Fixed design matrix
Modeling the design matrix
Concluding remarks

Knockoff filter

Patrick Breheny

April 7

Fixed design matrix
Modeling the design matrix
Concluding remarks

Introduction

- Today we will discuss one final approach to inference in high-dimensional regression models called the *knockoff filter*
- There are two approaches to the knockoff filter:
  - In its simplest form, we can generate knockoffs without any assumptions on $\mathbf{X}$; however this approach only works if $\mathbf{X}$ is full rank (Barber and Candès 2015)
  - A later paper (Candès et al. 2018) extended this idea to the $p > n$ case, although in order to do so, we need to make some assumptions about $\mathbf{X}$
- Both approaches are implemented in the R package knockoff

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

Step 1: Construct knockoffs

- The basic idea of the knockoff filter is that for each feature $\mathbf{x}_j$ in the original feature matrix, we construct a *knockoff* feature $\tilde{\mathbf{x}}_j$

- We'll go into specifics on constructing knockoffs later; for now, we specify the properties that a knockoff $\tilde{\mathbf{x}}_j$ must have:

$$\widetilde{\mathbf{X}}^\top \widetilde{\mathbf{X}} = \mathbf{X}^\top \mathbf{X}$$

$$\tilde{\mathbf{x}}_j^\top \mathbf{x}_k = \mathbf{x}_j^\top \mathbf{x}_k \qquad \text{for all } k \neq j$$

$$\tfrac{1}{n} \tilde{\mathbf{x}}_j^\top \mathbf{x}_j = 1 - s_j \qquad \text{where } 0 \leq s_j \leq 1$$

- In other words, the knockoff matrix $\widetilde{\mathbf{X}}$ differs from the original matrix $\mathbf{X}$, but has the same correlation structure and the same correlation with the original features

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

Step 2: Calculate test statistics

- With the knockoffs constructed, the next step is to fit a (lasso) model to the augmented $n \times 2p$ design matrix $[\mathbf{X} \; \widetilde{\mathbf{X}}]$

- At this point, we need some sort of test statistic that measures whether the original feature is better than the knockoff

- There are actually a variety of statistics we could use here, but in this lecture we'll focus on the point λ along the lasso path at which a feature enters the model, giving us a $2p$-dimensional vector $\{Z_1, \ldots, Z_p, \tilde{Z}_1, \ldots, \tilde{Z}_p\}$

- Our test statistic is then

$$W_j = \max(Z_j, \tilde{Z}_j) \cdot \text{sign}(Z_j - \tilde{Z}_j);$$

i.e., W_j will be positive if the original feature is selected before the knockoff, and negative if the knockoff is selected first

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

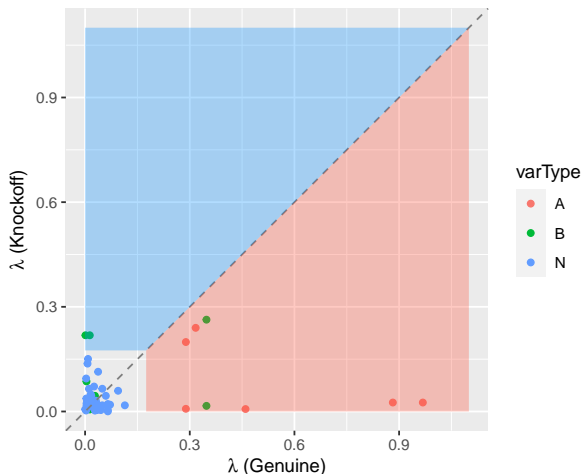## Step 3: Estimate false discovery rate

- Now, if we select features such that $W_j \geq t$ for some threshold t, we can use the knockoff features to estimate the false discovery rate

- Specifically, our knockoff estimate of the FDR is:

$$\widehat{\mathrm{FDR}} = \frac{\#\{j : W_j \leq -t\}}{\#\{j : W_j \geq t\}},$$

with the understanding that $\widehat{\mathrm{FDR}} = 1$ if the numerator is larger than the denominator, or if the denominator is zero

- Typically, we would specify the desired FDR q and then choose t to be the smallest value satisfying $\widehat{\mathrm{FDR}}(t) \leq q$

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

Illustration: Augmented example data ($n = 200, p = 60$)

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

Power and $\{s_j\}$

- So, how do we actually construct these knockoffs?
- As we will see, the knockoff filter is valid provided that the knockoffs have the correlation structure outlined earlier; its power, however, depends on $\{s_j\}$
- For the greatest power, we want the knockoffs to be as different from the original features as possible (i.e, we want the $\{s_j\}$ terms to be as large as possible)

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

Nullspace, n, and p

- Let $\mathbf{N}$ denote an $n \times p$ orthonormal matrix such that $\mathbf{N}^\top \mathbf{X} = \mathbf{0}$ (in other words, $\mathbf{N}\boldsymbol{\alpha}$ lies within the column null space of $\mathbf{X}$; note that this can be constructed using the QR decomposition)

- Note that the nullspace of $\mathbf{X}$ has dimension $n - \mathrm{rank}(\mathbf{X})$

- Thus, for the matrix $\mathbf{N}$ to exist, it is not enough for $\mathbf{X}$ to be full rank; we also need $n \geq p + \mathrm{rank}(\mathbf{X})$, so $n \geq 2p$ in the full-rank case

Fixed design matrix   Procedure
Modeling the design matrix   Constructing the knockoffs
Concluding remarks   Theoretical properties

Constructing knockoffs under equal correlation

- So, let's say we have a full rank $\mathbf{X}$ with $n \geq 2p$ and thus can construct an orthonormal $\mathbf{N}$ with $\mathbf{N}^\top \mathbf{X} = 0$

- Furthermore, suppose we require $s_j = s$ for all j and let $\frac{1}{n}\mathbf{C}^\top\mathbf{C} = 2s\mathbf{I} - s^2\mathbf{\Sigma}^{-1}$, where $\mathbf{\Sigma} = \frac{1}{n}\mathbf{X}^\top\mathbf{X}$

- **Proposition:** The matrix

$$\widetilde{\mathbf{X}} = \mathbf{X}(\mathbf{I} - s\mathbf{\Sigma}^{-1}) + \mathbf{N}\mathbf{C}$$

  satisfies the requirements of a knockoff matrix

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
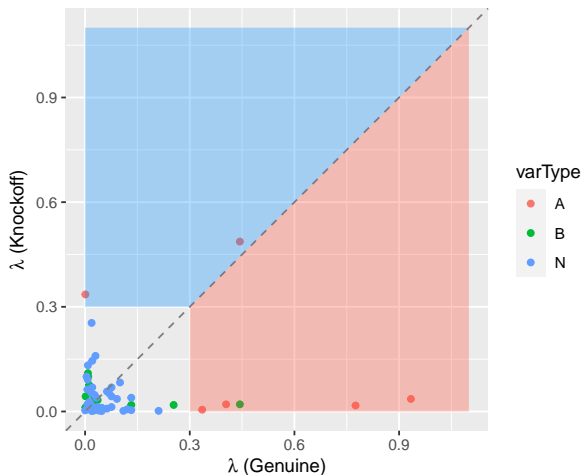**Constructing the knockoffs**
Theoretical properties

The non-full rank case

- What if $\mathbf{X}$ is not full rank?

- It turns out that the maximum value for s is 2 times the minimum eigenvalue of $\boldsymbol{\Sigma}$; thus, $s_j = s$ for all j cannot work in the case where $\mathbf{X}$ is not full rank

- In this case, we will have to set some of the $s_j = 0$ (meaning no power for those features) and try to maximize the rest as best we can

- In the knockoff package, a semidefinite programming approach is used to determine the values that minimize $\sum_j (1 - s_j)$ subject to the constraints (method='sdp'; the earlier approach is method='equi')

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

The $p < n < 2p$ case

- Now, what if $\mathbf{X}$ is full rank, but $n < 2p$?

- In this case, there is an interesting little data augmentation trick that can be used, provided that σ^2 can be estimated accurately

- To get our sample size up to $2p$, we can generate $2p - n$ additional rows of $\mathbf{X}$ that are simply all equal to $\mathbf{0}$ and $2p - n$ additional entries for $\mathbf{y}$ that are drawn from a $\mathrm{N}(0, \hat{\sigma}^2)$ distribution

- We now have a linear model with p features and $2p$ observations; the new observations carry no information about β, but are useful for generating knockoffs

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

$p < n < 2p$ data augmentation applied to example data

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

FDR control

- So does this knockoff procedure actually control the FDR? Not quite

- Instead, Barber and Candès show that it controls a modified version of the FDR:

$$\mathbb{E}\left(\frac{|\mathcal{N} \cap \hat{\mathcal{S}}|}{|\hat{\mathcal{S}}| + q^{-1}}\right) \leq q,$$

where $\hat{\mathcal{S}}$ is the set of features selected by the knockoff filter

- Alternatively, the knockoff filter controls the FDR if we add 1 to the numerator (i.e., to the number of knockoffs selected)

- The modifications have a nontrivial effect unless many features are selected

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

Coin flip lemma

- We won't go through the entire proof here, but just present a sketch of the main ideas

- The critical property that knockoffs have is a "coin flipping property": for $j \in \mathcal{N}$, we have $\mathrm{sign}(W_j) \stackrel{\perp\!\!\!\perp}{\sim} \mathrm{Bern}(1/2)$

- This coin flipping property derives from two exchangeability results:
  - $[\mathbf{X} \ \widetilde{\mathbf{X}}]^\top [\mathbf{X} \ \widetilde{\mathbf{X}}]$ is invariant to any exchange of original and knockoff features
  - The distribution of $[\mathbf{X} \ \widetilde{\mathbf{X}}]^\top \mathbf{y}$ is invariant to any exchange of *null* original and knockoff features

Fixed design matrix
Modeling the design matrix
Concluding remarks

Procedure
Constructing the knockoffs
Theoretical properties

Sketch of proof

- With these lemmas in place, the FDR control proof follows from the inequality

$$\text{FDR} \le q \cdot \frac{\#\{j : \beta_j = 0 \text{ and } W_j > t\}}{1 + \#\{j : \beta_j = 0 \text{ and } W_j < -t\}};$$

  the coin flipping property ensuring that the expected value of this quantity is below q

- The argument can be extended to a random threshold T through use of martingales and the optional stopping theorem similar to our FDR proof at the beginning of the course

Fixed design matrix
Modeling the design matrix
Concluding remarks

Modeling $\mathbf{X}$

- An obvious shortcoming of the previous approach is that it requires $n \geq p$

- Extending the idea to $p > n$ situations requires us to treat $\mathbf{X}$ as random and to model its distribution; Candès et al. refer to these as "model-X knockoffs" or just "MX" knockoffs

- Note that this is an interesting philosophical shift: the classical setup is to assume a very specific distribution for $\mathbf{y}$ but assume as little as possible about $\mathbf{X}$, whereas MX knockoffs assume that we know everything about the distribution of $\mathbf{X}$ but require no assumptions on the distribution of $Y|\mathbf{X}$

Fixed design matrix
Modeling the design matrix
Concluding remarks

Knockoff properties in the random case

- Recall our exchangeability results from earlier; with these in mind, we can define knockoff conditions in the case where $\mathbf{X}$ is treated as a random matrix with IID rows
- A knockoff matrix $\widetilde{\mathbf{X}}$ satisfies
  - The distribution of $[X \ \tilde{X}]$ is invariant to any exchange of original and knockoff features
  - $\tilde{X} \perp\!\!\!\perp Y | X$
- Note that the second condition is guaranteed if $\widetilde{\mathbf{X}}$ is constructed without looking at $\mathbf{y}$

Fixed design matrix
Modeling the design matrix
Concluding remarks

Gaussian case

- There are special cases in which we actually know something about the distribution of $\mathbf{X}$; in general, however, we would likely assume it follows a multivariate normal distribution

- The main challenge here is that now we must estimate $\mathbf{\Sigma}$, a $p \times p$ covariance matrix, or rather $\mathbf{\Sigma}^{-1}$, the precision matrix

- We will (time permitting) discuss this problem a bit later in the course; for now, although this is by no means trivial, let us assume that we can estimate $\mathbf{\Sigma}$ well enough to assume that we know $X \sim \mathrm{N}(\mathbf{0}, \mathbf{\Sigma})$

Fixed design matrix
Modeling the design matrix
Concluding remarks

MX knockoffs in the Gaussian case

- In order to satisfy the knockoff property, let us assume the joint distribution $[X \ \tilde{X}] \sim \mathrm{N}(\mathbf{0}, \mathbf{G})$ where

$$\mathbf{G} = \left[\begin{array}{cc} \boldsymbol{\Sigma} & \boldsymbol{\Sigma} - \mathbf{S} \\ \boldsymbol{\Sigma} - \mathbf{S} & \boldsymbol{\Sigma} \end{array} \right];$$
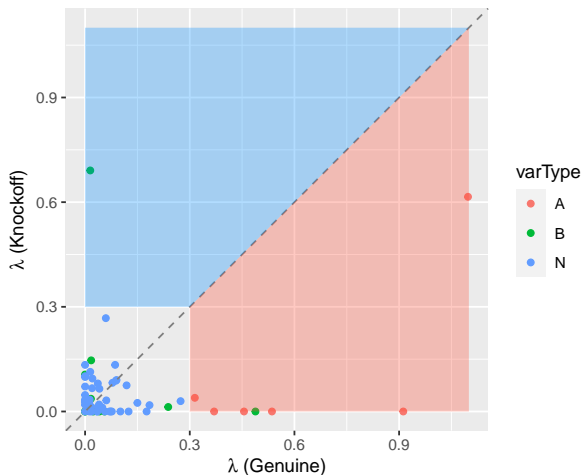
here $\mathbf{S}$ is a diagonal matrix with entries $\{s_j\}$

- Now, we can draw a random $\widetilde{\mathbf{X}}$ from the conditional distribution $\tilde{X}|X$, which is normal with

$$\mathbb{E}(\tilde{X}|X) = X - \mathbf{S}\boldsymbol{\Sigma}^{-1}X$$
$$\mathbb{V}(\tilde{X}|X) = 2\mathbf{S} - \mathbf{S}\boldsymbol{\Sigma}^{-1}\mathbf{S}$$

Fixed design matrix
Modeling the design matrix
Concluding remarks

Example data with modeled $\mathbf{X}$

Fixed design matrix
Modeling the design matrix
Concluding remarks

TCGA data

- I tried applying the MX knockoff approach to the TCGA data using the knockoff package, but this crashed, being unable to allocate the RAM for a $17,322 \times 17,322$ matrix

- I even tried running it on our HPC cluster, but this also crashed

- So it would seem safe to say that at least in the current version (0.3.3), the knockoff package is not particularly scalable to high dimensions

- However, it is worth noting that in their paper, Candès et al. applied the MX knockoff filter to a problem with $p = 400,000$ by taking advantage of a special correlation structure in $\mathbf{X}$

Fixed design matrix
Modeling the design matrix
Concluding remarks

Remarks: Some drawbacks

- The results I obtained for this example differed quite a bit depending on the random $\widetilde{\mathbf{X}}$ I drew; it would seem desirable to aggregate or average these results over the draws, although how exactly to do this is unclear

- In our examples, the knockoff filter seemed a bit less powerful than some of the other approaches we have discussed, although it's a fairly new method and at present it isn't fully understood (by me, at least) how it compares to the other inferential approaches we have discussed

- This is especially true in the MX case: "Admittedly, this paper may pose more problems than it solves"

- Furthermore, scaling the method to very high dimensions is not trivial

Fixed design matrix
Modeling the design matrix
Concluding remarks

Remarks: Some advantages

- However, the knockoff filter also has some nice advantages
- In particular, none of its theory involves any asymptotics, or anything special about the statistic W, or about the lasso, which means:
  - The theory holds exactly in finite dimensions
  - We can use other statistics, such as the lasso coefficient difference: $W_j = |\widehat{\beta}_j(\lambda)| - |\widehat{\beta}_{j+p}(\lambda)|$
  - Perhaps most appealing, we can apply this reasoning to all kinds of other methods – other penalties of course, but also much more ambitious problems: forward selection, random forests, even deep learning