Model-X Knockoff for Variable Selection

Linh Nguyen, Sungmin Park, Hexuan Zhang

December 16, 2021

Abstract

This paper explored the method of Model-X Knockoff for variable selection. Here, we provide a introductory review of the *model-X knockoff* and provide a simulation study of knockoff variable selection. Specifically, we conducted simulation studies to examine the robustness of knockoff methods and compare the performance of knockoffs with standard methods in FDR control under both moderate ($n > p$) and high-dimensional ($n \ll p$) setting. We also applied the method to analyze high-dimensional genetic data using single-nucleotide polymorphisms to predict height. The results show that knockoff is a powerful variable selection tool that works under various conditions where the traditional methods fail. However, there is significant computation cost associated with constructing knockoff variables compared to traditional methods.

Keywords: model-X, knockoff, variable selection, genome-wide association study, single-nucleotide polymorphism

1 Introduction

1.1 Variable selection

In this report, we look at the problem of variable selection: given a response variable Y and a set of predictor variables $\{X_1, \ldots, X_p\}$ we wish to identify a subset of predictor variables that is related to the response Y. More formally, a predictor variable X_j is called a *null* variable if

$$Y \perp\!\!\!\perp X_j | X_1, \ldots X_{j-1}, X_{j+1}, \ldots, X_p,$$

that is, X_j does not offer any additional information about Y. Let $\mathcal{H}_0$ be the index set of null variables. Different variable selection problems follow with different goals or criteria but here we will look at the false discovery rate (FDR)

$$\text{FDR} := \mathbf{E}[\text{FDP}] = \mathbf{E}\frac{|\hat{S} \cap \mathcal{H}_0|}{|\hat{S}|} \leq \alpha$$

as our main criterion. The goal of this problem is to identify a subset $\hat{S} \subset \{1, \ldots, p\}$ of non-null variables while controlling the level of FDR under some pre-specified value $\alpha \in (0, 1)$.

1.2 Benjamini-Hochberg

A traditional and well-known method that controls the FDR under multiple hypothesis testing regime is the Benjamini-Hochberg (BH) procedure. A brief description of the BH procedure to control the FDR at level $\alpha \in (0, 1)$ is as follows:

i) Given a set of p-values $\{p_1, \ldots, p_k\}$, arrange them in increasing order $p_{(1)} \leq \ldots \leq p_{(k)}$.

ii) Select largest $j \in \{1, \ldots, k\}$ that satisfies $p_{(j)} \leq \frac{j}{k}\alpha$.

iii) Only reject the null hypotheses that corresponds to the smallest j p-values $p_{(1)}, \ldots, p_{(j)}$.

iv) The proportion of falsely rejected null hypothesis (Type-I error) will be less than or equal to α.

Although this method is relatively simple and straight-forward, it comes with a few drawbacks. The method requires a collection of p-values, so it can only be applied to a problem with a known testing procedure. Moreover, many testing procedures are formulated using the asymptotic distribution of the test statistic rather than its exact distribution. In such case, if sample size is not large enough the p-values derived from the asymptotic distribution of the test statistic may not be accurate. If the p-values are inaccurate, then the control of the FDR cannot be guaranteed for the BH method.

1.3 Outline of this report

In Section 2, we introduce a novel method called *Model-X Knockoff* proposed by Candes et al. (2018) that can control the FDR without the use of p-values. In Section 3, we demonstrate the performance of Knockoff procedure. Application of Knockoff method in single-nucleotide polymorphisms (SNPs) data is provided in Section 4. We conclude the paper with discussion in Section 5.

2 Method

2.1 Knockoff methods

Knockoff methods are a class of variable selection methods designed to control the FDR. Knockoff was originally proposed in Barber and Candès (2015) where the authors considered the problem of variable selection for fixed design problem, that is, the predictor variables are assumed to be deterministic instead of being random. Candes et al. (2018) later proposed another version of knockoff called *model-X knockoff* in which the predictor variables were assumed to be random. The former method is often called *fixed-X knockoff* to differentiate from the latter method. In this report, we focus on model-X knockoff for the following strengths compared to fixed-X knockoff. Fixed-X knockoff requires the number of observations n to be larger than the number of variables p. However, model-X knockoff can also be applied to high-dimensional problems with $p \gg n$ which is typically the case for single nucleotide polymorphisms (SNP) data. The main assumption in model-X knockoff is that the predictor variable $(X_1, \ldots, X_p)$ are random and the joint distribution of $(X_1, \ldots, X_p)$ is known.

2.1.1 Knockoff variables

We first introduce the definition of *MX knockoffs* which is the central ingredient in the model-X knockoff variable selection procedure. *MX knockffs* for the family of random variables $X = (X_1, \ldots, X_p)$ are a new family of random variables $\tilde{X} = (\tilde{X}_1, \ldots, \tilde{X}_p)$ with the following two properties:

1. $(X, \tilde{X})_{\text{swap}(S)} \overset{d}{=} (X, \tilde{X})$, $\forall S \subset \{1, \ldots, p\}$,

2. $\tilde{X} \perp\!\!\!\perp Y | X$ if there is a response Y.

The random vector $(X, \tilde{X})_{\text{swap}(S)}$ is obtained from $(X, \tilde{X})$ by swapping the entries X_j and $\tilde{X}_j$ for each $j \in S$. Here is an example of swap operator

$$(X_1, X_2, X_3, \tilde{X}_1, \tilde{X}_2, \tilde{X}_3)_{\text{swap}(\{2,3\})} = (X_1, \tilde{X}_2, \tilde{X}_3, \tilde{X}_1, X_2, X_3).$$

Intuitively, knockoff variables $\tilde{X}$ behave similarly to X (Property 1) but are irrelevant to Y (Property 2) serving as a control group for variable selection.

2.1.2 Variable selection procedure

Now that we have MX knockoffs, we construct knockoff test statistic W_j for each variable X_j. Any function can be used as knockoff statistics as long as it satisfies the *flip sign property*

$$w_j\{(X, \tilde{X})_{\text{swap}(S)}, y\} = \begin{cases} w_j\{(X, \tilde{X}), y\}, & x \notin S, \\ -w_j\{(X, \tilde{X}), y\}, & x \in S. \end{cases}$$

One example of such statistic is the lasso coefficient difference (LCD) statistic. Given $\lambda > 0$, we first obtain the lasso shrinkage estimator

$$\hat{\beta}(\lambda) = \operatorname*{argmin}_{\beta \in \mathbb{R}^{2p}} \left\{ \frac{1}{2} \|y - (X, \tilde{X})\beta\|_2^2 + \lambda \|\beta\|_1 \right\}$$

where $\|\cdot\|_p$ denotes the ℓ_p-norm, and calculate the LCD statistic

$$W_j = |\hat{\beta}_j(\lambda)| - |\hat{\beta}_{j+p}(\lambda)|,$$

the difference in the magnitude of lasso regression coefficient for the j-th variable X_j and its MX knockoff counterpart $\tilde{X}_j$. The LCD statistic clearly satisfies the flip sign property and a large positive value of W_j indicates that X_j is not null. To control the FDR at level $\alpha \in (0, 1)$, we choose the threshold

$$\tau = \min \left\{ t > 0 : \frac{\#\{j : W_j \leq -t\}}{\#\{j : W_j \geq t\}} \leq \alpha \right\},$$

and select the subset of non-null variables $\hat{S} = \{j : W_j \geq \tau\}$. This procedure will control the modified FDR which is defined as

$$\text{mFDR} = \mathbf{E} \left[\frac{|\hat{S} \cap \mathcal{H}_0|}{|\hat{S}| + 1/\alpha} \right] \leq \alpha.$$

A slightly more conservative threshold (called *knockoff+*)

$$\tau_+ = \min \left\{ t > 0 : \frac{1 + \#\{j : W_j \leq -t\}}{\#\{j : W_j \geq t\}} \leq \alpha \right\}$$

will control the FDR in the usual sense.

2.1.3 Constructing MX knockoff variables

Although model-X knockoff is an appealing method, we may not know the true distribution of $X = (X_1, \dots, X_p)$ when in practice. Even if the distribution of X was known, it may not be clear how to generate MX knockoff $\tilde{X}$. Candes et al. (2018) provides a particular algorithm to construct MX knockoff using conditional distribution of $\mathcal{L}(X_j | X_{-j}, \tilde{X}_{1:j-1})$ where we use the notation $X_{-j} = (X_1, \dots, X_{j-1}, X_{j+1}, \dots, X_p)$ and $\tilde{X}_{1:j-1} = (\tilde{X}_1, \dots, \tilde{X}_{j-1})$. Bates et al. (2021) provides a sequential characterization of all knockoff distributions and also practical algorithms to generate MX knockoffs.

2.1.4 Approximate knockoff

As an alternative to generating exact MX knockoffs that satisfies $(X, \tilde{X})_{\text{swap}} \overset{d}{=} (X, \tilde{X})$, we may instead generate approximate MX knockoffs with matching first and second moments

$$\mathbf{E}(X, \tilde{X})_{\text{swap}} = \mathbf{E}(X, \tilde{X}), \quad \mathbf{E}(X, \tilde{X})^2_{\text{swap}} = \mathbf{E}(X, \tilde{X})^2.$$

If the first and second moments are specified, then MX knockoffs can be sampled from multivariate Gaussian distribution. It is worth emphasizing that if X follows multivariate Gaussian distribution, then the approximate procedure will in fact produce exact MX knockoffs. Suppose $\text{Cov}(X) = \Sigma$, then $(X, \tilde{X})_{\text{swap}(S)} \overset{d}{=} (X, \tilde{X})$ requires the covariance of $(X, \tilde{X})$ to have the form

$$\text{Cov}(X, \tilde{X}) = \begin{pmatrix} \Sigma & \Sigma - \text{diag}(s) \\ \Sigma - \text{diag}(s) & \Sigma \end{pmatrix}$$

where $\text{diag}(s)$ is a diagonal matrix with s as its diagonal components. Any choice of s will work as long as $\text{Cov}(X, \tilde{X})$ is positive semidefinite. Intuitively, we would want the components of s to be large so that $\text{Cov}(X_j, \tilde{X}_j)$ is small. The extreme case of $s_j = 0$ will produce $\tilde{X}_j$ that is perfectly correlated with X_j.

Candes et al. (2018) proposes three different methods to choose s. Assuming that X_j's are scaled to have zero mean and unit variance, one can use

1. equi-correlated: $s_j^{\text{EQ}} = 2\lambda_{\min}(\Sigma) \wedge 1 \; \forall j$ where $\lambda_{\min}(\Sigma)$ denotes the smallest eigenvalue of Σ.

2. semidefinite programme (SDP): obtain s^{SDP} as the solution to

$$\begin{aligned} \text{minimize} \quad & \sum_j |1 - s_j^{\text{SDP}}| \\ \text{subject to} \quad & s_j^{\text{SDP}} \geq 0, \\ & \operatorname{diag}(s^{\text{SDP}}) \preceq 2\Sigma. \end{aligned}$$

Equi-correlated method is relatively easier to compute compared to SDP method. However, s_j^{EQ} tends to be very small or nearly zero for Σ with large dimensions. On the other hand, SDP method will produce s_j than equi-correlated method at the cost of heavier computation. As a mixture of both methods, the authors proposed an approximate semidefinite programme (ASDP) method: obtain solution $\hat{s}$ from

$$\begin{aligned} \text{minimize} \quad & \sum_j |1 - \hat{s}_j| \\ \text{subject to} \quad & \hat{s}_j \geq 0, \\ & \operatorname{diag}(\hat{s}) \preceq 2\Sigma_{\text{approx}}. \end{aligned}$$

and set $s^{\text{ASDP}} = \hat{\gamma}\hat{s}$ where $\hat{\gamma} = \max\{\gamma : \operatorname{diag}(\gamma\hat{s}) \preceq 2\Sigma\}$. Choosing a good block diagonal approximation Σ_{approx} of Σ will save considerable computation time while not degrading the magnitude of s by a large extent. Trivial choices of $\Sigma_{\text{approx}} = I$ and Σ corresponds to equi-correlated and SDP method, respectively.

3 Simulation studies

3.1 High-dimensional data

High-dimensional data refer to when the number of predictors or features exceed the number of cases or observations in the data set, $p \gg n$. This is a common occurrence in genome-wide association studies, in which individuals provide their genotypes for hundreds of thousands to millions of single-nucleotide polymorphisms across the genome. For knockoff, we first fit a multivariate Gaussian distribution to observations of X, then generate Gaussian knockoff based on the estimated model using approximate semidefinite program (ASDP) construction. As a comparison, we apply BH to the same data. However, with $p \gg n$, the assumptions of BH are not satisfied, so we use Ridge regression to fit model with all covariates and extract p-values. Adjusted p-values using BH corrections are obtained and those lower than the FDR are ultimately selected.

In the first simulation, we generate the data 100 times and perform both knockoff and BH on them. FDR, power, and run time were calculated as the average value across 100 replications, separately for each procedure. Data were generated using the following conditions:

- $n = 400$ sample size
- $p = 1000$ predictors
- $k = 40$ predictors that are non-zero
- amp $= 5$ signal amplitude
- FDR $= 0.1$ false discovery rate
- $X_1, ..., X_n \sim \text{iid} N(0, 1)$
- $\beta = I(X \in \text{non-zero covariates}) \times \text{amp}/\sqrt{n}$
- $Y = X \times \beta + \epsilon$ where $\epsilon \sim N(0, 1)$

In this setting, we have: Across 100 replications, model-X knockoff has slightly better power and slightly worse FDR control than BH. However, there is a significant computation cost with almost 20s per run for model-X knockoff procedures (11 times longer than BH).

Method	FDR	Power	Time
Knockoff	0.128	0.205	19.744s
Benjamini-Hochberg	0.100	0.167	1.776s

Table 1: Simulation results with high-dimensional data

In a similar setting, we generate the data once, and perform knockoff 100 times. Because BH has no randomness setting, the results will always be the same over 100 replications, we only performed BH once. In this simulation, model-X knockoff has slightly better power, and has a FDR better than BH. However, knockoff took many times longer than BH which only performed once.

Method	FDR	Power
Knockoff	0.089	0.383
Benjamini-Hochberg	0.222	0.350

Table 2: Another simulation results with high-dimensional data

3.2 Robustness to non-Gaussian distribution

Approximate MX knockoffs are generated from Gaussian distributions. Our question was how robust these methods are when the underlying distribution of the predictor variable is not Gaussian. We simulated 1,000 observations and 400 predictors in which 40 have non-zero coefficients towards the dependent variable. We constructed the predictor variables i.i.d. from Gaussian, exponential, Poisson, logistic and Cauchy distribution. The first distributions are from exponential family and the latter two are not. Poisson distribution is a discrete random variable, and Cauchy distribution is heavy tailed and does not have any moments. The distributions are all scaled to have unit variance except for the Cauchy distribution. For each distribution, we perform knockoff with three different settings: ASDP, SDP, and equi-correlated. BH was also applied for comparison. The FDR and power presented in Table 3, 4, 5 are average over 100 repetitions.

In Table 3, we considered a linear regression model $Y = X\beta + \epsilon,\ \epsilon \sim N(0,1)$. In this case, the p-values from BH are exact which makes it a suitable choice to control the FDR. We can see that both BH and knockoff methods do well. We speculated that the power of SDP knockoff to be the largest, followed by ASDP and equi-correlated. But that did not seem to be the case.

In Table 4, we have a logistic regression model $\mathbf{P}(Y_i = 1|X_i) = \frac{\exp(X_i^T\beta)}{1+\exp(X_i^T\beta)}$. The p-values are obtained from the asymptotic theory of maximum likelihood estimator and we can clearly see that the BH method cannot control the FDR at level $\alpha = 0.1$. Among knockoff methods, only ASDP was able to control the FDR at 0.1. SDP and equi-correlated methods had a slightly higher FDR at 0.15 approximately. However, the power of the knockoff methods were generally very low because our simulation was based on low amplitude. we have: From the table above. We were surprised to see that the power of SDP and equi-correlated knockoff for Cauchy distribution was much higher compared to powers in different scenario. This result seems to suggest that the success of knockoff may depend more on the variability (information) contained in X, rather than the exact distribution of X.

	Distribution	BH	Knockoff		
			ASDP	SDP	Equi-correlated
FDR	Gaussian	0.089	0.085	0.108	0.104
	Exponential	0.073	0.077	0.110	0.107
	Poisson	0.085	0.086	0.105	0.114
$\alpha = 0.1$	Logistic	0.086	0.082	0.118	0.128
	Cauchy	0.087	0.013	0.072	0.070
	Gaussian	0.794	0.812	0.929	0.921
Power	Exponential	0.798	0.815	0.928	0.921
	Poisson	0.790	0.818	0.923	0.930
$\mathbb{E}\left[\frac{\lvert\hat{S}\cap\mathcal{H}_0^c\rvert}{\lvert\mathcal{H}_0^c\rvert}\right]$	Logistic	0.805	0.790	0.929	0.932
	Cauchy	1.000	0.711	0.958	0.959

Table 3: Linear regression (exact p-values)

	Distribution	BH	Knockoff		
			ASDP	SDP	Equi-correlated
FDR	Gaussian	0.694	0.086	0.138	0.148
	Exponential	0.598	0.109	0.167	0.151
	Poisson	0.643	0.112	0.175	0.147
$\alpha = 0.1$	Logistic	0.691	0.086	0.154	0.155
	Cauchy	0.720	0.011	0.021	0.015
	Gaussian	0.470	0.097	0.150	0.151
Power	Exponential	0.464	0.082	0.124	0.121
	Poisson	0.494	0.083	0.126	0.125
$\mathbb{E}\left[\frac{\lvert\hat{S}\cap\mathcal{H}_0^c\rvert}{\lvert\mathcal{H}_0^c\rvert}\right]$	Logistic	0.478	0.099	0.170	0.150
	Cauchy	0.800	0.166	0.334	0.308

Table 4: Logistic regression (asymptotic p-values)

3.3 Robustness to non-i.i.d. data

In the previous section, the predictor variables were generated i.i.d. We also wanted to see if we can obtain the same results when there is a correlation structure between the predictors. Except for Gaussian distribution, generating a structured multivariate distribution is not straight-forward. Here, we used the fact that the sum of independent Gaussian, Gamma, and Poisson distributions still has the same distribution respectively. Namely, we created generated the predictors as

$$X_i = Z_0 + Z_i, \quad Z_0 \text{ and } Z_1, \dots, Z_p \sim \text{ i.i.d. } \{\text{Gaussian, Gamma, Poisson}\},$$

for an equi-correlated covariance

$$\mathrm{Cov}(X) = 2 \begin{pmatrix} 1 & .5 & \cdots & .5 \\ .5 & 1 & \cdots & .5 \\ \vdots & \ddots & \ddots & \vdots \\ .5 & \cdots & .5 & 1 \end{pmatrix}$$

The outcome of this method is contained in Table 5. It is not much different from Table 3 controlling the FDR at 0.1 and having power approximately 0.8.

	Distribution	BH	Knockoff		
			ASDP	SDP	Equi-correlated
FDR	Gaussian	0.094	0.095	0.074	0.093
	Gamma	0.088	0.081	0.099	0.090
	Poisson	0.092	0.088	0.089	0.097
Power	Gaussian	0.813	0.836	0.841	0.852
	Gamma	0.796	0.825	0.858	0.841
	Poisson	0.803	0.824	0.845	0.846

Table 5: Linear regression (exact p-values)

3.4 Impact of Amplitude

When we generate our data, we generally set our amplitude to 5. Might sound self-evident, if the signal is stronger, our knockoff can capture it more easily. We simulate 1000 observations with

400 predictors, in which 60 are important. We calculate the FDR and power by taking average over 3 repetitions on each amplitude.

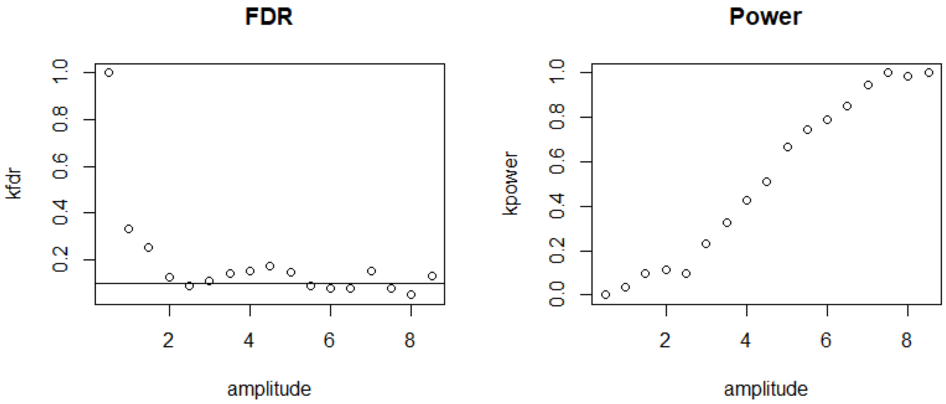


Figure 1: FDR and power on different amplitude

From Figure 1, we can see that FDR is somewhat controlled after amplitude 2, and power appears to have a logistic growth with the increase of amplitude. Power reaches and stays at 1 after amplitude of 7.5. This proves the common sense: stronger signals are easier to catch. FDR and power appear to wiggle in the same pattern, because we generally have more false discoveries when we have more true discoveries. When we scale up noise as well, neither FDR and power change greatly.

3.5 Divide-and-conquer algorithm on knockoff

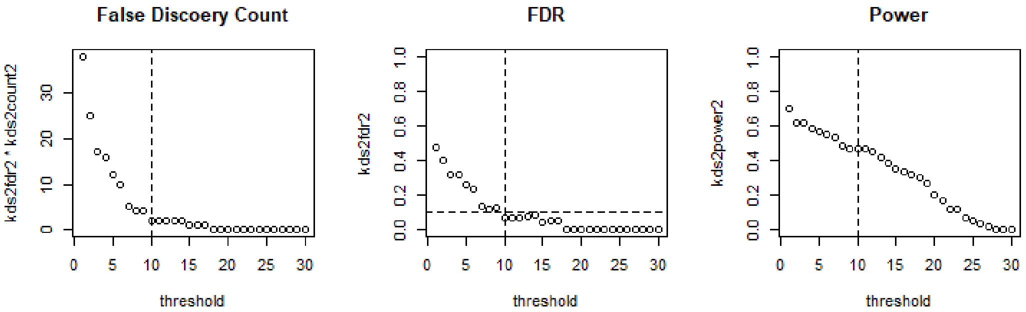
Knockoff are often time used on high-dimensional data, and with the size of data getting larger, it gets harder to construct knockoff. Running time is not the main issue. Storage, on the other hand, is the real problem. When over ten thousands or even millions of features are considered, the size of vector is generally measured in GB and cannot be held in a normal computer's RAM. Thus, we look out for divide-and-conquer (DAC) techniques which can reduce the burden of RAM and store the data in disk.

Unlike the usual DAC as outlined by Battey et al. (2018), we divide features instead of the observations. By assumption, the distribution of features should be i.i.d. and dividing features will not lead to problems. When we split the data, we get higher FDR and lower power because of the limitation of the data size. We can solve this problem by doing, say B repetitions, and combine the selected variables. Instead of deciding which variables are selected in each repetitions, we choose variables after all repetitions. For a set of predictor variables $X_1, ...X_p$ where k of them are relevant features, we do the following step:

1. For b in 1:B:
   (a) (Randomly) split $X_1, ...X_p$ into r groups.
   (b) Construct knockoff on each groups.
   (c) Select variables $\hat{S}_{b1}, ..., \hat{S}_{br}$ as described in 2.1.2 for each groups
   (d) Take all the selected variables as $\hat{S}_b = \hat{S}_{b1} \cup ... \cup \hat{S}_{br}$
2. Take the final selected variables as $\hat{S} = \{X_i : \#\{b : X_i \in \hat{S}_b\} \geq t\}$ for some threshold t.

The average weighted FDR among all thresholds, calculated as $\#FP_{1:B}/(\#FP_{1:B} + \#TP_{1:B})$, is the same as the expected weighted FDR we calculate within each repetition because for a feature appears s times, it will be included in s sets with threshold $1 : s$. The average power among all thresholds is also the same as the expected power within each repetition. Most of the features and false discoveries only appear once or twice, empirically, a threshold of one third of the total repetitions can control the FDR at the desired rate as we target in each split(0.1), and the corresponding power, which appears to be decreasing logistically, will be higher than the expected power we have in each split. The exact threshold still needs further investigation.

In a similar setting as before, we use random split for the simulated data. We have 500 observations, 1000 predictors in which 60 are important features with amplitude of 5.



False discovery count and power decrease step by step, and FDR decreases in general, and slightly increases within each step due to less discovery made.

The threshold that controls FDR at 0.1 is 10, about one third, and the corresponding power is 0.467, not as good as the knockoff construct on the entire data 0.786. But when the data get larger, it might not be possible to have one on the entire data.

According to DAC, it is theoretically better to split randomly, but when we are at the genetic settings, splitting the data with some specific knowledge in advance would help improving the results. When we do not split the data randomly, the number of important features might be different from one split to another. In this simulation, we have similar setting as before, but we split the features in advance. To observe the behavior of knockoff under different density, we set the important features in each split to be 16, 12, 12, 12, and 8 out of 200. The middle three splits have the same density as the entire data. We construct knockoff on each of the splits and calculate FDR and power by taking average over 30 repetitions. As a comparison, we apply knockoff on the entire data as well. We can see that for the middle three splits, the FDR is controlled, and that of the first split and the entire data are a little above 0.1. FDR of the fifth split is even larger than the power. Actually, the power for all the splits are lower than that of applied on the entire data. The average FDR and power, weighted by the number of selected and the number of important features, is 0.113 and 0.317. The reason for the low power is that in each split, knockoff selects way less variables than it should. In the fifth split, one false feature was repeatedly selected, and because on average there were only 2 feature selected, the FDR is larger than we expected.

	S1	S2	S3	S4	S5	Entire
FDR	0.138	0.078	0.051	0.094	0.319	0.125
Power	0.308	0.2	0.411	0.472	0.137	0.77
No. of non-zero	16	12	12	12	8	60
No. of selected	6.23	3.06	5.56	6.9	2.1	53.16

Table 6: Divide-and-Conquer Results

With the previous data and selected variables in 30 repetitions, we combine all the selected and record the time of appearance. A total of 72 variables are selected at least once, and feature 22 appears most frequently, 28 out of 30. We set threshold at several level and calculate FDR and power.

We can see that with the threshold at 11 or 12 (about one third), the FDR is controlled around

Threshold	3	6	11	12
FDR	0.35	0.28	0.12	0.08
Power	0.58	0.5	0.36	0.35

Table 7: Divide-and-Conquer Threshold

0.1, and the power is better than the weighted average of the previous method which is 0.317.

4 Genetic data

4.1 Introduction to GWAS

Genome-wide association studies (GWAS) aim to find associations between genotypes and some phenotype of interest. The procedures generally include collecting a DNA sample from a large pool of subjects for whom there are also information on the phenotype. Phenotypes may be dichotomous or categorical, such as the presence or absence of some disease, or continuous, such as height, weight, or IQ scores. The focal genetic component in GWAS is single-nucleotide polymorphisms (SNPs), which refer to the variant at each nucleotide in the genome. For instance, the nucleotide cytosine (C) might be replaced with thymine (T) at a certain location in the genome. These locations, at which there might be variation across individuals, are named and documented in a large international database for reference across research teams (dnSNP; Sherry et al. (1999)). GWAS are a powerful tool to find potential genetic markers for certain phenotypes, and it far surpassed its predecessor - the candidate gene model, in which only a subset of pre-specified genes are studied and thus highly prone to false positive errors Tabor et al. (2002). However, there are still several important considerations and assumptions made by this model. First, this is a high-dimensional problem. There are usually many more features (SNPs) than there are observations because GWAS aim to cover the entire genome. In order to efficiently run GWAS, the standard approach is to perform many simple linear regression analysis, one for each feature. Second, this solution creates a multiple testing problem. When upward of 1 million tests are run, we need to adjust α to account for family-wise error rate. As a result, GWAS conventionally adjust the threshold to $\alpha_{\text{adj}} = 5 \times 10^{-8}$. GWAS results are thus often displayed in a Manhattan plot of all individual p-values extracted from simple linear regression for each SNP, and those lower than

α_{adj} are considered significant hits.

An important assumption behind GWAS is the common disease - common variant hypothesis, as described by Pritchard and Cox (2002). Common and complex diseases are best understood in contrast with Mendelian diseases, which are usually very rare and a result of mutations at a single location in the genome. On the other hand, most diseases or traits that we routinely encounter are much more complex. There is no one mutation, for instance, that could fully explain the variability in height, or account for the onset of depression. Past research using the candidate gene approach have largely failed for these phenotypes precisely due to a misunderstanding of the complexity of the genetic architecture behind their expression. If dichotomous, these phenotypes (e.g., presence of heart diseases, depression, schizophrenia) are much more common than Mendelian diseases and they are a result of an aggregation of numerous mutations across the genome. In addition, these mutations themselves are considered very common in the population ($> 1\%$), which is why we usually refer to them as "variants" instead. Data on SNPs capture these variants, and it is hypothesized that given sufficient genotyped data across the genome and sufficient observations in the population, we can identify a collection of many SNPs that, in aggregate, can account for variability in these complex phenotypes. This means that the effect of each individual SNP is very small, thus resulting in a low-power setting.

4.2 Obtaining and cleaning data

For the current project, we used the open dataset retrieved from openSNP (Greshake et al. (2014)). This contains user-uploaded data where individuals, upon receiving their genotyped results from companies such as 23andme, would upload the text file of their genotypes and answer some additional questions on a list of phenotypes. Here, we use the phenotype height as the predicted variable in our model.

Data were obtained in their raw forms as individual text files of genotypes for each person and a separate text file of all phenotypes across all people. We only selected genotyped data of individuals for whom there was available height data, which resulted in a total sample size of $N = 1,282$. Due to size constraints, a subset $N = 250$ was randomly selected for analysis. After further cleaning due to corrupted files or questionable data (e.g., having only one allele when two should be present), the final focal sample size of $N = 224$ was used for all further analyses. In addition, only documented SNPs and those not from the sex chromosome were considered for simplicity. Across all individuals, this resulted in a total of $p > 2,000,000$ features. However, there were high missingness due to non-overlapping data across people. We then created two focal datasets:

1. Only non-missing genotypes: $N = 224, p = 6,019$
2. At most 5%-missing genotypes: $N = 224, p = 99,979$

Each SNP is coded as numeric $X \in \{1, 2, 3\}$ which corresponds to three potential values: two types of homozygosity (e.g., CC or TT) and heterozygosity (e.g., CT). Height is measured in centimeters, $M = 173.0, SD = 9.87$ and appears relative normally distributed in Figure 2.

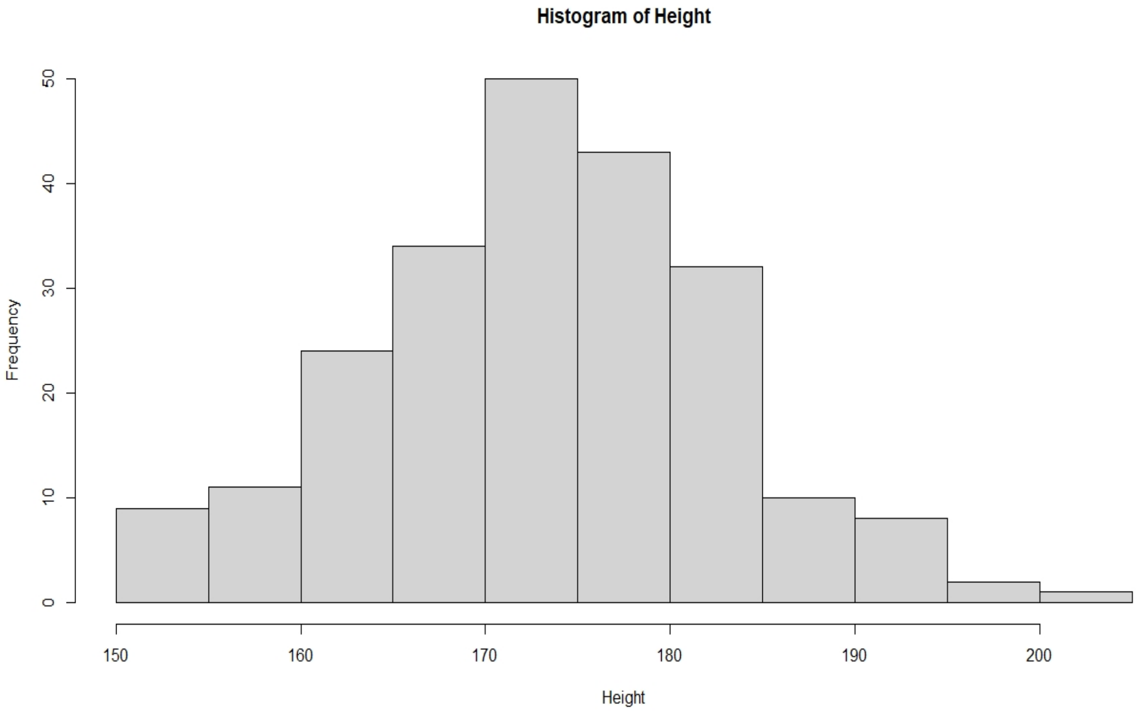


Figure 2: FDR and power on different amplitude

4.3 Data analysis

Three approaches were compared for both datasets:

a) Model-X knockoff: $create.second\_order()$ with $method = asdp$ for knockoff construction with FDR $= 0.1$

b) Benjamini-Hochberg with ridge regression: $linearRidge()$ and $p.adjust()$ to fit model with all covariates and extracted corrected p-value for FDR $= 0.1$

c) Bonferroni adjusted $\alpha_{\text{adj}} = 5 \times 10^{-8}$: Height is regressed on each individual SNP and only p-values lower than α_{adj} were selected

Unfortunately, there was no hit for any procedure. In particular, BH with ridge regression failed to run in either dataset due to memory limitations. All extracted p-values using Bonferroni were larger than α_{adj} in both datasets. MX knockoff took significantly longer than Bonferroni: 8.34 minutes vs. 8.26 seconds for the smaller non-missing dataset. Although this was disappointing, the results were not surprising due to our low-power setting. As previously discussed, one main assumption for GWAS is the common disease-common variant model, which specifies that there are numerous SNPs that might contribute to the variance in the phenotype, each with a very small effect. This is why GWAS generally use very large sample upwards of > 1 million individuals. In comparison, our sample of $N = 224$ or even the full sample of $N = 1,282$ is very lacking. As demonstrated in section 3.4, both FDR and power seemed to suffer at amplitude lower than 2 for MX knockoff.

5 Discussions

5.1 Simulation studies

We conducted simulation studies to examine performance of MX knockoff and compared it to Benjamini-Hochberg method. We examined robustness across non-Gaussian distributions, performance in high-dimensional settings, the effect of amplitudes, and potential application of the divide-and-conquer technique.

In high-dimensional setting ($n \ll p$), many traditional methods of variable selections would fail. We thus compared MX knockoff to a modified Benjamini-Hochberg method with ridge regression instead of linear regression to control FDR at 10%. Results showed that MX knockoff had slightly better power yet slightly worse FDR control than BH. However, MX knockoff was much more costly, with run-time being 11 times longer than BH per iteration. As a result, we did not find strong evidence in favor of using MX knockoff in this setting.

Our simulation showed that approximate knockoff works well under non-Gaussian distributions and also under different covariance structures. However, we had mixed results for the power of different knockoff methods (ASDP, SDP, equi-correlated), which requires further theoretical and empirical studies.

Amplitude of simulated data has a great impact on the FDR and power. FDR starts to be stable after a certain amplitude. Power appears to have a logistic growth as amplitude. However, when noise and signal both get stronger, FDR and power will not change.

When there are too many features, we can apply DAC on the features. However, the FDR increases and power decreases when we split our data. We can use repetitions to recover some of the loss of FDR and power. The choice of threshold still needs more investigation.

5.2 Genetic data analysis

In short, neither MX knockoff or other traditional methods for variable selection succeeded in identifying important features in the genetic dataset. Our project suffered from logistical constraints, such as storage limitations, that made full analyses of the data impossible on local machines. Nonetheless, even with the full dataset, MX knockoff would have likely struggled because we did not have the usual GWAS power of hundreds of thousands to millions of observations. Genetic data are difficult to obtain, and although the open dataset allowed us to explore the methods, it had several issues that made analyses difficult. For instance, sex data is very lacking, with only 42 women and 61 men (of the total 224 observations), and height systematically differs between the sexes. Using available data in our sample, men and women did differ significantly in height ($t_{(101)} = -6.56, p < .001$). However, when we modified the Bonferroni procedure to include the covariate sex in multiple regression models (instead of simple linear regression with just the SNP value), there was still no significant hit for either dataset.

Further, the current coding scheme for our predictors means that the joint distribution of X is not multivariate Gaussian. Although previous simulation studies in section 3.2 showed good robustness of MX knockoff for non-Gaussian distribution, there exists a method to construct knockoff using hidden Markov chain that are specific for SNPs data as described by Sesia et al.

(2019). However, due to time constraint and computing limitations, we did not include this method in the current project. Our analyses of genetic data using the openSNP archive thus did not successfully demonstrate the added utility of MX knockoff, particularly not compared to the traditional Bonferroni technique due to its significant computation cost. Future work might benefit from using hidden Markov chain to more accurately model the features and applying parallel computing or DAC algorithm to speed up computation.

6 Author contributions

- Linh Nguyen: simulation study of high-dimensional data, genetic data analysis
- Sungmin Park: simulation of approximate knockoffs to test robustness against non-Gaussian predictors.
- Hexuan Zhang: simulation of high-dimensional data, amplitude impact, DAC on knockoff.

SUPPLEMENTAL MATERIALS

Analysis code: `https://github.com/nguyenllpsych/8053-knockoff`

Non-missing data: `https://github.com/nguyenllpsych/8053-knockoff/blob/main/data_clean.RData`

5%-missing data: `https://github.com/nguyenllpsych/8053-knockoff/blob/main/data_5.RData`

References

Barber, R. F. and Candès, E. J. (2015). Controlling the false discovery rate via knockoffs. *The Annals of Statistics*, 43(5):2055–2085.

Bates, S., Candès, E., Janson, L., and Wang, W. (2021). Metropolized knockoff sampling. *Journal of the American Statistical Association*, 116(535):1413–1427.

Battey, H., Fan, J., Liu, H., Lu, J., and Zhu, Z. (2018). Distributed testing and estimation under sparse high dimensional models. *Annals of statistics*, 46(3):1352.

Candes, 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.

Greshake, B., Bayer, P. E., Rausch, H., and Reda, J. (2014). opensnp–a crowdsourced web resource for personal genomics. *PloS one*, 9(3):e89204.

Pritchard, J. K. and Cox, N. J. (2002). The allelic architecture of human disease genes: common disease–common variant… or not? *Human molecular genetics*, 11(20):2417–2423.

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

Sherry, S. T., Ward, M., and Sirotkin, K. (1999). dbsnp—database for single nucleotide polymorphisms and other classes of minor genetic variation. *Genome research*, 9(8):677–679.

Tabor, H. K., Risch, N. J., and Myers, R. M. (2002). Candidate-gene approaches for studying complex genetic traits: practical considerations. *Nature Reviews Genetics*, 3(5):391–397.