High-Dimensional Variable Selection via Model-X Knockoffs

Lucas Janson

Harvard University Department of Statistics

CHASC Talk, Apr 9, 2019

Problem Statement

Controlled Variable Selection

Given:

- Y an outcome of interest (AKA response or dependent variable),
- $X_1, \ldots, X_p$ a set of p potential explanatory variables (AKA covariates, features, or independent variables),

How can we select important explanatory variables with few mistakes?

Controlled Variable Selection

Given:

- Y an outcome of interest (AKA response or dependent variable),
- $X_1, \ldots, X_p$ a set of p potential explanatory variables (AKA covariates, features, or independent variables),

How can we select important explanatory variables with few mistakes?

Applications to:

- Biology/genomics/health care

Controlled Variable Selection

Given:

- Y an outcome of interest (AKA response or dependent variable),
- $X_1, \ldots, X_p$ a set of p potential explanatory variables (AKA covariates, features, or independent variables),

How can we select important explanatory variables with few mistakes?

Applications to:

- Biology/genomics/health care
- Economics/political science

Controlled Variable Selection

Given:

- Y an outcome of interest (AKA response or dependent variable),
- $X_1, \ldots, X_p$ a set of p potential explanatory variables (AKA covariates, features, or independent variables),

How can we select important explanatory variables with few mistakes?

Applications to:

- Biology/genomics/health care
- Economics/political science
- Industry/technology
- **Astronomy?**

Controlled Variable Selection (cont'd)

What is an important variable?

Controlled Variable Selection (cont'd)

What is an important variable?

We consider X_j to be unimportant if the conditional distribution of Y given $X_1, \ldots, X_p$ does not depend on X_j. Formally, X_j is unimportant if it is conditionally independent of Y given $X_{\text{-}j}$:

$$Y \perp\!\!\!\perp X_j \mid X_{\text{-}j}$$

Controlled Variable Selection (cont'd)

What is an important variable?

We consider X_j to be unimportant if the conditional distribution of Y given $X_1, \ldots, X_p$ does not depend on X_j. Formally, X_j is unimportant if it is conditionally independent of Y given $X_{\text{-}j}$:

$$Y \perp\!\!\!\perp X_j \mid X_{\text{-}j}$$

Markov Blanket of Y: smallest set S such that $Y \perp\!\!\!\perp X_{\text{-}S} \,|\, X_S$

Controlled Variable Selection (cont'd)

What is an important variable?

We consider X_j to be unimportant if the conditional distribution of Y given $X_1, \dots, X_p$ does not depend on X_j. Formally, X_j is unimportant if it is conditionally independent of Y given $X_{\text{-}j}$:

$$Y \perp\!\!\!\perp X_j \mid X_{\text{-}j}$$

Markov Blanket of Y: smallest set S such that $Y \perp\!\!\!\perp X_{\text{-}S} \mid X_S$

For GLMs with no stochastically redundant covariates, equivalent to $\{j : \beta_j = 0\}$

Controlled Variable Selection (cont'd)

What is an important variable?

We consider X_j to be unimportant if the conditional distribution of Y given $X_1, \ldots, X_p$ does not depend on X_j. Formally, X_j is unimportant if it is conditionally independent of Y given $X_{\text{-}j}$:

$$Y \perp\!\!\!\perp X_j \mid X_{\text{-}j}$$

Markov Blanket of Y: smallest set S such that $Y \perp\!\!\!\perp X_{\text{-}S} \mid X_S$

For GLMs with no stochastically redundant covariates, equivalent to $\{j : \beta_j = 0\}$

To make sure we do not make too many mistakes, we seek to select a set $\hat{S}$ to control the **false discovery rate (FDR)**:

$$\text{FDR} = \mathbb{E}\left[\frac{\#\{j \text{ in } \hat{S} \ : \ X_j \text{ unimportant}\}}{\#\{j \text{ in } \hat{S}\}}\right] \leq q \quad \text{(e.g., 10\%)}$$

"Here is a set of variables $\hat{S}$, 90% of which I expect to be important"

Group Knockoffs

"What if two variables are so correlated as to be indistinguishable?"

Group Knockoffs

“What if two variables are so correlated as to be indistinguishable?”

Insufficient info to select either variable confidently (needed for FDR control)

Group Knockoffs

"What if two variables are so correlated as to be indistinguishable?"

Insufficient info to select either variable confidently (needed for FDR control)

Single-variable resolution impossible: **wrong question**

- Group variables with their highly-correlated neighbors: $\biguplus_{k=1}^{m} G_k = \{1, \ldots, p\}$

Group Knockoffs

"What if two variables are so correlated as to be indistinguishable?"

Insufficient info to select either variable confidently (needed for FDR control)

Single-variable resolution impossible: **wrong question**

- Group variables with their highly-correlated neighbors: $\biguplus_{k=1}^{m} G_k = \{1, \ldots, p\}$
- Redefine null hypothesis on per-group basis: group G_k is unimportant if

$$Y \perp\!\!\!\perp X_{G_k} \mid X_{\text{-}G_k}$$

Group Knockoffs

"What if two variables are so correlated as to be indistinguishable?"

Insufficient info to select either variable confidently (needed for FDR control)

Single-variable resolution impossible: **wrong question**

- Group variables with their highly-correlated neighbors: $\biguplus_{k=1}^{m} G_k = \{1, \dots, p\}$
- Redefine null hypothesis on per-group basis: group G_k is unimportant if

$$Y \perp\!\!\!\perp X_{G_k} \mid X_{\text{-}G_k}$$

- Redefine FDR: for selected set of groups $\hat{S}_G$,

$$\text{FDR}_G = \mathbb{E}\left[\frac{\#\{k \text{ in } \hat{S}_G \;:\; G_k \text{ contains no important variables}\}}{\#\{j \text{ in } \hat{S}_G\}}\right] \leq q \quad \text{(e.g., 10\%)}$$

Group Knockoffs

"What if two variables are so correlated as to be indistinguishable?"

Insufficient info to select either variable confidently (needed for FDR control)

Single-variable resolution impossible: **wrong question**

- Group variables with their highly-correlated neighbors: $\biguplus_{k=1}^{m} G_k = \{1, \ldots, p\}$
- Redefine null hypothesis on per-group basis: group G_k is unimportant if

$$Y \perp\!\!\!\perp X_{G_k} \mid X_{\text{-}G_k}$$

- Redefine FDR: for selected set of groups $\hat{S}_G$,

$$\text{FDR}_G = \mathbb{E}\left[\frac{\#\{k \text{ in } \hat{S}_G \ : \ G_k \text{ contains no important variables}\}}{\#\{j \text{ in } \hat{S}_G\}}\right] \le q \quad \text{(e.g., 10\%)}$$

Straightforward extension to group knockoffs (Dai and Barber, 2016)

Outline

- Review of (model-X) **knockoffs**, which uses knowledge of X's distribution to solve the controlled variable selection problem with
  - Any model for Y and $X_1, \dots, X_p$
  - Any dimension (including $p > n$)
  - Finite-sample control (non-asymptotic) of FDR
  - Practical performance on real problems ($\approx 2\times$ power in real GWAS)

Outline

- Review of (model-X) **knockoffs**, which uses knowledge of X's distribution to solve the controlled variable selection problem with
  - Any model for Y and $X_1, \ldots, X_p$
  - Any dimension (including $p > n$)
  - Finite-sample control (non-asymptotic) of FDR
  - Practical performance on real problems ($\approx 2\times$ power in real GWAS)
- **Metropolized Knockoff Sampling**
  - New extremely general way to generate knockoffs
  - Needs only an unnormalized density function

Outline

- Review of (model-X) **knockoffs**, which uses knowledge of X's distribution to solve the controlled variable selection problem with
  - Any model for Y and $X_1, \ldots, X_p$
  - Any dimension (including $p > n$)
  - Finite-sample control (non-asymptotic) of FDR
  - Practical performance on real problems ($\approx 2\times$ power in real GWAS)
- **Metropolized Knockoff Sampling**
  - New extremely general way to generate knockoffs
  - Needs only an unnormalized density function
- **Conditional Knockoffs**
  - Relaxes requirement on the knowledge of X's distribution
  - Same exact guarantees, and almost identical power

Existing Methods for Controlled Variable Selection

- Marginal p-values
  - Excellent exploratory tool
  - Answer low-dimensional question $Y \perp\!\!\!\perp X_j$ instead of $Y \perp\!\!\!\perp X_j \mid X_{\text{-}j}$
  - Can lose power, interpretation, and FDR control when X_j are correlated

Existing Methods for Controlled Variable Selection

- Marginal p-values
  - Excellent exploratory tool
  - Answer low-dimensional question $Y \perp\!\!\!\perp X_j$ instead of $Y \perp\!\!\!\perp X_j \mid X_{\text{-}j}$
  - Can lose power, interpretation, and FDR control when X_j are correlated
- Bayesian inference
  - Great way of incorporating prior information
  - Computation constrains to simple priors which may not match actual prior knowledge
  - Inference (esp. in high dimensions) is sensitive to choice of prior

Existing Methods for Controlled Variable Selection

- Marginal p-values
  - Excellent exploratory tool
  - Answer low-dimensional question $Y \perp\!\!\!\perp X_j$ instead of $Y \perp\!\!\!\perp X_j \mid X_{\text{-}j}$
  - Can lose power, interpretation, and FDR control when X_j are correlated
- Bayesian inference
  - Great way of incorporating prior information
  - Computation constrains to simple priors which may not match actual prior knowledge
  - Inference (esp. in high dimensions) is sensitive to choice of prior
- Machine learning
  - Excellent for prediction
  - Cross-validation comes with no statistical guarantees
  - Statistical analysis exists only for simplest methods (lasso) and makes unrealistic assumptions

Model-X Knockoffs
(Candès, Fan, **J.**, Lv, JRSSB, 2018)

View from 10,000 feet

You have:

- n data samples of Y and X stacked into $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{X} \in \mathbb{R}^{n \times p}$

View from 10,000 feet

You have:

- n data samples of Y and X stacked into $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{X} \in \mathbb{R}^{n \times p}$
- Algorithm to compute **variable importance measure** Z_j of each X_j for Y
  - This need not be based on any statistical model, or have any statistical properties at all
  - For instance, you could fit any machine learning method and use the drop in prediction accuracy when $\boldsymbol{X}_j$ is removed from the data

View from 10,000 feet

You have:

- n data samples of Y and X stacked into $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{X} \in \mathbb{R}^{n \times p}$
- Algorithm to compute **variable importance measure** Z_j of each X_j for Y
  - This need not be based on any statistical model, or have any statistical properties at all
  - For instance, you could fit any machine learning method and use the drop in prediction accuracy when $\boldsymbol{X}_j$ is removed from the data
- Desired FDR level q but no way to use Z_j to control it

View from 10,000 feet

You have:

- n data samples of Y and X stacked into $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{X} \in \mathbb{R}^{n \times p}$
- Algorithm to compute **variable importance measure** Z_j of each X_j for Y
  - This need not be based on any statistical model, or have any statistical properties at all
  - For instance, you could fit any machine learning method and use the drop in prediction accuracy when $\boldsymbol{X}_j$ is removed from the data
- Desired FDR level q but no way to use Z_j to control it

If you can model X's distribution, knockoffs allows you to:

- Select a subset of the variables based on your variable importance measure and nothing else, while controlling the FDR exactly (no asymptotics)

View from 10,000 feet

You have:

- n data samples of Y and X stacked into $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{X} \in \mathbb{R}^{n \times p}$
- Algorithm to compute **variable importance measure** Z_j of each X_j for Y
  - This need not be based on any statistical model, or have any statistical properties at all
  - For instance, you could fit any machine learning method and use the drop in prediction accuracy when $\boldsymbol{X}_j$ is removed from the data
- Desired FDR level q but no way to use Z_j to control it

If you can model X's distribution, knockoffs allows you to:

- Select a subset of the variables based on your variable importance measure and nothing else, while controlling the FDR exactly (no asymptotics)

$$\boldsymbol{y}, \quad \boldsymbol{X}_1, \ldots, \boldsymbol{X}_p$$

View from 10,000 feet

You have:

- n data samples of Y and X stacked into $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{X} \in \mathbb{R}^{n \times p}$
- Algorithm to compute **variable importance measure** Z_j of each X_j for Y
  - This need not be based on any statistical model, or have any statistical properties at all
  - For instance, you could fit any machine learning method and use the drop in prediction accuracy when $\boldsymbol{X}_j$ is removed from the data
- Desired FDR level q but no way to use Z_j to control it

If you can model X's distribution, knockoffs allows you to:

- Select a subset of the variables based on your variable importance measure and nothing else, while controlling the FDR exactly (no asymptotics)

$$
\begin{array}{rc}
 & \boldsymbol{y}, \quad \boldsymbol{X}_1, \ldots, \boldsymbol{X}_p \\
 & \downarrow \\
\text{Variable importances} & Z_1, \ldots, Z_p
\end{array}
$$

View from 10,000 feet

You have:

- n data samples of Y and X stacked into $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{X} \in \mathbb{R}^{n \times p}$
- Algorithm to compute **variable importance measure** Z_j of each X_j for Y
  - This need not be based on any statistical model, or have any statistical properties at all
  - For instance, you could fit any machine learning method and use the drop in prediction accuracy when $\boldsymbol{X}_j$ is removed from the data
- Desired FDR level q but no way to use Z_j to control it

If you can model X's distribution, knockoffs allows you to:

- Select a subset of the variables based on your variable importance measure and nothing else, while controlling the FDR exactly (no asymptotics)

$$\begin{array}{rc} \boldsymbol{y}, & \boldsymbol{X}_1, \ldots, \boldsymbol{X}_p \\ & \downarrow \\ \text{Variable importances} & Z_1, \ldots, Z_p \end{array}$$

View from 10,000 feet

You have:

- n data samples of Y and X stacked into $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{X} \in \mathbb{R}^{n \times p}$
- Algorithm to compute **variable importance measure** Z_j of each X_j for Y
  - This need not be based on any statistical model, or have any statistical properties at all
  - For instance, you could fit any machine learning method and use the drop in prediction accuracy when $\boldsymbol{X}_j$ is removed from the data
- Desired FDR level q but no way to use Z_j to control it

If you can model X's distribution, knockoffs allows you to:

- Select a subset of the variables based on your variable importance measure and nothing else, while controlling the FDR exactly (no asymptotics)

$$\boxed{\begin{array}{rc} \boldsymbol{y}, & \boldsymbol{X}_1, \ldots, \boldsymbol{X}_p \\ & \downarrow \\ \text{Variable importances} & Z_1, \ldots, Z_p \end{array}} \overset{\text{knockoffs}}{\longrightarrow} \hat{S} \subseteq \{1, \ldots, p\} \text{ s.t. FDR} \le q$$

Overview of the Knockoffs Procedure

(1) **Construct knockoffs**:

- Artificial versions ("knockoffs") of each variable
- Act as controls for assessing importance of original variables

Overview of the Knockoffs Procedure

(1) **Construct knockoffs**:

- Artificial versions ("knockoffs") of each variable
- Act as controls for assessing importance of original variables

(2) **Compute variable importance statistics**:

- Compute statistics measuring variable importance for all variables and knockoffs

Overview of the Knockoffs Procedure

(1) **Construct knockoffs**:
- Artificial versions ("knockoffs") of each variable
- Act as controls for assessing importance of original variables

(2) **Compute variable importance statistics**:
- Compute statistics measuring variable importance for all variables and knockoffs

(3) **Select variables**:
- Select variables whose importance statistic sufficiently larger than its knockoff
- "Sufficiently larger" is well-defined through a concrete step-up procedure

Overview of the Knockoffs Procedure

(1) **Construct knockoffs**:

- Artificial versions ("knockoffs") of each variable
- Act as controls for assessing importance of original variables

(2) **Compute variable importance statistics**:

- Compute statistics measuring variable importance for all variables and knockoffs

(3) **Select variables**:

- Select variables whose importance statistic sufficiently larger than its knockoff
- "Sufficiently larger" is well-defined through a concrete step-up procedure

Symmetry of null variables and their knockoffs guarantees **exchangeability** of their corresponding importance statistics

Overview of the Knockoffs Procedure

(1) **Construct knockoffs**:

- Artificial versions ("knockoffs") of each variable
- Act as controls for assessing importance of original variables

(2) **Compute variable importance statistics**:

- Compute statistics measuring variable importance for all variables and knockoffs

(3) **Select variables**:

- Select variables whose importance statistic sufficiently larger than its knockoff
- "Sufficiently larger" is well-defined through a concrete step-up procedure

Symmetry of null variables and their knockoffs guarantees **exchangeability** of their corresponding importance statistics

That symmetry leads to selection in step (3) controlling the FDR exactly

A Picture for Intuition

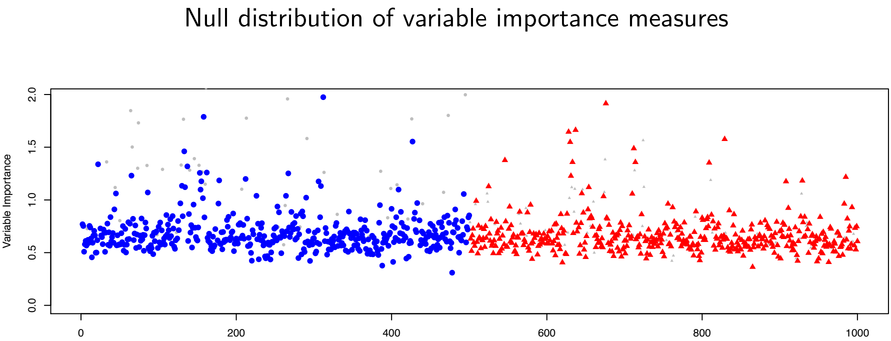


Figure: Variable importance measures for 500 variables and their knockoffs. Colored points are nulls, grey are non-nulls.

Knockoff Construction

Valid knockoffs are defined by

(1) Swap exchangeability:

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$
$$\overset{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

Knockoff Construction

Valid knockoffs are defined by

(1) Swap exchangeability:

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p] \stackrel{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

(2) Nullity: $\widetilde{\boldsymbol{X}} \perp\!\!\!\perp \boldsymbol{y} \mid \boldsymbol{X}$ (don't look at $\boldsymbol{y}$ when constructing $\widetilde{\boldsymbol{X}}$)

Knockoff Construction

Valid knockoffs are defined by

(1) Swap exchangeability:

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$
$$\overset{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

(2) Nullity: $\widetilde{\boldsymbol{X}} \perp\!\!\!\perp \boldsymbol{y} \mid \boldsymbol{X}$ (don't look at $\boldsymbol{y}$ when constructing $\widetilde{\boldsymbol{X}}$)

Example: $(X_1, \ldots, X_p) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, need

$$\mathrm{Cov}(X_1, \ldots, X_p, \widetilde{X}_1, \ldots, \widetilde{X}_p) = \begin{bmatrix} \boldsymbol{\Sigma} & \boldsymbol{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} \\ \boldsymbol{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} & \boldsymbol{\Sigma} \end{bmatrix}$$

Knockoff Construction

Valid knockoffs are defined by

(1) Swap exchangeability:

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$
$$\overset{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

(2) Nullity: $\widetilde{\boldsymbol{X}} \perp\!\!\!\perp \boldsymbol{y} \mid \boldsymbol{X}$ (don't look at $\boldsymbol{y}$ when constructing $\widetilde{\boldsymbol{X}}$)

Example: $(X_1, \ldots, X_p) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, need

$$\mathrm{Cov}(X_1, \ldots, X_p, \widetilde{X}_1, \ldots, \widetilde{X}_p) = \begin{bmatrix} \boldsymbol{\Sigma} & \boldsymbol{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} \\ \boldsymbol{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} & \boldsymbol{\Sigma} \end{bmatrix}$$

Efficient knockoff constructions for the following X distributions:

- Multivariate Gaussian (Candès et al., 2018)
- Discrete Markov chains (Sesia et al., 2019)
- Hidden Markov models (Sesia et al., 2019)
- Gaussian mixture models (Gimenez et al., 2018)

Robustness

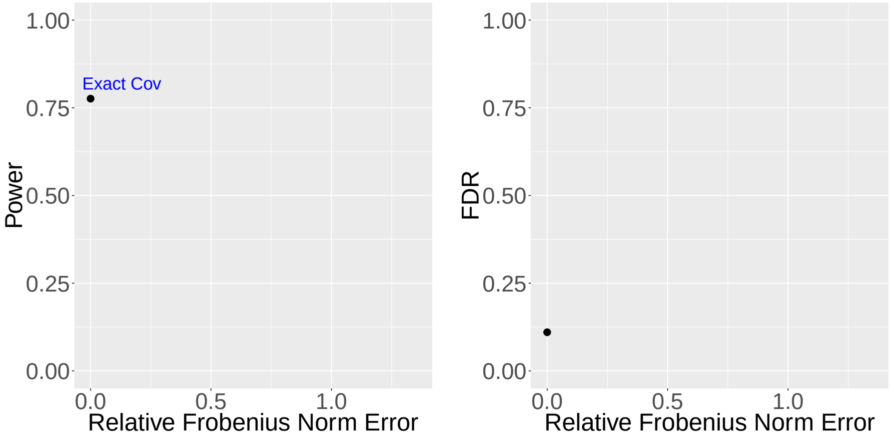


Figure: Covariates are **AR(1) with autocorrelation coefficient 0.3**. $n = 800$, $p = 1500$, and target FDR is 10%. Y comes from a binomial linear model with logit link function with 50 nonzero entries.

Robustness

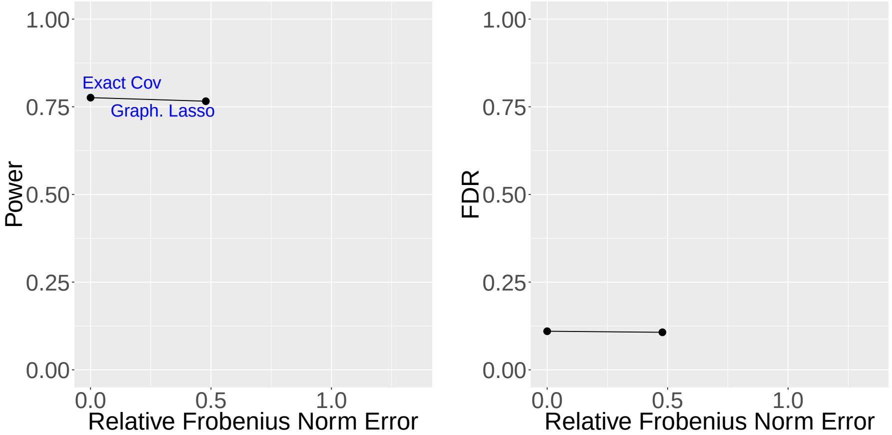


Figure: Covariates are **AR(1) with autocorrelation coefficient 0.3**. $n = 800$, $p = 1500$, and target FDR is 10%. Y comes from a binomial linear model with logit link function with 50 nonzero entries.

Robustness

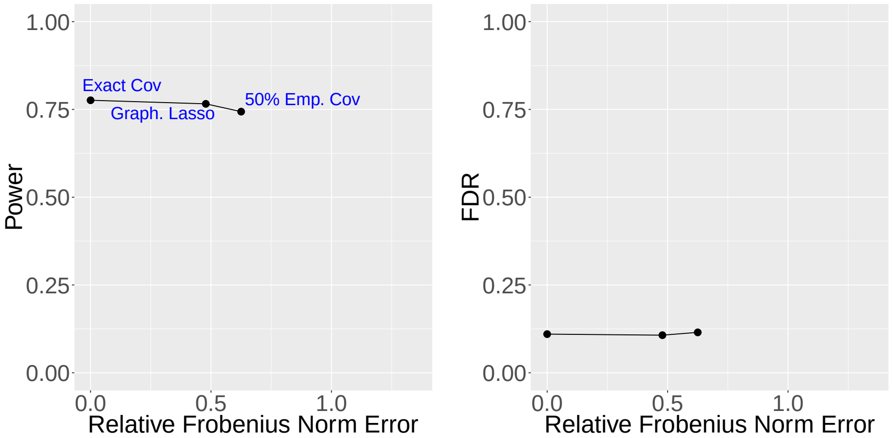


Figure: Covariates are **AR(1) with autocorrelation coefficient 0.3**. $n = 800$, $p = 1500$, and target FDR is 10%. Y comes from a binomial linear model with logit link function with 50 nonzero entries.

Robustness

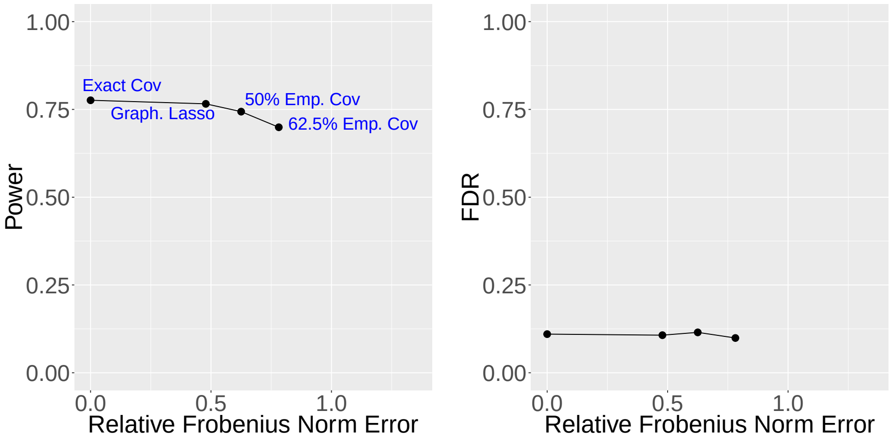


Figure: Covariates are **AR(1) with autocorrelation coefficient 0.3**. $n = 800$, $p = 1500$, and target FDR is 10%. Y comes from a binomial linear model with logit link function with 50 nonzero entries.

Robustness

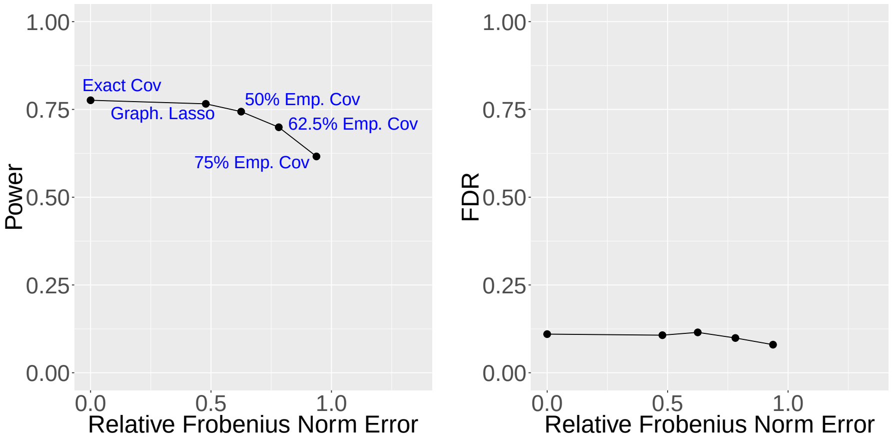


Figure: Covariates are **AR(1) with autocorrelation coefficient 0.3**. $n = 800$, $p = 1500$, and target FDR is 10%. Y comes from a binomial linear model with logit link function with 50 nonzero entries.

Robustness

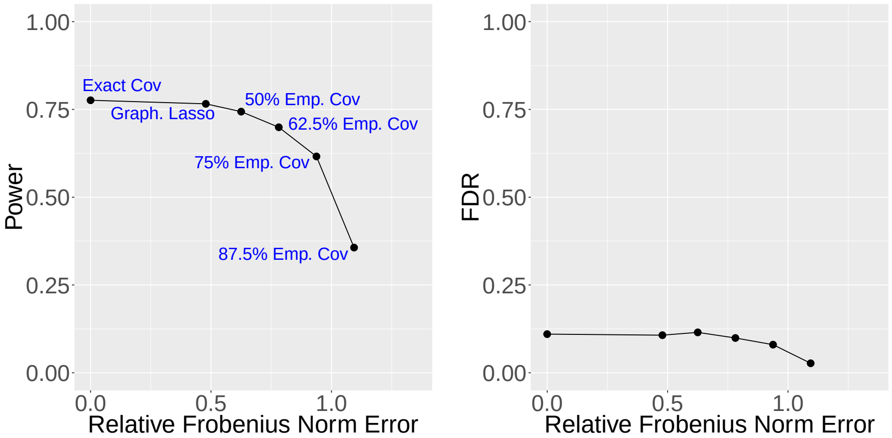


Figure: Covariates are **AR(1) with autocorrelation coefficient 0.3**. $n = 800$, $p = 1500$, and target FDR is 10%. Y comes from a binomial linear model with logit link function with 50 nonzero entries.

Robustness

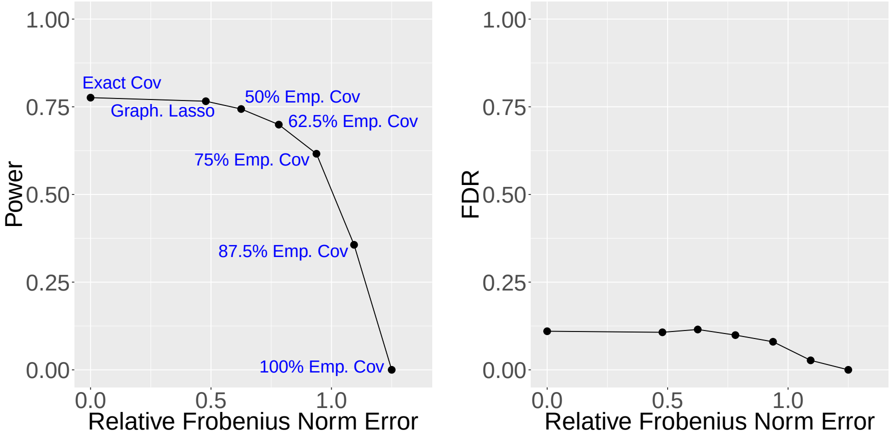


Figure: Covariates are **AR(1) with autocorrelation coefficient 0.3**. $n = 800$, $p = 1500$, and target FDR is 10%. Y comes from a binomial linear model with logit link function with 50 nonzero entries.

Variable Importance Statistics

Variable importance measures for all original and knockoff variables

$$Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$$

Variable Importance Statistics

Variable importance measures for all original and knockoff variables

$$Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$$

Examples:

- Magnitude of fitted coefficient β from a lasso regression of $\boldsymbol{y}$ on $[\boldsymbol{X}\,\widetilde{\boldsymbol{X}}\,]$

Variable Importance Statistics

Variable importance measures for all original and knockoff variables

$$Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$$

Examples:

- Magnitude of fitted coefficient β from a lasso regression of $\boldsymbol{y}$ on $[\boldsymbol{X}\, \widetilde{\boldsymbol{X}}]$
- CV error increase when variable dropped, using any machine learning method

Variable Importance Statistics

Variable importance measures for all original and knockoff variables

$$Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$$

Examples:

- Magnitude of fitted coefficient β from a lasso regression of $\boldsymbol{y}$ on $[\boldsymbol{X}\, \widetilde{\boldsymbol{X}}]$
- CV error increase when variable dropped, using any machine learning method

Adaptivity

- Higher-level adaptivity: CV to choose best-fitting model for inference

Variable Importance Statistics

Variable importance measures for all original and knockoff variables

$$Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$$

Examples:

- Magnitude of fitted coefficient β from a lasso regression of $\boldsymbol{y}$ on $[\boldsymbol{X}\,\widetilde{\boldsymbol{X}}]$
- CV error increase when variable dropped, using any machine learning method

Adaptivity

- Higher-level adaptivity: CV to choose best-fitting model for inference
  - E.g., fit random forest and ℓ_1-penalized regression; derive feature importance from whichever has lower CV error—**still strict FDR control**

Variable Importance Statistics

Variable importance measures for all original and knockoff variables

$$Z_1, \dots, Z_p, \widetilde{Z}_1, \dots, \widetilde{Z}_p$$

Examples:

- Magnitude of fitted coefficient β from a lasso regression of $\boldsymbol{y}$ on $[\boldsymbol{X}\,\widetilde{\boldsymbol{X}}]$
- CV error increase when variable dropped, using any machine learning method

Adaptivity

- Higher-level adaptivity: CV to choose best-fitting model for inference
  - E.g., fit random forest and ℓ_1-penalized regression; derive feature importance from whichever has lower CV error—**still strict FDR control**
- Can even let analyst look at (masked version of) data to choose Z function

Variable Importance Statistics

Variable importance measures for all original and knockoff variables

$$Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$$

Examples:

- Magnitude of fitted coefficient β from a lasso regression of $\boldsymbol{y}$ on $[\boldsymbol{X}\, \widetilde{\boldsymbol{X}}]$
- CV error increase when variable dropped, using any machine learning method

Adaptivity

- Higher-level adaptivity: CV to choose best-fitting model for inference
  - E.g., fit random forest and ℓ_1-penalized regression; derive feature importance from whichever has lower CV error—**still strict FDR control**
- Can even let analyst look at (masked version of) data to choose Z function

Prior information

- **Bayesian approach**: choose prior and model, and Z_j could be the posterior probability that X_j contributes to the model

Variable Importance Statistics

Variable importance measures for all original and knockoff variables

$$Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$$

Examples:

- Magnitude of fitted coefficient β from a lasso regression of $\boldsymbol{y}$ on $[\boldsymbol{X} \, \widetilde{\boldsymbol{X}}]$
- CV error increase when variable dropped, using any machine learning method

Adaptivity

- Higher-level adaptivity: CV to choose best-fitting model for inference
  - E.g., fit random forest and ℓ_1-penalized regression; derive feature importance from whichever has lower CV error—**still strict FDR control**
- Can even let analyst look at (masked version of) data to choose Z function

Prior information

- **Bayesian approach**: choose prior and model, and Z_j could be the posterior probability that X_j contributes to the model
- Still strict FDR control, **even if wrong prior or MCMC has not converged**

Tracking the FDR

Compute $W_1, \ldots, W_p$, where

$$W_j = Z_j - \widetilde{Z}_j$$

and select variables with W_j above a positive threshold $\hat{\tau}$

Tracking the FDR

Compute $W_1, \ldots, W_p$, where

$$W_j = Z_j - \widetilde{Z}_j$$

and select variables with W_j above a positive threshold $\hat{\tau}$

$$\begin{aligned} \text{FDR} &= \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] \\ &= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \end{aligned}$$

Tracking the FDR

Compute $W_1, \ldots, W_p$, where

$$W_j = Z_j - \widetilde{Z}_j$$

and select variables with W_j above a positive threshold $\hat{\tau}$

$$\begin{aligned} \text{FDR} &= \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] \\ &= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\ &\approx \mathbb{E}\left[\frac{\#\{\text{null negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \end{aligned}$$

Tracking the FDR

Compute $W_1, \ldots, W_p$, where

$$W_j = Z_j - \widetilde{Z}_j$$

and select variables with W_j above a positive threshold $\hat{\tau}$

$$\begin{aligned}
\text{FDR} &= \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] \\
&= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&\approx \mathbb{E}\left[\frac{\#\{\text{null negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&\leq \mathbb{E}\left[\frac{\#\{\text{negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]
\end{aligned}$$

Tracking the FDR

Compute $W_1, \ldots, W_p$, where

$$W_j = Z_j - \widetilde{Z}_j$$

and select variables with W_j above a positive threshold $\hat{\tau}$

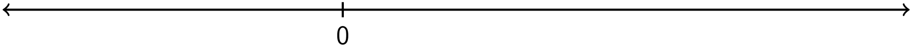
$$\begin{aligned}
\text{FDR} &= \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] \\
&= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&\approx \mathbb{E}\left[\frac{\#\{\text{null negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&\leq \mathbb{E}\left[\frac{\#\{\text{negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&= \mathbb{E}\left[\widehat{\text{FDR}}\right]
\end{aligned}$$

Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:

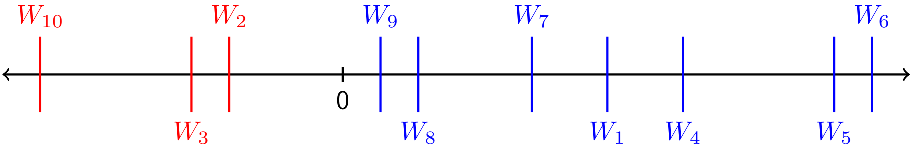
Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:



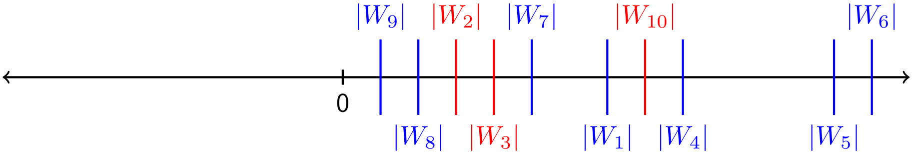
Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:



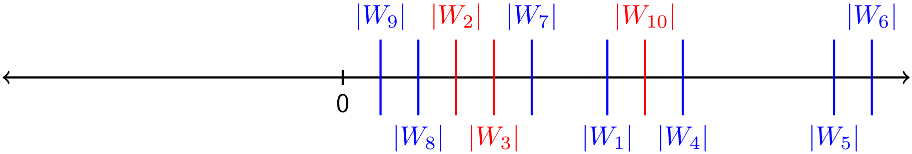
Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:



Selecting Variables

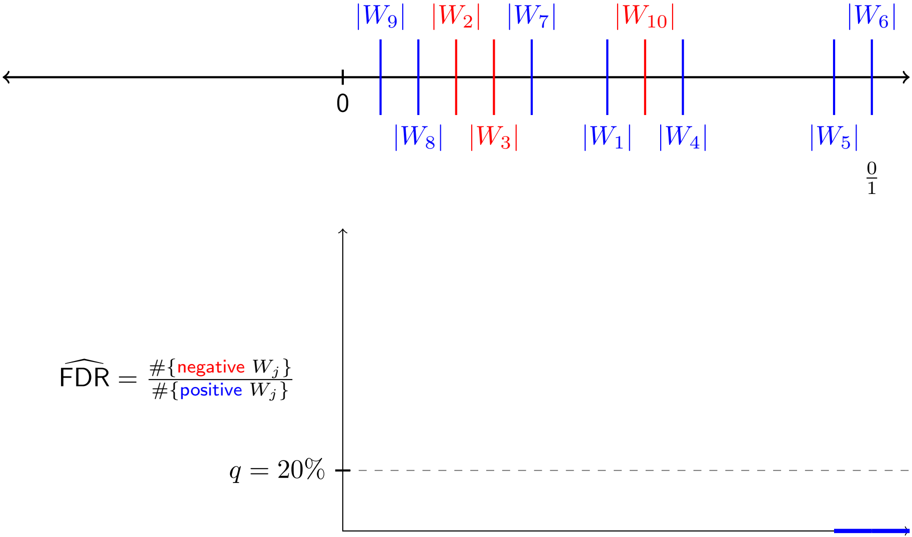
Example with $p = 10$ and $q = 20\% = 1/5$:



$$\widehat{\mathsf{FDR}} = \frac{\#\{\text{negative } W_j\}}{\#\{\text{positive } W_j\}}$$

Selecting Variables

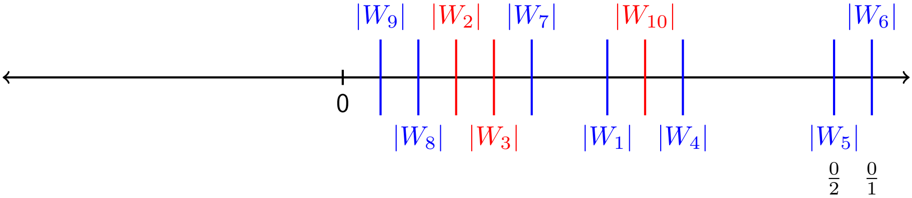
Example with $p = 10$ and $q = 20\% = 1/5$:



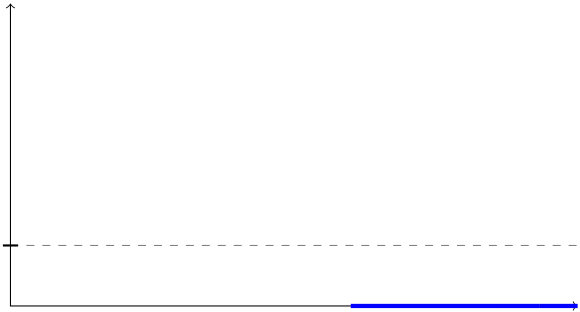
$$\widehat{\mathsf{FDR}} = \frac{\#\{\text{negative } W_j\}}{\#\{\text{positive } W_j\}}$$

Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:

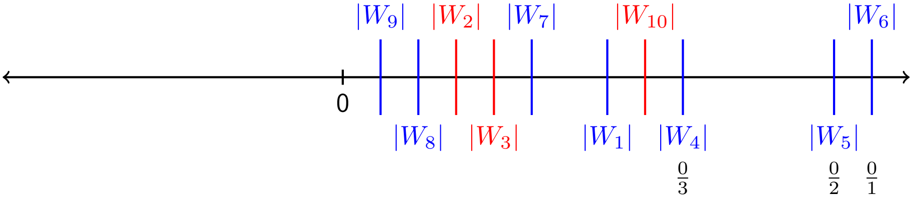


$\widehat{\text{FDR}} = \frac{\#\{\text{negative } W_j\}}{\#\{\text{positive } W_j\}}$

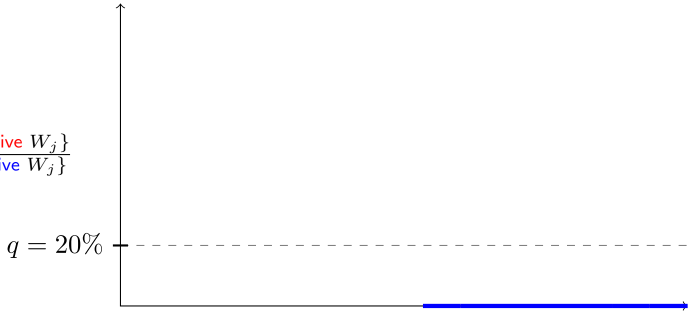


Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:

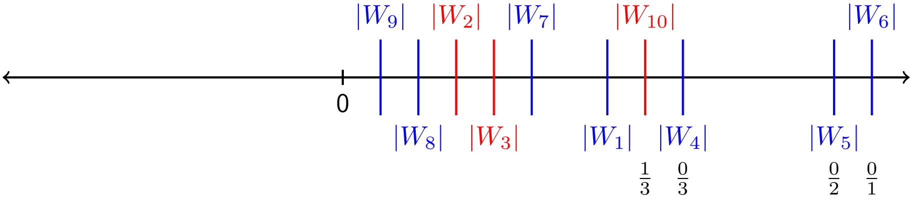


$$\widehat{\text{FDR}} = \frac{\#\{\text{negative } W_j\}}{\#\{\text{positive } W_j\}}$$

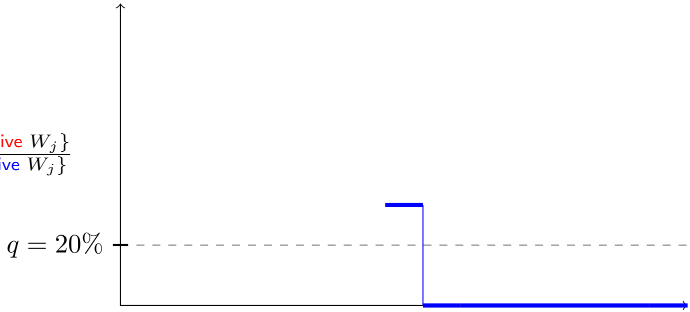


Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:

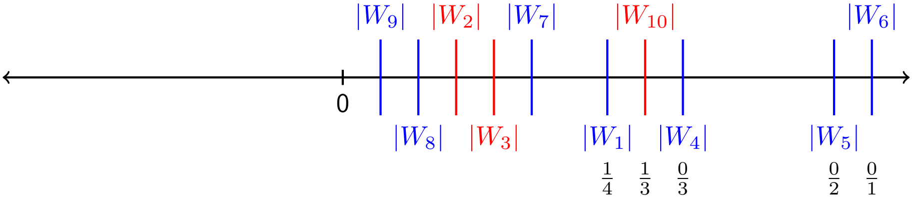


$$\widehat{\text{FDR}} = \frac{\#\{\text{negative } W_j\}}{\#\{\text{positive } W_j\}}$$



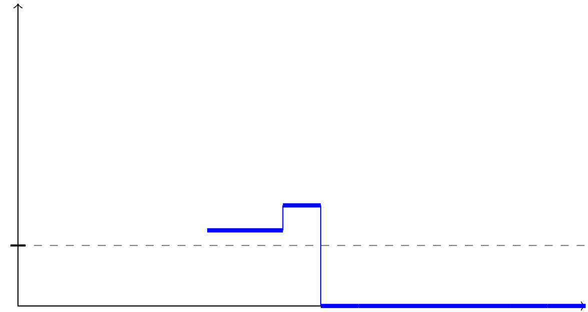
Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:



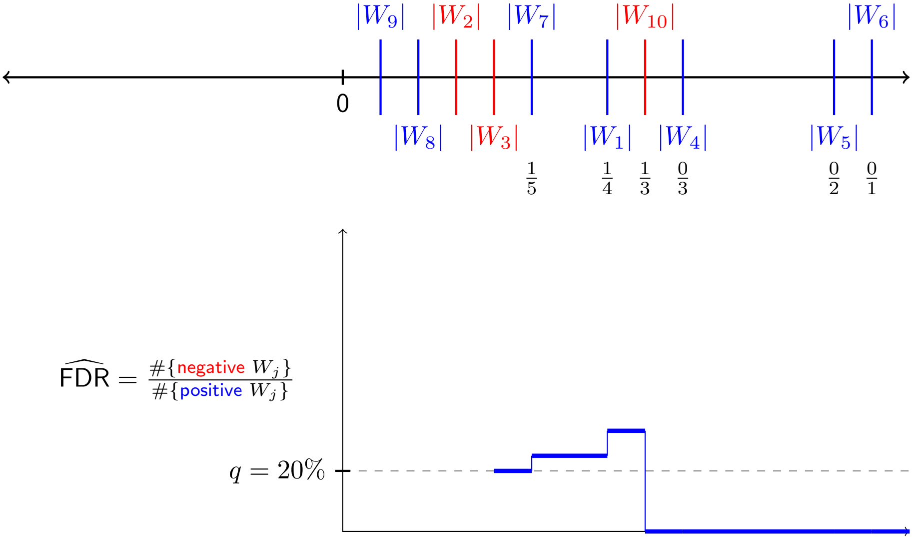
$$\widehat{\text{FDR}} = \frac{\#\{\text{negative } W_j\}}{\#\{\text{positive } W_j\}}$$

$q = 20\%$



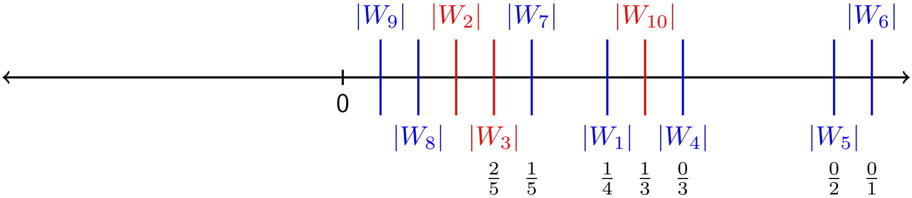
Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:

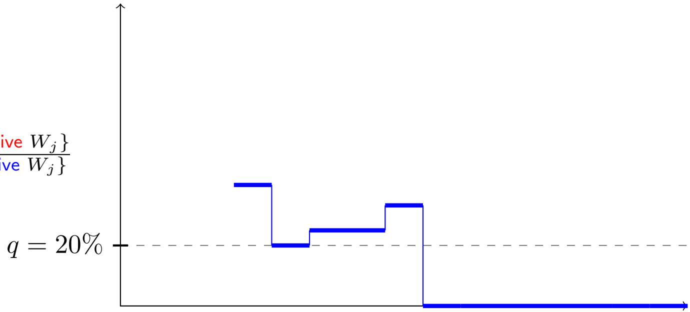


Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:

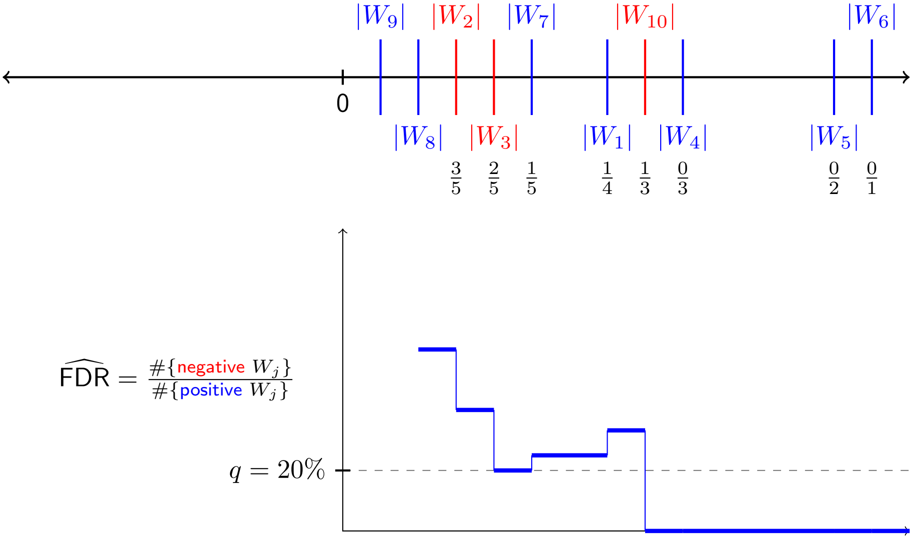


$$\widehat{\text{FDR}} = \frac{\#\{\text{negative } W_j\}}{\#\{\text{positive } W_j\}}$$



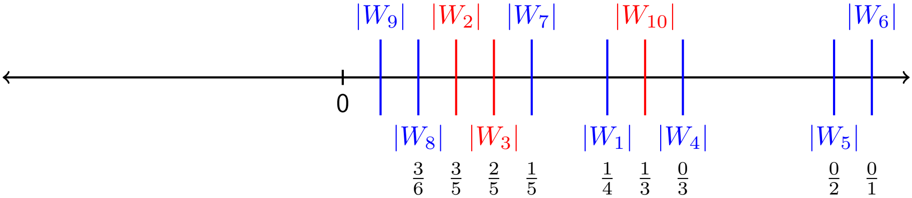
Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:

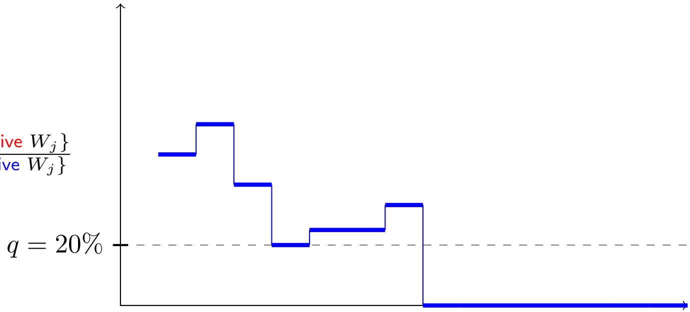


Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:

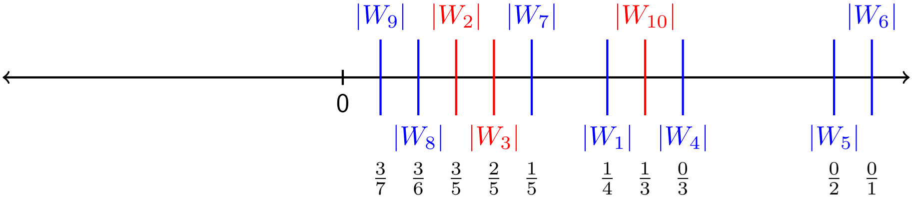


$$\widehat{\text{FDR}} = \frac{\#\{\text{negative } W_j\}}{\#\{\text{positive } W_j\}}$$

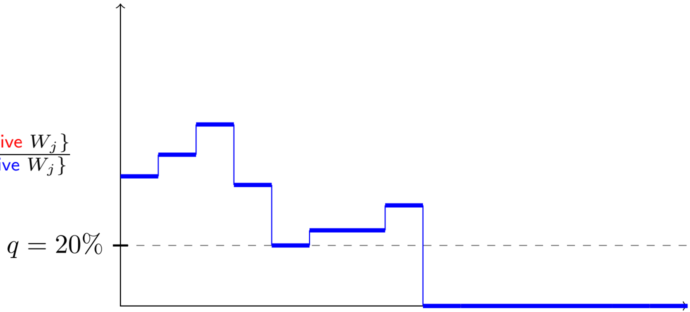


Selecting Variables

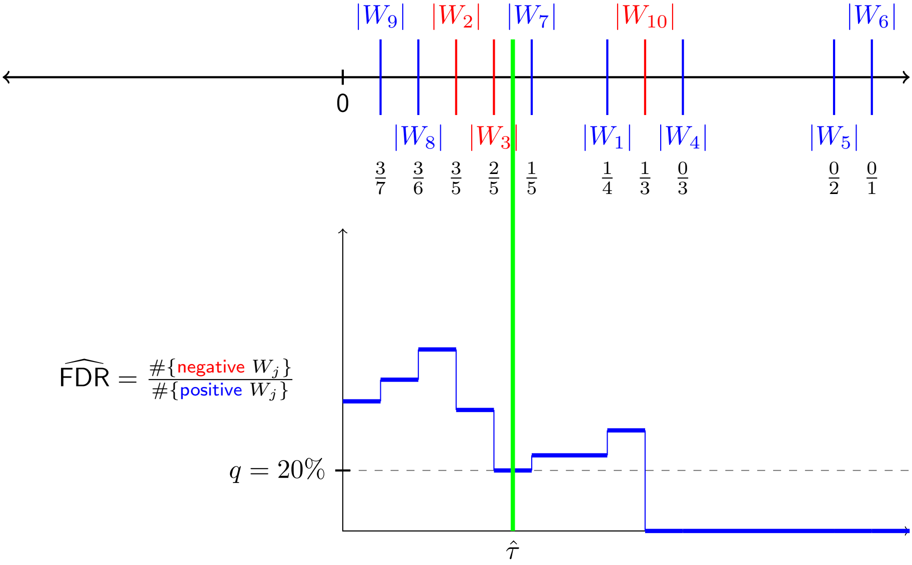
Example with $p = 10$ and $q = 20\% = 1/5$:



$\widehat{\text{FDR}} = \frac{\#\{\text{negative } W_j\}}{\#\{\text{positive } W_j\}}$

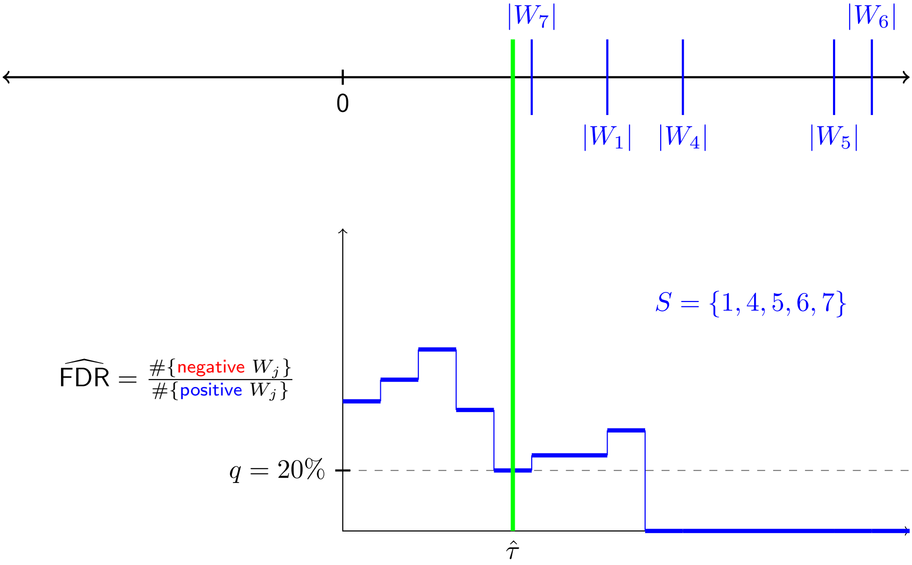


Example with $p = 10$ and $q = 20\% = 1/5$:



Selecting Variables

Example with $p = 10$ and $q = 20\% = 1/5$:



Simulations in Low-Dimensional Linear Model

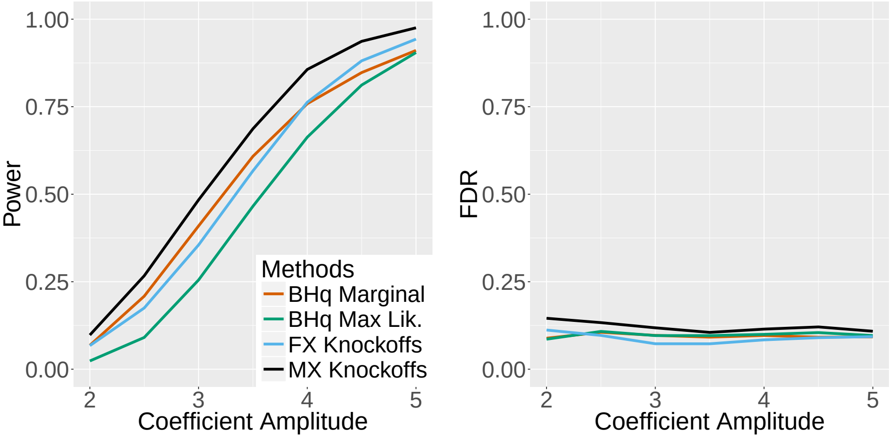


Figure: Power and FDR (target is 10%) for knockoffs and alternative procedures. The design matrix is i.i.d. $\mathcal{N}(0, 1/n)$, $n = 3000$, $p = 1000$, and y comes from a Gaussian linear model with 60 nonzero regression coefficients having equal magnitudes and random signs. The noise variance is 1.

Computation and Software

- R, Python, and Matlab packages available depending on knockoff construction; link on my website

Computation and Software

- R, Python, and Matlab packages available depending on knockoff construction; link on my website
- Knockoff construction algorithms generally scale linearly in p and n

Computation and Software

- R, Python, and Matlab packages available depending on knockoff construction; link on my website
- Knockoff construction algorithms generally scale linearly in p and n
- Given variable importances $Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$, computation trivial

Computation and Software

- R, Python, and Matlab packages available depending on knockoff construction; link on my website
- Knockoff construction algorithms generally scale linearly in p and n
- Given variable importances $Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$, computation trivial
- Need to compute $Z_1, \ldots, Z_p, \widetilde{Z}_1, \ldots, \widetilde{Z}_p$
  - Just compute variable importances for twice as many variables
  - Generally only constant times slower than computing variable importances without knockoffs

Metropolized Knockoff Sampling
(Bates, Candès, **J.**, Wang, arXiv, 2019)

Metropolized Knockoff Sampling

S. Bates, E. Candès, L. Janson, and W. Wang. **Metropolized Knockoff Sampling**. 2019. [https://arxiv.org/abs/1903.00434]

Metropolized Knockoff Sampling

S. Bates, E. Candès, L. Janson, and W. Wang. **Metropolized Knockoff Sampling**. 2019. [`https://arxiv.org/abs/1903.00434`]

Solves computational problem of sampling knockoffs for any X distribution

- Reframes knockoff sampling problem in terms of reversible Markov chains

Metropolized Knockoff Sampling

S. Bates, E. Candès, L. Janson, and W. Wang. **Metropolized Knockoff Sampling**. 2019. [https://arxiv.org/abs/1903.00434]

Solves computational problem of sampling knockoffs for any X distribution

- Reframes knockoff sampling problem in terms of reversible Markov chains
- Enables huge body of tools from MCMC to be used for the problem

Metropolized Knockoff Sampling

S. Bates, E. Candès, L. Janson, and W. Wang. **Metropolized Knockoff Sampling**. 2019. [https://arxiv.org/abs/1903.00434]

Solves computational problem of sampling knockoffs for any X distribution

- Reframes knockoff sampling problem in terms of reversible Markov chains
- Enables huge body of tools from MCMC to be used for the problem
- Yet, unlike MCMC, Metropolized knockoff sampling is **exact**!

Sequential Knockoff Sampling

We introduce a flexible way to generate knockoffs called **Sequential Conditional Exchangeable Pairs (SCEP)**:

For $j = 1, \ldots, p$

- Condition on everything except X_j so far: $X_{1:(j-1)}$, $X_{(j+1):p}$, $\widetilde{X}_{1:(j-1)}$

Sequential Knockoff Sampling

We introduce a flexible way to generate knockoffs called **Sequential Conditional Exchangeable Pairs (SCEP)**:

For $j = 1, \ldots, p$

- Condition on everything except X_j so far: $X_{1:(j-1)}$, $X_{(j+1):p}$, $\widetilde{X}_{1:(j-1)}$
- Generate $\widetilde{X}_j$ **conditionally-exchangeably** with X_j

Sequential Knockoff Sampling

We introduce a flexible way to generate knockoffs called **Sequential Conditional Exchangeable Pairs (SCEP)**:

For $j = 1, \ldots, p$

- Condition on everything except X_j so far: $X_{1:(j-1)}$, $X_{(j+1):p}$, $\widetilde{X}_{1:(j-1)}$
- Generate $\widetilde{X}_j$ **conditionally-exchangeably** with X_j
- Make sure that $(\widetilde{X}_j, X_j)$'s distribution is **invariant to swapping** previously-sampled knockoff pairs

Sequential Knockoff Sampling

We introduce a flexible way to generate knockoffs called **Sequential Conditional Exchangeable Pairs (SCEP)**:

For $j = 1, \dots, p$

- Condition on everything except X_j so far: $X_{1:(j-1)}$, $X_{(j+1):p}$, $\widetilde{X}_{1:(j-1)}$
- Generate $\widetilde{X}_j$ **conditionally-exchangeably** with X_j
- Make sure that $(\widetilde{X}_j, X_j)$'s distribution is **invariant to swapping** previously-sampled knockoff pairs

This is **completely general**: all knockoff distributions are a special case

Sequential Knockoff Sampling

We introduce a flexible way to generate knockoffs called **Sequential Conditional Exchangeable Pairs (SCEP)**:

For $j = 1, \ldots, p$

- Condition on everything except X_j so far: $X_{1:(j-1)}$, $X_{(j+1):p}$, $\widetilde{X}_{1:(j-1)}$
- Generate $\widetilde{X}_j$ **conditionally-exchangeably** with X_j
- Make sure that $(\widetilde{X}_j, X_j)$'s distribution is **invariant to swapping** previously-sampled knockoff pairs

This is **completely general**: all knockoff distributions are a special case

Can think of $\widetilde{X}_j$ being one step from X_j in a reversible Markov chain with stationary distribution given by X_j's (conditional) distribution

Using Tools from Markov Chain Monte Carlo

The reversible Markov chain formulation of knockoff sampling allows us to draw from MCMC literature, e.g., Metropolis–Hastings

Using Tools from Markov Chain Monte Carlo

The reversible Markov chain formulation of knockoff sampling allows us to draw from MCMC literature, e.g., Metropolis–Hastings

Metropolized knockoff sampling (Metro):
For $j = 1, \ldots, p$

- Sample $X_j^* = x_j^*$ from a faithful, symmetric proposal distribution q_j
- Accept the proposal with probability

$$\min\left(1, \frac{\mathbb{P}\left(X_j = x_j^*, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}{\mathbb{P}\left(X_j = x_j, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}\right)$$

- Upon acceptance, set $\tilde{X}_j = X_j^*$; otherwise, set $\tilde{X}_j = X_j$

Computational Complexity

Any completely general knockoff sampler has time complexity at least 2^p

Computational Complexity

Any completely general knockoff sampler has time complexity at least 2^p

Indeed the ratio

$$\frac{\mathbb{P}\left(X_j = x_j^*, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}{\mathbb{P}\left(X_j = x_j, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}$$

in Metro will in general be hard to compute

Computational Complexity

Any completely general knockoff sampler has time complexity at least 2^p

Indeed the ratio

$$\frac{\mathbb{P}\left(X_j = x_j^*, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}{\mathbb{P}\left(X_j = x_j, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}$$

in Metro will in general be hard to compute

X's distribution often has conditional independence / graphical model structure

Computational Complexity

Any completely general knockoff sampler has time complexity at least 2^p

Indeed the ratio

$$\frac{\mathbb{P}\left(X_j = x_j^*, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}{\mathbb{P}\left(X_j = x_j, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}$$

in Metro will in general be hard to compute

X's distribution often has conditional independence / graphical model structure

Metro's complexity only exponential in the width of a **junction tree** for the graph; we show this is optimal in some cases

Computational Complexity

Any completely general knockoff sampler has time complexity at least 2^p

Indeed the ratio

$$\frac{\mathbb{P}\left(X_j = x_j^*, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}{\mathbb{P}\left(X_j = x_j, X_{\text{-}j} = x_{\text{-}j}, \tilde{X}_{1:(j-1)} = \tilde{x}_{1:(j-1)}, X^*_{1:(j-1)} = x^*_{1:(j-1)}\right)}$$

in Metro will in general be hard to compute

X's distribution often has conditional independence / graphical model structure

Metro's complexity only exponential in the width of a **junction tree** for the graph; we show this is optimal in some cases

Enables sampling in, e.g.,

- Continuous graphical models (e.g., Markov chains) that can have skewness or heavy tails
- Discrete graphical models with any number of states, e.g., Ising models or, more generally, Gibbs measures

Conditional Knockoffs
(Huang and **J.**, arXiv, 2019)

Relaxing the Assumptions of Knockoffs by Conditioning

D. Huang and L. Janson. **Relaxing the Assumptions of Knockoffs by Conditioning**. 2019. [https://arxiv.org/abs/1903.02806]

Relaxing the Assumptions of Knockoffs by Conditioning

D. Huang and L. Janson. **Relaxing the Assumptions of Knockoffs by Conditioning**. 2019. [https://arxiv.org/abs/1903.02806]

Removes assumption that X's distribution known

- Allows X's distribution to be known only up to a model

Relaxing the Assumptions of Knockoffs by Conditioning

D. Huang and L. Janson. **Relaxing the Assumptions of Knockoffs by Conditioning**. 2019. [https://arxiv.org/abs/1903.02806]

Removes assumption that X's distribution known

- Allows X's distribution to be known only up to a model
- Model can have $O(n^*p)$ free parameters, where n^* is the total number of covariate samples, labeled and unlabeled

Relaxing the Assumptions of Knockoffs by Conditioning

D. Huang and L. Janson. **Relaxing the Assumptions of Knockoffs by Conditioning**. 2019. [https://arxiv.org/abs/1903.02806]

Removes assumption that X's distribution known

- Allows X's distribution to be known only up to a model
- Model can have $O(n^*p)$ free parameters, where n^* is the total number of covariate samples, labeled and unlabeled
- Retains exact same error control guarantees as model-X knockoffs, and barely any power loss in simulations

Relaxing the Assumptions of Knockoffs by Conditioning

D. Huang and L. Janson. **Relaxing the Assumptions of Knockoffs by Conditioning**. 2019. [https://arxiv.org/abs/1903.02806]

Removes assumption that X's distribution known

- Allows X's distribution to be known only up to a model
- Model can have $O(n^*p)$ free parameters, where n^* is the total number of covariate samples, labeled and unlabeled
- Retains exact same error control guarantees as model-X knockoffs, and barely any power loss in simulations
- Note $O(n^*p)$ parameters is far more than allowed in fixed-X inference, which is typically $o(n)$

Conditional Knockoffs

Recall definition of valid knockoffs: for any j,

$$[\boldsymbol{X}, \widetilde{\boldsymbol{X}}]_{\mathsf{swap(j)}} \stackrel{\mathcal{D}}{=} [\boldsymbol{X}, \widetilde{\boldsymbol{X}}]$$

Conditional Knockoffs

Recall definition of valid knockoffs: for any j,

$$[\boldsymbol{X}, \widetilde{\boldsymbol{X}}]_{\mathsf{swap(j)}} \overset{\mathcal{D}}{=} [\boldsymbol{X}, \widetilde{\boldsymbol{X}}]$$

Note by law of total probability, a sufficient condition is that for any j,

$$[\boldsymbol{X}, \widetilde{\boldsymbol{X}}]_{\mathsf{swap(j)}} \overset{\mathcal{D}}{=} [\boldsymbol{X}, \widetilde{\boldsymbol{X}}] \;\Big|\; T(\boldsymbol{X})$$

for some statistic $T(\boldsymbol{X})$

Conditional Knockoffs

Recall definition of valid knockoffs: for any j,

$$[\boldsymbol{X}, \widetilde{\boldsymbol{X}}]_{\mathsf{swap(j)}} \stackrel{\mathcal{D}}{=} [\boldsymbol{X}, \widetilde{\boldsymbol{X}}]$$

Note by law of total probability, a sufficient condition is that for any j,

$$[\boldsymbol{X}, \widetilde{\boldsymbol{X}}]_{\mathsf{swap(j)}} \stackrel{\mathcal{D}}{=} [\boldsymbol{X}, \widetilde{\boldsymbol{X}}] \;\Big|\; T(\boldsymbol{X})$$

for some statistic $T(\boldsymbol{X})$

Now suppose $\boldsymbol{X}$'s rows are i.i.d. from a model with **sufficient statistic** $T(\boldsymbol{X})$

- E.g., if $X \sim \mathcal{N}(\mu, \boldsymbol{\Sigma})$, then $(\hat{\mu}, \hat{\boldsymbol{\Sigma}})$ are sufficient

Conditional Knockoffs

Recall definition of valid knockoffs: for any j,

$$[\boldsymbol{X}, \widetilde{\boldsymbol{X}}]_{\mathsf{swap(j)}} \stackrel{\mathcal{D}}{=} [\boldsymbol{X}, \widetilde{\boldsymbol{X}}]$$

Note by law of total probability, a sufficient condition is that for any j,

$$[\boldsymbol{X}, \widetilde{\boldsymbol{X}}]_{\mathsf{swap(j)}} \stackrel{\mathcal{D}}{=} [\boldsymbol{X}, \widetilde{\boldsymbol{X}}] \,\Big|\, T(\boldsymbol{X})$$

for some statistic $T(\boldsymbol{X})$

Now suppose $\boldsymbol{X}$'s rows are i.i.d. from a model with **sufficient statistic** $T(\boldsymbol{X})$

- E.g., if $X \sim \mathcal{N}(\mu, \boldsymbol{\Sigma})$, then $(\hat{\mu}, \hat{\boldsymbol{\Sigma}})$ are sufficient

Then by sufficiency, the distribution $\boldsymbol{X} \mid T(X)$ is model-parameter-free

Conditional Knockoffs

Recall definition of valid knockoffs: for any j,

$$[\boldsymbol{X}, \widetilde{\boldsymbol{X}}]_{\mathsf{swap(j)}} \stackrel{\mathcal{D}}{=} [\boldsymbol{X}, \widetilde{\boldsymbol{X}}]$$

Note by law of total probability, a sufficient condition is that for any j,

$$[\boldsymbol{X}, \widetilde{\boldsymbol{X}}]_{\mathsf{swap(j)}} \stackrel{\mathcal{D}}{=} [\boldsymbol{X}, \widetilde{\boldsymbol{X}}] \;\Big|\; T(\boldsymbol{X})$$

for some statistic $T(\boldsymbol{X})$

Now suppose $\boldsymbol{X}$'s rows are i.i.d. from a model with **sufficient statistic** $T(\boldsymbol{X})$

- E.g., if $X \sim \mathcal{N}(\mu, \boldsymbol{\Sigma})$, then $(\hat{\mu}, \hat{\boldsymbol{\Sigma}})$ are sufficient

Then by sufficiency, the distribution $\boldsymbol{X} \mid T(X)$ is model-parameter-free

Sample knockoffs as when $\boldsymbol{X}$'s distribution known, but **valid for any distribution in a model**

Example Models

- **Low-dimensional arbitrary Gaussian model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p \times p},\ \boldsymbol{\Sigma} \succ \mathbf{0}\right\},$$

when $n > 2p$

Example Models

- **Low-dimensional arbitrary Gaussian model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p \times p},\ \boldsymbol{\Sigma} \succ \mathbf{0}\right\},$$

when $n > 2p$ [can have $p = \Omega(n)$, number of parameters is $\Omega(np)$]

Example Models

- **Low-dimensional arbitrary Gaussian model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p \times p},\ \boldsymbol{\Sigma} \succ \mathbf{0}\right\},$$

when $n > 2p$ [can have $p = \Omega(n)$, number of parameters is $\Omega(np)$]

- **Gaussian graphical model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p \times p},\ \boldsymbol{\Sigma} \succ \mathbf{0},\ \left(\boldsymbol{\Sigma}^{-1}\right)_{j,k} = 0 \text{ for all } (j,k) \notin E\right\}$$

for some known sparsity pattern E

Example Models

- **Low-dimensional arbitrary Gaussian model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p\times p},\ \boldsymbol{\Sigma} \succ \mathbf{0}\right\},$$

  when $n > 2p$ [can have $p = \Omega(n)$, number of parameters is $\Omega(np)$]

- **Gaussian graphical model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p\times p},\ \boldsymbol{\Sigma} \succ \mathbf{0},\ \left(\boldsymbol{\Sigma}^{-1}\right)_{j,k} = 0 \text{ for all } (j,k) \notin E\right\}$$

  for some known sparsity pattern E [$\boldsymbol{\Sigma}^{-1}$ can be banded with bandwidth $\Omega(n)$, number of parameters is $\Omega(np)$]

Example Models

- **Low-dimensional arbitrary Gaussian model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p \times p},\ \boldsymbol{\Sigma} \succ \mathbf{0}\right\},$$

when $n > 2p$ [can have $p = \Omega(n)$, number of parameters is $\Omega(np)$]

- **Gaussian graphical model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p \times p},\ \boldsymbol{\Sigma} \succ \mathbf{0},\ \left(\boldsymbol{\Sigma}^{-1}\right)_{j,k} = 0 \text{ for all } (j,k) \notin E\right\}$$

for some known sparsity pattern E [$\boldsymbol{\Sigma}^{-1}$ can be banded with bandwidth $\Omega(n)$, number of parameters is $\Omega(np)$]

- **Discrete graphical model**:

$$\left\{\text{distribution on } \prod_{j=1}^{p}[K_j] : X_j \perp\!\!\!\perp X_k \mid X_{[p]\setminus\{j,k\}} \text{ for all } (j,k) \notin E\right\}$$

for some known positive integers K_j and known sparsity pattern E

Example Models

- **Low-dimensional arbitrary Gaussian model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p\times p},\ \boldsymbol{\Sigma} \succ \mathbf{0}\right\},$$

when $n > 2p$ [can have $p = \Omega(n)$, number of parameters is $\Omega(np)$]

- **Gaussian graphical model**:

$$\left\{\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \boldsymbol{\mu} \in \mathbb{R}^p,\ \boldsymbol{\Sigma} \in \mathbb{R}^{p\times p},\ \boldsymbol{\Sigma} \succ \mathbf{0},\ \left(\boldsymbol{\Sigma}^{-1}\right)_{j,k} = 0 \text{ for all } (j,k) \notin E\right\}$$

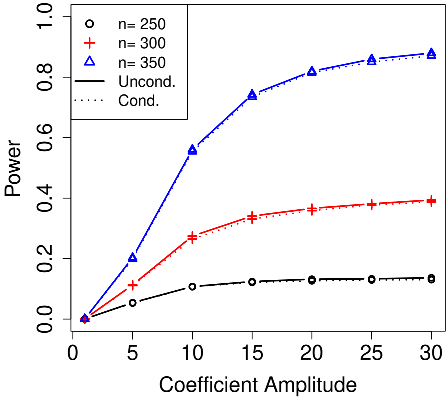
for some known sparsity pattern E [$\boldsymbol{\Sigma}^{-1}$ can be banded with bandwidth $\Omega(n)$, number of parameters is $\Omega(np)$]

- **Discrete graphical model**:

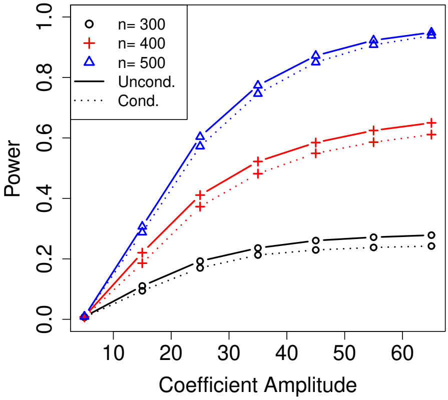
$$\left\{\text{distribution on } \prod_{j=1}^{p}[K_j] : X_j \perp\!\!\!\perp X_k \mid X_{[p]\setminus\{j,k\}} \text{ for all } (j,k) \notin E\right\}$$

for some known positive integers K_j and known sparsity pattern E [X can be $\Omega(\sqrt{n})$-state Markov chain, number of parameters is $\Omega(np)$]

Simulations in Low-Dimensional Linear Model



(a)



(b)

Figure: (a) is time-varying AR(1) with $p = 2000$ totaling 5,999 parameters in model, (b) is time-varying AR(10) with $p = 2000$ totaling 23,945 parameters in model

Takeaways

Can run knockoffs when $Y \mid X$ is completely unknown and X's distribution is only known up to a model with $\Omega(np)$ parameters

Takeaways

Can run knockoffs when $Y \mid X$ is completely unknown and X's distribution is only known up to a model with $\Omega(np)$ parameters

- Compare to results for asymptotic p-values with penalized GLMs: X's distribution unknown and $Y \mid X$ known up to model with $o(n)$ parameters

Takeaways

Can run knockoffs when $Y \mid X$ is completely unknown and X's distribution is only known up to a model with $\Omega(np)$ parameters

- Compare to results for asymptotic p-values with penalized GLMs: X's distribution unknown and $Y \mid X$ known up to model with $o(n)$ parameters

Can actually replace n with n^*, which includes **unlabeled samples** of X

Takeaways

Can run knockoffs when $Y \mid X$ is completely unknown and X's distribution is only known up to a model with $\Omega(np)$ parameters

- Compare to results for asymptotic p-values with penalized GLMs: X's distribution unknown and $Y \mid X$ known up to model with $o(n)$ parameters

Can actually replace n with n^*, which includes **unlabeled samples** of X

By conditioning on $T(\boldsymbol{X})$, sampling and exchangeability hold on measure-zero manifold of $\mathbb{R}^{2p}$

- We use **topological measure theory** to prove our results

Summary

Model-X knockoffs is a powerful and flexible tool for high-dimensional controlled variable selection

Summary

Model-X knockoffs is a powerful and flexible tool for high-dimensional controlled variable selection

Beyond knockoffs, I am interested in all types of high-dimensional inference—**please reach out** if you think this work or something like it could help with work you're doing!

http://lucasjanson.fas.harvard.edu
ljanson@fas.harvard.edu

Summary

Model-X knockoffs is a powerful and flexible tool for high-dimensional controlled variable selection

Beyond knockoffs, I am interested in all types of high-dimensional inference—**please reach out** if you think this work or something like it could help with work you're doing!

`http://lucasjanson.fas.harvard.edu`
ljanson@fas.harvard.edu

Thank you!

Appendix

References

Bates, S., Candès, E. J., Janson, L., and Wang, W. (2019). Metropolized knockoff sampling. *arXiv preprint arXiv:1903.00434*.

Candès, E., Fan, Y., Janson, L., and Lv, J. (2018). Panning for gold: 'model-X' knockoffs for high dimensional controlled variable selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(3):551–577.

Dai, R. and Barber, R. F. (2016). The knockoff filter for FDR control in group-sparse and multitask regression. In *Proceedings of the 33nd International Conference on Machine Learning (ICML 2016)*.

Gimenez, J. R., Ghorbani, A., and Zou, J. (2018). Knockoffs for the mass: new feature importance statistics with false discovery guarantees. *arXiv preprint arXiv:1807.06214*.

Huang, D. and Janson, L. (2019). Relaxing the assumptions of knockoffs by conditioning. *arXiv preprint arXiv:1903.02806*.

Sesia, M., Sabatti, C., and Candès, E. J. (2019). Gene hunting with hidden Markov model knockoffs. *Biometrika*, 106(1):1–18.

Existing Methods: Low-Dimensional Linear Model

Suppose we assume that our data:

- follows a **linear model**:

$$Y = X_1\beta_1 + \cdots + X_p\beta_p + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2),$$

- has more observations that variables: $n \geq p$ (**low-dimensional**).

Existing Methods: Low-Dimensional Linear Model

Suppose we assume that our data:

- follows a **linear model**:

$$Y = X_1\beta_1 + \cdots + X_p\beta_p + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2),$$

- has more observations that variables: $n \geq p$ (**low-dimensional**).

Classical problem:

- Ordinary least squares (OLS) theory gives exact p-values for testing whether each $\beta_j = 0$ or not (under very mild assumptions, $\beta_j = 0 \;\Leftrightarrow\; Y \perp\!\!\!\perp X_j \,|\, X_{\text{-}j}$)
- The Benjamini-Hochberg procedure (BHq) applied to the p-values will essentially control the FDR

Existing Methods: Low-Dimensional Linear Model

Suppose we assume that our data:

- follows a **linear model**:

$$Y = X_1\beta_1 + \cdots + X_p\beta_p + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2),$$

- has more observations that variables: $n \geq p$ (**low-dimensional**).

Classical problem:

- Ordinary least squares (OLS) theory gives exact p-values for testing whether each $\beta_j = 0$ or not (under very mild assumptions, $\beta_j = 0 \Leftrightarrow Y \perp\!\!\!\perp X_j \mid X_{\text{-}j}$)
- The Benjamini-Hochberg procedure (BHq) applied to the p-values will essentially control the FDR

Minor caveats:

- FDR control not exact (but good enough in practice)
- Sparsity not used (reduces power to find important variables)

Nonlinearity and High Dimensions

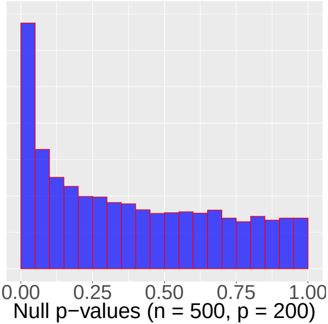
Low-dimensional ($n \geq p$) generalized linear model

- Apply BHq to asymptotic p-values

Nonlinearity and High Dimensions

Low-dimensional ($n \geq p$) generalized linear model

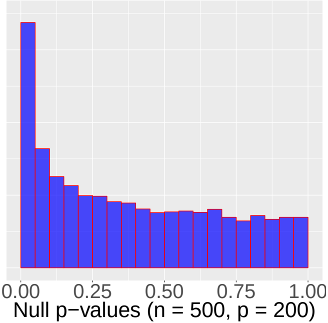
- Apply BHq to asymptotic p-values
- Can be far from valid in practice



Nonlinearity and High Dimensions

Low-dimensional ($n \geq p$) generalized linear model

- Apply BHq to asymptotic p-values
- Can be far from valid in practice



High-dimensional ($n < p$) generalized linear models

- Apply BHq to p-values from
  - Debiased lasso, e.g., Zhang and Zhang (2014), Javanmard and Montanari (2014), van de Geer et al. (2014), Cai and Guo (2015)
  - Causal inference, e.g., Belloni et al. (2014), Athey et al. (2016), Farrell (2015)
  - Inference after selection, e.g., Berk et al. (2013), Lee et al. (2016), Fithian et al. (2014)
- Asymptotic, require sparsity and random design assumptions

Why all the Fuss?

Why all the Fuss?

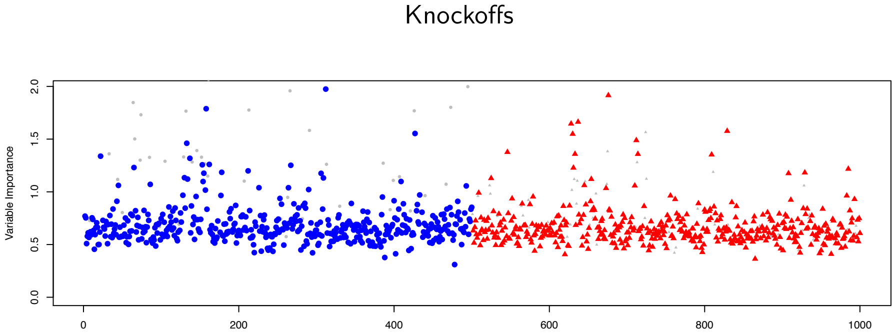


Figure: Variable importance measures for 500 variables and their knockoffs. Colored points are nulls, grey are non-nulls.

Why all the Fuss?

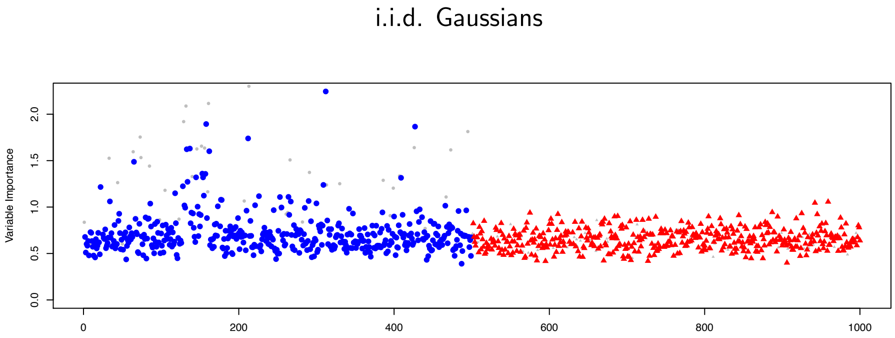


Figure: Variable importance measures for 500 variables and their knockoffs. Colored points are nulls, grey are non-nulls.

Why all the Fuss?

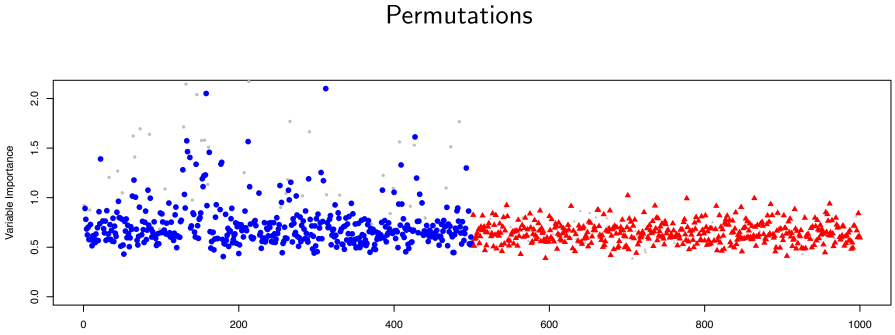


Figure: Variable importance measures for 500 variables and their knockoffs. Colored points are nulls, grey are non-nulls.

Sequential Independent Pairs Generates Valid Knockoffs

Algorithm 1 Sequential Conditional Independent Pairs

for $j = \{1, \ldots, p\}$ **do**

  Sample $\tilde{X}_j$ from $\mathcal{L}(X_j \,|\, X_{\text{-}j}, \tilde{X}_{1:j-1})$ conditionally independently of X_j

end

Sequential Independent Pairs Generates Valid Knockoffs

Algorithm 1 Sequential Conditional Independent Pairs

for $j = \{1, \ldots, p\}$ **do**

  Sample $\tilde{X}_j$ from $\mathcal{L}(X_j \,|\, X_{\text{-}j}, \tilde{X}_{1:j-1})$ conditionally independently of X_j

end

Proof sketch (discrete case):

- Denote PMF of $(X_{1:p}, \tilde{X}_{1:j-1})$ by $\mathcal{L}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})$

Sequential Independent Pairs Generates Valid Knockoffs

Algorithm 1 Sequential Conditional Independent Pairs

for $j = \{1, \ldots, p\}$ **do**

Sample $\tilde{X}_j$ from $\mathcal{L}(X_j \mid X_{\text{-}j}, \tilde{X}_{1:j-1})$ conditionally independently of X_j

end

Proof sketch (discrete case):

- Denote PMF of $(X_{1:p}, \tilde{X}_{1:j-1})$ by $\mathcal{L}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})$
- Conditional PMF of $\tilde{X}_j \mid X_{1:p}, \tilde{X}_{1:j-1}$ is

$$\frac{\mathcal{L}(X_{\text{-}j}, \tilde{X}_j, \tilde{X}_{1:j-1})}{\sum_u \mathcal{L}(X_{\text{-}j}, u, \tilde{X}_{1:j-1})}.$$

Sequential Independent Pairs Generates Valid Knockoffs

Algorithm 1 Sequential Conditional Independent Pairs

for $j = \{1, \ldots, p\}$ **do**

  Sample $\tilde{X}_j$ from $\mathcal{L}(X_j \,|\, X_{\text{-}j}, \tilde{X}_{1:j-1})$ conditionally independently of X_j

end

Proof sketch (discrete case):

- Denote PMF of $(X_{1:p}, \tilde{X}_{1:j-1})$ by $\mathcal{L}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})$
- Conditional PMF of $\tilde{X}_j \,|\, X_{1:p}, \tilde{X}_{1:j-1}$ is

$$\frac{\mathcal{L}(X_{\text{-}j}, \tilde{X}_j, \tilde{X}_{1:j-1})}{\sum_u \mathcal{L}(X_{\text{-}j}, u, \tilde{X}_{1:j-1})}.$$

- Joint PMF of $(X_{1:p}, \tilde{X}_{1:j})$ is

$$\frac{\mathcal{L}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})\mathcal{L}(X_{\text{-}j}, \tilde{X}_j, \tilde{X}_{1:j-1})}{\sum_u \mathcal{L}(X_{\text{-}j}, u, \tilde{X}_{1:j-1})}$$

Sequential Independent Pairs Generates Valid Knockoffs

Algorithm 1 Sequential Conditional Independent Pairs

for $j = \{1, \ldots, p\}$ **do**

Sample $\tilde{X}_j$ from $\mathcal{L}(X_j \mid X_{\text{-}j}, \tilde{X}_{1:j-1})$ conditionally independently of X_j

end

Proof sketch (discrete case):

- Denote PMF of $(X_{1:p}, \tilde{X}_{1:j-1})$ by $\mathcal{L}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})$
- Conditional PMF of $\tilde{X}_j \mid X_{1:p}, \tilde{X}_{1:j-1}$ is

$$\frac{\mathcal{L}(X_{\text{-}j}, \tilde{X}_j, \tilde{X}_{1:j-1})}{\sum_u \mathcal{L}(X_{\text{-}j}, u, \tilde{X}_{1:j-1})}.$$

- Joint PMF of $(X_{1:p}, \tilde{X}_{1:j})$ is

$$\frac{\mathcal{L}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})\mathcal{L}(X_{\text{-}j}, \tilde{X}_j, \tilde{X}_{1:j-1})}{\sum_u \mathcal{L}(X_{\text{-}j}, u, \tilde{X}_{1:j-1})}$$

Sequential Independent Pairs Generates Valid Knockoffs

Algorithm 1 Sequential Conditional Independent Pairs

for $j = \{1, \ldots, p\}$ **do**
  Sample $\tilde{X}_j$ from $\mathcal{L}(X_j \mid X_{\text{-}j}, \tilde{X}_{1:j-1})$ conditionally independently of X_j
end

Proof sketch (discrete case):

- Denote PMF of $(X_{1:p}, \tilde{X}_{1:j-1})$ by $\mathcal{L}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})$
- Conditional PMF of $\tilde{X}_j \mid X_{1:p}, \tilde{X}_{1:j-1}$ is

$$\frac{\mathcal{L}(X_{\text{-}j}, \tilde{X}_j, \tilde{X}_{1:j-1})}{\sum_u \mathcal{L}(X_{\text{-}j}, u, \tilde{X}_{1:j-1})}.$$

- Joint PMF of $(X_{1:p}, \tilde{X}_{1:j})$ is

$$\frac{\mathcal{L}(X_{\text{-}j}, \tilde{X}_j, \tilde{X}_{1:j-1})\mathcal{L}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})}{\sum_u \mathcal{L}(X_{\text{-}j}, u, \tilde{X}_{1:j-1})}$$

Sequential Independent Pairs Generates Valid Knockoffs

Algorithm 1 Sequential Conditional Independent Pairs

for $j = \{1, \ldots, p\}$ **do**

  Sample $\tilde{X}_j$ from $\mathcal{L}(X_j \mid X_{\text{-}j}, \tilde{X}_{1:j-1})$ conditionally independently of X_j

end

Proof sketch (discrete case):

- Denote PMF of $(X_{1:p}, \tilde{X}_{1:j-1})$ by $\mathcal{L}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})$
- Conditional PMF of $\tilde{X}_j \mid X_{1:p}, \tilde{X}_{1:j-1}$ is

$$\frac{\mathcal{L}(X_{\text{-}j}, \tilde{X}_j, \tilde{X}_{1:j-1})}{\sum_u \mathcal{L}(X_{\text{-}j}, u, \tilde{X}_{1:j-1})}.$$

- Joint PMF of $(X_{1:p}, \tilde{X}_{1:j})$ is

$$\frac{\color{red}{\mathcal{L}}(X_{\text{-}j}, X_j, \tilde{X}_{1:j-1})\color{red}{\mathcal{L}}(X_{\text{-}j}, \tilde{X}_j, \tilde{X}_{1:j-1})}{\sum_u \color{red}{\mathcal{L}}(X_{\text{-}j}, u, \tilde{X}_{1:j-1})}$$

Computation of Second-Order Knockoffs

$\mathrm{Cov}(X_1, \ldots, X_p) = \mathbf{\Sigma}$, need:

$$\mathrm{Cov}(X_1, \ldots, X_p, \tilde{X}_1, \ldots, \tilde{X}_p) = \begin{bmatrix} \mathbf{\Sigma} & \mathbf{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} \\ \mathbf{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} & \mathbf{\Sigma} \end{bmatrix}$$

Computation of Second-Order Knockoffs

$\mathrm{Cov}(X_1, \ldots, X_p) = \mathbf{\Sigma}$, need:

$$\mathrm{Cov}(X_1, \ldots, X_p, \tilde{X}_1, \ldots, \tilde{X}_p) = \begin{bmatrix} \mathbf{\Sigma} & \mathbf{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} \\ \mathbf{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} & \mathbf{\Sigma} \end{bmatrix}$$

- **Equicorrelated (EQ)** (fast, less powerful): $s_j^{\mathsf{EQ}} = 2\lambda_{\mathsf{min}}(\mathbf{\Sigma}) \wedge 1$ for all j

Computation of Second-Order Knockoffs

$\text{Cov}(X_1, \ldots, X_p) = \mathbf{\Sigma}$, need:

$$\text{Cov}(X_1, \ldots, X_p, \tilde{X}_1, \ldots, \tilde{X}_p) = \begin{bmatrix} \mathbf{\Sigma} & \mathbf{\Sigma} - \text{diag}\{\boldsymbol{s}\} \\ \mathbf{\Sigma} - \text{diag}\{\boldsymbol{s}\} & \mathbf{\Sigma} \end{bmatrix}$$

- **Equicorrelated (EQ)** (fast, less powerful): $s_j^{\mathsf{EQ}} = 2\lambda_{\mathsf{min}}(\mathbf{\Sigma}) \wedge 1$ for all j
- **Semidefinite program (SDP)** (slower, more powerful):

$$\begin{array}{rl} \text{minimize} & \sum_j |1 - s_j^{\mathsf{SDP}}| \\ \text{subject to} & s_j^{\mathsf{SDP}} \geq 0 \\ & \text{diag}\{s^{\mathsf{SDP}}\} \preceq 2\mathbf{\Sigma}, \end{array}$$

Computation of Second-Order Knockoffs

$\mathrm{Cov}(X_1, \ldots, X_p) = \mathbf{\Sigma}$, need:

$$\mathrm{Cov}(X_1, \ldots, X_p, \tilde{X}_1, \ldots, \tilde{X}_p) = \begin{bmatrix} \mathbf{\Sigma} & \mathbf{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} \\ \mathbf{\Sigma} - \mathrm{diag}\{\boldsymbol{s}\} & \mathbf{\Sigma} \end{bmatrix}$$

- **Equicorrelated (EQ)** (fast, less powerful): $s_j^{\mathsf{EQ}} = 2\lambda_{\mathsf{min}}(\mathbf{\Sigma}) \wedge 1$ for all j
- **Semidefinite program (SDP)** (slower, more powerful):

$$\begin{array}{rl} \text{minimize} & \sum_j |1 - s_j^{\mathsf{SDP}}| \\ \text{subject to} & s_j^{\mathsf{SDP}} \geq 0 \\ & \mathrm{diag}\{s^{\mathsf{SDP}}\} \preceq 2\mathbf{\Sigma}, \end{array}$$

- **(New) Approximate SDP**:
  - Approximate $\mathbf{\Sigma}$ as block diagonal so that SDP separates
  - Bisection search scalar multiplier of solution to account for approximation
  - faster than SDP, more powerful than EQ, and easily parallelizable

Why Does it Work?

Recall swap exchangeability property: for any j,

$$
\begin{aligned}
&[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p] \\
\stackrel{\mathcal{D}}{=}\, &[\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]
\end{aligned}
$$

Why Does it Work?

Recall swap exchangeability property: for any j,

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$
$$\overset{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

Coin-flipping property for W_j:

Why Does it Work?

Recall swap exchangeability property: for any j,

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$
$$\overset{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

Coin-flipping property for W_j: for any *unimportant* variable j,

$$\left(Z_j, \widetilde{Z}_j\right) := \left(Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right),\ \ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right)$$

Why Does it Work?

Recall swap exchangeability property: for any j,

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$
$$\overset{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

Coin-flipping property for W_j: for any *unimportant* variable j,

$$\left(Z_j, \widetilde{Z}_j\right) := \left(Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right), \;\; \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right)$$
$$\overset{\mathcal{D}}{=} \left(Z_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right), \;\; \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right)\right)$$

Why Does it Work?

Recall swap exchangeability property: for any j,

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p] \stackrel{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

Coin-flipping property for W_j: for any *unimportant* variable j,

$$\begin{aligned} \left(Z_j, \widetilde{Z}_j\right) &:= \left(Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right), \ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \\ &\stackrel{\mathcal{D}}{=} \left(Z_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right), \ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right)\right) \\ &= \left(\widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right), \ Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \end{aligned}$$

Why Does it Work?

Recall swap exchangeability property: for any j,

$$\begin{aligned}&[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p] \\ \overset{\mathcal{D}}{=}\, &[\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]\end{aligned}$$

Coin-flipping property for W_j: for any *unimportant* variable j,

$$\begin{aligned}\left(Z_j, \widetilde{Z}_j\right) &:= \left(Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right),\ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \\ &\overset{\mathcal{D}}{=} \left(Z_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right),\ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right)\right) \\ &= \left(\widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right),\ Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \\ &= \left(\widetilde{Z}_j, Z_j\right)\end{aligned}$$

Why Does it Work?

Recall swap exchangeability property: for any j,

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$
$$\stackrel{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

Coin-flipping property for W_j: for any *unimportant* variable j,

$$\begin{aligned}
\left(Z_j, \widetilde{Z}_j\right) &:= \left(Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right),\ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \\
&\stackrel{\mathcal{D}}{=} \left(Z_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right),\ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right)\right) \\
&= \left(\widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right),\ Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \\
&= \left(\widetilde{Z}_j, Z_j\right)
\end{aligned}$$

$$W_j = f_j(Z_j, \widetilde{Z}_j) \stackrel{\mathcal{D}}{=} f_j(\widetilde{Z}_j, Z_j)$$

Why Does it Work?

Recall swap exchangeability property: for any j,

$$[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$
$$\stackrel{\mathcal{D}}{=} [\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p]$$

Coin-flipping property for W_j: for any *unimportant* variable j,

$$\begin{aligned}
\left(Z_j, \widetilde{Z}_j\right) &:= \left(Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right),\ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \\
&\stackrel{\mathcal{D}}{=} \left(Z_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right),\ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right)\right) \\
&= \left(\widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right),\ Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \\
&= \left(\widetilde{Z}_j, Z_j\right)
\end{aligned}$$

$$W_j = f_j(Z_j, \widetilde{Z}_j) \stackrel{\mathcal{D}}{=} f_j(\widetilde{Z}_j, Z_j) = -f_j(Z_j, \widetilde{Z}_j) = -W_j$$

Why Does it Work?

Recall swap exchangeability property: for any j,

$$\begin{aligned} &[\boldsymbol{X}_1, \cdots, \boldsymbol{X}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \widetilde{\boldsymbol{X}}_p] \\ \overset{\mathcal{D}}{=} &[\boldsymbol{X}_1, \cdots, \widetilde{\boldsymbol{X}}_j, \cdots, \boldsymbol{X}_p, \widetilde{\boldsymbol{X}}_1, \cdots, \boldsymbol{X}_j, \cdots, \widetilde{\boldsymbol{X}}_p] \end{aligned}$$

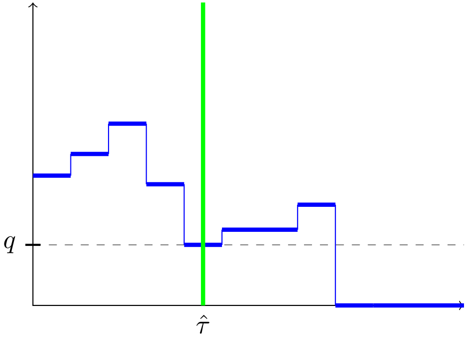
Coin-flipping property for W_j: for any *unimportant* variable j,

$$\begin{aligned} \left(Z_j, \widetilde{Z}_j\right) &:= \left(Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right),\ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \\ &\overset{\mathcal{D}}{=} \left(Z_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right),\ \widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \widetilde{\boldsymbol{X}}_j \cdots \boldsymbol{X}_j \cdots\right]\right)\right) \\ &= \left(\widetilde{Z}_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right),\ Z_j\left(\boldsymbol{y}, \left[\cdots \boldsymbol{X}_j \cdots \widetilde{\boldsymbol{X}}_j \cdots\right]\right)\right) \\ &= \left(\widetilde{Z}_j, Z_j\right) \end{aligned}$$

$$W_j \overset{\mathcal{D}}{=} -W_j$$

Proof of Control

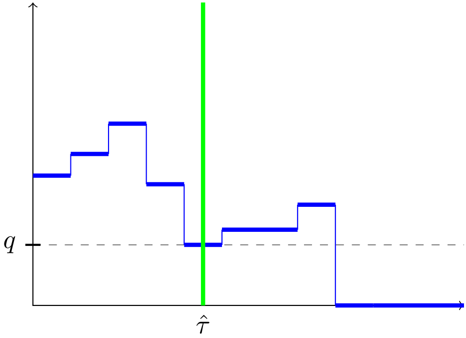
$$\text{FDR} = \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right]$$



Proof of Control

$$\text{FDR} = \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right]$$

$$= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$

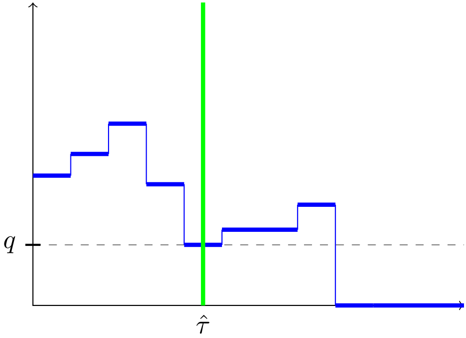


Proof of Control

$$\text{FDR} = \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right]$$

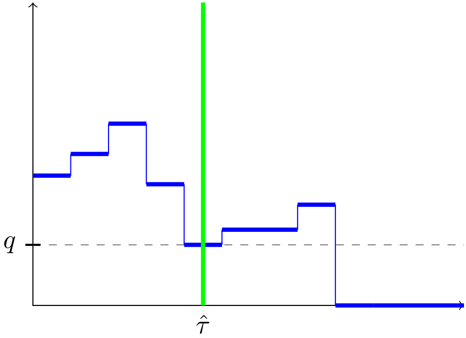
$$= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$

$$\approx \mathbb{E}\left[\frac{\#\{\text{null negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$



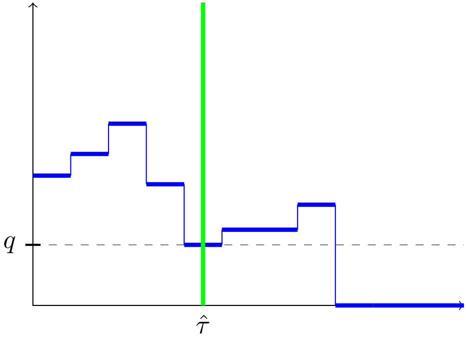
Proof of Control

$$\begin{aligned}\text{FDR} &= \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] \\ &= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\ &\approx \mathbb{E}\left[\frac{\#\{\text{null negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\ &\leq \mathbb{E}\left[\frac{\#\{\text{negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]\end{aligned}$$



Proof of Control

$$
\begin{aligned}
\text{FDR} &= \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] \\
&= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&\approx \mathbb{E}\left[\frac{\#\{\text{null negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&\leq \mathbb{E}\left[\frac{\#\{\text{negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]
\end{aligned}
$$



More precisely:

$$
\text{mFDR} = \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{q^{-1} + \#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] = \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{q^{-1} + \#\{\text{positive } |W_j| > \hat{\tau}\}}\right]
$$

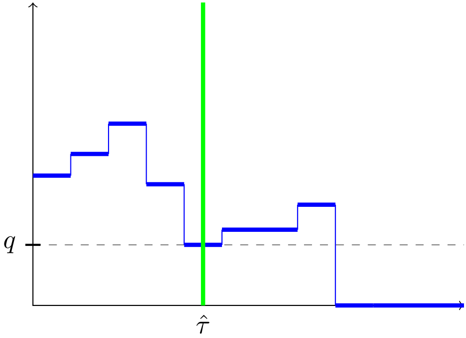
Proof of Control

$$\text{FDR} = \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right]$$

$$= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$

$$\approx \mathbb{E}\left[\frac{\#\{\text{null negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$

$$\leq \mathbb{E}\left[\frac{\#\{\text{negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$



More precisely:

$$\text{mFDR} = \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{q^{-1} + \#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] = \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{q^{-1} + \#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$

$$= \mathbb{E}\left(\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{1 + \#\{\text{null negative } |W_j| > \hat{\tau}\}} \cdot \frac{1 + \#\{\text{null negative } |W_j| > \hat{\tau}\}}{q^{-1} + \#\{\text{positive} |W_j| > \hat{\tau}\}}\right)$$

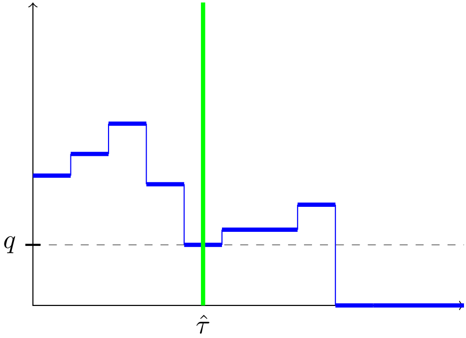
Proof of Control

$$\text{FDR} = \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right]$$

$$= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$

$$\approx \mathbb{E}\left[\frac{\#\{\text{null negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$

$$\leq \mathbb{E}\left[\frac{\#\{\text{negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$



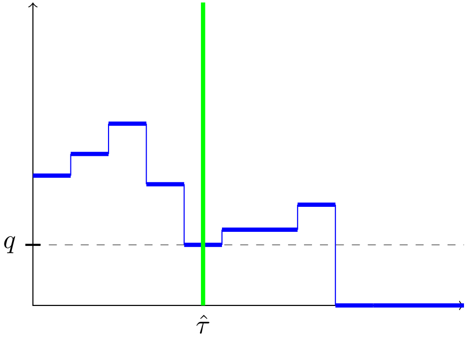
More precisely:

$$\text{mFDR} = \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{q^{-1} + \#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] = \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{q^{-1} + \#\{\text{positive } |W_j| > \hat{\tau}\}}\right]$$

$$= \mathbb{E}\left(\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{1 + \#\{\text{null negative } |W_j| > \hat{\tau}\}} \cdot \underbrace{\frac{1 + \#\{\text{null negative } |W_j| > \hat{\tau}\}}{q^{-1} + \#\{\text{positive} |W_j| > \hat{\tau}\}}}_{\leq q \text{ by definition of } \hat{\tau}}\right)$$

Proof of Control

$$\begin{aligned}
\text{FDR} &= \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{\#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] \\
&= \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&\approx \mathbb{E}\left[\frac{\#\{\text{null negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&\leq \mathbb{E}\left[\frac{\#\{\text{negative } |W_j| > \hat{\tau}\}}{\#\{\text{positive } |W_j| > \hat{\tau}\}}\right]
\end{aligned}$$



More precisely:

$$\begin{aligned}
\text{mFDR} &= \mathbb{E}\left[\frac{\#\{\text{null } \boldsymbol{X}_j \text{ selected}\}}{q^{-1} + \#\{\text{total } \boldsymbol{X}_j \text{ selected}\}}\right] = \mathbb{E}\left[\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{q^{-1} + \#\{\text{positive } |W_j| > \hat{\tau}\}}\right] \\
&= \mathbb{E}\left(\underbrace{\frac{\#\{\text{null positive } |W_j| > \hat{\tau}\}}{1 + \#\{\text{null negative } |W_j| > \hat{\tau}\}}}_{\substack{\text{Supermartingale } \leq 1 \\ \text{with } \hat{\tau} \text{ a stopping time}}} \cdot \underbrace{\frac{1 + \#\{\text{null negative } |W_j| > \hat{\tau}\}}{q^{-1} + \#\{\text{positive} |W_j| > \hat{\tau}\}}}_{\leq q \text{ by definition of } \hat{\tau}}\right)
\end{aligned}$$

Simulations in Low-Dimensional Nonlinear Model

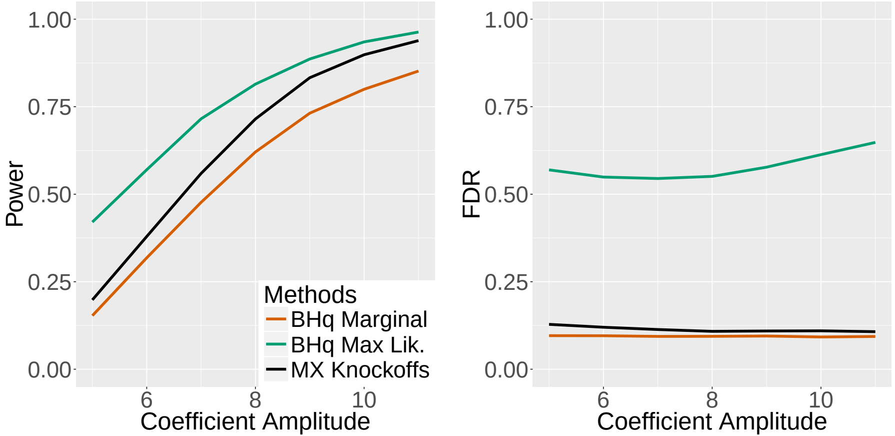


Figure: Power and FDR (target is 10%) for knockoffs and alternative procedures. The design matrix is i.i.d. $\mathcal{N}(0, 1/n)$, $n = 3000$, $p = 1000$, and y comes from a binomial linear model with logit link function, and 60 nonzero regression coefficients having equal magnitudes and random signs.

Simulations in High Dimensions

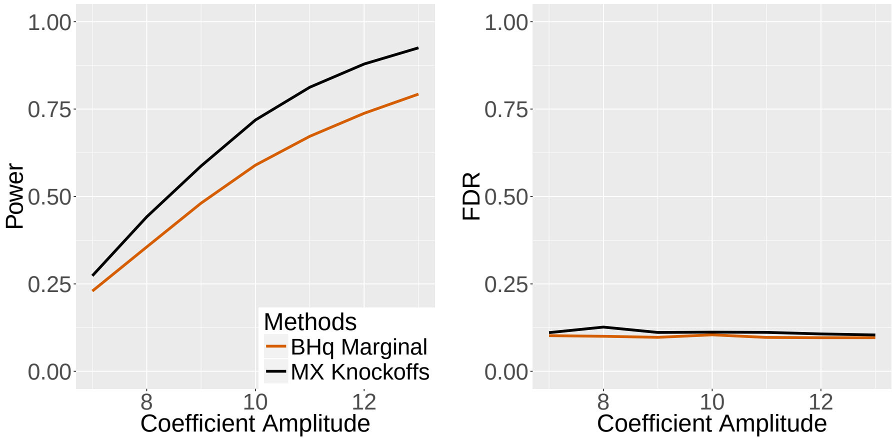


Figure: Power and FDR (target is 10%) for knockoffs and alternative procedures. The design matrix is i.i.d. $\mathcal{N}(0, 1/n)$, $n = 3000$, $p = 6000$, and y comes from a binomial linear model with logit link function, and 60 nonzero regression coefficients having equal magnitudes and random signs.

Simulations in High Dimensions with Dependence

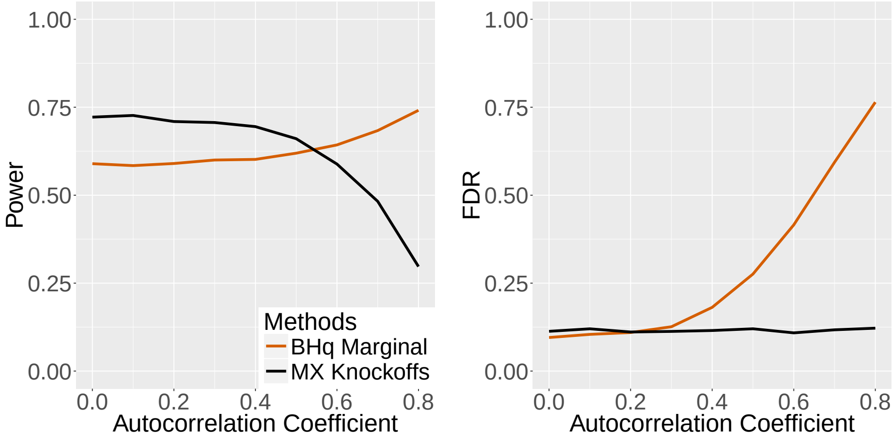


Figure: Power and FDR (target is 10%) for knockoffs and alternative procedures. The design matrix has AR(1) columns, and marginally each $X_j \sim \mathcal{N}(0, 1/n)$. $n = 3000$, $p = 6000$, and y follows a binomial linear model with logit link function, and 60 nonzero coefficients with random signs and randomly selected locations.

Genetic Analysis of Crohn's Disease

2007 case-control study by WTCCC

- $n \approx 5,000$, $p \approx 375,000$; preprocessing mirrored original analysis

Genetic Analysis of Crohn's Disease

2007 case-control study by WTCCC

- $n \approx 5{,}000$, $p \approx 375{,}000$; preprocessing mirrored original analysis
- **Strong spatial structure**: second-order knockoffs generated on genetic covariance estimate

Genetic Analysis of Crohn's Disease

2007 case-control study by WTCCC

- $n \approx 5,000$, $p \approx 375,000$; preprocessing mirrored original analysis
- **Strong spatial structure**: second-order knockoffs generated on genetic covariance estimate
- Entire analysis took 6 hours of serial computation time; **1 hour** in parallel

Genetic Analysis of Crohn's Disease

2007 case-control study by WTCCC

- $n \approx 5,000$, $p \approx 375,000$; preprocessing mirrored original analysis
- **Strong spatial structure**: second-order knockoffs generated on genetic covariance estimate
- Entire analysis took 6 hours of serial computation time; **1 hour** in parallel
- Knockoffs made **twice as many discoveries** as original analysis

Genetic Analysis of Crohn's Disease

2007 case-control study by WTCCC

- $n \approx 5,000$, $p \approx 375,000$; preprocessing mirrored original analysis
- **Strong spatial structure**: second-order knockoffs generated on genetic covariance estimate
- Entire analysis took 6 hours of serial computation time; **1 hour** in parallel
- Knockoffs made **twice as many discoveries** as original analysis
  - Some new discoveries confirmed in larger study

Genetic Analysis of Crohn's Disease

2007 case-control study by WTCCC

- $n \approx 5,000$, $p \approx 375,000$; preprocessing mirrored original analysis
- **Strong spatial structure**: second-order knockoffs generated on genetic covariance estimate
- Entire analysis took 6 hours of serial computation time; **1 hour** in parallel
- Knockoffs made **twice as many discoveries** as original analysis
  - Some new discoveries confirmed in larger study
  - Some corroborated by work on nearby genes: promising candidates

Genetic Analysis of Crohn's Disease

2007 case-control study by WTCCC

- $n \approx 5,000$, $p \approx 375,000$; preprocessing mirrored original analysis
- **Strong spatial structure**: second-order knockoffs generated on genetic covariance estimate
- Entire analysis took 6 hours of serial computation time; **1 hour** in parallel
- Knockoffs made **twice as many discoveries** as original analysis
  - Some new discoveries confirmed in larger study
  - Some corroborated by work on nearby genes: promising candidates
- Similar result obtained with X model taken from **existing genomic imputation software**

Checking Sensitivity to Misspecification Error

	Concern about misspecification	
	$Y \mid X$	X
Canonical (fixed-X)	Yes	No
Model-X	No	Yes

Checking Sensitivity to Misspecification Error

	Concern about misspecification	
	$Y \mid X$	X
Canonical (fixed-X)	Yes	No
Model-X	No	Yes
Misspecification replicated in simulation?	No	Yes

Checking Sensitivity to Misspecification Error

	Concern about misspecification $Y \mid X$	Concern about misspecification X
Canonical (fixed-X)	Yes	No
Model-X	No	Yes
Misspecification replicated in simulation?	No	Yes

Model-X: can actually **check sensitivity** to misspecification error!

Robustness on Real Data

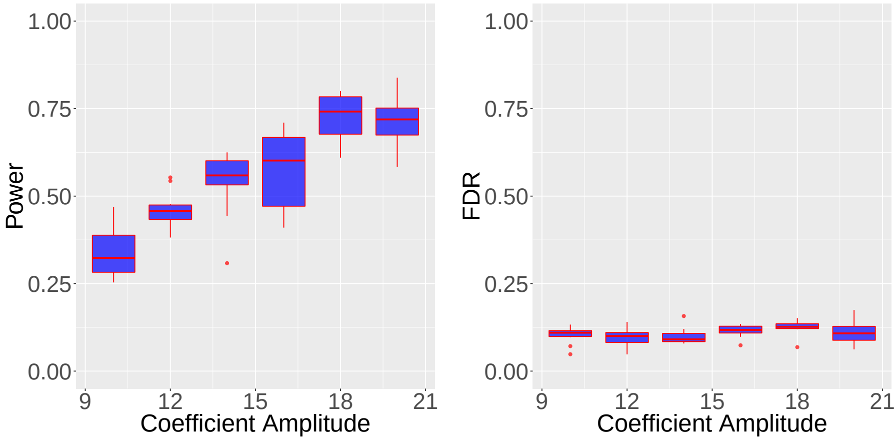


Figure: Power and FDR (target is 10%) for knockoffs applied to subsamples of a chromosome 1 of real genetic design matrix; $n \approx 1,400$.