2) Luminal B
3) HER2-enriched
4) Basal-like
5) Normal-like

Parker et al. "Supervised risk predictor of breast cancer based on intrinsic subtypes." *Journal of Clinical Oncology* (2009)

Case Study Objectives

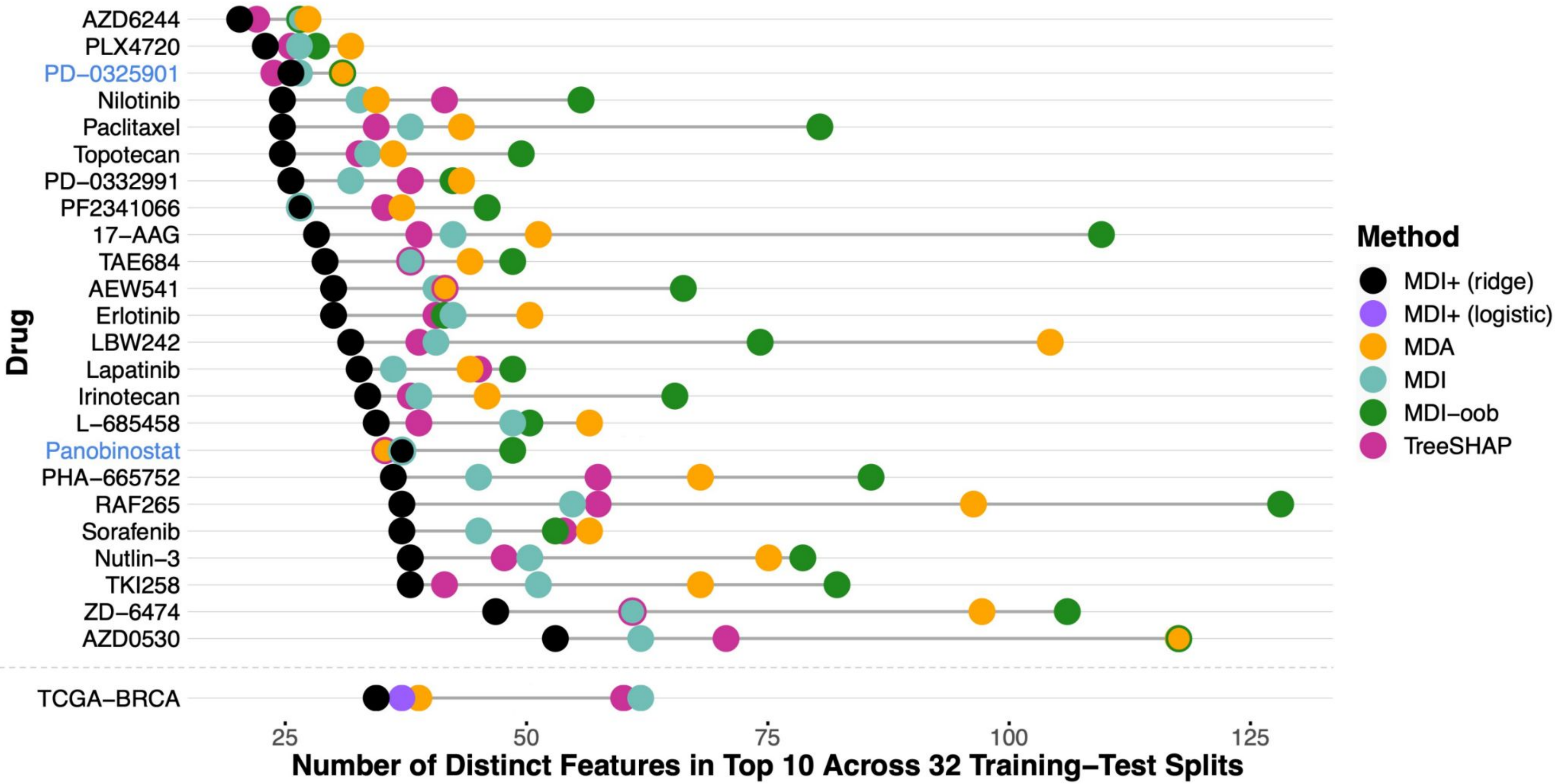
If we apply the feature importance method to 32 different RF fits (all trained on the same real X and y), are the feature rankings **accurate** and **stable**?

Accuracy: MDI+ identified all top gene expression predictors from the original CCLE paper [Barretina et al. (2012)]

- NQO1 gene for 17-AAG; EGFR gene for Erlotinib; ERBB2 gene for Lapatinib; MDM2 gene for Nutlin-3; MET, HGF genes for PF2341066

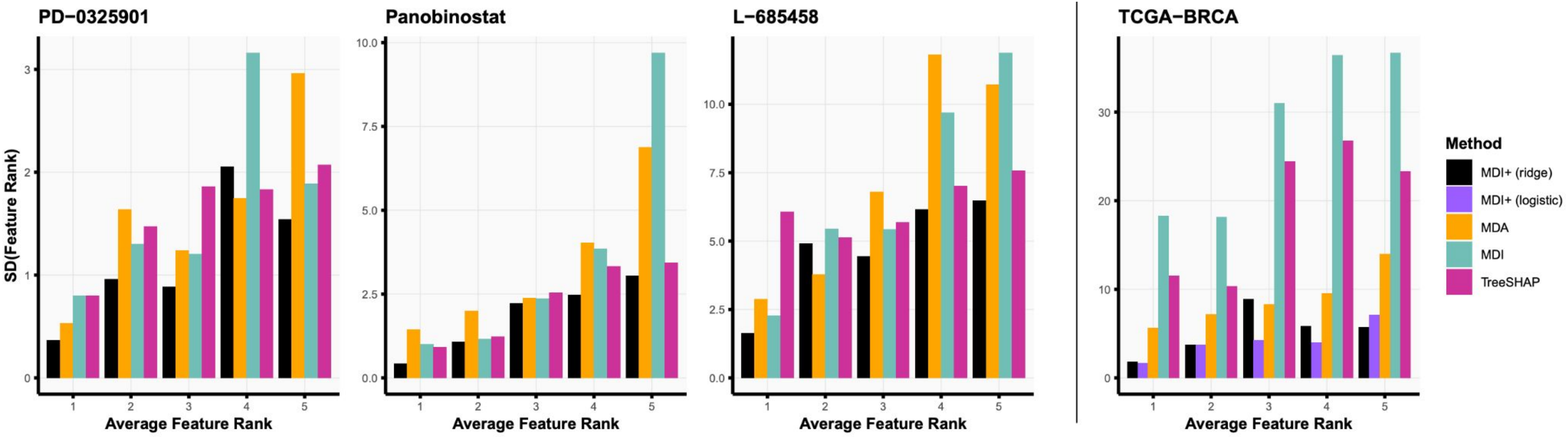
Stability: The feature rankings from MDI+ are more stable across the different RF fits, compared to competing methods (MDI, MDI-oob, MDA, TreeSHAP)

MDI+ is more stable w.r.t. randomness in RF fits

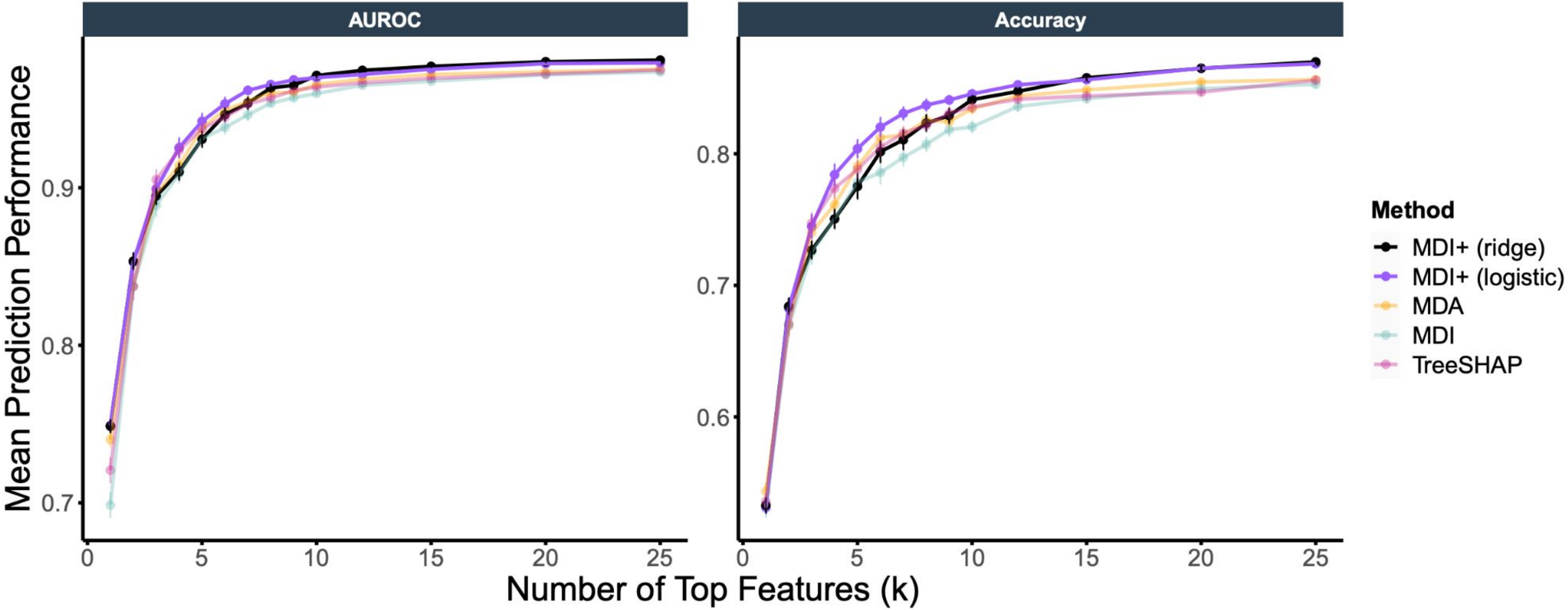


MDI+ is more stable w.r.t. randomness in RF fits

A closer look at the top 5 features shows their ranking distribution is tighter (i.e., more stable) for MDI+ relative to competitors.



Top MDI+ features are predictive of breast cancer subtypes



Summary and Discussion

- **RF+ and MDI+:** provide a flexible random forest-based framework that
  - Overcomes many of the inductive biases of RF/decision trees and limitations of MDI
  - Allows the analyst to tailor the feature importance computation to the data/problem structure
- **Key building block:** rethinking RF/MDI as a linear model
- **Connection between decision trees and linear regression** opens the door to many interesting future directions
  - A new class of prediction algorithms that leverage the tree basis/stump features
  - Possibility to build upon familiar linear regression tools (e.g., for inference)

Thank you!

Email: ttang4@nd.edu
Website: tiffanymtang.github.io

Code in **`imodels`** python package: https://github.com/csinva/imodels
Preprint (RF+/MDI+): https://arxiv.org/abs/2307.01932
Preprint (NeRF+): in progress

Collaborators:

Abhineet Agarwal

Ana Kenney

Yan Shuo Tan

Bin Yu

Ji Zhu

Elizaveta (Liza) Levina

Appendix

Correlation bias simulation setup

X generated with block covariance structure

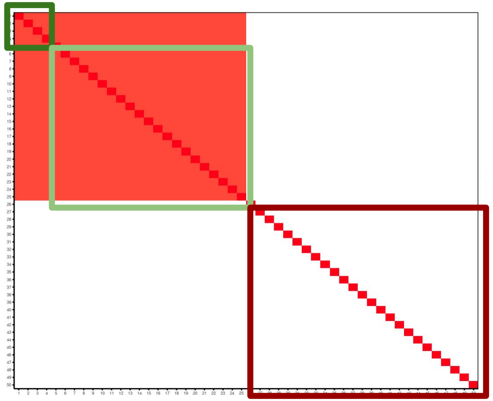
$\boldsymbol{X} \sim N(0, \boldsymbol{\Sigma})$ with n = 250, p = 100

5 “Correlated Signal” features (Sig)

45 “Correlated Non-signal” features (C-NSig)

50 “Uncorrelated Non-signal” features (NSig)

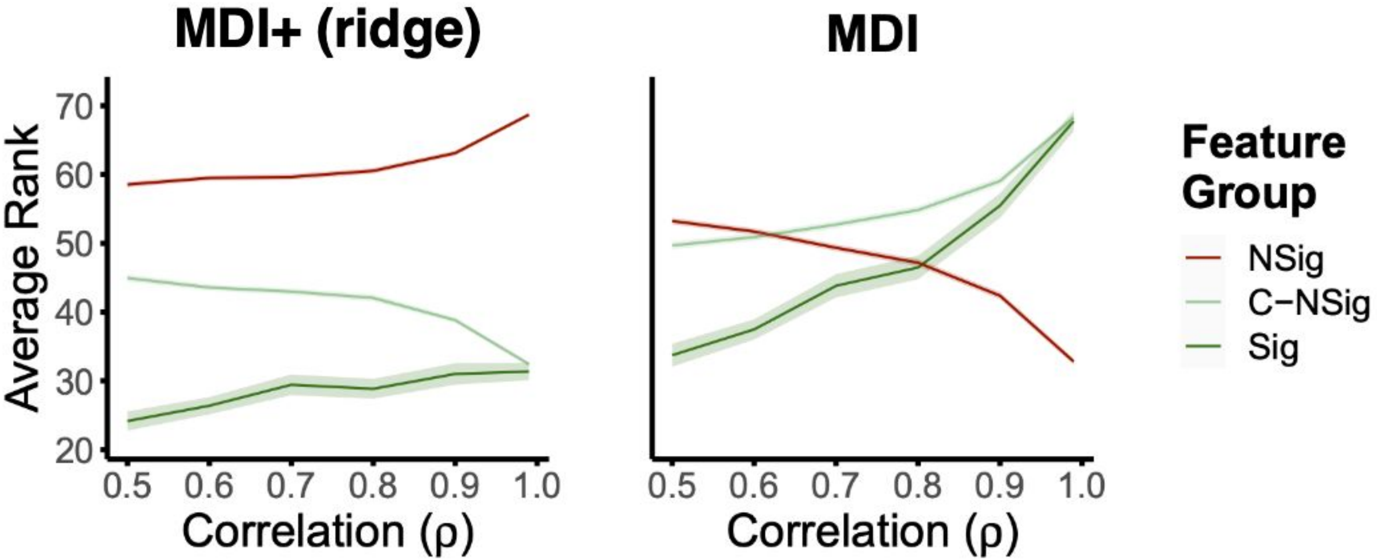
$\boldsymbol{\Sigma} =$



y generated from sparse linear function

$$y = x_1 + x_2 + x_3 + x_4 + x_5 + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2)$$

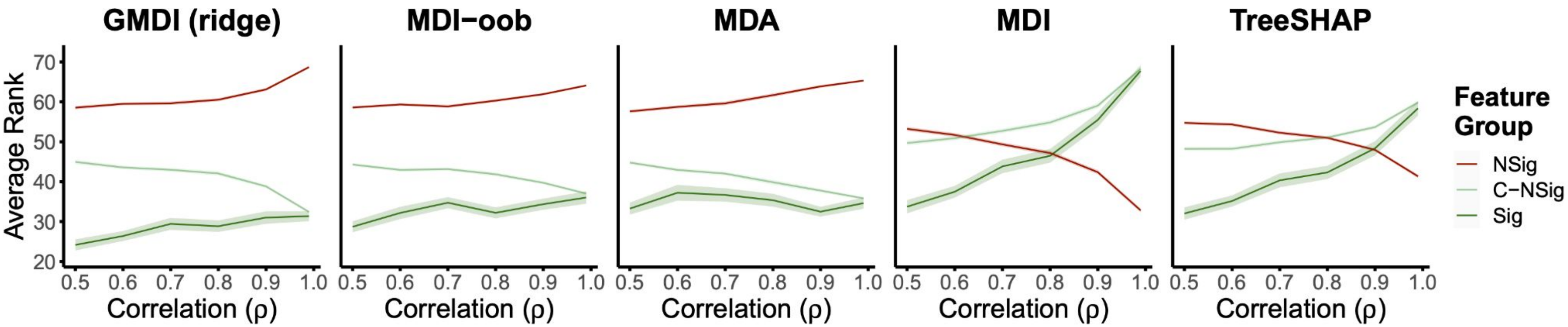
GMDI mitigates **correlated** feature bias



MDI+ overcomes bias by using **out-of-sample prediction**

MDI ranks non-signal features as more important than signal features

GMDI mitigates correlated feature bias



MDI+, MDI-oob, and MDA overcome bias by using **out-of-sample prediction**

MDI and TreeSHAP rank non-signal features as more important than signal features