Swiss Federal Institute of Technology Zurich

Seminar for Statistics

Department of Mathematics

Master Thesis Spring 2021

Zipei Geng

Nonparametric Variable Selection under Latent Confounding

Submission Date: October 2nd 2021

Co-supervisor: Dr. Mona Azadkia & Dr. Armeen Taeb
Supervisor: Prof. Dr. Peter Bühlmann

Preface

ACKNOWLEDGEMENT

This is a section for a short note of thanks to Dr. Mona Azadkia and Dr. Armeen Taeb who inspired much of the enthusiasm and provide great help and thoughts to make me continue on and finalize the thesis. Besides, I want to express my gratitude to Prof. Dr. Peter Bühlmann who provides me with the enlightening research topics of the thesis. Finally, I would like to thank my parents for their unconditional love and encouragements during my studies in Switzerland.

Abstract

Variable selection under latent confounding is a classic problem in causal inference. Recently, Chatterjee (2020) proposed a rank correlation and Azadkia and Chatterjee (2021) laid out an simple but effective approach based on Chatterjee's rank correlation to define a new measure of conditional dependence and a new algorithm for variable selection. This fully non-parametric approach is based on rankings and the method of nearest neighbors. In this thesis, we incorporated the measure, namely *Conditional Dependence Coefficient*, and the algorithm, namely *Feature Ordering by Conditional Independence* (FOCI), to conduct variable selection using generated data with latent confounding. Upon several exploratory simulation experiments, we proposed to use proper confounder estimation method together with FOCI to not only provides a view of the empirical distribution of latent confounders and computationally control the false discovery proportion when selecting the signal variables, but also theoretically remove the spurious dependence between the non-signal predictors and the response. In the light of this founding, we developed a new FOCI function to include the estimated confounders always in the conditioning set which is different from the original function, and a resampling scheme with a heuristic threshold.

As for the confounder estimation, we proposed two methods which are *principal component analysis* (PCA) and *variational autoencoder* (VAE). We then theoretically justified the use of PCA in the sense of *relative information loss*. Our simulation experiment results have shown that FOCI with VAE will be better and more efficient to nonlinear relationships between the predictors and confounders than using FOCI with PCA, given sufficient sample size. While PCA is efficient when the hidden confounders have linear relationships with predictors, it also performs well when we only have a relatively small set of data, no matter the relationships between predictors and confounders are linear or not. In the context of magnitudes of latent confounders and the signals, we found that if signals are dominated by confounders, this will result in the FOCI's failure to select Markov blankets. We also developed comparison between FOCI family and classic methods on real dataset, which had shown competitive MSPE and generalization ability when using FOCI family. Finally, several future research directions are given at the end of the thesis.

Keywords: nonparametric variable selection, confounder, false discovery proportion, principal component analysis, variational autoencoders

Contents

List of Figures

List of Tables

Notation

This list describes several notations that will be used later within the body of the thesis.

1_A	Indicator function of a set A
$\mathcal{N}(\mu, \sigma^2)$	Gaussian distribution with mean μ and variance σ^2
$a.s.$	Almost surely
$\mathcal{U}(a, b)$	Continuous uniform distribution with bounds a and b
I_n	Identity matrix of size $n \times n$
R^2	Coefficient of determination

Chapter 1

Introduction

It is clear that causal inference and variable selection are very basic but crucial counterparts of statistical learning. Their applications lie in a great variety of areas including health, social, and behavioral studies. There are quite some causal questions in daily life which require some knowledge of the data-generating process such as the efficacy of COVID-19 *mRNA* vaccine on different kinds of population, e.g. male and female. However, they cannot be fully solved from the given data alone. In other words, one normally cannot know the underlying data generating distribution. As the result, the statistical research community has developed many causal inference theories (Shanmugam, 2018).

As for the variable selection, it is an old problem in statistics. Variable selection is widely used for prediction and forecasting in the specific disciplines that attempts to determine the relationship between one dependent variable and a series of other variables (Hastie, Tibshirani, and Friedman, 2009). We often call these dependent variables *signals*. From linear regression to LASSO (Tibshirani, 1996), there are a variety of models to use. But all of them are model-based. For example, linear regression assumes that the data follows a pattern of $Y = X\beta + \varepsilon$, where β is the regression coefficients and ε is the noise term with some distributions. The shortcoming of model-based method is conspicuous since one should have lots of assumption in prior which could not be reasonable in reality. In other words, model-based variable selection often loses generality.

From what has been discussed, it naturally comes out a question. Is that possible to use model-free methods to implement variable selections under hidden counfoundings and still generates competitive results? Recently, Chatterjee (2020) proposed a new measure of statistical association which as simple as the classical coefficients such as Pearson's correlation (Benesty, Chen, Huang, and Cohen, 2009) and consistently estimates a simple and interpretable measure of the degree of dependence. It replaces the traditional measure like Spearman's ρ (Myers and Sirois, 2006) and Kendall's τ (Abdi, 2007), since they are not that very powerful for all cases, especially for detecting associations that are not monotonic, or even in the complete absence of noise.

As for testing the conditional dependence, which is basically checking whether $P(X, Y|Z) = P(X|Z)P(Y|Z)$. There are several method proposed recent years. For example, Linton and Gozalo (1996) proposed a method based on conditional cumulative distribution estimation. Poczos and Schneider (2012) proposed to use mutual information estimation. Also, couplas (Dette, Siburg, and Stoimenov, 2013) are also widely used for finding dependence of time series datasets. For the new measure of correlation proposed by Chatterjee (2020), it is then extended by Azadkia and Chatterjee (2021) by defining a new measure of *conditional dependence* called *Conditional Dependence Coefficient (CODEC)* and accordingly designed an algorithm *Feature Ordering by Conditional Independence (FOCI)* for variable selection, which is suitable to deal with the datasets with causal relationship.

In this thesis, we conducted an exploratory research with numerical simulations using pseudo-random sampling scheme (Morris, White, and Crowther, 2019). We will first introduce this brand new measure of conditional dependence, *CODEC*, and go over the most basic concepts in causal inference which is *confounding* and then introducing how FOCI selects the variables by feature ordering. Then, we will use this newly defined algorithm FOCI based on CODEC to select variables, i.e. the Markov blanket, that are deciding the response. To recapitalize, we will explore the most plain case of confounding, which is given Y as response, $\mathbf{X} = (X_1, ..., X_p)$ as signal and $\mathbf{Z} = (Z_1, ..., Z_q)$ as potential confounders, we want a subset of $\mathbf{X}$ called $(X_j)_{j \in S}$, where S refers to sufficient if Y and $(X_j)_{j \notin S}$ are conditionally independent with Y given $(X_j)_{j \in S}$ and design different application scenarios to understand the behaviour of FOCI. Among all the experiments, we mainly discussed two different kinds of confounding relationship between $\mathbf{X}$ and $\mathbf{Z}$, which is linear or nonlinear. For linear case, we used the *Principal Component Analysis (PCA)* (Bro and K. Smilde, 2014) to estimate the hidden confounders; for nonlinear case, we naturally think about using variational autoencoders (Kingma and Welling, 2014) to find the latent space in a probabilistic way. Both methods are not only give us a view of empirical distribution of latent confounders, but also remove the spurious association between the response and the non-signal predictors which providing aids for the FOCI algorithm to select correct signals and control on the false discovery proportion compared to the case that we do not estimate the latent confounders and directly run FOCI algorithm. Furthermore, we gave out the theoretical justification of using principle components as estimator of latent confounders under some conditions. In addition, we investigated on the magnitudes difference between the generated signals and confounders, it has then been shown by simulations that the magnitude difference matters while selecting the Markov blankets. Last but not the least, we implemented our methods on a real dataset and compare with some classic feature selection models. The results turned out to be surprising since in the sense of MSPE and generalization ability, FOCI with latent confounder estimators is competitive with the classic models. Finally, we discussed the results and proposed some future research directions.

Paper organization.

i.) Chapter 2 introduces statistical concepts and theorems.

ii.) Chapter 3 reviews the conditional dependence estimator CODEC & feature selection method FOCI proposed by Azadkia and Chatterjee (2021).

iii.) Chapter 4 formalizes the problems we want to solve and introduces the simulation experiments and corresponding results.

iv.) Chapter 5 implemented our methods and classic models of feature selection on a real dataset.

v.) Chapter 6 summarizes the experimental results and makes the conclusion.

vi.) Chapter 7 provides several thoughts and improvements for future works.

Chapter 2

Mathematical Background

2.1 Graphical models

The following introductions are taken from Clark (2018) with some adjustments.

A graphical model can be seen as a mathematical or statistical construct to connect the vertices via edges. When pertaining to statistical models, the vertices will normally represent variables of interest (which is generated by some probability distribution) in our dataset, and edges will specify the causal relationships among them. Visually they are depicted in the style of the following example.

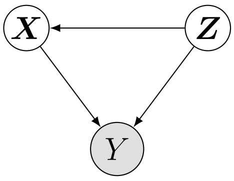

Figure 2.1: A typical causal graph with **confounding**: Random variables **X** and the confounders **Z** both contributes to Y. Besides, the confounders **Z** also decides **X**.

Any statistical model you have used can be expressed as a graphical model. The above graph with vertices **X**, Y, and **Z** could represent a model in which **X** and **Z** predict Y and **X** is also dependent on **Z**.

A key idea of a graphical model is that of conditional independence and Bayesian network.

Definition 2.1.1 (Conditional independence)**.** *Two random variables A and B are conditionally independent given a random variable C iff. they are independent in their conditional probability distribution given C, i.e.*

$$(X \perp\!\!\!\perp Y|Z) \iff F_{X,Y|Z=z}(x,y) = F_{X|Z=z}(x)F_{Y|Z=z}(y), \ \ \forall x, y, z$$

Definition 2.1.2 (Bayesian network)**.** *Let $G = (V, E)$, where V stands for number set of vertices, E stands for the set of directed edges and p be the distribution of X_V. The pair*

(G, p) is a Bayesian network if

$$p(x_V) = \prod_{i \in V} p(x_i | x_{pa(i)})$$

*Note that **pa** here means the parent nodes.*

Definition 2.1.3 (Confounding)**.** *As shown in the Figure 2.1, random variables $\mathbf{Z}$ contributes to predictors $\mathbf{X}$ and the response Y simultaneously. In the sense of causal inference, we say that this phenomenon will cause Y and $\mathbf{X}$ spuriously associated. In other words, $\mathbf{Z}$ is the confounder.*

2.2 Variable selection

In this part, we will define the linear and nonlinear relationship between two variable sets $\mathbf{A}$ and $\mathbf{B}$ (Friedman, 2017).

Definition 2.2.1. *Given $\mathbf{A} = (A_1, ..., A_k)$ and $\mathbf{B} = (B_1, ..., B_j)$, and $\mathbf{A}$ is dependent of $\mathbf{B}$, we can write*

$$\mathbf{A} = f(\mathbf{B}) + \varepsilon,$$

where ε is random noise matrix. We say that $\mathbf{A}$ has linear relationship with $\mathbf{B}$ if $f(\cdot)$ is a linear map that maps from the space of $\mathbf{A}$, i.e. $\mathcal{A}$ to the space of $\mathbf{B}$, i.e. $\mathcal{B}$. Similarly, the nonlinear relationship relies on the nonlinear function $g(\cdot)$ which maps from $\mathcal{A}$ to $\mathcal{B}$.

In this thesis, our main focus is to select variables that contributes to the response, *i.e.* selecting signal variables (Markov blanket $\mathcal{S}$) from the predictors $\mathbf{X}$ where there are latent confounders that may mislead us to select non-signals predictors, *i.e.* $\mathbf{X} \backslash \mathcal{S}$.

After introducing the above concepts, we are going to introduce the definition of principal component analysis.

2.3 Principal Component Analysis

Principal Component Analysis, or simply PCA, is a statistical procedure concerned with elucidating the covariance structure of a set of variables (Gillies, 2018). In particular it allows us to identify the principal directions in which the data varies.

If the variation in the data is caused by some relationship then PCA gives us a way of reducing the dimensionality of a data set. That is a very normal method of dimension reduction which returns a low-dimensional representation of the original dataset.

Assume we have a $n \times p$ data matrix X, where p stands for its number of initial variables. The PCA process can be divided into the following steps:

i.) Compute each column's sample mean and shift them to zero, then scale each column to make them have the same standard deviation. (Note: This procedure is necessary in order to find the correct principal components, since PCA is a method maximizing the variance.)

ii.) Calculating the $p \times p$ data covariance matrix $\Sigma = (E[X_i, X_j])_{1 \leq i,j \leq p}$ and do the eigendecomposition, which is finding an orthogonal matrix Φ whose columns are eigenvectors of Σ and a diagonal matrix A whose diagonal elements are eigenvalues of Σ, *i.e.*,

$$\Phi^T \Sigma \Phi = A$$

iii.) Assume we want the initial p dimensions reduction to k dimensions. We then order the eigenvalues in the diagonal matrix A from largest to smallest and select the first k eigenvalues.

iv.) Select the corresponding eigenvetors and form a $p \times k$ matrix P. Let $Y = XP$, which is our final result.

To summarize, this process is equivalent to finding a new axis system in which the covariance matrix is diagonal. The eigenvector with the largest eigenvalue is the direction of greatest variation, the one with the second largest eigenvalue is of the next highest variation and so on.

2.4 Variational Autoencoders

In this section, we are going to introduce the ideas behind the variational autoencoder and how to use it practically (Kingma and Welling, 2014).

VAE is a latent variable model. Such model relies on the idea that the data generated by a model can be parametrized by some latent variables that will generate some specific characteristics of the given data.

One of the most crucial idea behind VAE is that instead of trying to construct a space of latent variables explicitly. We aim to sample in order to find samples that could actually generate proper outputs which are as close as possible to our distribution.

For that reason, we constructed an encoder-decoder-like network which can be split in two parts:

- The encoder learns to generate a distribution $Q_\Phi(z|x)$ depending on input samples X from which we can sample a latent variable Z that is highly likely to generate X samples. In other words, we learn a set of parameters Φ which rules the distribution Q.
- The decoder part learns to generate the output which belongs to the real data distribution given latent variable Z as input. In other words, we learn a set of parameters Θ that generates a function $P_\Theta(x|z)$ that maps Z to X.

The VAE objective is to maximizing the evidence lower bound (ELBO)

$$\mathcal{L}(p_\theta, q_\Phi; x) = E_{q_\Phi(z|x)}[\log p_\theta(x, z) - \log q_\Phi(z|x)]$$

over the space of q_Φ. In order to further explain the ELBO, we now introduce the definition of Kullback-Leibler divergence between two probability distributions.

Definition 2.4.1 (Kullback-Leibler divergence (KL divergence))**.** *Given two random variables P and Q with probability distribution p and q defined on the same probability space. The KL divergence is given by*

$$KL(P||Q) = \int p(x)\log(\frac{p(x)}{q(x)})dx$$

Using Definition 2.4.1, we can measure the differences between two probability distributions. Thus we have that ELBO can be written as the equation

$$\mathcal{L}(p_\theta, q_\Phi; x) = -KL(q_\Phi(z|x)||p_\theta(z)) + \mathbb{E}_{q_\Phi(z|x)}[\log p_\theta(x|z)] \tag{2.4.1}$$

Our goal is to differentiate and maximize this ELBO w.r.t. Φ and θ. However, the gradient of the ELBO w.r.t. Φ cannot be calculated directly. We then are required to estimate of the ELBO in order to calculate the gradients. Usually, the KL divergence of Equation 2.4.1 can be integrated analytically, such that we only need to estimate the expected reconstruction error $\mathbb{E}_{q_\Phi(z|x)}[\log p_\theta(x|z)]$ by sampling. This leads to the estimation of

$$\mathcal{L}^E(p_\theta, q_\Phi) = -KL(q_\Phi(z|x)||p_\theta(z)) + \frac{1}{M}\sum_{i=1}^{M}(\log p_\theta(x|z^{(i)})) \tag{2.4.2}$$

Given N samples, we can construct the estimator of the ELBO based on mini-batches:

$$\mathcal{L}(p_\theta, q_\Phi; \mathbf{X}) = \mathcal{L}^M(p_\theta, q_\Phi; \mathbf{X}^M) = \frac{N}{M}\sum_{j=1}^{M}\mathcal{L}^E(p_\theta, q_\Phi; x^{(j)}) \tag{2.4.3}$$

In order to solve the problem we will hereby use an alternative method generating samples from $q_\Phi(z|x)$. In the context of this thesis, we assume z be a continuous Gaussian random variable where $Z \sim q_\Phi(Z|x) = N(\vec{\mu}, \sigma^2 I) = \vec{\mu} + \vec{\sigma} \otimes \vec{\varepsilon}$, where $\varepsilon \sim N(0,1)$ and $\otimes$ refers to element-wise multiplication. To specify, we use the learnt encoder to generate mean $\vec{\mu}(X)$ and standard deviation $\vec{\sigma}(X)$. Then construct the generated Z as $\vec{\mu}(X) + \vec{(}X) \otimes \vec{\varepsilon}$. This process is called reparametrization trick to allow us to do the backpropagation during the training process.

Based on above information, we can train a typical variational autoencoder with Gaussian prior. The illustration can be found in Figure 2.2.

All in all, we can use this method to discover the nonlinear relationship between the latent variables and the predictors, since we can use deep learning to summarize the nonlinearilty, which is not possible using PCA. Though kernel PCA is another interesting choice of discovering nonlinearities, it is very hard to choose what kind of kernel to include, since we know nothing about the underlying nonlinear relationships.

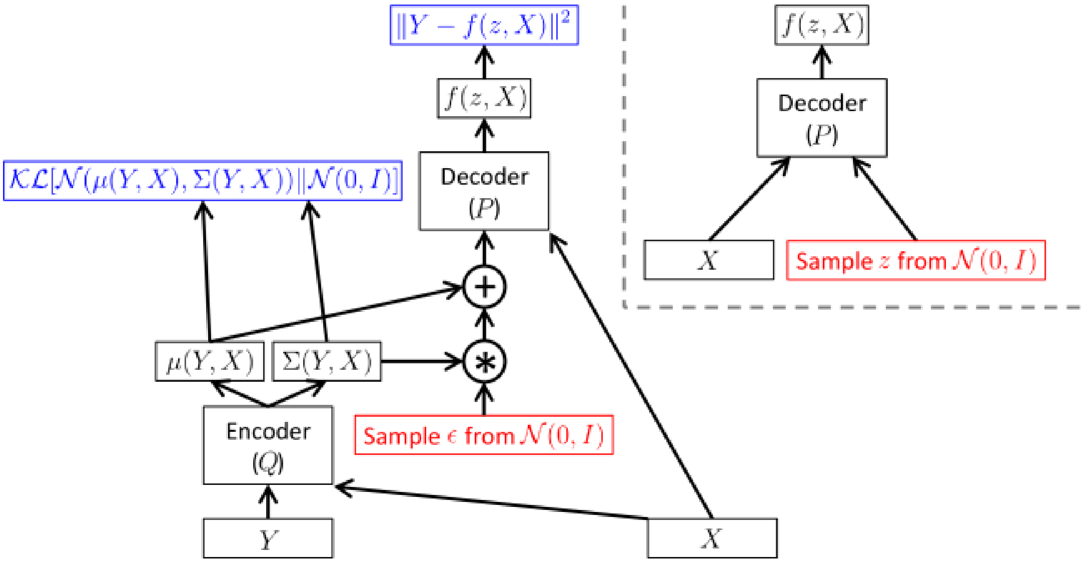
$\|Y - f(z,X)\|^2$
$f(z,X)$
Decoder (P)
$\mathcal{KL}[\mathcal{N}(\mu(Y,X),\Sigma(Y,X))\|\mathcal{N}(0,I)]$
$\mu(Y,X)$
$\Sigma(Y,X)$
Sample ϵ from $\mathcal{N}(0,I)$
Encoder (Q)
Y
X
$f(z,X)$
Decoder (P)
X
Sample z from $\mathcal{N}(0,I)$

Figure 2.2: Typical VAE architecture with standard normal prior and reparametrization trick (Sinitambirivoutin, 2020)

Chapter 3

Introduction to CODEC and FOCI

3.1 The Coefficient CODEC

3.1.1 Background

Suppose we are going to estimate the degree of conditional dependence between the random variable Y and random vector $\mathbf{X} = (X_1, ..., X_s)$ given random vector $\mathbf{Z} = (Z_1, ..., Z_t)$, which are all defined in the same probability space $\mathcal{X}$. To make this coefficient meaningful, we must have $s \geq 1$ and $t \geq 0$, which is saying that we must have at least one signal X to measure, and there can be no conditioning random variables. In that case, we are estimating the degree of dependence without any conditions.

Let μ be the probability measure of Y. It is proposed that the following quantity T is the estimation of degree of conditional dependence of Y and $\mathbf{X}$ given $\mathbf{Z}$:

$$
\begin{aligned}
T &= T(Y, \mathbf{X}|\mathbf{Z}) && (3.1.1)\\
&:= \frac{\int \mathbb{E}[\mathrm{Var}\,(\mathbb{P}(Y \geq v|\mathbf{X}, \mathbf{Z})|\mathbf{Z})]d\mu(v)}{\int \mathbb{E}[\mathrm{Var}\,(1_{\{Y \geq v\}}|\mathbf{Z})]d\mu(v)} && (3.1.2)
\end{aligned}
$$

It looks a bit complicated at first glance but its has natural explanation which is a nonlinear generalization of the partial R^2 statistic for measuring the proportion of variation in Y that is explained by $(\mathbf{X}, \mathbf{Z})$ but cannot be explained solely by $\mathbf{X}$ (Azadkia and Chatterjee, 2021).

Besides, we can also prove that this measure is well-defined, which is saying that no matter what v value is given, T will return either a valid value or an invalid (undefined) value with proper explanations. To emphasize, we are considering the case of

$$Y \overset{a.s.}{=} \text{some measurable function of } \mathbf{Z},$$

then $\mathbb{P}(Y \geq v|\mathbf{X}, \mathbf{Z}) = 1_{\{Y \geq v\}}$ for any v. Therefore in this case, $T = 1$ and this is a degeneration case so that we can ignore.

In addition, CODEC also have a very promising property that in nondegenerate cases, $0 \leq T \leq 1$. Moreover, $(T = 0) \iff$ (Y and $\mathbf{X}$ are conditionally independent given $\mathbf{Z}$), and $(T = 1) \iff$ (Y is almost surely equal to a measurable function of $\mathbf{X}$ given $\mathbf{Z}$).

3.1.2 Estimation of T

After defining the quantity T, a consistent estimator $\hat{T}$ of T is constructed by Azadkia and Chatterjee (2021), from which we can directly estimate using the real data. It is defined in the following way:

Considering we have a dataset of n ($n \geq 2$) independently and identically distributed copies of $(Y, \mathbf{X}, \mathbf{Z})$. For each i, let $N(i)$ be the index j *s.t.* Z_j is the nearest neighbor of Z_i *w.r.t.* Euclidean metric on $\mathbb{R}^s$, where ties are broken uniformly at random. Let $M(i)$ be the index j such that (X_j, Z_j) is the nearest neighbor of (X_i, Z_i) in $\mathbb{R}^{s+t}$, with ties broken uniformly at random.

Let R_i be the rank of Y_i, *i.e.*, the number of j *s.t.* $Y_j \leq Y_i$.

Assume we have n copies of samples, then comes the final result of $\hat{T}$:

$$\hat{T} = \hat{T}(Y, \mathbf{X}|\mathbf{Z}) \tag{3.1.3}$$

$$:= \frac{\sum_{i=1}^{n}(\min\{R_i, R_{M(i)}\} - \min\{R_i, R_{N(i)}\})}{\sum_{i=1}^{n}(R_i - \min\{R_i, R_{N(i)}\})}, \quad \text{if } t \neq 0 \tag{3.1.4}$$

If $t = 0$, let L_i be the number of j such that $Y_j \leq Y_i$, let $M(i)$ denote the j *s.t.* Z_j is the nearest neighbor of Z_i, with ties broken uniformly at random. In this case,

$$\hat{T} = \hat{T}(Y, \mathbf{X}) \tag{3.1.5}$$

$$:= \frac{\sum_{i=1}^{n}(n \min\{R_i, R_{M(i)}\} - L_i^2)}{\sum_{i=1}^{n} L_i(n - L_i)}, \quad \text{if } t = 0 \tag{3.1.6}$$

The following theorem proves that $\hat{T}$ is indeed the consistent estimator of T.

Theorem 3.1.1 (Consistency of estimator $\hat{T}$ (Azadkia and Chatterjee, 2021))**.** *Suppose that Y is not almost surely equal to a measurable function of* $\mathbf{X}$. *Then as $n \to \infty, \hat{T} \to T$ almost surely.*

Remark. i.) There is no assumptions on the joint probability of $(Y, \mathbf{X}, \mathbf{Z})$ are needed other than the non-degenerate condition that Y is not almost surely equal to a measurable function of $\mathbf{X}$. This condition is naturally resonable. Since if this does not hold, then given $\mathbf{X}, Y$ is a constant; in this circumstance, Y is both a function of $\mathbf{X}$ given $\mathbf{Z}$ and independent of X given Z. In this case, the degree of conditional dependency is meaningless.

ii.) Though the limit of $\hat{T}$ is guaranteed to be in $[0, 1]$, the actual value could be any value outside of the interval, since its an asymptotic property but the dataset is finite.

3.2 The Algorithm of Feature Ordering by Conditional Independence

After defining the CODEC, we hereby define the core algorithm FOCI which originally aims to solve the problem of the following:

- **Data**: n i.i.d. copies of $(Y_1, \mathbf{X}_1), ..., ((Y_n, \mathbf{X}_n))$, where $\mathbf{X}$ are predictors and Y is the response
- **Aim**: Using the dataset, find the Markov blanket of predictors, preferably the small subset.

The original algorithm of FOCI is given below:

Algorithm 1 Algorithm of Feature Ordering by Conditional Independence (*original version* (Azadkia and Chatterjee, 2021))

```
Input: The dataset $(Y, \mathbf{X})$        ▷ $\mathbf{X} = (X_j)_{1 \le j \le s}$; Data consists $n$ i.i.d. copies
  Start with $S_0 \leftarrow \emptyset, i \leftarrow 1$
  Let $j_1$ be the index $j$ hat maximizes $\hat{T}(Y, X_j)$, $S_1 \leftarrow S_0 \cup \{j_1\}$
  $i \leftarrow i + 1$
  while $\hat{T}(Y, X_{j_{i+1}} | X_{j_1}, ..., X_{j_i}) > 0$ do
      Let $j_{i+1}$ be the index $j \notin \{j_1, ..., j_i\}$ that maximizes $\hat{T}(Y, X_j | X_{j_1}, ..., X_{j_i})$
      $S_{i+1} \leftarrow S_i \cup \{j_{i+1}\}$
      $i \leftarrow i + 1$
  end while
Output: $S := \{j_1, ..., j_k\}$        ▷ Assume we stopped at $k$
```

Remark. i.) The consistency of FOCI is assured under certain conditions (Theorem 6.1 in Azadkia and Chatterjee (2021)).

ii.) If the stopping condition does not happen over the whole process, we will select all the given $\mathbf{X}$ consequently.

Given the original algorithm, we slightly adjusted the main function of the R package `foci` to fit our problem setting. It is clear that since we are assuming there will always be latent variables $\mathbf{Z}$ which is undiscoverable but affect the response variable Y, $\mathbf{Z}$ (in fact the estimated $\hat{\mathbf{Z}}$) should always be in the conditioning set while the FOCI do the variable selection, the adjusted FOCI algorithm is given below:

Algorithm 2 Algorithm of Feature Ordering by Conditional Independence (*edited version*)

```
Input: The dataset $(Y, \mathbf{X}, \hat{\mathbf{Z}})$        ▷ $\mathbf{X} = (X_j)_{1 \le j \le s}$; Data consists $n$ i.i.d. copies
  Start with $S_0 \leftarrow \emptyset, i \leftarrow 1$
  Let $j_1$ be the index $j$ hat maximizes $\hat{T}(Y, X_j|\hat{\mathbf{Z}})$, $S_1 \leftarrow S_0 \cup \{j_1\}$
  $i \leftarrow i + 1$
  while $\hat{T}(Y, X_{j_{i+1}}|X_{j_1}, ..., X_{j_i}, \hat{\mathbf{Z}}) > 0$ do
      Let $j_{i+1}$ be the index $j \notin \{j_1, ..., j_i\}$ that maximizes $\hat{T}(Y, X_j|X_{j_1}, ..., X_{j_i}, \hat{\mathbf{Z}})$
      $S_{i+1} \leftarrow S_i \cup \{j_{i+1}\}$
      $i \leftarrow i + 1$
  end while
Output: $S := \{j_1, ..., j_k\}$        ▷ Assume we stopped at $k$
```

Remark. 1. This edited version of FOCI algorithm is suitable for the problem of variable selection under hidden confounding.

2. For the estimation of $\hat{T}$, we proposed a new measurement method with increased computational efficiency.

Before stating how it has been improved, we first define the following functions.

Definition 3.2.1. *We know that*

$$\hat{T}_n(Y, \mathbf{X}|\mathbf{Z}) = \frac{\sum_{i=1}^{n}(\min\{R_i, R_{M(i)}\} - \min\{R_i, R_{N(i)}\})}{\sum_{i=1}^{n}(R_i - \min\{R_i, R_{N(i)}\})},$$

based on this fact, we define

$$Q_n(Y, \mathbf{X}|\mathbf{Z}) = \frac{1}{n^2}\sum_{i=1}^{n}(\min\{R_i, R_{M(i)}\} - \min\{R_i, R_{N(i)}\}),$$

and

$$S_n(Y, \mathbf{Z}) = \frac{1}{n^2}\sum_{i=1}^{n}(R_i - \min\{R_i, R_{N(i)}\}).$$

Thus,

$$\hat{T}_n(Y, \mathbf{X}|\mathbf{Z}) = \frac{Q_n(Y, \mathbf{X}|\mathbf{Z})}{S_n(Y, \mathbf{Z})}.$$

If the dimension of conditioning set is equal to zero, then

$$Q_n(Y, \mathbf{X}) = \frac{1}{n^2}\sum_{i=1}^{n}(\min\{R_i, R_{M(i)}\} - L_i^2/n)$$

and

$$S_n(Y) = \frac{1}{n^3}\sum_{i=1}^{n} L_i(n - L_i).$$

Thus,

$$\hat{T}_n(Y, \mathbf{X}) = \frac{Q_n(Y, \mathbf{X})}{S_n(Y)}.$$

Our method is to calculate the $Q_n(Y, \mathbf{X}|\mathbf{Z})$ for each iteration since the denominator remains unchanged. Besides, calculating $Q_n(Y, \mathbf{X}|\mathbf{Z})$ with fixed conditioning set $\mathbf{Z}$ requires additional computation time. To reduce it, we use an equivalent measure $Q_n(Y, (\mathbf{X}, \mathbf{Z}))$ which selects exactly the same Markov blanket as using the $Q_n(Y, \mathbf{X}|\mathbf{Z})$. We give a brief proof in the following.

By the Definition 3.2.1, we have

$$Q_n(Y, \mathbf{X}|\mathbf{Z}) = Q_n(Y, (\mathbf{X}, \mathbf{Z})) - Q_n(Y, \mathbf{Z}).$$

Since $Q_n(Y, \mathbf{Z})$ remains unchanged during iteration, we can just compute the $Q_n(Y, (\mathbf{X}, \mathbf{Z}))$ to do the feature ordering process. It saves time of running FOCI.

Chapter 4

Problems and Simulation Studies

4.1 Problems and Datasets

In the previous chapter, we went over the working scheme and the theoretical properties of CODEC and FOCI. In this chapter, we will try to use synthesized datasets in order to discover what FOCI performs under different types of hidden confounding. In addition, we will mainly use two hidden variable estimators: principal Components Analysis (Section 2.3) and Variational Autoencoders (Section 2.4).

4.1.1 Problems

In this paper, we are going to do the simulations about two basic cases in causal inference, which are given in the following Figure 4.1 and 4.2:

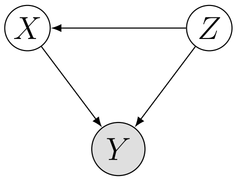

Figure 4.1: A random variable X and the a latent variable Z both contributes to Y. Besides, the hidden variable Z also decides X

.

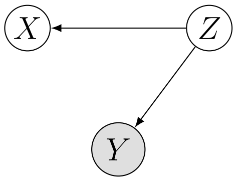

Figure 4.2: Only the latent variable Z contributes to Y. Besides, the hidden variable Z also decides X. This case is also called confounding

.

Based on these two basic cases, we will do several different simulations to see how FOCI performs.

Recapitalizing our problem in a detailed manner, we are using the FOCI to select the Markov blanket of the predictors $\mathbf{X} = (X_1, ..., X_s)$ and response Y given hidden variable(s) $\mathbf{Z} = (Z_1, ..., Z_t)$.

After using the designed settings to generate the artificial datasets (Note: the causal setting is to choose one from the above two relationships (Figure 4.1 and 4.2) to generate the $X_i,\ i \in \{1, ..., s\}$), we will discuss how FOCI performs and what could be improved.

4.1.2 Datasets and Experimental Environment

In this paper, we mainly use R to generate the datasets with fixed random seeds to ensure the reproducibility. We are using the extant R package `FOCI` to do the selection. However, in the section of simulating nonlinear functional relationships, we are going to use Python and the package of `PyTorch` to build and train the variational autoencoders in order to simulate the latent variables.

4.2 Simulation Studies and Results

Before going into our exploratory experiments, we first define the notations that will appear in this chapter for clarification. $\mathbf{Z}$ stands for the latent variables with dimension $n \times$ `num_hidden`, where n is the number of generated samples and `num_hidden` is number of latent variables. B stands for the coefficient matrix with dimension `num_hidden` $\times\ p$, where p is the column dimension of $\mathbf{X}$, *i.e.*, the number of predictors. β stands for the vector that adds all the latent variables up and contributes to the value of response Y.

4.2.1 Experiment 1: Variable selection over single signal & single latent variable

Settings

Generate 1000 samples with 100 variables $X_1, ..., X_{100} \sim \mathcal{N}(0, 1)$ except for X_{77} and one hidden variable $Z \sim \mathcal{U}(0, 1)$. Assuming that Y is a function of Z, we can say that in the setting of causality that Y is depending on Z. At the same time, we selected a variable, X_{77}, to be depending on Z by the following way:

$$X_{77} = 10\sqrt{Z} + \varepsilon_0,$$

where $\varepsilon_0 \sim \mathcal{N}(0, 1)$.

In addition, the Y is depending on some of the signal variables $\in \{X_1, ..., X_{100}\}$. In the first setting,

$$Y = X_1 \times X_{10} + X_{60} \times Z + \varepsilon,$$

where $\varepsilon \sim \mathcal{N}(0, 1)$

In the second setting,

$$Y = X_1 \times X_{10} + X_{77} \times Z + \varepsilon,$$

where $\varepsilon \sim \mathcal{N}(0, 1)$

Here, ε_0 and ε are both random noises which aim to simulate the noise of a real data.

Methods & Objectives

Using PCA to estimate the single latent variable Z as Z_{est} and regard it as a prediction variable, *i.e.* $X_{new} = X$ column binding Z_{est}. We repeat this simulation for 50 times.

Results

The results are given below, while the total simulation counts is 50.

X_1	X_3	X_{60}	Z_{est}	X_{77}
44	40	50	18	23

Table 4.1: Results of the first setting

X_1	X_3	X_{77}	Z_{est}
46	45	49	47

Table 4.2: Results of the second setting

Discussions

Overall, FOCI selects correctly when there is dependency between the response and the predictor. The frequencies of signal selection are very closed to 50 times in both cases, which is satisfying.

However, when there is confounding (the first setting), FOCI wrongly selects X_{77} to be regarded as an element in Markov blanket for nearly 50% of the total cases. Besides, if we regard Z_{est} as our predictor, the case with confounding will selects only 18 out of 50 times, which is unsatisfying.

I am conjecturing the reason why the case with confounding performs worse:

- It is due to the nonlinear relationship between $\mathbf{X}$ and Z, and Y and $\mathbf{X}$. In this setting, using PCA will only discover a linear relationship, and hence lead to a sub-optimal estimation of Z.
- Since we are only using 1000 samples per simulation. This can lead to an inaccurate estimate of CODEC value $\hat{T}$, and hence influence the process of FOCI.
- The latent variable is affecting only one of the predictors X. Under this circumstance, if one want to make the latent variable estimation, PCA will not be a good choice because the projection from original X space to the hidden variable Z space is not dense. Here, the dense is defined as the notion that $\mathbf{X}$ and $\mathbf{Z}$ are densely related. In other words, every latent confounder contributes to all the predictors. In that case, PCA would possibly be a sharp weapon for us to deal with estimating the latent confounding. We will give a formal proof later.

Furthermore, we have realized that including the estimated hidden variable in the predictors may not be a good idea, it is worthwhile to try to let the estimated hidden variable not attending the selection process, but always stay in the conditioning set. This change can motivates our edited version of FOCI main function.

Finally, from this basic experiment, we found that only calculating the correct frequencies of signals will restrict our insights about the other predictors. For example, in the first setting, we only looked at the statistic of X_3, X_{10}, and X_{60}. Since we do not know whether the FOCI algorithm selects other redundant variables. It is hard to say whether the FOCI performs well. For instance, if we simulate for 50 times, and got near 50 frequencies selecting X_3, X_{10}, and X_{60}. However, while selecting the correct sufficient subsets, it simultaneously includes lots more redundant predictors, which will lead to a mess.

Those thoughts lead to the following Experiment 2.

4.2.2 Experiment 2: Variable selection over multiple signals & multiple latent variables

New FOCI function

As we mentioned before, it is worthwhile to always include the estimated hidden variables in the conditioning set to seek for a better variable selection result. We hereby introduce a crucial additional arguments in the new version of FOCI function `foci_new` based on the original version `foci_main` (Azadkia, Chatterjee, and Matloff, 2021). This argument is called `num_hidden`, when `num_hidden` $= 0$, that means we do not care about the latent confounders, and directly use the previous version `foci_main`. However, if `num_hidden` is equal to some integer $l > 0$, then we will use the estimated $\mathbf{Z}_{est}$ with dimension $N \times$ `num_hidden`.

Settings

Generate $N = 10000$ samples with 100 variables $X_1, ..., X_{100})$. Assume that $\mathbf{X} = (X_1, ..., X_{100})$ are all generated from $\mathbf{Z} = (Z_1, Z_2, Z_3)$, where $Z_i \sim \mathcal{U}(10, 20)$, $\forall i \in [3]$, and the relationship between $\mathbf{X}$ and $\mathbf{Z}$ are as follows:

$$\mathbf{X} = \mathbf{Z} \cdot B + \varepsilon_0,$$

where B is the coefficient matrix generated randomly from $\mathcal{U}(0, 1)$ (which makes the relationship linear) and $(\varepsilon_0)_{ij} \sim \mathcal{N}(0, 1), i \in [10000], j \in [100]$ is noise term matrix.

Besides, we define the response $Y = X_3 + X_{10} + X_{77} + \mathbf{Z}\beta + \varepsilon$, where $\beta_i \sim \mathcal{U}(0, 1)$, $\forall i \in [3]$.

Methods & Objectives

Our goal is to use the FOCI to select the correct Markov blanket, i.e. X_3, X_{10}, X_{77}, based on the fact that $\mathbf{X}$ is depend on $\mathbf{Z}$. Since in reality we do not know how many hidden variables are affecting $\mathbf{X}$ and the information about the hidden variables are not our main focus, we hereby focusing on the question that: *By including how many estimated hidden variables Z_{est} in the conditioning set can we achieve the best precision $= \frac{TP}{TP+FP}$ regarding selecting X_3, X_{10}, X_{77}? Or in other words, control the false discovery proportion (FDP) $= \frac{FP}{TP+FP}$ by adjusting the conditioning set.* Here, TP refers to true positives, which in our case is the number of correctly selected Markov blankets. On the other hand, FP refers to false positives, which in our case means that FOCI selects them but they should not appear in the Markov blanket since they are independent with response Y conditional on hidden variables $\mathbf{Z}$.

We again using the PCA to estimate the hidden variables, i.e. how many principal components relates to how many hidden variables, with decreasing order of explained variance.

We start from including PC0, which is to do nothing, PC1, which is to include the first principal component, and increases the principal components included by adding one PC each time to see the results.

Results

We run from including PC0 to including PC7 (all previous PCs included), with each run 100 times of repetition. The result is as following:

PCs	*TP*	*FP*	*Precision*
PC0	300	249	0.547
PC1	300	116	0.721
(PC1 , PC2)	300	63	0.826
(PC1, ... , PC3)	300	23	0.929
(PC1 , ... , PC4)	300	25	0.923
(PC1 , ... , PC5)	300	42	0.877
(PC1 , ... , PC6)	300	57	0.840
(PC1 , ... , PC7)	300	66	0.820

Table 4.3: The TP, FP and Precision table of including different amount of estimated principal components, with $N = 10000$ samples

Discussions

In the above Table 4.3, TP is the abbreviation of true positive, which stands for the correct selection of X_3, X_{10}, X_{77}; FP is the abbreviation of false positive, which stands for the wrong selection of the previous three variables but in reality they should not be selected; precision stands for the ratio of $\frac{TP}{TP+FP}$.

We have discovered that including three hidden variables will lead to a better result of precision, which coincides with the truth that we have defined three hidden variables. While including more and more principal components (starting from PC3), it did not lead to a better result but the precision kept decreasing.

This perfect result tell us that when the estimation method and selection process is designed properly. FOCI will reach approximately 93% of precision.

Finally, this experiment provide the empirical justification for including the $\mathbf{Z}_{est}$ in the conditioning set as it decreases the false discovery proportion for predicting the signal variables given sufficient sample size (in our case $N = 10000$).

Additional Experiment I

After seeing the high precision the FOCI has reached, we are wondering if this precision good enough? The question can be solved comparing the including $\mathbf{Z}_{est}$ version and the version including the true latent confounders $\mathbf{Z}$ (which should never be known in reality).

We assume that

$$Y = X_3 + X_{10} + X_{77} + \mathbf{Z}\beta + \varepsilon$$

.

The comparison of precision is among these two cases:

i.) True $\mathbf{Z}$ are considered as conditioning set, ii.) Estimated **Z** are considered as conditioning set, the number of hidden variables included `= 0, 1, 2, 3, 4, 5, 6, 7`.

We use the same random seed as the previous experiment, so the part two's result should be exactly the same. The result is given in the following table:

PCs	*TP*	*FP*	*Precision*
PC0	300	249	0.547
PC1	300	116	0.721
(PC1 , PC2)	300	63	0.826
(PC1, ... , PC3)	300	23	0.929
(PC1 , ... , PC4)	300	25	0.923
(PC1 , ... , PC5)	300	42	0.877
(PC1 , ... , PC6)	300	57	0.840
(PC1 , ... , PC7)	300	66	0.820
True Z	300	5	0.984

Table 4.4: The TP, FP and Precision table of including different amount of estimated principal components and including the true **Z**, with $N = 10000$ samples

Remark. i.) We can see including the true hidden variables **Z** will lead to the best precision, which is not surprising. On the other hand, it demonstrates that with more accurate estimation of hidden variables, we will have better variable selection results. This empirically proves that FOCI is a strong tool without any regression model assumptions to deal with sufficient dimension reduction problem especially in the area of causal inference.

ii.) It illustrated that PCA is a proper estimation method here dealing with latent variables, which should be considered primarily when we are dealing with real-life datasets with latent variables.

iii.) The loop over PC0 to PC7 provides us with a good estimation scheme to find the amount of the latent variables. In our case, we can infer from the loop results that the number of hidden variables could be three or four.

Additional Experiment II

Under the same setting, we also wants to see whether increasing the inclusion of principal components to up to PC100 would harm the FOCI selection process. Since when we are talking about including all 100 PCs, it is just an transformation from the original Euclidean space to another one.

Besides, we also want to see whether decreasing the generated i.i.d. samples from 10000 to 1000 will lead to a **much** worse variable selection results.

We this time settled the $N = 1000$ and each iteration the repetition time reduces from 100 to 20, but will go over including all the principal components (in our case to PC100).

The result is given below in the Table 4.5 (Note: for the sake of simplicity, (PC1, ..., PCk) will be denoted by [PCk])

PCs	*TP*	*FP*	*Precision*
PC0	54	60	0.474
PC1	58	51	0.532
[PC2]	58	48	0.547
[PC3]	58	36	0.617
[PC4]	59	48	0.551
[PC5]	60	51	0.541
[PC6]	60	59	0.504
[PC7]	60	61	0.496
[PC8]	59	62	0.488
[PC9]	60	54	0.526
[PC10]	60	47	0.561
[PC11]	59	54	0.522
[PC12]	59	58	0.504
[PC13]	59	59	0.500
[PC14]	58	67	0.464
[PC15]	59	71	0.454
[PC16]	59	54	0.522
[PC17]	59	79	0.428
[PC18]	58	67	0.464
[PC19]	60	67	0.472
[PC20]	59	70	0.457
[PC25]	58	61	0.487
[PC50]	57	59	0.491
[PC75]	59	52	0.532
[PC100]	59	71	0.454
True	60	9	0.870

Table 4.5: The TP, FP and Precision table of including different amount of estimated principal components (up to PC100) and including the true $\mathbf{Z}$, with $N = 1000$ samples

Remark. i.) We can see that even we always couldn't find a good number of FP (ideally the less the better). There does not exist a very conspicuous decreasing trend of precision while PCs included are increasing.

ii.) Besides, we also observed that while the included PCs are increasing, the TP will not change a lot, even including 100 PCs. It is hard to say what causes this, but I am pretty sure that the random noise between ($\mathbf{X}$ and $\mathbf{Z}$) and (Y and $\mathbf{X}$) will not make the PCA rotation a strictly linear one, this helps the FOCI to judge which is belong to Markov blanket.

iii.) All of the precision values went down conspicuously, we believe that less samples will lead to worse variable selection results.

4.2.3 Experiment 3: Comparison between Z and $\mathbf{Z}_{est}$

In the last experiment, we proposed a viewpoint that the more accurate our estimate of latent variable $\mathbf{Z}$ is, the more precision and less false discovery proportion we have when using new FOCI to select the Markov blankets. This leads to a question: *Does our estimate $\mathbf{Z}_{est}$ really close to $\mathbf{Z}$ in distribution without considering the variable selection?*

Settings

Generate 10000 copies of data, $\mathbf{Z} \sim \mathcal{N}(0, I_t)$, where X is generated from Z which is a linear transformation with random noise (as described previously $X = Z \cdot B + \varepsilon_0$, see Section 4.2.2).

Methods & Objectives

Straightforwardly, we can use illustrations of empirical cumulative distribution function (ECDF) to compare the similarity of underlying distributions between two given $n \times r$ data matrices, where n stands for the amount of copies, r stands for the number of hidden variables. There are also many other methods to measure and judge the distribution similarities between samples. Kolmogorov-Smirnov test (Massey, 1951) is one of the most used, but it is rather like a test which judges whether two are similar. Consequently, we chose another measure which is estimating the overlapping areas to illustrate the extent of similarity, which is called *overlapping index* (Pastore and Calcagnì, 2019).

Definition 4.2.1 (Overlapping index). *Let us assume two real probability density functions $f_A(x)$ and $f_B(x)$. The overlapping index $\eta : \mathbb{R}^n \times \mathbb{R}^n \to [0,1]$ is defined as follows:*

$$\eta(A, B) = \int_{\mathbb{R}^n} \min[f_A(x), f_B(x)]dx,$$

where the integral can be replaced by summation in the discrete case.

The estimation of overlapping index $\hat{\eta}$ is established and have already formed an R package `overlapping`. We will use it during the experiment.

Results

The illustration is given in the following Figure 4.3. Besides, the corresponding overlapping index in this figure is $\hat{\eta} = 0.9080391$.

Discussions

i.) $\hat{\eta}$ is close to 1 indicates that two set of random vectors have similar underlying distribution. It is a good sign that PCA really works in the context of pure empirical CDF comparison but not under the context of FOCI feature selection process.

ii.) We can see from the Figure 4.3 that the standardized data is very close to the original Z's empirical CDF. That gives us justification of standardize / normalize the datasets prior to estimate the hidden confounders Z_{est}.

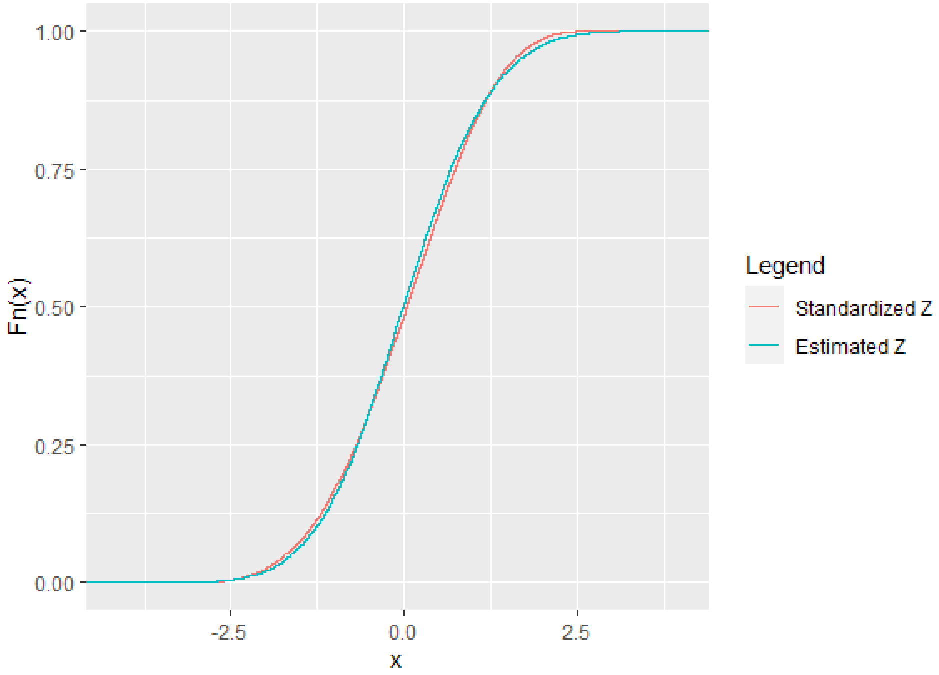

Figure 4.3: Illustration of standardized Z and standardized Z_{est}

4.2.4 Experiment 4: Invertible transformation P on $\mathbf{Z}$

Since we have already shown that after putting the estimated hidden variables Z_{est} into the conditioning set of CODEC function $\hat{T}$ will do a better job than putting them into the predictors, we now want to see does the invertible (orthogonal) transformation P on $\mathbf{Z}_{est}$ will change the CODEC value. It is worth noting that if there exists such extent of invariance, it will computationally justify that PCA is proper to use for dense relationship between $\mathbf{Z}$ and $\mathbf{X}$, since by the nature of PCA, estimating $\mathbf{Z}_{est}$ is approximately equivalent to estimate $\mathbf{Z}_{est} \times M \approx \mathbf{Z}$, where $M \approx P^{-1}$.

Settings

We assume that

$$Y = X_3 + X_{10} + X_{77} + \mathbf{Z}\beta + \varepsilon,$$

where $\mathbf{Z} = (Z_1, Z_2, Z_3)$. All of the remaining settings are the same as that of Experiment 2 (See Section 4.2.2).

We generate random invertible square (transformation) matrix using the property that the product of matrices is invertible if each of the multipliers are invertible.

Let Λ be a 3×3 diagonal matrix whose square product is not I_3, U be a orthogonal matrix, so that

$$P := U\Lambda U^T$$

is an invertible matrix but not orthogonal since $P^TP = U\Lambda U^TU\Lambda U^T = U\Lambda^2U^T$, which is definitely not I_3 (If so, we must have $P^TP = I = U\Lambda^2U^T$, which indicates that $U^TU = \Lambda^2$, and is contradicted to the previous assumption).

Methods & Objectives

This experiment aim to investigate how the number of data samples affect the absolute difference between CODEC function conditioning on original $\mathbf{Z}$ and transformed $\hat{P}(\mathbf{Z})$, where $\hat{P}$ is a randomly chosen invertible transformation, whose transformation matrix P is invertible but not orthogonal.

Specifically, this experiment is aim to calculate the absolute difference between CODEC(Y, X, Z) and CODEC$(Y, X, \hat{P}(Z))$, where $\hat{P}$ is invertible yet not orthogonal while changing the amount of samples n.

Results

In our setting, the number of samples N ranges from 100 to 20000 with an increment of 100.

We particularly calculate

$$C_0 = \text{CODEC}(Y, X_{3:10}, \hat{\mathbf{Z}})$$

and

$$C_1 = \text{CODEC}(Y, X_{3:10}, \hat{P}(\hat{\mathbf{Z}})),$$

where $\mathbf{Z}$ is estimated from the first three PCs.

The result is shown in Figure 4.4.

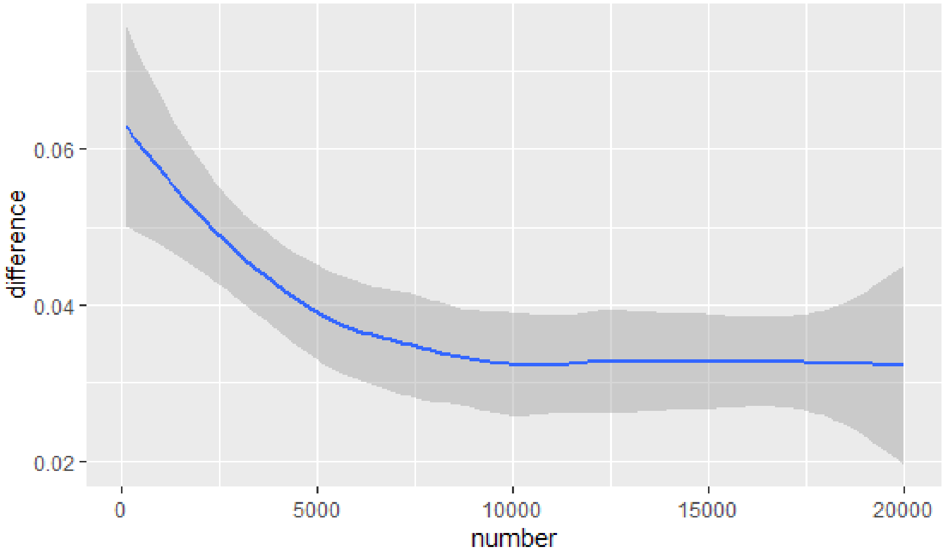

Figure 4.4: The absolute difference between C_0 and C_1 with the increasing number of samples, the increment of increasing is 100

Discussions

i.) It is clear that with the sample increasing, the difference between C_0 and C_1 are getting closer to zero. It could be a correct hypothesis that with $n \to \infty, \Delta C = |C_0 - C_1| \to 0$.

ii.) The gray area is represents the 95% confidence interval since we are using a LOESS smoother (Cleveland, 1979) to illustrate.

iii.) We also tried the experiments on orthogonal transformations, which returns us with exactly the same CODEC values.

Robustness of variable selection on invertible transformations

Using the same setting as the previous experiment, with $N = 5000, p = 100,$ repetitions $=$ 20. We tried to compare the selection results without transformation and the one with arbitrary invertible transformation on the estimated hidden Z.

The result is as following:

PCs	*TP*	*FP*	*Precision*
PC0	59	44	0.573
PC1	60	25	0.706
[PC2]	60	18	0.769
[PC3]	60	17	0.779
[PC4]	60	12	0.833
[PC5]	60	18	0.769
[PC6]	60	17	0.779

Table 4.6: TP, FP and precision table without transformation, with $N = 5000$

PCs	*TP*	*FP*	*Precision*
[PC2]	60	21	0.741
[PC3]	60	20	0.750
[PC4]	60	24	0.714
[PC5]	60	34	0.638
[PC6]	60	25	0.706

Table 4.7: TP, FP and precision table with some invertible but not orthogonal transformations, and $N = 5000$

Remark. i.) Table 4.6 refers to the selection results without any transformations, each PCs with 20 repetitions and we calculate the accumulated results.

ii.) Table 4.7 refers to the selection results with random designed transformations which are introduced before, *i.e.* $P = U\Lambda U^T$, where U is orthogonal matrix and Λ is a diagonal matrix. Here our Λ is

$$\begin{bmatrix} 1+\delta_1 & 0 & \cdots & 0 \\ 0 & 1+\delta_2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & 1+\delta_k \end{bmatrix}$$

k refers to the dimension of our estimated hidden variables, e.g., if we have 5 estimated hidden variables, k should be 5 in order to conform the dimensions of transformation matrix. Here,

$$\delta_i \sim \mathcal{U}(0,1), \quad i = 1, ..., k$$

iii.) Table 4.7 is starting from PC2 since 0×0 and 1×1 matrices are meaningless.

iv.) The result shows that with invertible projection on the estimated Z, the algorithm FOCI still selects correct variables but with a tiny bit more false positives. The precision is fairly good compared to the selection results without transformations.

From above discussions, we computationally proved the justification of using principal components as estimator of latent confounders. In the next section, we will theoretically justify the statement.

4.2.5 Theoretical support of using principal components as estimator

From what we have discussed before, we state our assumptions and then begin to justify the reasonableness of using principal components.

Assumptions:

i.) $\mathbf{X} \in \mathbb{R}^{n \times p}$ and $\mathbf{Z} \in \mathbb{R}^{n \times l}$ are densely associated without random error term, *i.e.*,

$$\mathbf{X} = F(\mathbf{Z}),$$

where F is an invertible affine transformation (definition given below) from Euclidean space $\mathbb{R}^{n \times p}$ to $\mathbb{R}^{n \times l}$.

ii.) $\mathbf{X}$ has zero mean and a positive definite population covariance matrix $C_{\mathbf{X}} = \mathbb{E}[\mathbf{X}\mathbf{X}^T]$.

Definition 4.2.2 (1-Nearest Neighbors (1-NN) (Kramer, 2013))**.** *For a given vector $x' \in \mathbb{R}^n$, the 1-nearest neighbors are defined as the vector $x \in \mathbb{R}^n$ that minimize the Euclidean distance between the two vectors, i.e.,*

$$x \in \arg\min_{v} \{\|v - x'\|_2\},$$

and the ties broken uniformly at random.

Definition 4.2.3 (Isometry)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}^n$ is an isometry if every $x, y \in \mathbb{R}^n$ satisfies*

$$\|f(x) - f(y)\|_2 = \|x - y\|_2 \, .$$

In other words, we say f is isometry if it preserves the norms.

Definition 4.2.4 (Orthogonal transformation)**.** *Given any two nontrivial Euclidead spaces E and F of the same finite dimension n, a function $f : E \to F$ ia an orthogonal transformation iff. it is linear and*

$$\|f(u)\|_2 = \|u\|_2 \, ,$$

for all $u \in E$.

Thus, an orthogonal transformation is a linear map that preserves the norms.

Definition 4.2.5 (Linearity)**.** *A function $\phi : \mathbb{R}^n \to \mathbb{R}^m$ is called linear if it satisfies the following properties:*

(1) For every $x \in \mathbb{R}^n$ and every $\alpha \in \mathbb{R}$, we have $\phi(\alpha x) = \alpha\phi(x)$.

(2) For every $x, y \in \mathbb{R}^n$, we have $\phi(x + y) = \phi(x) + \phi(y)$.

Note that if ϕ is linear, then there exists an $m \times n$ matrix L such that $\phi(x) = Lx$.

Definition 4.2.6 (Affine transformation in $\mathbb{R}^n$)**.** *An affine transformation of is a map $f : \mathbb{R}^n \to \mathbb{R}^m$ of the form*

$$f(p) = Ap + q,$$

for every $p \in \mathbb{R}^m$ and $q \in \mathbb{R}^m$.

The inverse transformation of an invertible affine transformation is denoted by f^{-1}.

Definition 4.2.7 (Translation)**.** *A translation T_y defined in $\mathbb{R}^n$ is that $T_y(x) = x + y$, for any $x, y \in \mathbb{R}^n$.*

Lemma 4.2.8. *Every isometry F that satisfies $F(0) = 0$ is linear.*

Proof. The proof can be found in page 3 of DeVos (2018). □

Theorem 4.2.9. *Every isometry F is an affine transformation.*

Proof. Let F be an isometry and let $u = F(0)$. We then define a new transformation $G = T_{-u} \circ F$ and we have

$$G(0) = T_{-u} \circ F(0) = T_{-u}(u) = u - u = 0.$$

From Lemma 4.2.8, we have G is a linear map, which indicates that there exists a matrix L such that $G(x) = Lx$. It follows naturally that $F(x) = Lx + y$, which indicates that F is an affine transformation. □

Remark. 1. Orthogonal transformation is a subset of affine transformation, which preserves Euclidean norm naturally.

2. It follows that the composite of two affine transformations are still affine transformation, which indicates that composite affine transformations still preserves the Euclidean norm.

Next, we will review some of the key notions to define a CODEC function in Azadkia and Chatterjee (2021).

Recall the Equation 3.1.4 and the definition of $N(i)$ to be the index j *s.t.* Z_j is the nearest neighbor of Z_i *w.r.t.* Euclidean norm on $\mathbb{R}^s$, where ties are broken uniformly at random.

Besides, let $M(i)$ be the index j such that (X_j, Z_j) is the nearest neighbor of (X_i, Z_i) in $\mathbb{R}^{s+t}$, with ties broken uniformly at random.

Finally, let R_i be the rank of Y_i, *i.e.*, the number of j *s.t.* $Y_j \leq Y_i$.

We propose the following theorem.

Theorem 4.2.10. *If F is an affine transform, then CODEC($Y, X, F(Z)$) = CODEC(Y, X, Z).*

Proof. From the definition of CODEC function, only $N(i)$ and $M(i)$ would possibly change after conditioning on $F(Z)$. In the sense of finding 1-NN, we will focus on calculating the Euclidean norm, which is preserved by affine transformation using Theorem 4.2.9. This leads to the results of 1-NN unchanged.

Hence, $R_{M(i)}$ and $R_{N(i)}$ will remain unchanged, which indicates that the CODEC values will be the same. □

Based on the fact that PCA is a process of orthogonal transformation $\mathbf{X}' = \mathbf{X} \times M$ and $\mathbf{X} = F(\mathbf{Z})$ by assumption, we have

$$\mathbf{X}' = F(\mathbf{Z}) \times M = G(F(\mathbf{Z})) = H(\mathbf{Z}),$$

where G is a orthogonal transformation and $H = G \circ F$.

Noticing that M is orthogonal which belongs to the family of invertible affine transformation (by setting A an $n \times n$ orthogonal matrix and $q = 0$), and F is an invertible affine transformation. We have H is invertible affine transformation, and thus $H^{-1}(X') = \mathbf{Z}$.

Using Theorem 4.2.10, we have

$$\mathrm{CODEC}(Y, \mathbf{X}, \mathbf{X}') = \mathrm{CODEC}(Y, \mathbf{X}, H^{-1}(\mathbf{X}')) = \mathrm{CODEC}(Y, \mathbf{X}, \mathbf{Z}).$$

This gives the justification of using PCA as latent confounder estimator. Before going on, it has also proposed several issue with respect to variable selection.

i.) Since $X' = G(X)$, this also indicates that $\mathrm{CODEC}(Y, \mathbf{X}, \mathbf{X}') = \mathrm{CODEC}(Y, \mathbf{X}, \mathbf{X})$, which is equal to zero undoubtedly.

ii.) In the sense of dimension reduction (find the correct number of latent confounders), we will always choose $k < p$ principle components where p is the number of predictors. In this case, the output will be truncated after performing PCA and the information is lost which makes it not an affine transformtion.

For the first issue, it can be explained heuristically and computationally.

Heuristically, we know that there is typically an error term E that makes $\mathbf{X}$ the noisy signals generated from $\mathbf{Z}$, *i.e.*,

$$\mathbf{X} = F(\mathbf{Z}) + E.$$

Under this condition, we will have

$$\begin{aligned}\mathbf{X}' &= (F(\mathbf{Z}) + E) \times M = G(F(\mathbf{Z}) + E) \\ &= G \circ F(\mathbf{Z}) + G(E) \\ &= H(\mathbf{Z}) + G(E)\end{aligned}$$

Thus, with the noise term, we could actually selects the variable through including the principal components in the conditioning set. In this sense, we are using a approximate version of original $\mathbf{Z}$. It remains to discover which type of random noise and how large of the sample size will affect the CODEC value in the future.

Computationally, the Experiment 7 has shown that with this random noise term E, FOCI actually selects without losing much of the ability finding the conditional dependence when conditioning on full principle components. (See Experiment 7 Discussion 4.2.8 iii.)

For the second issue, we will introduce some new notions presented by Geiger and Kubin (2012) which relates to the accurate definition of *loss of information*. In the context of information, we will show that when the sample size increases to infinity, the information loss will converge to zero, which makes the low-dimension representation feasible as well as adds to the credibility of retrieving original latent confounder $\mathbf{Z}$ using $\mathbf{X}_{truncated}$.

The following definitions and theorems are a review of Geiger and Kubin (2012) and Rényi (1959). All the related proofs can be found in these two papers.

Definition 4.2.11 (Relative information loss)**.** *Let $X \in \mathcal{X}$ be an N-dimensional random variable and let $Y = g(X)$, where $g(\cdot)$ is a function (map). We define the **relative information loss** induced by this map as*

$$l(X \to Y) = \lim_{n\to\infty} \frac{H(\hat{X}_n|Y)}{H(\hat{X}_n)},$$

where H stands for the Shannon (Shannon, 1948) entropy

$$H(X) = -\sum_{x\in\mathcal{X}} p_X(x)\log(p_X(x)),$$

and $\hat{X}_n = \frac{[nX]}{n}$ (element-wise round-up)

Definition 4.2.12 (Information dimension)**.** *The information dimension of a random variable $\mathbf{X}$ is given as*

$$id(X) = \lim_{n\to\infty} \frac{H(\hat{X}_n)}{\log n},$$

provided the limit exists.

Lemma 4.2.13. *Let X be a random variable with positive and finite information dimension $id(X)$. Then, if $id(X|Y = y)$ exists and is finite P_Y- almost surely, the relative information loss equals*

$$l(X \to Y) = \frac{id(X|Y)}{id(X)},$$

where $id(X|Y) = \int id(X|Y = y)dP_Y(y)$.

Theorem 4.2.14. *Assume $\mathbf{X}$ is a matrix where each of its n rows represents an indepedent sample of N-dimensional Gaussian random variable X with sample covariance matrix $\hat{C_{\mathbf{X}}} = \frac{1}{n}\mathbf{X}\mathbf{X}^T$ and $n \geq N$. Besides, we assume that after PCA transformation, we truncated the PCA transformed $Y^N = F_{PCA}(X)$ to a lower dimension Y^K. We then by Lemma 4.2.13 getting the relative information loss, which is equal to*

$$l(X \to Y^K) = \frac{N-1}{2n} \times \frac{N-M}{N}.$$

Remark. 1. It indicates that while sample size $n \to \infty$, the information loss of truncated PCA will converge to zero, which lead us to accurately retrieving the latent confounders $\mathbf{Z}$.

2. Combine this theorem with the above discussions about the CODEC value retrieving

with random error terms, we can conclude that while sample size is sufficiently large, we can use first-K ($K < N$) principle components to represent the latent confounders and conditioning on them without worrying the vanishing conditional dependence (zero-valued CODEC).

4.2.6 Experiment 5: Feature selection when the relationship between X and Z is not dense

Settings

Suppose we have $Y = X_3 + X_{10} + X_{77} + \mathbf{Z}\beta + \varepsilon$, where $\beta_i \sim \mathcal{U}(0,1), \forall i \in [3]$ and $\varepsilon \sim \mathcal{N}(0,1)$ is the random noise. But this time we have

$$X_1, ..., X_{10} = \sqrt{\mathbf{Z} \cdot B} + \varepsilon_1,$$

where $\mathbf{Z} = (Z_1, Z_2, Z_3)$, where $Z_i \sim \mathcal{U}(10, 20)$, $\forall i \in [3]$, B is the coefficient matrix generated randomly from $\mathcal{U}(0,1)$ and ε_1 is the random noise with standard normal distribution.

Besides, we have

$$X_{11}, ..., X_{100} \perp\!\!\!\perp \mathbf{Z},$$

this makes the effect of Z on $\mathbf{X}$ not dense.

To summarize, we have $\mathbf{X} = (X_1, ..., X_{100})$, and

$$X_i = \begin{cases} \sqrt{\mathbf{Z} \cdot B} & \text{when } i = 1, ..., 10 \\ \sim \mathcal{N}(0,1) & \text{when } i = 11, ..., 100 \end{cases} + \varepsilon_1,$$

where ε_1 is the random noise vector with standard normal distribution, which the same random noise as the previously defined one.

Methods & Objectives

In this experiment, we want to see whether using PCA as hidden variable estimate still performs well under this condition.

Results

We run from including PC0 to including PC15 (all previous principal components included) with each iteration 20 times of repetitions. The result is given in Table 4.2.6.

Discussions

i.) We can see that we always couldn't find a good number of TP (ideally it should be 60). The best result of TP now is to include nothing and the best result of precision is to only include one principal component, which counters the fact that we should have three latent confounders. This suggests that while the relationship between **X** and **Z** is nonliear and not dense (Note: piecewise relationship between **X** and **Z** has additionally add the nonlinearity), PCA may not be a good estimator of hidden variables for selecting Markov blankets. We probably need to find another estimator of latent variables, for example variational autoencoders to ease the problem of nonlinearity.

ii.) Besides, using different generated samples at each iteration and calculate the total true positives and false positives may not be meaningful since it is impossible to get

PCs	*TP*	*FP*	*Precision*
PC0	54	56	0.513
PC1	45	41	0.523
[PC2]	43	40	0.518
[PC3]	42	55	0.433
[PC4]	39	54	0.419
[PC5]	35	55	0.389
[PC6]	53	71	0.427
[PC7]	42	87	0.326
[PC8]	42	83	0.336
[PC9]	44	88	0.333
[PC10]	48	102	0.320
[PC11]	53	99	0.349
[PC12]	36	83	0.303
[PC13]	42	92	0.313
[PC14]	44	109	0.288
[PC15]	41	114	0.265

Table 4.8: TP, FP and precision table, with $N = 5000$

real-life datasets in this way, it is more meaningful in the sense of variable selection from my point of view to generate one piece of data and resampling subsamples from it. This scheme will be introduced in the next experiment.

4.2.7 Experiment 6: Using permuted subsamples during selection process

Settings

Suppose we have $Y = X_3 + X_{10} + X_{17} + \mathbf{Z}\beta + \varepsilon$. We also have

$$X_1, ..., X_{10} = \sqrt{\mathbf{Z} \cdot B},$$

where B is coefficient matrix with entries generated from $\mathcal{U}(0,1)$, ε_1 is the random noise with standard normal distribution. Besides, we have

$$X_{11}, ..., X_{20} \perp\!\!\!\perp \mathbf{Z},$$

this makes the effect of $\mathbf{Z}$ on $\mathbf{X}$ not dense.

To summarize, we have $\mathbf{X} = (X_1, ..., X_{20})$, and

$$X_i = \begin{cases} \sqrt{\mathbf{Z} \cdot B} & \text{when } i = 1, ..., 10 \\ \sim N(0,1) & \text{when } i = 11, ..., 20 \end{cases} + \varepsilon_1,$$

where ε_1 is the random noise vector with standard normal distribution.

Methods & Objectives

Since we are not getting good results of variable selection while there is nonlinear and non-dense relationship between $\mathbf{X}$ and $\mathbf{Z}$, we decided to change to data generating scheme in order to reduce the misleading effect of the nonlinear and non-dense relationship.

The data generating scheme is easily to summarize, which is resampling the permuted sub-samples (without replacement) to not only reduce the computation time and thus increases the repetition times of the same dataset which will add stability to the variable selection process, but also can spread the false positives over every non-Markov blanket random variables. The advantage of this scheme is that one can clearly distinguish the difference between the signals and non-signals, which certainly helps to select signals.

We hereby define our resampling scheme.

Definition 4.2.15 (Resampling scheme)**.**

i.) *Generate a large sample S with N entries using the given relationship between Y and X.*

ii.) *Randomly choosing subsamples S_i, $i = 1, ..., R$ with size $N/2$ without replacement.*

iii.) *Total R sets of the subsamples will then be used to extract the principal components and then implement the FOCI algorithm.*

Actually, for every set of subsamples, we will run the current algorithm and record a vector indicating whether the feature is selected for each repetition, *i.e.* a vector with entries 0 and 1, 0 stands for the corresponding feature is not selected by FOCI, 1 stands for the corresponding feature is selected by FOCI. We then add them column by column and get the frequencies across all the features. In the end, we will get a vector of size equal to the number of predictors that counts the frequencies that a variable appears out of R times.

Results

We run from PC0 to PC20 with $N = 10000$ and $R = 20$. The result is as following:

PCs	Frequency vectors																			
	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	X_{16}	X_{17}	X_{18}	X_{19}	X_{20}
0	6	3	**20**	6	6	8	11	5	8	**20**	1	2	1	0	1	0	**20**	0	0	1
1	2	0	**20**	1	2	1	3	2	4	**20**	2	0	1	4	2	3	**20**	0	1	1
2	4	0	**20**	3	2	2	4	2	1	**20**	0	1	0	1	2	1	**20**	1	0	0
3	2	2	**20**	1	2	0	3	6	5	**18**	0	3	1	1	1	2	**19**	1	0	1
4	1	1	**20**	1	6	0	1	5	3	**19**	1	0	0	0	1	1	**19**	1	1	0
5	4	3	**17**	2	3	3	2	3	3	**17**	2	0	0	0	0	1	**20**	0	0	2
6	5	2	**19**	2	4	0	3	3	3	**17**	4	3	2	1	1	4	**20**	2	0	0
7	1	2	**20**	0	2	3	6	2	4	**19**	2	4	3	1	0	0	**19**	1	1	0
8	2	2	**17**	1	2	1	1	4	0	**19**	1	0	1	2	0	2	**19**	1	2	0
9	3	1	**16**	2	3	3	4	2	0	**18**	3	3	2	0	1	0	**16**	2	3	2
10	1	4	**18**	3	3	3	5	3	3	**17**	3	5	1	1	1	0	**18**	2	3	1
11	0	3	**20**	3	3	2	4	1	3	**17**	2	2	5	2	1	2	**17**	1	3	3
12	2	1	**20**	4	3	0	5	3	4	**20**	1	1	1	0	4	1	**16**	3	5	3
13	2	2	**20**	2	4	1	4	6	3	**18**	3	2	2	1	2	1	**17**	0	3	3
14	2	3	**18**	0	5	6	3	5	3	**17**	2	3	2	2	2	1	**16**	2	2	0
15	2	2	**18**	2	2	5	3	4	3	**18**	4	3	2	3	0	0	**16**	2	1	1
16	3	2	**17**	1	5	3	4	3	1	**19**	3	1	4	4	2	0	**14**	1	3	1
17	3	3	**18**	5	3	4	3	3	1	**18**	3	1	3	3	2	2	**14**	0	2	1
18	4	3	**19**	4	2	5	5	2	4	**19**	2	2	1	4	3	2	**17**	5	0	0
19	3	2	**19**	4	3	4	4	7	5	**20**	2	1	2	2	3	0	**17**	2	2	1
20	3	2	**16**	4	2	4	2	5	5	**17**	0	2	1	1	4	2	**17**	3	1	2

Table 4.9: Frequecy table of different number of principal components included, with $N = 10000$ and signals $X_3, X_{10},$ and X_{17} bold-faced and highlighted as light indigo.

Discussions

i.) Including zero to two principal components as the estimator of hidden variables gives us the best true positive results. It is assumed that including 3 PCs would give us the best result but in fact not, this could be due to the non-dense relationship between $\mathbf{X}$ and hidden $\mathbf{Z}$ which leads to a unprecise estimation of latent confounders.

ii.) Although including nothing which is to use the original FOCI algorithm without considering the latent confounders also selects all the true positives, it perform worse than including 1 PC or 2 PCs in the sense of controlling the false discovery proportion.

iii.) There is a significant threshold τ between the frequencies of **single** true positives and **single** false positives. Though when adding the frequencies of TP and FP up, the different will be non-significant, but in this table if we choose $\tau = 10$, we will do a perfect job of finding the Markov blanket. I believe in general case, choosing a threshold as selection criterion

$$\tau_0 = R/2 \tag{4.2.1}$$

would work well.

iv.) False positives decreases at first, and then increases with the increase of principal components (See Figure 4.5)

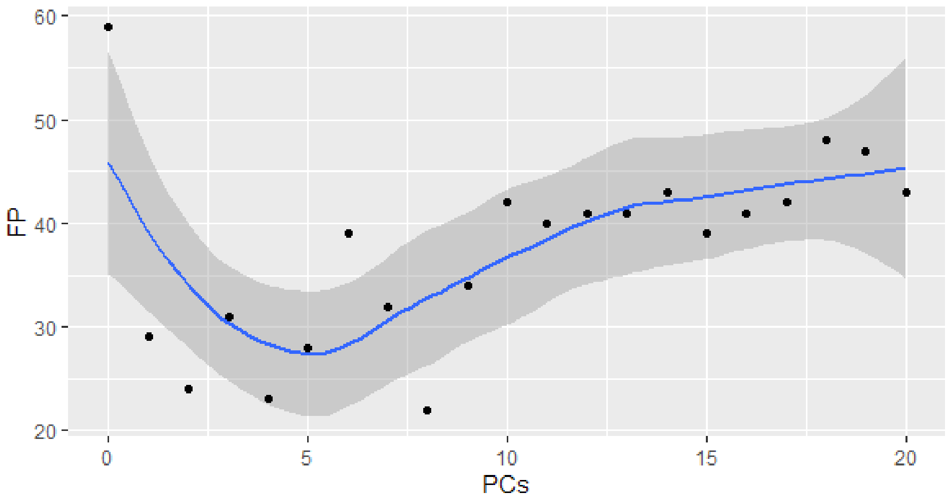

Figure 4.5: PCs vs. False Positives

v.) This experiment can be improved which is to include the corresponding average of CODEC values of selected predictors to see how likely the FOCI will select them. This may give us insights about how FOCI works.

4.2.8 Experiment 7: Variable selection in nonlinear functional relationship

Settings

Suppose we have $Y = X_3^2 + X_{10} + \sqrt{|X_{17}|} + \mathbf{Z}\beta + \varepsilon$. We also have

$$X_1, ..., X_{10} = \mathbf{Z} \cdot B + \varepsilon_1,$$

where $\mathbf{Z} = (Z_1, Z_2, Z_3) \sim \mathcal{U}(0,1)$, B is the coefficient matrix with elements generated randomly from $\mathcal{U}(0,1)$, ε and ε_1 are the random noises with standard normal distribution. Other definitions are the same as that of the Experiment 5 (Section 4.2.6). Besides, we have

$$X_{11}, ..., X_{20} \perp\!\!\!\perp \mathbf{Z},$$

this makes the effect of $\mathbf{Z}$ on X not dense.

To summarize, we have $\mathbf{X} = (X_1, ..., X_{20})$, and

$$X_i = \begin{cases} \mathbf{Z} \cdot B & \text{when } i = 1, ..., 10 \\ \sim \mathcal{N}(0,1) & \text{when } i = 11, ..., 20 \end{cases} + \varepsilon_1,$$

Methods & Objectives

We desired to use the same resampling scheme without replacement to select the Markov blanket. Besides, in this experiment we will try to calculate the average CODEC value of the selected variables, which is

$$\text{Average CODEC value of } X_k = \Gamma := \frac{\sum \gamma}{\text{The size of the set of selected } X_k},$$

where γ is the CODEC value when $X_k, k \in [s]$ is selected and its expression can be written as $\text{CODEC}(Y, X_k|\mathbf{Z}_{est})$.

We also want to check whether the PCA estimation still works when there is nonlinearity in the functional relationship between response Y and $\mathbf{X}$.

What's more, we tried the variational autoencoders with Gaussian prior which is introduced in Section 2.4 to simulate the latent variables and put them directly as conditioning set when running the FOCI algorithm.

Finally, we also want to check if the difference between the magnitudes of each signals contributed to the response Y and the magnitudes of hidden confounders $\mathbf{Z}$ contributed to the response affect the FOCI selection process. We re-state the formula of response Y here:

$$Y = X_3^2 + X_{10} + \sqrt{|X_{17}|} + \mathbf{Z}\beta + \varepsilon.$$

Here X_3^2 refers to 'variable 1', X_{10} refers to 'variable 2', $\sqrt{|X_{17}|}$ refers to 'variable 3', and $\mathbf{Z}\beta$ refers to 'hidden confounder'.

To specify, we define the following three cases by changing β:

i.) If $\beta \sim \mathcal{U}(0, 1)$, we say that hidden confounders and signals have the same magnitudes, which is illustrated in Figure 4.6.

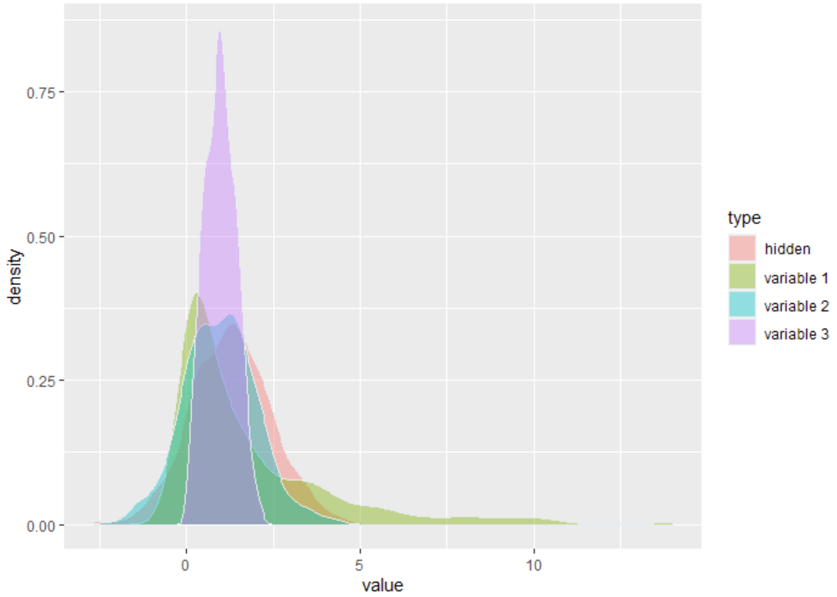

Figure 4.6: Kernel density of signals and confounders where they have same magnitudes

.

ii.) If $\beta \sim \mathcal{U}(0, 0.01)$, we say that hidden confounders are dominated by the signals in the context of magnitudes, which is illustrated in Figure 4.7.

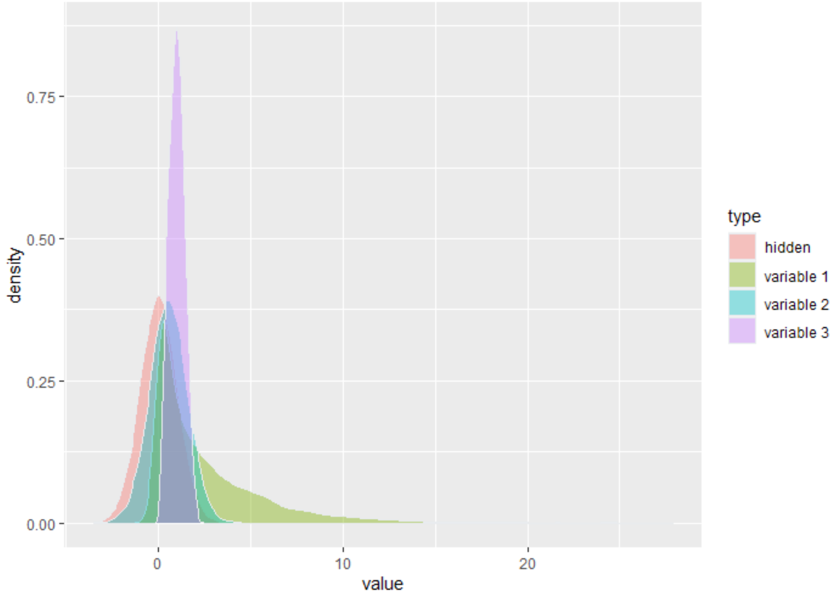

Figure 4.7: Kernel density of signals and confounders where confounders are dominated by the signals in the context of magnitude

.

iii.) If $\beta \sim \mathcal{U}(10, 20)$, we say that the signals are dominated by hidden confounders in the context of magnitudes, which is illustrated in Figure 4.8.

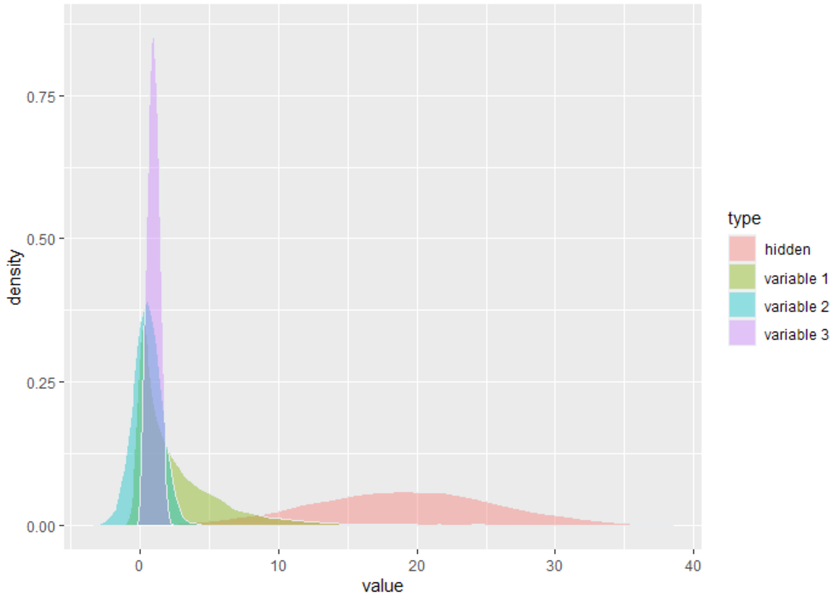

Figure 4.8: Kernel density of signals and confounders where the signals are dominated by confounders in the context of magnitude

.

Results using PCA

As for PCA estimator, we run from PC0 to PC20 with $N_1 = 10000$ / $N_2 = 500$ and using resampling schme with $R = 20$. Besides, we will compare between the three magnitude cases for N_1 and N_2.

Accordingly we generated six tables, which are given in the Table 4.10 - 4.15.

X's	PC0		PC1		PC2		PC3		PC4		PC5		PC6		PC7		PC8		PC9		PC10		PC11		PC12		PC13		PC14		PC15		PC16		PC17		PC18		PC19		PC20	
X_1	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.19	0	0.00	1	0.20	0	0.00	0	0.00	0	0.00	2	0.38
X_2	1	0.02	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.20	1	0.20	1	0.18	0	0.00	2	0.18	2	0.18	1	0.18	3	0.17	1	0.27	3	0.29	5	0.35	5	0.42
X_3	20	0.47	20	0.57	20	0.58	20	0.57	20	0.56	20	0.53	20	0.52	20	0.50	20	0.48	20	0.47	20	0.46	20	0.45	20	0.44	20	0.44	20	0.43	20	0.43	20	0.43	20	0.43	20	0.47	20	0.51	20	0.55
X_4	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.17	0	0.00	0	0.00	1	0.18	1	0.22	0	0.00	1	0.22	1	0.22	3	0.20	2	0.23	1	0.23	1	0.37	1	0.38
X_5	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.19	0	0.00	0	0.00	2	0.17	3	0.17	2	0.19	1	0.18	3	0.18	2	0.16	2	0.19	1	0.18	1	0.19	1	0.20	1	0.31	3	0.39	1	0.44
X_6	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.20	0	0.00	1	0.21	0	0.00	0	0.00	0	0.00	0	0.00	2	0.19	1	0.16	2	0.20	0	0.00	1	0.29	0	0.00	1	0.41
X_7	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.18	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	4	0.22	1	0.26	3	0.23	1	0.26	0	0.00	2	0.42
X_8	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.21	1	0.20	2	0.19	2	0.21	4	0.18	7	0.19	2	0.20	3	0.19	3	0.19	1	0.13	2	0.18	0	0.00	0	0.00	0	0.00	1	0.42
X_9	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	2	0.20	1	0.15	1	0.19	1	0.19	3	0.20	0	0.00	2	0.19	4	0.22	3	0.28	3	0.27	5	0.37	1	0.44
X_{10}	20	0.11	20	0.23	20	0.24	20	0.24	20	0.24	20	0.23	20	0.23	20	0.22	20	0.22	20	0.22	20	0.22	20	0.22	20	0.22	20	0.22	20	0.22	20	0.22	20	0.23	20	0.24	20	0.31	19	0.37	20	0.45
X_{11}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.16	0	0.00	1	0.18	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00
X_{12}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.22	0	0.00	1	0.16	0	0.00	1	0.14	2	0.18	3	0.20	1	0.18	0	0.00	0	0.00	1	0.40
X_{13}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.22	1	0.24	0	0.00	1	0.34	1	0.42
X_{14}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.19	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	2	0.45	1	0.43	0	0.00
X_{15}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.18	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.24	0	0.00	0	0.00	0	0.00
X_{16}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.21	0	0.00	2	0.20	1	0.18	1	0.16	2	0.17	1	0.24	0	0.00	1	0.42
X_{17}	20	0.01	16	0.18	12	0.19	6	0.20	4	0.20	8	0.19	6	0.21	6	0.20	6	0.17	7	0.19	13	0.19	10	0.20	7	0.19	12	0.19	13	0.20	10	0.20	12	0.22	15	0.24	17	0.30	15	0.38	19	0.44
X_{18}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.20	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	2	0.18	1	0.25	0	0.00	0	0.00
X_{19}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.19	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.21	2	0.20	0	0.00	0	0.00	1	0.21	1	0.18	0	0.00
X_{20}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	2	0.18	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	2	0.16	1	0.18	0	0.00	0	0.00	1	0.21	2	0.26	1	0.32	0	0.00

Table 4.10: Frequency & CODEC value table from PC0 to PC20 when $N = 10000$ and $\beta \sim \mathcal{U}(0,1)$ using PCA as latent variable estimator. The signals X_3, X_{10}, and X_{17} are highlighted with light indigo. For each principal components included, i.e. PCk, the left side with bold face shows the frequencies of FOCI selection (max. 20), the right side shows the corresponding averaged CODEC value γ. The signals X_3, X_{10}, and X_{17} are highlighted. The following tables follow the same rule.

X's	PC0		PC1		PC2		PC3		PC4		PC5		PC6		PC7		PC8		PC9		PC10		PC11		PC12		PC13		PC14		PC15		PC16		PC17		PC18		PC19		PC20	
X_1	0	0.00	0	0.00	2	0.17	2	0.20	4	0.19	2	0.17	3	0.21	2	0.16	2	0.19	2	0.16	3	0.20	5	0.20	0	0.00	5	0.23	4	0.30	4	0.24	1	0.17	2	0.18	4	0.15	3	0.17	0	0.00
X_2	0	0.00	1	0.08	0	0.00	1	0.19	0	0.00	4	0.20	0	0.00	5	0.15	6	0.19	2	0.15	5	0.24	1	0.27	4	0.27	0	0.00	2	0.18	2	0.25	4	0.27	3	0.32	4	0.23	5	0.26	5	0.18
X_3	20	0.37	20	0.40	20	0.41	19	0.39	19	0.39	20	0.34	19	0.32	19	0.31	17	0.30	18	0.30	17	0.31	17	0.29	17	0.32	15	0.28	17	0.28	16	0.28	18	0.28	16	0.28	17	0.28	17	0.28	16	0.27
X_4	3	-0.04	1	0.19	2	0.23	2	0.20	0	0.00	1	0.24	0	0.00	3	0.16	4	0.15	3	0.09	2	0.27	6	0.17	9	0.22	8	0.25	5	0.24	3	0.30	7	0.23	8	0.21	4	0.21	3	0.16	3	0.22
X_5	1	0.06	0	0.00	3	0.11	1	0.18	2	0.08	3	0.09	2	0.23	2	0.11	2	0.16	0	0.00	2	0.19	0	0.00	3	0.20	2	0.22	2	0.26	0	0.00	3	0.26	3	0.26	1	0.30	1	0.31	3	0.27
X_6	0	0.00	4	0.11	3	0.14	5	0.17	4	0.19	5	0.21	1	0.06	2	0.22	3	0.26	1	0.06	1	0.01	3	0.21	2	0.12	1	0.14	3	0.24	2	0.17	2	0.21	4	0.16	0	0.00	0	0.00	0	0.00
X_7	1	-0.09	0	0.00	0	0.00	2	0.29	3	0.19	4	0.08	2	0.11	2	0.16	3	0.20	5	0.19	2	0.27	3	0.25	4	0.19	2	0.10	5	0.28	4	0.21	3	0.27	3	0.32	4	0.23	3	0.35	2	0.20
X_8	0	0.00	2	0.13	4	0.19	4	0.19	4	0.21	3	0.14	3	0.17	3	0.21	3	0.18	2	0.19	2	0.28	2	0.15	3	0.24	4	0.19	0	0.00	4	0.25	2	0.25	3	0.17	3	0.14	4	0.24	5	0.18
X_9	3	-0.03	1	0.15	0	0.00	1	0.16	3	0.15	2	0.25	4	0.24	5	0.21	2	0.02	4	0.18	3	0.19	4	0.17	2	0.26	5	0.17	3	0.22	1	0.27	1	0.24	3	0.10	1	0.16	0	0.00	3	0.16
X_{10}	19	0.16	19	0.23	18	0.22	16	0.29	16	0.28	16	0.28	14	0.22	15	0.24	14	0.24	12	0.23	15	0.23	13	0.27	14	0.25	11	0.24	9	0.24	13	0.24	16	0.25	13	0.24	12	0.20	9	0.21	13	0.21
X_{11}	0	0.00	0	0.00	2	0.14	2	0.26	3	0.13	2	0.17	3	0.17	2	0.25	1	0.14	0	0.00	2	0.28	1	0.30	3	0.17	1	0.19	4	0.25	0	0.00	3	0.25	5	0.23	6	0.29	6	0.22	3	0.25
X_{12}	0	0.00	0	0.00	0	0.00	2	0.14	1	0.22	3	0.21	2	0.16	2	0.24	1	0.12	2	0.17	2	0.14	1	0.12	1	0.21	1	0.27	1	0.26	0	0.00	1	0.27	2	0.13	1	0.01	0	0.00	1	0.20
X_{13}	1	-0.01	0	0.00	0	0.00	1	0.26	0	0.00	0	0.00	0	0.00	0	0.00	2	0.14	0	0.00	0	0.00	0	0.00	1	0.11	2	0.19	1	0.32	0	0.00	0	0.00	2	0.18	0	0.00	1	0.20	0	0.00
X_{14}	1	-0.10	2	0.06	1	0.11	0	0.00	2	0.15	2	0.29	3	0.22	3	0.25	2	0.21	3	0.23	6	0.19	1	0.17	3	0.21	2	0.22	4	0.23	5	0.25	5	0.19	3	0.31	6	0.18	5	0.21	3	0.18
X_{15}	2	-0.00	0	0.00	1	0.08	2	0.25	0	0.00	1	0.31	6	0.22	4	0.17	1	0.15	2	0.19	2	0.18	2	0.16	0	0.00	2	0.09	3	0.21	1	0.23	4	0.20	3	0.15	4	0.11	2	0.16	2	0.11
X_{16}	1	0.03	1	0.03	3	0.18	1	0.22	3	0.16	1	0.19	1	0.08	1	0.27	3	0.11	1	0.21	3	0.29	3	0.20	3	0.26	2	0.37	2	0.23	4	0.29	5	0.29	4	0.23	3	0.31	3	0.16	2	0.08
X_{17}	3	0.05	0	0.00	1	-0.04	2	0.24	3	0.14	3	0.19	4	0.28	3	0.26	3	0.17	1	0.23	1	0.25	5	0.15	5	0.20	5	0.26	8	0.28	6	0.23	3	0.28	8	0.22	4	0.20	4	0.18	3	0.29
X_{18}	0	0.00	0	0.00	3	0.06	1	0.19	2	0.16	1	0.30	2	0.30	0	0.00	0	0.00	0	0.00	2	0.29	3	0.27	1	0.08	3	0.26	2	0.17	3	0.27	2	0.30	0	0.00	1	0.08	1	0.39	6	0.24
X_{19}	0	0.00	2	0.13	0	0.00	0	0.00	0	0.00	1	0.19	2	0.16	1	0.16	0	0.00	2	0.19	0	0.00	0	0.00	1	0.12	2	0.17	1	0.28	1	0.10	1	0.06	1	0.16	4	0.16	3	0.20	3	0.16
X_{20}	0	0.00	0	0.00	1	0.09	2	0.16	0	0.00	3	0.21	2	0.11	5	0.14	4	0.15	3	0.13	2	0.26	2	0.13	2	0.23	3	0.20	2	0.26	4	0.25	7	0.22	4	0.14	1	0.35	4	0.27	3	0.20

Table 4.11: Frequency & CODEC value table from PC0 to PC20 when $N = 500$ and $\beta \sim \mathcal{U}(0,1)$ using PCA as latent variable estimator. The signals X_3, X_{10}, and X_{17} are highlighted with light indigo.

X's	PC0		PC1		PC2		PC3		PC4		PC5		PC6		PC7		PC8		PC9		PC10		PC11		PC12		PC13		PC14		PC15		PC16		PC17		PC18		PC19		PC20	
X_1	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.18	0	0.00	1	0.19	0	0.00	0	0.00	1	0.27	2	0.37
X_2	1	0.02	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.18	1	0.16	0	0.00	2	0.17	4	0.18	2	0.17	4	0.17	2	0.20	2	0.30	5	0.33	5	0.41
X_3	20	0.46	20	0.55	20	0.56	20	0.55	20	0.54	20	0.52	20	0.50	20	0.48	20	0.47	20	0.45	20	0.44	20	0.43	20	0.43	20	0.42	20	0.42	20	0.41	20	0.41	20	0.42	20	0.46	20	0.50	20	0.54
X_4	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.16	0	0.00	0	0.00	1	0.17	1	0.20	0	0.00	0	0.00	1	0.20	3	0.19	1	0.23	2	0.22	1	0.36	1	0.37
X_5	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.17	0	0.00	0	0.00	3	0.16	2	0.14	2	0.17	1	0.16	3	0.16	2	0.14	2	0.17	1	0.17	1	0.17	1	0.19	1	0.30	3	0.38	1	0.43
X_6	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.19	0	0.00	0	0.00	0	0.00	0	0.00	1	0.15	0	0.00	2	0.18	0	0.00	3	0.22	0	0.00	1	0.40
X_7	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	4	0.20	0	0.00	2	0.23	1	0.24	0	0.00	3	0.41
X_8	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	2	0.18	1	0.18	2	0.17	3	0.18	4	0.17	7	0.18	3	0.17	3	0.17	2	0.16	2	0.13	2	0.17	0	0.00	0	0.00	0	0.00	1	0.41
X_9	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.18	0	0.00	0	0.00	2	0.18	0	0.00	2	0.17	1	0.18	3	0.18	1	0.21	1	0.21	4	0.20	3	0.27	3	0.26	4	0.33	1	0.43
X_{10}	20	0.11	20	0.21	20	0.22	20	0.22	20	0.22	20	0.22	20	0.22	20	0.21	20	0.21	20	0.20	20	0.21	20	0.21	20	0.21	20	0.20	20	0.21	20	0.21	20	0.22	20	0.23	20	0.29	19	0.36	20	0.44
X_{11}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.15	0	0.00	1	0.16	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00
X_{12}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.20	0	0.00	1	0.15	0	0.00	1	0.13	2	0.17	4	0.19	0	0.00	0	0.00	0	0.00	1	0.39
X_{13}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.15	0	0.00	0	0.00	1	0.20	1	0.22	1	0.25	1	0.33	0	0.00
X_{14}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.17	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.24	2	0.43	1	0.42	0	0.00
X_{15}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.17	1	0.15	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.22	0	0.00	0	0.00	0	0.00
X_{16}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.20	0	0.00	2	0.19	1	0.17	1	0.14	1	0.17	3	0.21	0	0.00	1	0.40
X_{17}	20	0.01	15	0.17	11	0.17	5	0.19	5	0.18	9	0.18	5	0.20	6	0.19	6	0.15	7	0.17	12	0.18	11	0.18	8	0.17	11	0.18	12	0.18	11	0.19	12	0.21	15	0.22	16	0.30	15	0.36	18	0.43
X_{18}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.19	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	2	0.17	1	0.23	0	0.00	0	0.00
X_{19}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.17	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	2	0.18	0	0.00	0	0.00	1	0.20	1	0.16	0	0.00
X_{20}	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	2	0.17	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.15	1	0.16	0	0.00	0	0.00	1	0.20	2	0.24	1	0.31	0	0.00

Table 4.12: Frequency & CODEC value table from PC0 to PC20 when $N = 10000$ and $\beta \sim \mathcal{U}(0, 0.01)$ using PCA as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo.

X's	PC0		PC1		PC2		PC3		PC4		PC5		PC6		PC7		PC8		PC9		PC10		PC11		PC12		PC13		PC14		PC15		PC16		PC17		PC18		PC19		PC20	
X_1	1	-0.01	0	0.00	3	0.09	2	0.16	4	0.14	4	0.16	4	0.19	1	0.14	3	0.11	2	0.07	5	0.17	7	0.16	2	0.22	3	0.15	1	0.20	2	0.27	2	0.13	5	0.13	3	0.15	2	0.15	0	0.00
X_2	0	0.00	0	0.00	0	0.00	1	0.12	1	0.10	2	0.20	0	0.00	1	-0.05	4	0.18	4	0.11	3	0.20	2	0.15	3	0.24	1	0.26	1	0.24	2	0.23	5	0.21	8	0.17	3	0.19	5	0.24	5	0.17
X_3	20	0.38	20	0.37	20	0.38	19	0.35	19	0.34	20	0.30	19	0.28	18	0.28	18	0.26	18	0.26	19	0.26	18	0.26	18	0.28	17	0.25	19	0.26	15	0.26	18	0.26	18	0.24	18	0.26	17	0.27	17	0.25
X_4	1	-0.06	2	0.11	2	0.19	2	0.13	0	0.00	1	0.17	1	0.12	3	0.02	3	0.12	6	0.13	4	0.13	6	0.12	7	0.17	6	0.21	3	0.13	3	0.18	5	0.15	5	0.14	5	0.20	5	0.18	4	0.23
X_5	1	0.00	0	0.00	4	0.09	1	0.13	3	0.05	1	0.03	2	0.20	5	0.18	2	0.10	1	0.14	2	0.08	0	0.00	2	0.17	3	0.24	3	0.20	0	0.00	2	0.22	1	0.23	1	0.24	1	0.13	3	0.22
X_6	0	0.00	3	0.05	2	0.12	4	0.12	5	0.16	4	0.16	1	0.19	4	0.14	2	0.30	1	0.33	1	-0.02	3	0.17	1	0.07	1	0.16	0	0.00	1	0.26	1	0.31	1	0.12	0	0.00	1	0.04	3	0.14
X_7	0	0.00	1	0.21	0	0.00	2	0.20	2	0.21	1	0.12	0	0.00	1	-0.02	3	0.06	5	0.17	1	0.24	5	0.20	3	0.24	2	0.21	5	0.17	3	0.14	4	0.16	4	0.16	3	0.24	5	0.22	4	0.15
X_8	2	-0.03	2	0.09	5	0.15	1	0.19	3	0.11	3	0.06	2	0.07	3	0.15	5	0.12	4	0.10	4	0.19	2	0.12	4	0.16	2	0.22	1	0.36	3	0.20	4	0.19	6	0.21	5	0.20	5	0.15	6	0.19
X_9	2	-0.03	3	0.11	0	0.00	0	0.00	1	0.19	4	0.15	2	0.20	4	0.13	1	0.11	4	0.13	5	0.17	4	0.08	2	0.18	4	0.18	4	0.16	0	0.00	0	0.00	3	0.13	0	0.00	2	0.18	3	0.08
X_{10}	19	0.16	20	0.20	19	0.20	16	0.26	14	0.26	15	0.25	13	0.21	17	0.21	16	0.21	15	0.19	17	0.20	16	0.23	17	0.21	12	0.20	11	0.21	11	0.21	9	0.20	14	0.18	11	0.17	12	0.19	14	0.21
X_{11}	0	0.00	0	0.00	1	0.20	1	0.33	1	0.04	4	0.11	3	0.13	5	0.17	0	0.00	0	0.00	3	0.21	3	0.18	1	0.24	2	0.18	4	0.17	0	0.00	3	0.20	4	0.16	6	0.24	3	0.15	5	0.18
X_{12}	1	-0.06	0	0.00	0	0.00	1	0.01	0	0.00	5	0.16	2	0.09	2	0.19	1	0.04	1	0.15	1	-0.01	1	0.25	0	0.00	1	0.25	2	0.23	1	0.08	1	0.04	0	0.00	1	0.23	1	0.23	4	0.17
X_{13}	1	-0.06	0	0.00	0	0.00	1	0.18	0	0.00	0	0.00	0	0.00	0	0.00	1	0.11	1	0.12	0	0.00	0	0.00	2	0.10	2	0.15	2	0.24	1	0.33	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00
X_{14}	2	-0.10	1	0.05	1	0.06	0	0.00	1	0.08	0	0.00	2	0.18	4	0.16	1	0.18	2	0.19	4	0.21	1	0.14	2	0.16	4	0.23	4	0.20	8	0.20	5	0.23	6	0.20	6	0.22	5	0.18	4	0.21
X_{15}	0	0.00	0	0.00	0	0.00	1	0.13	0	0.00	1	0.24	4	0.23	3	0.20	1	0.07	2	0.15	3	0.04	1	0.14	5	0.14	3	0.13	2	0.16	1	0.20	3	0.14	4	0.10	3	0.11	3	0.18	1	0.14
X_{16}	1	0.01	0	0.00	1	0.39	1	0.17	3	0.13	1	0.16	1	0.04	1	0.21	2	0.12	4	0.12	3	0.24	4	0.18	3	0.20	2	0.33	2	0.27	2	0.28	1	0.38	1	0.32	4	0.22	4	0.20	2	0.07
X_{17}	2	0.06	0	0.00	1	0.12	3	0.19	2	0.08	5	0.15	4	0.22	5	0.17	4	0.13	4	0.14	3	0.17	5	0.11	4	0.21	6	0.21	8	0.24	7	0.17	7	0.24	9	0.21	5	0.16	5	0.19	6	0.21
X_{18}	0	0.00	1	0.03	2	0.15	1	0.18	2	0.13	3	0.17	2	0.26	1	0.16	4	0.11	1	0.18	2	0.24	2	0.22	1	0.17	3	0.20	1	0.21	5	0.16	3	0.25	0	0.00	3	0.11	3	0.23	6	0.24
X_{19}	0	0.00	2	0.08	1	-0.05	1	0.13	0	0.00	1	0.12	1	0.13	2	0.10	0	0.00	1	0.12	0	0.00	1	0.18	4	0.13	2	0.15	2	0.13	1	0.24	3	0.10	2	0.18	1	0.18	2	0.15	3	0.11
X_{20}	0	0.00	0	0.00	1	0.04	3	0.09	4	0.15	3	0.12	4	0.05	4	0.10	3	0.14	2	0.09	2	0.25	4	0.15	2	0.13	3	0.13	2	0.16	3	0.21	3	0.17	4	0.14	3	0.24	3	0.23	4	0.17

Table 4.13: Frequency & CODEC value table from PC0 to PC20 when $N = 500$ and $\beta \sim \mathcal{U}(0, 0.01)$ using PCA as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo.

X's	PC0		PC1		PC2		PC3		PC4		PC5		PC6		PC7		PC8		PC9		PC10		PC11		PC12		PC13		PC14		PC15		PC16		PC17		PC18		PC19		PC20	
X_1	11	0.05	0	0.00	2	0.40	1	0.42	1	0.41	1	0.41	0	0.00	0	0.00	2	0.35	2	0.35	0	0.00	1	0.33	1	0.37	1	0.32	1	0.29	3	0.31	1	0.31	1	0.30	0	0.00	2	0.31	5	0.33
X_2	5	0.05	1	0.40	1	0.41	1	0.36	2	0.37	1	0.40	2	0.39	2	0.35	1	0.38	0	0.00	1	0.31	2	0.30	5	0.31	3	0.34	2	0.34	3	0.31	3	0.31	7	0.31	5	0.33	9	0.32	9	0.30
X_3	20	0.21	20	0.51	20	0.51	20	0.50	20	0.49	20	0.47	20	0.46	20	0.44	20	0.44	20	0.43	20	0.41	20	0.40	20	0.39	20	0.38	20	0.37	20	0.37	20	0.37	20	0.36	20	0.35	20	0.34	20	0.34
X_4	13	0.04	2	0.39	2	0.39	0	0.00	2	0.41	1	0.41	2	0.36	0	0.00	0	0.00	1	0.36	2	0.33	2	0.31	4	0.31	4	0.32	5	0.33	0	0.00	6	0.31	7	0.30	9	0.29	6	0.31	6	0.30
X_5	2	0.02	0	0.00	2	0.41	1	0.36	1	0.36	2	0.35	0	0.00	0	0.00	1	0.37	2	0.36	4	0.33	3	0.35	2	0.32	2	0.34	3	0.30	3	0.32	1	0.32	3	0.30	5	0.31	1	0.30	5	0.32
X_6	17	0.05	2	0.40	0	0.00	4	0.40	1	0.38	3	0.37	0	0.00	1	0.38	3	0.37	3	0.35	2	0.34	2	0.32	4	0.32	3	0.31	3	0.31	3	0.32	4	0.31	2	0.29	5	0.29	2	0.30	5	0.31
X_7	5	0.03	2	0.42	2	0.45	3	0.38	2	0.38	3	0.37	4	0.38	2	0.34	1	0.35	1	0.34	2	0.32	0	0.00	2	0.32	1	0.35	2	0.29	4	0.30	1	0.28	4	0.29	4	0.33	3	0.30	4	0.29
X_8	2	0.04	3	0.41	1	0.38	1	0.42	2	0.39	1	0.43	2	0.37	3	0.37	1	0.37	3	0.37	2	0.35	1	0.32	1	0.32	1	0.34	9	0.32	6	0.32	1	0.31	5	0.30	4	0.30	5	0.30	1	0.30
X_9	1	0.02	2	0.41	1	0.43	2	0.40	1	0.40	1	0.35	4	0.36	1	0.34	2	0.37	2	0.35	2	0.34	0	0.00	2	0.31	3	0.31	6	0.32	1	0.31	8	0.30	3	0.33	5	0.31	8	0.31	8	0.30
X_{10}	20	0.05	2	0.41	4	0.38	6	0.41	6	0.41	6	0.39	7	0.38	9	0.36	6	0.36	9	0.36	9	0.35	8	0.34	6	0.34	7	0.33	8	0.31	10	0.31	11	0.32	10	0.31	15	0.31	10	0.31	10	0.32
X_{11}	0	0.00	2	0.40	0	0.00	2	0.38	1	0.41	1	0.38	0	0.00	2	0.36	1	0.36	0	0.00	0	0.00	1	0.31	0	0.00	0	0.00	0	0.00	1	0.33	0	0.00	1	0.25	1	0.33	1	0.31	1	0.31
X_{12}	0	0.00	0	0.00	1	0.41	0	0.00	0	0.00	1	0.43	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.34	1	0.35	2	0.31	0	0.00	1	0.29	1	0.31	0	0.00	2	0.33	0	0.00	0	0.00
X_{13}	0	0.00	0	0.00	0	0.00	1	0.39	0	0.00	1	0.33	1	0.35	2	0.35	3	0.34	1	0.35	4	0.33	2	0.35	9	0.33	3	0.33	1	0.33	5	0.31	3	0.31	2	0.30	4	0.30	1	0.27	1	0.30
X_{14}	0	0.00	1	0.42	2	0.39	0	0.00	0	0.00	1	0.39	0	0.00	0	0.00	1	0.35	0	0.00	2	0.33	3	0.35	1	0.34	0	0.00	1	0.34	0	0.00	3	0.30	2	0.30	2	0.33	3	0.31	1	0.31
X_{15}	0	0.00	0	0.00	1	0.42	0	0.00	0	0.00	0	0.00	0	0.00	0	0.00	1	0.32	1	0.36	2	0.32	0	0.00	1	0.32	1	0.35	4	0.32	1	0.28	1	0.34	1	0.30	1	0.33	0	0.00	1	0.28
X_{16}	1	-0.01	1	0.36	1	0.43	1	0.39	2	0.38	2	0.40	2	0.40	3	0.37	1	0.36	3	0.34	1	0.34	2	0.33	2	0.31	1	0.31	0	0.00	1	0.36	5	0.32	3	0.30	0	0.00	3	0.30	4	0.31
X_{17}	0	0.00	4	0.40	0	0.00	2	0.41	1	0.38	3	0.36	2	0.38	3	0.39	1	0.33	4	0.35	0	0.00	2	0.31	3	0.33	3	0.32	3	0.33	2	0.32	3	0.32	6	0.31	4	0.28	3	0.31	7	0.32
X_{18}	0	0.00	2	0.40	0	0.00	1	0.37	2	0.38	1	0.38	0	0.00	1	0.34	0	0.00	0	0.00	1	0.33	0	0.00	1	0.32	1	0.34	2	0.33	0	0.00	2	0.30	1	0.27	0	0.00	0	0.00	0	0.00
X_{19}	2	-0.02	0	0.00	2	0.41	1	0.40	1	0.39	1	0.37	2	0.36	3	0.40	3	0.39	1	0.33	0	0.00	1	0.35	2	0.34	2	0.33	3	0.30	2	0.28	2	0.32	1	0.29	1	0.29	2	0.31	1	0.32
X_{20}	0	0.00	4	0.39	0	0.00	1	0.40	0	0.00	0	0.00	2	0.38	5	0.36	3	0.37	4	0.36	2	0.36	3	0.33	3	0.33	2	0.31	2	0.32	2	0.32	3	0.31	3	0.29	2	0.30	4	0.30	2	0.31

Table 4.14: Frequency & CODEC value table from PC0 to PC20 when $N = 10000$ and $\beta \sim \mathcal{U}(10, 20)$ using PCA as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo.

X's	PC0		PC1		PC2		PC3		PC4		PC5		PC6		PC7		PC8		PC9		PC10		PC11		PC12		PC13		PC14		PC15		PC16		PC17		PC18		PC19		PC20	
X_1	**7**	0.12	**2**	0.41	**3**	0.43	**1**	0.35	**0**	0.00	**2**	0.38	**4**	0.27	**3**	0.33	**1**	0.29	**4**	0.30	**3**	0.23	**6**	0.29	**6**	0.25	**4**	0.23	**9**	0.22	**2**	0.22	**6**	0.20	**4**	0.28	**6**	0.25	**2**	0.26	**3**	0.30
X_2	**0**	0.00	**0**	0.00	**2**	0.38	**0**	0.00	**4**	0.34	**3**	0.20	**4**	0.37	**3**	0.28	**6**	0.23	**5**	0.31	**6**	0.28	**4**	0.23	**3**	0.27	**4**	0.25	**4**	0.29	**3**	0.34	**2**	0.15	**4**	0.19	**4**	0.25	**3**	0.24	**5**	0.21
X_3	**9**	0.13	**8**	0.38	**3**	0.31	**5**	0.43	**7**	0.42	**6**	0.32	**6**	0.30	**3**	0.34	**6**	0.27	**8**	0.31	**8**	0.31	**5**	0.27	**4**	0.34	**11**	0.25	**4**	0.25	**6**	0.24	**8**	0.30	**6**	0.29	**7**	0.28	**7**	0.25	**9**	0.24
X_4	**3**	0.13	**3**	0.39	**5**	0.35	**5**	0.40	**4**	0.26	**1**	0.35	**2**	0.31	**3**	0.24	**3**	0.17	**3**	0.24	**4**	0.33	**4**	0.27	**3**	0.19	**3**	0.26	**6**	0.24	**5**	0.24	**6**	0.26	**0**	0.00	**7**	0.29	**9**	0.24	**10**	0.25
X_5	**5**	0.07	**3**	0.30	**0**	0.00	**4**	0.40	**4**	0.24	**4**	0.29	**7**	0.24	**5**	0.29	**4**	0.41	**5**	0.40	**4**	0.30	**4**	0.31	**3**	0.20	**2**	0.32	**5**	0.27	**2**	0.27	**5**	0.24	**4**	0.22	**8**	0.20	**6**	0.24	**4**	0.23
X_6	**5**	0.13	**4**	0.41	**2**	0.30	**5**	0.33	**3**	0.38	**5**	0.22	**3**	0.27	**5**	0.29	**3**	0.22	**5**	0.25	**3**	0.31	**5**	0.31	**2**	0.40	**8**	0.30	**6**	0.29	**4**	0.33	**6**	0.20	**5**	0.25	**5**	0.24	**7**	0.18	**8**	0.22
X_7	**2**	-0.00	**4**	0.38	**3**	0.45	**5**	0.37	**0**	0.00	**0**	0.00	**2**	0.20	**2**	0.40	**2**	0.41	**2**	0.47	**2**	0.23	**4**	0.25	**2**	0.34	**6**	0.21	**4**	0.29	**3**	0.16	**3**	0.29	**1**	0.35	**1**	0.11	**5**	0.12	**1**	0.23
X_8	**4**	0.05	**1**	0.37	**2**	0.43	**3**	0.41	**1**	0.23	**1**	0.19	**4**	0.31	**2**	0.41	**4**	0.30	**5**	0.34	**2**	0.29	**2**	0.26	**6**	0.30	**5**	0.30	**5**	0.22	**1**	0.28	**1**	0.08	**1**	0.37	**2**	0.29	**2**	0.11	**2**	0.15
X_9	**2**	-0.10	**3**	0.33	**2**	0.23	**2**	0.31	**1**	0.18	**6**	0.33	**1**	0.47	**2**	0.29	**4**	0.29	**3**	0.28	**6**	0.25	**5**	0.18	**3**	0.26	**2**	0.27	**4**	0.26	**2**	0.32	**5**	0.20	**5**	0.26	**6**	0.21	**4**	0.22	**3**	0.24
X_{10}	**6**	0.10	**1**	0.37	**3**	0.35	**2**	0.36	**3**	0.36	**3**	0.29	**2**	0.31	**5**	0.25	**3**	0.37	**3**	0.44	**3**	0.34	**3**	0.31	**3**	0.30	**4**	0.27	**4**	0.19	**3**	0.34	**6**	0.18	**5**	0.27	**4**	0.26	**2**	0.34	**3**	0.24
X_{11}	**0**	0.00	**1**	0.32	**3**	0.44	**1**	0.38	**4**	0.32	**4**	0.23	**0**	0.00	**1**	0.15	**3**	0.31	**3**	0.29	**3**	0.27	**4**	0.26	**3**	0.23	**4**	0.20	**4**	0.24	**4**	0.25	**5**	0.28	**5**	0.28	**4**	0.22	**5**	0.19	**5**	0.27
X_{12}	**1**	0.02	**3**	0.38	**1**	0.40	**1**	0.47	**2**	0.29	**1**	0.20	**1**	0.22	**2**	0.19	**3**	0.26	**4**	0.33	**4**	0.25	**4**	0.30	**1**	0.34	**3**	0.25	**2**	0.30	**1**	0.24	**1**	0.24	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_{13}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**2**	0.22	**1**	0.16	**2**	0.34	**5**	0.30	**2**	0.32	**1**	0.20	**1**	0.25	**0**	0.00	**4**	0.30	**4**	0.18	**3**	0.18	**2**	0.34	**1**	0.18	**1**	0.27	**2**	0.29	**2**	0.39	**1**	0.15
X_{14}	**0**	0.00	**2**	0.35	**1**	0.28	**2**	0.34	**2**	0.33	**6**	0.38	**2**	0.27	**4**	0.28	**2**	0.31	**3**	0.22	**2**	0.26	**2**	0.27	**6**	0.29	**2**	0.28	**3**	0.22	**3**	0.32	**3**	0.26	**1**	0.11	**3**	0.25	**3**	0.20	**2**	0.29
X_{15}	**1**	0.17	**2**	0.41	**2**	0.41	**0**	0.00	**3**	0.24	**2**	0.30	**3**	0.37	**2**	0.42	**2**	0.23	**2**	0.38	**2**	0.15	**5**	0.27	**6**	0.30	**3**	0.26	**6**	0.31	**5**	0.19	**4**	0.32	**3**	0.32	**5**	0.28	**5**	0.19	**4**	0.30
X_{16}	**1**	0.03	**2**	0.45	**2**	0.50	**4**	0.34	**4**	0.35	**4**	0.37	**4**	0.33	**1**	0.34	**3**	0.23	**3**	0.31	**2**	0.33	**4**	0.30	**0**	0.00	**5**	0.21	**4**	0.25	**5**	0.31	**3**	0.34	**3**	0.26	**3**	0.22	**2**	0.18	**3**	0.21
X_{17}	**1**	0.09	**3**	0.32	**1**	0.17	**1**	0.25	**2**	0.33	**3**	0.27	**4**	0.35	**5**	0.35	**5**	0.30	**6**	0.30	**6**	0.30	**5**	0.32	**6**	0.33	**6**	0.26	**6**	0.27	**7**	0.29	**4**	0.29	**4**	0.26	**6**	0.21	**4**	0.20	**0**	0.00
X_{18}	**1**	-0.02	**1**	0.23	**2**	0.31	**2**	0.41	**1**	0.47	**4**	0.29	**2**	0.36	**3**	0.34	**4**	0.29	**0**	0.00	**2**	0.26	**1**	0.29	**2**	0.13	**2**	0.33	**1**	0.31	**2**	0.26	**2**	0.14	**0**	0.00	**1**	0.18	**0**	0.00	**2**	0.22
X_{19}	**1**	-0.03	**2**	0.25	**1**	0.47	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.32	**0**	0.00	**2**	0.24	**2**	0.33	**1**	0.19	**2**	0.29	**1**	0.17	**2**	0.30	**2**	0.27	**3**	0.28	**5**	0.20	**4**	0.24	**4**	0.27	**4**	0.20	**1**	0.27
X_{20}	**0**	0.00	**2**	0.32	**1**	0.29	**3**	0.35	**0**	0.00	**1**	0.29	**2**	0.12	**1**	0.30	**0**	0.00	**4**	0.37	**1**	0.25	**0**	0.00	**1**	0.35	**1**	0.29	**2**	0.30	**1**	0.17	**1**	0.09	**2**	0.28	**1**	0.30	**1**	0.10	**2**	0.28

Table 4.15: Frequency & CODEC value table from PC0 to PC20 when $N = 500$ and $\beta \sim \mathcal{U}(10, 20)$ using PCA as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo.

Results using variational autoencoder

Our VAE training process can be summarized as following:

- First sample data with given functional relationship and randomly split the whole dataset, which is the data matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ where the cardinality of training set : the cardinality of validation set $= 4 : 1$.
- Feed the mini-batch data into the variational autoencoder and started to train using the ELBO objective 2.4.3 for each epoch.
- Update the parameters using gradient-based optimization, in our context we used Adam optimizer (Kingma and Ba, 2017).
- Stop training if the validation loss does not keep decreasing within five epochs.

Our VAE architecture can be summarized as the following table and figure:

Layer (type)	Output Shape	Remarks
Input_layer	(`batch_size`, 20)	`batch_size` refers to size of mini-batch
Linear_1	(`batch_size`, 256)	
Softplus_1	(`batch_size`, 256)	Activation function
Linear_2	(`batch_size`, 256)	
BatchNorm1d_1	(`batch_size`, 128)	Batch normalization
Softplus_2	(`batch_size`, 128)	Activation function
Linear_3	(`batch_size`, 64)	
BatchNorm1d_2	(`batch_size`, 64)	Batch normalization
Softplus_3	(`batch_size`, 64)	Activation function
Linear_4	(`batch_size`, 8)	
Linear_μ	(`batch_size`, `lat_dim`)	Encode the mean of Z
Linear_σ	(`batch_size`, `lat_dim`)	Encode the std. variance of Z
Linear_mapping	(`batch_size`, 8)	Map the (-1, `lat_dim`) to (-1, 8) tensor
Linear_5	(`batch_size`, 8)	
Linear_6	(`batch_size`, 16)	
Softplus_4	(`batch_size`, 16)	Activation function
BatchNorm1d_3	(`batch_size`, 16)	Batch normalization
Linear_7	(`batch_size`, 64)	
Softplus_5	(`batch_size`, 64)	Activation function
BatchNorm1d_4	(`batch_size`, 64)	Batch normalization
Linear_8	(`batch_size`, 128)	
Softplus_6	(`batch_size`, 128)	Activation function
BatchNorm1d_5	(`batch_size`, 128)	Batch normalization
Linear_9	(`batch_size`, 20)	Output layer

Table 4.16: Variational autoencoder architecture with mini-batch input. The latent space encoding and mapping layers are highlighted with light indigo and activation functions highlighted with light gray. Note that our dataset has the dimension $n \times p$. Besides, `lat_dim` refers to the dimension of latent space, $(-1, x)$ refers to a tensor (with N data points) with dimension $\frac{N}{x} \times x$.

After training the VAE, we then save its architecture and sampling from the estimated posterior distribution to generate $\mathbf{Z}_{est}$.

As for VAE estimator, we run from including the 1-dimensional estimated confounders to including 20-dimensional estimated confounders with $N_1 = 10000$ / $N_2 = 500$ and using resampling scheme with $R = 20$. In addition, we use three different type of β which is introduced beforehand.

Accordingly we generated another six tables, which are given in the Table 4.17 - 4.22.

X's	D0		D1		D2		D3		D4		D5		D6		D7		D8		D9		D10		D11		D12		D13		D14		D15		D16		D17		D18		D19		D20	
X_1	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.03	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**2**	-0.00	**1**	-0.01	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_2	**2**	0.02	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.01	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.00	**0**	0.00
X_3	**20**	0.47	**20**	0.46	**20**	0.47	**20**	0.46	**20**	0.46	**20**	0.47	**20**	0.47	**20**	0.45	**20**	0.46	**20**	0.44	**20**	0.45	**20**	0.45	**20**	0.44	**20**	0.44	**20**	0.44	**20**	0.42	**20**	0.44	**20**	0.44	**20**	0.43	**20**	0.43	**20**	0.42
X_4	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.02	**2**	0.01	**0**	0.00	**1**	-0.02
X_5	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.01	**1**	0.02
X_6	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.01	**0**	0.00	**1**	-0.01	**1**	0.01	**0**	0.00	**1**	0.01	**5**	0.01	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_7	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.02	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_8	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**2**	0.01	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.02	**0**	0.00	**0**	0.00	**1**	0.01	**0**	0.00	**1**	0.01	**0**	0.00	**0**	0.00
X_9	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.01	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_{10}	**20**	0.11	**20**	0.11	**20**	0.11	**20**	0.10	**20**	0.11	**20**	0.11	**20**	0.10	**20**	0.10	**20**	0.11	**20**	0.10	**20**	0.11	**20**	0.11	**20**	0.11	**20**	0.11	**20**	0.10	**20**	0.10	**20**	0.11	**20**	0.11	**20**	0.10	**20**	0.11	**20**	0.10
X_{11}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_{12}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_{13}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_{14}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_{15}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**2**	0.01	**2**	0.00	**0**	0.00
X_{16}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.01	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_{17}	**20**	0.01	**20**	0.01	**18**	0.01	**19**	0.01	**18**	0.02	**19**	0.01	**14**	0.02	**14**	0.02	**15**	0.02	**17**	0.01	**11**	0.01	**14**	0.03	**17**	0.01	**16**	0.02	**8**	0.02	**19**	0.01	**16**	0.00	**9**	0.01	**10**	-0.00	**6**	-0.00	**20**	0.01
X_{18}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.01	**0**	0.00
X_{19}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.00	**0**	0.00	**1**	-0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_{20}	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.01	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.01	**0**	0.00	**2**	0.01	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00

Table 4.17: Frequency & CODEC value table from D1 (1-dimensional) to D20 (20-dimensional) when $N = 10000$ and $\beta \sim \mathcal{U}(0,1)$ using VAE as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo. For each principal components included, i.e. Dk, the left side with bold face shows the frequencies of FOCI selection (max. 20), the right side shows the corresponding averaged CODEC value γ. The following tables follow the same rule.

X's	D0		D1		D2		D3		D4		D5		D6		D7		D8		D9		D10		D11		D12		D13		D14		D15		D16		D17		D18		D19		D20	
X_1	**0**	0.00	**2**	-0.00	**2**	0.05	**0**	0.00	**1**	0.07	**0**	0.00	**1**	0.04	**2**	0.06	**0**	0.00	**2**	0.02	**2**	0.03	**2**	0.10	**3**	0.18	**0**	0.00	**3**	0.01	**1**	0.14	**0**	0.00	**2**	0.03	**2**	0.05	**1**	0.03	**3**	0.09
X_2	**0**	0.00	**1**	-0.06	**1**	0.11	**1**	0.07	**0**	0.00	**2**	0.00	**0**	0.00	**1**	-0.15	**1**	-0.15	**0**	0.00	**0**	0.00	**1**	-0.03	**1**	0.04	**0**	0.00	**1**	0.02	**1**	0.06	**0**	0.00	**0**	0.00	**1**	0.13	**0**	0.00	**1**	0.18
X_3	**20**	0.37	**20**	0.34	**20**	0.32	**20**	0.35	**20**	0.34	**20**	0.29	**20**	0.31	**20**	0.28	**20**	0.34	**20**	0.36	**20**	0.34	**20**	0.29	**20**	0.35	**20**	0.32	**20**	0.30	**20**	0.33	**20**	0.25	**20**	0.21	**20**	0.35	**20**	0.32	**20**	0.34
X_4	**3**	-0.04	**1**	0.02	**1**	-0.02	**4**	0.02	**0**	0.00	**1**	-0.13	**0**	0.00	**0**	0.00	**2**	-0.06	**1**	0.20	**2**	-0.01	**1**	-0.03	**1**	0.15	**1**	0.10	**0**	0.00	**1**	0.04	**1**	-0.12	**0**	0.00	**2**	0.05	**0**	0.00	**0**	0.00
X_5	**1**	0.06	**2**	-0.01	**1**	0.02	**1**	-0.07	**4**	-0.02	**0**	0.00	**0**	0.00	**4**	-0.03	**0**	0.00	**2**	0.04	**0**	0.00	**2**	-0.02	**0**	0.00	**0**	0.00	**1**	0.11	**2**	0.01	**1**	0.09	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.05
X_6	**0**	0.00	**3**	0.02	**1**	0.06	**3**	0.11	**1**	0.00	**5**	0.09	**2**	0.03	**3**	0.06	**1**	0.14	**1**	0.01	**3**	0.08	**3**	0.06	**1**	0.06	**6**	0.06	**2**	-0.01	**4**	-0.01	**5**	0.05	**6**	-0.04	**0**	0.00	**3**	0.12	**5**	0.12
X_7	**1**	-0.09	**1**	0.01	**1**	0.05	**1**	0.01	**2**	-0.06	**0**	0.00	**0**	0.00	**2**	-0.03	**1**	0.07	**3**	0.06	**1**	0.03	**0**	0.00	**0**	0.00	**0**	0.00	**2**	-0.01	**4**	-0.02	**1**	0.02	**1**	-0.03	**0**	0.00	**0**	0.00	**1**	-0.01
X_8	**0**	0.00	**1**	0.02	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.04	**0**	0.00	**1**	0.04	**1**	0.01	**0**	0.00	**3**	-0.06	**0**	0.00	**0**	0.00	**1**	-0.03	**0**	0.00	**1**	0.10	**4**	-0.04	**1**	-0.01	**1**	0.03	**1**	0.23
X_9	**3**	-0.03	**3**	-0.08	**2**	-0.06	**1**	0.06	**1**	-0.08	**2**	0.02	**1**	0.00	**0**	0.00	**2**	0.10	**2**	0.05	**1**	0.04	**2**	0.10	**1**	0.17	**0**	0.00	**1**	0.04	**1**	-0.14	**1**	0.00	**4**	-0.07	**0**	0.00	**3**	-0.06	**2**	0.02
X_{10}	**19**	0.16	**19**	0.17	**20**	0.16	**18**	0.15	**20**	0.14	**20**	0.15	**20**	0.15	**20**	0.13	**20**	0.16	**18**	0.14	**19**	0.08	**20**	0.13	**20**	0.20	**18**	0.10	**20**	0.08	**20**	0.15	**20**	0.14	**20**	0.06	**20**	0.18	**20**	0.17	**20**	0.17
X_{11}	**0**	0.00	**1**	0.03	**1**	0.02	**1**	0.05	**1**	0.04	**1**	-0.02	**1**	0.01	**1**	-0.11	**1**	-0.03	**1**	0.01	**4**	0.07	**1**	-0.05	**1**	-0.10	**0**	0.00	**4**	-0.01	**1**	-0.00	**0**	0.00	**3**	-0.01	**2**	0.05	**0**	0.00	**2**	0.17
X_{12}	**0**	0.00	**1**	-0.04	**1**	0.05	**3**	0.01	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.04	**0**	0.00	**0**	0.00	**0**	0.00	**1**	0.07	**4**	0.03	**0**	0.00	**1**	0.08	**0**	0.00	**0**	0.00	**2**	0.01	**0**	0.00
X_{13}	**1**	-0.01	**0**	0.00	**0**	0.00	**1**	0.02	**2**	-0.11	**1**	0.05	**2**	0.00	**2**	0.05	**0**	0.00	**3**	-0.01	**2**	-0.09	**2**	0.05	**1**	-0.09	**2**	0.04	**1**	-0.09	**1**	0.08	**2**	-0.01	**0**	0.00	**0**	0.00	**4**	0.04	**3**	0.11
X_{14}	**1**	-0.10	**3**	-0.01	**3**	-0.02	**1**	-0.07	**0**	0.00	**2**	0.09	**1**	-0.05	**3**	-0.08	**2**	0.07	**1**	0.05	**1**	0.02	**2**	0.09	**1**	0.20	**0**	0.00	**1**	-0.02	**3**	-0.01	**0**	0.00	**0**	0.00	**1**	0.28	**1**	0.15	**2**	0.03
X_{15}	**2**	-0.00	**0**	0.00	**3**	-0.07	**1**	0.12	**3**	-0.11	**1**	-0.01	**0**	0.00	**1**	-0.12	**1**	0.09	**0**	0.00	**4**	-0.01	**1**	0.00	**0**	0.00	**2**	-0.06	**0**	0.00	**0**	0.00	**0**	0.00	**5**	-0.07	**6**	0.00	**3**	0.04	**3**	0.08
X_{16}	**1**	0.03	**0**	0.00	**1**	0.06	**2**	0.04	**2**	0.01	**1**	0.00	**1**	0.05	**3**	0.01	**1**	0.09	**2**	0.09	**0**	0.00	**2**	-0.03	**0**	0.00	**3**	0.00	**1**	-0.04	**1**	0.00	**2**	-0.07	**1**	-0.11	**2**	-0.00	**0**	0.00	**2**	0.03
X_{17}	**3**	0.05	**2**	0.01	**3**	0.01	**2**	0.16	**2**	-0.06	**1**	-0.01	**2**	0.09	**1**	0.03	**1**	0.05	**4**	0.01	**0**	0.00	**3**	0.06	**1**	0.08	**2**	-0.07	**1**	0.00	**1**	0.02	**9**	-0.08	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00
X_{18}	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.05	**0**	0.00	**1**	-0.06	**1**	-0.11	**0**	0.00	**5**	0.04	**0**	0.00	**1**	-0.10	**0**	0.00	**2**	-0.05	**2**	0.02	**3**	-0.04	**2**	0.04	**2**	-0.06	**1**	-0.17	**1**	0.14	**0**	0.00	**0**	0.00
X_{19}	**0**	0.00	**1**	0.10	**0**	0.00	**0**	0.00	**1**	-0.18	**0**	0.00	**1**	0.04	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**2**	0.09	**0**	0.00	**0**	0.00	**0**	0.00	**0**	0.00	**1**	-0.02	**1**	-0.11	**0**	0.00	**0**	0.00	**1**	0.07
X_{20}	**0**	0.00	**0**	0.00	**1**	0.07	**1**	0.02	**1**	0.07	**4**	-0.01	**1**	-0.03	**0**	0.00	**0**	0.00	**2**	0.09	**0**	0.00	**2**	0.13	**3**	0.13	**0**	0.00	**1**	0.08	**1**	-0.09	**1**	0.10	**6**	-0.07	**1**	0.06	**2**	-0.06	**1**	0.14

Table 4.18: Frequency & CODEC value table from D1 (1-dimensional) to D20 (20-dimensional) when $N = 500$ and $\beta \sim \mathcal{U}(0,1)$ using VAE as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo.

X's	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20
X_1	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.03	0 0.00	0 0.00	0 0.00	0 0.00	2 -0.00	1 -0.01	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_2	2 0.02	0 0.00	0 0.00	0 0.00	1 0.01	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.01	0 0.00
X_3	20 0.46	20 0.46	20 0.47	20 0.46	20 0.46	20 0.47	20 0.47	20 0.45	20 0.46	20 0.44	20 0.45	20 0.45	20 0.43	20 0.44	20 0.44	20 0.42	20 0.44	20 0.44	20 0.42	20 0.43	20 0.41
X_4	0 0.00	0 0.00	0 0.00	1 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.02	2 0.01	0 0.00	1 -0.02
X_5	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.01	0 0.00
X_6	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.01	0 0.00	0 0.00	1 0.01	5 0.01	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_7	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.00	0 0.00	0 0.00	0 0.00	1 0.02	0 0.00	0 0.00	0 0.00	1 -0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_8	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	2 0.01	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.02	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.03	0 0.00
X_9	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.01	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_{10}	20 0.11	20 0.11	20 0.11	20 0.10	20 0.11	20 0.11	20 0.10	20 0.10	20 0.11	20 0.10	20 0.11	20 0.11	20 0.11	20 0.11	20 0.10	20 0.10	20 0.11	20 0.10	20 0.10	20 0.11	20 0.10
X_{11}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_{12}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_{13}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.00	0 0.00	0 0.00	0 0.00	1 -0.01	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_{14}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_{15}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.01	2 0.00	0 0.00
X_{16}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	2 -0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_{17}	20 0.01	20 0.01	19 0.01	20 0.01	18 0.02	19 0.01	14 0.02	14 0.02	17 0.02	17 0.01	11 0.01	14 0.03	16 0.01	16 0.02	8 0.02	19 0.01	18 0.01	9 0.01	11 -0.00	7 0.00	20 0.01
X_{18}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.01	0 0.00
X_{19}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.00	0 0.00	1 -0.00	0 0.00	0 0.00	0 0.00	0 0.00
X_{20}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.01	0 0.00	0 0.00	0 0.00	1 -0.01	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00

Table 4.19: Frequency & CODEC value table from D1 (1-dimensional) to D20 (20-dimensional) when $N = 10000$ and $\beta \sim \mathcal{U}(0, 0.01)$ using VAE as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo.

X's	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20
X_1	1 -0.01	2 -0.00	0 0.00	0 0.00	2 0.03	1 -0.08	1 0.05	3 0.04	1 0.21	1 0.12	2 0.03	3 0.02	2 0.09	0 0.00	2 -0.03	1 0.11	1 -0.10	1 0.02	0 0.00	2 -0.01	3 0.10
X_2	0 0.00	2 -0.05	2 0.02	0 0.00	0 0.00	2 -0.02	1 0.06	0 0.00	2 -0.06	1 0.09	0 0.00	0 0.00	1 0.07	1 0.03	1 0.03	1 0.02	1 0.05	0 0.00	1 0.27	0 0.00	0 0.00
X_3	20 0.38	20 0.34	20 0.33	20 0.37	20 0.34	20 0.29	20 0.32	20 0.28	20 0.34	20 0.38	20 0.34	20 0.29	20 0.36	20 0.34	20 0.31	20 0.35	20 0.25	20 0.23	20 0.32	20 0.31	20 0.34
X_4	1 -0.06	0 0.00	1 0.00	2 0.03	0 0.00	1 -0.11	0 0.00	2 -0.03	1 -0.01	1 0.22	2 -0.02	1 -0.05	1 0.15	0 0.00	0 0.00	0 0.00	0 0.00	3 -0.04	1 0.08	0 0.00	0 0.00
X_5	1 0.00	1 0.00	1 -0.00	0 0.00	1 -0.01	0 0.00	0 0.00	2 -0.08	1 0.01	0 0.00	0 0.00	1 -0.04	0 0.00	0 0.00	0 0.00	1 -0.14	0 0.00	0 0.00	1 0.02	0 0.00	0 0.00
X_6	0 0.00	2 0.03	1 0.04	1 0.10	1 0.01	6 0.09	3 -0.03	3 0.04	1 0.12	1 -0.02	5 0.05	2 -0.01	1 0.14	6 0.05	3 -0.01	4 -0.00	1 0.05	5 -0.11	1 0.28	1 0.05	4 0.10
X_7	0 0.00	1 0.03	1 0.05	1 0.04	0 0.00	0 0.00	1 -0.01	1 -0.00	0 0.00	4 0.02	1 0.02	0 0.00	0 0.00	0 0.00	2 0.03	3 -0.06	1 0.02	0 0.00	1 0.03	0 0.00	0 0.00
X_8	2 -0.03	1 0.11	0 0.00	2 -0.00	1 -0.04	0 0.00	2 0.05	1 -0.03	1 0.02	1 0.02	3 0.01	2 -0.03	1 0.04	0 0.00	3 -0.02	0 0.00	2 0.01	7 -0.06	3 0.01	1 0.02	3 0.15
X_9	2 -0.03	5 -0.05	4 0.00	2 0.02	3 -0.01	1 0.05	1 0.02	5 -0.06	2 0.09	2 0.03	1 0.08	3 0.03	0 0.00	0 0.00	1 0.03	1 -0.17	1 -0.02	3 -0.04	0 0.00	3 -0.05	1 -0.07
X_{10}	19 0.16	20 0.17	19 0.16	19 0.16	20 0.13	20 0.14	20 0.16	20 0.14	20 0.16	19 0.13	20 0.09	20 0.13	20 0.21	19 0.10	20 0.07	20 0.15	20 0.13	20 0.06	20 0.17	20 0.17	20 0.18
X_{11}	0 0.00	1 0.02	3 -0.03	1 0.13	0 0.00	2 -0.05	1 0.02	1 -0.12	2 0.02	0 0.00	4 0.06	2 -0.02	3 -0.05	0 0.00	4 -0.02	1 0.01	0 0.00	2 0.00	3 -0.03	0 0.00	3 0.15
X_{12}	1 -0.06	0 0.00	0 0.00	1 0.03	1 -0.06	1 -0.05	1 -0.02	0 0.00	0 0.00	1 -0.06	0 0.00	0 0.00	0 0.00	1 0.09	4 -0.00	2 0.07	1 0.06	0 0.00	0 0.00	2 0.01	0 0.00
X_{13}	1 -0.06	0 0.00	0 0.00	0 0.00	0 0.00	1 0.02	0 0.00	2 -0.02	0 0.00	3 -0.00	2 -0.07	1 0.09	1 -0.07	0 0.00	2 -0.05	1 0.02	4 -0.01	0 0.00	0 0.00	4 0.01	2 0.08
X_{14}	2 -0.10	2 -0.01	2 -0.01	0 0.00	1 -0.02	0 0.00	1 -0.02	1 -0.07	1 0.13	0 0.00	1 0.05	2 0.07	0 0.00	0 0.00	1 0.04	2 -0.01	0 0.00	1 0.03	1 0.28	1 0.12	0 0.00
X_{15}	0 0.00	1 0.13	3 -0.09	1 0.09	4 -0.09	0 0.00	0 0.00	1 -0.05	1 0.01	0 0.00	5 -0.01	0 0.00	1 -0.02	1 0.06	3 0.00	1 -0.03	0 0.00	5 -0.08	3 -0.07	4 0.04	4 0.10
X_{16}	1 0.01	0 0.00	1 0.04	2 0.04	2 -0.14	2 -0.01	1 -0.01	3 0.01	1 0.08	3 0.07	0 0.00	3 0.13	0 0.00	2 0.02	1 -0.03	2 0.09	1 -0.10	1 -0.08	1 0.00	0 0.00	2 0.02
X_{17}	2 0.06	3 0.00	3 0.02	0 0.00	3 -0.03	2 -0.03	2 0.09	1 0.04	2 0.06	3 -0.00	0 0.00	3 0.04	2 0.04	0 0.00	2 0.00	2 0.04	4 -0.05	1 -0.13	1 -0.15	0 0.00	0 0.00
X_{18}	0 0.00	0 0.00	0 0.00	3 -0.03	0 0.00	1 -0.10	1 -0.00	1 -0.02	5 0.01	0 0.00	1 -0.10	2 -0.03	1 -0.02	3 0.03	2 -0.04	1 -0.01	2 -0.03	0 0.00	2 0.09	0 0.00	0 0.00
X_{19}	0 0.00	2 0.08	0 0.00	0 0.00	1 -0.04	1 0.05	1 0.04	0 0.00	0 0.00	0 0.00	0 0.00	1 0.07	0 0.00	1 0.18	0 0.00	0 0.00	4 0.03	0 0.00	0 0.00	0 0.00	0 0.00
X_{20}	0 0.00	0 0.00	0 0.00	2 -0.05	0 0.00	4 -0.03	0 0.00	0 0.00	1 -0.15	2 0.10	0 0.00	0 0.00	1 0.18	1 0.00	0 0.00	0 0.00	0 0.00	3 -0.12	1 0.07	2 -0.02	2 0.08

Table 4.20: Frequency & CODEC value table from D1 (1-dimensional) to D20 (20-dimensional) when $N = 500$ and $\beta \sim \mathcal{U}(0, 0.01)$ using VAE as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo.

X's	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20
X_1	11 0.05	13 0.04	9 0.03	12 0.05	15 0.04	17 0.05	12 0.05	9 0.02	15 0.04	17 0.04	12 0.04	16 0.04	16 0.05	14 0.05	15 0.05	20 0.03	17 0.03	16 0.05	15 0.03	14 0.03	14 0.04
X_2	5 0.05	9 0.04	11 0.03	11 0.03	12 0.04	7 0.03	13 0.04	13 0.02	10 0.04	10 0.04	9 0.04	6 0.05	8 0.04	13 0.04	7 0.03	14 0.04	12 0.03	11 0.03	7 0.02	11 0.02	13 0.02
X_3	20 0.21	20 0.22	20 0.22	20 0.22	20 0.21	20 0.22	20 0.21	20 0.22	20 0.21	20 0.20	20 0.20	20 0.21	20 0.22	20 0.22	20 0.20	20 0.20	20 0.20	20 0.20	20 0.20	20 0.20	20 0.19
X_4	13 0.04	8 0.04	5 0.03	11 0.04	3 0.02	10 0.02	7 0.03	14 0.02	9 0.05	12 0.03	7 0.00	10 0.04	12 0.03	9 0.06	10 0.03	12 0.03	10 0.02	15 0.02	11 0.03	8 0.03	11 0.03
X_5	2 0.02	4 0.01	2 0.01	5 0.03	8 0.00	6 0.03	10 0.02	8 0.01	2 0.01	11 0.03	3 0.01	7 0.02	5 0.01	6 0.02	8 0.03	9 0.02	7 0.01	6 0.02	11 0.02	4 -0.01	6 0.02
X_6	17 0.05	20 0.04	18 0.04	12 0.04	18 0.05	17 0.04	19 0.04	18 0.04	20 0.04	17 0.05	17 0.04	13 0.03	15 0.04	15 0.05	17 0.05	15 0.05	16 0.03	19 0.04	18 0.04	17 0.03	20 0.03
X_7	5 0.03	1 -0.00	4 0.03	4 0.02	5 0.03	7 0.03	9 0.02	4 0.03	5 0.02	14 0.02	2 -0.01	9 0.04	10 0.03	8 0.04	7 0.04	7 0.04	5 0.01	5 0.02	5 0.01	9 0.04	3 0.02
X_8	2 0.04	4 0.02	1 -0.01	1 0.01	3 0.02	4 0.01	3 0.01	0 0.00	5 0.02	1 0.02	4 0.01	2 0.02	4 0.02	1 0.02	1 0.01	1 0.01	1 0.04	3 -0.00	2 0.01	2 0.02	4 -0.00
X_9	1 0.02	6 0.02	1 0.01	2 0.02	8 0.02	2 0.01	4 0.01	3 0.01	3 0.01	4 0.02	5 0.01	3 0.01	2 -0.00	2 0.00	6 0.01	5 -0.00	1 0.02	5 0.00	4 0.00	5 0.00	7 0.01
X_{10}	20 0.05	20 0.05	20 0.04	20 0.05	20 0.05	20 0.06	20 0.05	20 0.04	19 0.05	20 0.05	19 0.05	19 0.05	18 0.06	18 0.05	20 0.05	20 0.04	20 0.03	20 0.04	19 0.03	20 0.04	20 0.03
X_{11}	0 0.00	1 0.01	1 0.02	2 -0.01	0 0.00	0 0.00	0 0.00	1 0.01	1 0.02	0 0.00	0 0.00	0 0.00	1 0.02	2 0.01	0 0.00	0 0.00	2 -0.01	0 0.00	0 0.00	0 0.00	0 0.00
X_{12}	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.02	0 0.00	0 0.00	0 0.00	2 0.02	1 -0.00	0 0.00	0 0.00	1 0.00	1 0.04	1 -0.01	1 0.02	0 0.00	1 -0.02	0 0.00	1 -0.01	1 -0.05
X_{13}	0 0.00	2 -0.00	2 -0.01	1 -0.01	1 -0.03	0 0.00	3 0.01	2 -0.04	0 0.00	0 0.00	2 0.01	0 0.00	1 0.01	0 0.00	2 0.01	0 0.00	2 0.00	0 0.00	0 0.00	1 -0.06	2 0.00
X_{14}	0 0.00	0 0.00	0 0.00	1 -0.01	0 0.00	0 0.00	1 -0.01	0 0.00	1 -0.00	1 0.01	0 0.00	0 0.00	1 -0.01	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	2 -0.02	0 0.00
X_{15}	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.01	0 0.00	0 0.00	1 -0.02	0 0.00	0 0.00	0 0.00	1 -0.01	0 0.00	0 0.00	0 0.00	0 0.00	2 0.00	1 -0.01	1 -0.01	0 0.00	1 -0.01
X_{16}	1 -0.01	0 0.00	1 -0.01	2 -0.01	0 0.00	1 0.03	0 0.00	2 -0.01	0 0.00	3 0.02	0 0.00	0 0.00	0 0.00	2 0.01	3 0.00	3 0.01	2 -0.02	0 0.00	1 0.01	1 0.02	0 0.00
X_{17}	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.01	1 -0.00	3 0.01	1 0.00	1 0.02	0 0.00	1 0.00	0 0.00	0 0.00	0 0.00	0 0.00	1 0.03	1 0.01	0 0.00	1 0.01	0 0.00
X_{18}	0 0.00	0 0.00	2 -0.00	1 -0.01	0 0.00	1 0.02	1 0.00	2 -0.01	1 0.02	1 0.01	0 0.00	1 0.01	1 -0.01	1 0.01	1 -0.02	1 0.03	0 0.00	1 -0.00	0 0.00	1 -0.02	4 -0.01
X_{19}	2 -0.02	1 -0.00	1 -0.04	2 -0.00	2 0.03	2 -0.00	3 -0.02	2 -0.02	3 0.01	0 0.00	2 0.01	2 0.01	1 -0.01	1 0.01	0 0.00	0 0.00	1 0.02	1 -0.01	0 0.00	1 -0.04	1 -0.00
X_{20}	0 0.00	0 0.00	0 0.00	0 0.00	1 -0.00	0 0.00	1 0.00	0 0.00	0 0.00	1 -0.05	1 0.01	0 0.00	0 0.00	3 -0.01	0 0.00	2 0.01	1 -0.01	0 0.00	3 -0.02	0 0.00	0 0.00

Table 4.21: Frequency & CODEC value table from D1 (1-dimensional) to D20 (20-dimensional) when $N = 10000$ and $\beta \sim \mathcal{U}(10, 20)$ using VAE as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo.

X's	D0		D1		D2		D3		D4		D5		D6		D7		D8		D9		D10		D11		D12		D13		D14		D15		D16		D17		D18		D19		D20	
X_1	**7**	0.12	**6**	0.17	**9**	0.07	**2**	0.05	**2**	0.03	**5**	-0.00	**6**	0.09	**7**	0.14	**5**	0.10	**7**	0.08	**6**	0.03	**6**	0.10	**4**	0.14	**4**	0.13	**9**	0.04	**7**	0.09	**5**	0.09	**5**	0.01	**2**	0.00	**2**	0.14	**9**	0.14
X_2	**0**	0.00	**6**	0.06	**1**	0.16	**1**	0.02	**3**	0.17	**7**	0.05	**6**	0.10	**4**	0.05	**4**	0.08	**7**	0.10	**4**	-0.03	**3**	0.11	**2**	0.02	**4**	0.10	**1**	-0.04	**8**	0.11	**1**	0.04	**3**	0.10	**7**	0.18	**3**	0.06	**8**	0.19
X_3	**9**	0.13	**12**	0.06	**13**	0.14	**11**	0.11	**12**	0.14	**16**	0.12	**7**	0.08	**10**	0.10	**6**	0.01	**10**	0.10	**11**	0.04	**6**	0.12	**7**	0.17	**12**	0.11	**12**	0.09	**10**	0.07	**10**	0.07	**9**	0.03	**12**	0.14	**7**	0.13	**12**	0.20
X_4	**3**	0.13	**5**	0.00	**5**	0.00	**7**	0.01	**1**	0.20	**4**	-0.01	**1**	0.21	**10**	0.14	**5**	0.02	**5**	0.02	**2**	-0.12	**2**	-0.00	**9**	0.03	**2**	0.06	**10**	0.02	**7**	0.03	**5**	0.04	**9**	0.03	**5**	0.11	**4**	0.13	**5**	0.08
X_5	**5**	0.07	**5**	-0.00	**6**	0.03	**7**	0.07	**7**	0.05	**8**	0.10	**2**	-0.00	**6**	0.08	**5**	0.06	**6**	0.01	**7**	0.08	**6**	0.13	**6**	0.06	**5**	0.12	**8**	0.07	**12**	0.12	**5**	0.07	**6**	-0.01	**6**	0.15	**10**	0.16	**2**	0.13
X_6	**5**	0.13	**10**	0.12	**7**	0.09	**8**	0.07	**11**	0.14	**9**	0.09	**10**	0.07	**5**	0.14	**9**	0.13	**5**	0.08	**12**	0.09	**4**	0.14	**8**	0.11	**10**	0.12	**9**	0.06	**3**	0.03	**7**	0.09	**11**	0.07	**10**	0.18	**9**	0.16	**6**	0.10
X_7	**2**	-0.00	**2**	0.04	**5**	0.15	**3**	0.09	**4**	0.05	**4**	-0.01	**1**	0.04	**1**	0.02	**2**	0.05	**4**	0.04	**1**	-0.08	**4**	0.06	**2**	-0.01	**2**	-0.03	**3**	0.00	**2**	0.05	**3**	0.11	**4**	0.03	**4**	0.10	**4**	0.07	**3**	0.07
X_8	**4**	0.05	**3**	0.04	**3**	0.07	**5**	0.08	**2**	-0.00	**6**	0.11	**1**	-0.03	**1**	0.15	**4**	0.07	**2**	0.09	**2**	-0.02	**2**	0.06	**2**	0.08	**6**	0.09	**7**	-0.03	**4**	0.11	**7**	0.16	**5**	0.01	**5**	0.04	**2**	0.19	**3**	0.13
X_9	**2**	-0.10	**0**	0.00	**3**	0.05	**4**	-0.01	**3**	0.04	**1**	0.14	**3**	0.03	**1**	0.08	**1**	0.23	**3**	0.10	**4**	-0.03	**8**	0.15	**3**	0.07	**2**	-0.06	**2**	0.09	**4**	0.07	**2**	-0.02	**5**	-0.04	**2**	0.07	**2**	0.14	**3**	-0.00
X_{10}	**6**	0.10	**4**	0.05	**2**	-0.00	**5**	0.08	**8**	0.06	**3**	0.20	**8**	0.12	**5**	0.05	**5**	0.08	**8**	0.12	**3**	0.05	**5**	0.16	**2**	0.07	**7**	0.13	**4**	0.04	**7**	0.09	**8**	0.09	**9**	0.06	**7**	0.12	**5**	0.08	**2**	0.01
X_{11}	**0**	0.00	**1**	0.04	**1**	0.04	**3**	-0.00	**1**	-0.02	**3**	0.03	**2**	0.02	**0**	0.00	**4**	-0.03	**3**	0.07	**2**	-0.11	**1**	0.08	**0**	0.00	**3**	-0.01	**6**	0.05	**1**	0.17	**4**	0.04	**1**	-0.02	**3**	0.07	**4**	0.11	**1**	0.12
X_{12}	**1**	0.02	**2**	0.12	**3**	0.11	**4**	0.05	**2**	0.06	**1**	-0.04	**0**	0.00	**0**	0.00	**1**	0.06	**0**	0.00	**2**	-0.02	**0**	0.00	**3**	0.09	**5**	0.03	**1**	-0.15	**2**	0.05	**4**	-0.02	**0**	0.00	**1**	0.17	**3**	0.14	**0**	0.00
X_{13}	**0**	0.00	**5**	0.08	**1**	-0.12	**0**	0.00	**0**	0.00	**1**	0.04	**0**	0.00	**2**	0.11	**1**	0.11	**2**	0.04	**1**	0.04	**0**	0.00	**1**	-0.02	**2**	0.03	**3**	-0.09	**1**	-0.09	**3**	0.12	**1**	-0.02	**1**	-0.01	**3**	0.02	**0**	0.00
X_{14}	**0**	0.00	**1**	-0.11	**2**	0.12	**1**	-0.04	**0**	0.00	**0**	0.00	**1**	-0.07	**2**	0.17	**1**	-0.03	**1**	0.11	**0**	0.00	**1**	0.16	**1**	0.12	**1**	-0.07	**2**	0.12	**2**	0.08	**2**	-0.05	**0**	0.00	**0**	0.00	**2**	0.29	**2**	0.07
X_{15}	**1**	0.17	**2**	0.08	**0**	0.00	**1**	0.09	**0**	0.00	**1**	0.04	**0**	0.00	**0**	0.00	**3**	-0.01	**0**	0.00	**3**	0.08	**6**	0.18	**1**	-0.05	**1**	0.09	**3**	-0.04	**1**	-0.03	**2**	0.00	**2**	-0.03	**2**	0.10	**2**	0.03	**2**	0.20
X_{16}	**1**	0.03	**1**	-0.00	**1**	0.00	**1**	-0.01	**1**	-0.06	**3**	-0.01	**2**	-0.07	**0**	0.00	**2**	0.04	**3**	0.03	**0**	0.00	**0**	0.00	**1**	0.07	**1**	-0.07	**2**	0.07	**1**	0.11	**2**	0.09	**3**	0.01	**1**	0.19	**2**	0.18	**1**	0.04
X_{17}	**1**	0.09	**3**	-0.03	**1**	-0.04	**1**	0.14	**0**	0.00	**1**	0.17	**3**	0.05	**1**	0.03	**0**	0.00	**2**	0.07	**5**	-0.07	**2**	0.03	**2**	-0.04	**1**	-0.05	**1**	-0.07	**1**	0.23	**2**	0.10	**3**	-0.07	**1**	-0.00	**1**	0.15	**0**	0.00
X_{18}	**1**	-0.02	**0**	0.00	**1**	-0.10	**1**	0.21	**0**	0.00	**2**	-0.02	**1**	0.03	**1**	0.02	**2**	0.17	**1**	0.17	**1**	-0.08	**2**	0.13	**1**	-0.02	**0**	0.00	**2**	-0.12	**1**	0.19	**0**	0.00	**3**	0.04	**0**	0.00	**2**	0.19	**2**	0.06
X_{19}	**1**	-0.03	**1**	-0.02	**0**	0.00	**0**	0.00	**1**	0.00	**2**	0.04	**1**	-0.18	**3**	-0.07	**0**	0.00	**0**	0.00	**1**	-0.15	**0**	0.00	**3**	0.17	**1**	0.07	**0**	0.00	**1**	-0.09	**1**	-0.13	**2**	-0.07	**1**	0.03	**1**	-0.02	**0**	0.00
X_{20}	**0**	0.00	**3**	0.07	**1**	0.15	**2**	0.10	**1**	-0.11	**1**	0.05	**4**	0.08	**2**	0.16	**1**	0.06	**9**	0.12	**3**	0.04	**0**	0.00	**2**	0.03	**0**	0.00	**2**	-0.12	**1**	0.11	**4**	0.05	**1**	-0.07	**3**	0.09	**0**	0.00	**4**	0.10

Table 4.22: Frequency & CODEC value table from D1 (1-dimensional) to D20 (20-dimensional) when $N = 500$ and $\beta \sim \mathcal{U}(10, 20)$ using VAE as latent variable estimator. The signals $X_3, X_{10},$ and X_{17} are highlighted with light indigo.

Discussions

Before discussion, we define the signals $S_1 := X_3^2$, $S_2 := X_{10}$, and $S_3 := \sqrt{|X_{17}|}$.

i.) When the given sample size $N = 10000$, PCA methods selects S_1 perfectly when the term $\mathbf{Z}\beta$ is dominating the magnitude of the response, but seldomly selects S_2 when including the estimated latent confounders (see Table 4.14). It is natural since if the latent variables make the most magnitude contributions to the response Y, FOCI would miss its direction in finding the signals with the smaller magnitude since the smaller the magnitude is, the CODEC value will become more closer to zero, which is the natural property derived form its definition. As we can see from Figure 4.8, S_1 is more close to hidden counfounder compared to S_2. As the result, we select less times of S_2 compared to S_1.

ii.) When the given sample size $N = 10000$, and $\mathbf{Z}\beta$ is not dominating the magnitude, and the included principal components are not too many (say at most we include 9 PCs), FOCI using PCA estimator does not selects S_3 very often which is not in the set of variables depending on $\mathbf{Z}$ but selects the ones which depend on perfectly. However, under the same conditions mention before, the VAE estimator selects the S_3 for near 20 frequencies from including 1-dimensional Z_{est} to including 5-dimensional Z_{est}. Using the heuristic threshold 4.2.1 that choose $\tau_0 = R/2 = 10$ (in our case), we found that S_3 will often be selected as the estimated Markov blanket including from 0-dimensional to 20-dimensional confounder estimates using VAE. However, if using PCA as confouder estimator, the selection results become worse since when we include different numbers of PCs, it is not significant enough to select S_3. That empirically gives us the justification that the VAE estimator is better than PCA estimator in the sense of the nonlinearity between the response and predictors and the sparsity between the predictors and the confounders.

iii.) When the given sample size $N = 10000$, $\mathbf{Z}\beta$ is not dominating the magnitude, and the included principal components (dimensions) is greater than or equal to 10 PCs (10-dimensional space), we can see from the Table 4.10, 4.12, 4.17, and 4.19 that while increasing the dimension of included $\mathbf{Z}_{est}$, the frequencies of selected signals started to increase, which means we are performing better with increased PC (dimension).

After verifying the values of CODEC$(Y, X_{17}|X_{17})$ and CODEC$(Y, X_{17}|\mathbf{X}^{'})$, and reckoning them both to be zero, the result turns out to be surprising since the latter one has strictly positive value and showing an increasing trend when the dimension of $\mathbf{X}^{'}$ increases. Note that $X_{17} \in \mathbf{X'} \subseteq \mathbf{X} = (X_1, ..., X_{20})$. It partly gives us the answer to the issue about including the full principle components $\mathbf{X'}$ as conditioning set, since including $\mathbf{X'}$ does not lead to a decrease of performance.

However, this phenomena remains to be discovered. As mentioned in Azadkia and Chatterjee (2021), a new consistent stopping rule could be designed in the future.

iv.) We also note that not including the estimated latent confounders in the conditioning set also gives us perfect results in the sense of pure signal selection. If the sample size is large enough, and our purpose is to just select distinguished variables, original

FOCI could possibly be used regardless of latent confounding.

This remains to be tested in real dataset, which will be implemented and discussed in the next section.

v.) When the given sample amount $N = 500$, both PCA and VAE methods with different settings of β perform poor results. Fixing the included principal components (dimensions), FOCI selects the Markov blanket quite by random. Thus, the size of data is of great importance when using FOCI for variable selection.

We also learn from this that when given less samples, original FOCI would not perform well anymore compared to the new FOCI which includes the estimate of hidden confounders in the conditioning set. It is in accordance with our expectation.

What's more, speaking of comparison between using PCA and using VAE, we found that FOCI with PCA performs generally well than using VAE when we have less sample size.

vi.) When the $\mathbf{Z}\beta$ is dominating the other signals, FOCI shows poor performance of selecting S_2 and S_3, while perform well given sufficient sample size when selecting S_1 since the magnitude of S_1 is greater than that of S_2 and S_3. The only exception is the case of FOCI with VAE estimator when $N = 10000$, it also selects S_2 all the time.

This reveals the great importance of balancing the magnitude between the signals and confounders, though using VAE estimators will ease this problem under specific conditions. In real-life datasets, we may need to learn some prior knowledge about the predictors and potential confounders as well as investigating thoroughly about their importance to the response.

vii.) Speaking of corresponding CODEC values, we found an interesting phenomenon that although PCA method selects much less of X_{17}, but the corresponding average CODEC value will be around 0.2, which is sufficient to say that there exists conditional dependencies.

While VAE are selecting X_{17}'s, the average CODEC values will be a positive number which is much smaller compared to that of PCA, e.g. 0.01, 0.02.

viii.) It can be concluded that the nonlinear functional relationship between response Y and signals $\mathbf{X}$ and the small magnitude of signals together add difficulties to variable selection.

ix.) Using VAE can help us further distinguish the true Markov blanket while PCA sometimes abandoned the signals that are independent of latent confounder $\mathbf{Z}$ and choose to select the non-signals as signals.

x.) In terms of computational complexity, plain PCA method under big datasets will certain take much more time to run. VAE is utilizing the GPU to train the network and thus is much faster compared to PCA estimator. It is worth noting that while the sample size N grows linearly, the computation time of PCA grows cubically, *i.e.*

$O(N^3)$. It is necessary to utilize GPU parallel computation and deriving new PCA algorithms (Andrecut, 2009; Yu, Gu, Li, Liu, and Li, 2017) when we have a large dataset while using PCA as estimator.

Chapter 5

Application to Real Dataset

5.1 Dataset and Methods

In this section, we apply FOCI with PCA and FOCI with VAE on the UCI Wine Quality dataset (Dua and Graff, 2017). This dataset describes the dependence between the wine quality score (0 - 10 integers) Y given by human raters of red and white wine, and 11 different predictors: X_1: fixed acidity, X_2: volatile acidity, X_3: citric acid, X_4: residual sugar, X_5: chlorides, X_6: free sulfur dioxide, X_7: total sulfur dioxide, X_8: density, X_9: pH, X_{10}: sulphates, and X_{11}: alcohol. It seems that it is a simple variable selection and regression task for us, however, Janzing and Schölkopf (2018) found that there might exist one confounding variable X_{11} (alcohol) since its estimated confounding strength is 0.55, which indicates that X_{11} is plausibly correlated with the remaining predictors $(X_1, ..., X_{10})$ and at the same time influences Y.

In that sense, we will assume that we do not know about X_{11} and use the first 10 predictors to select and predict. Then, we use the training mean squared prediction errors (MSPE) and the test MSPE to evaluate the performance.

We will compare those metrics among the following methods:

- **FOCI**: pure FOCI without any adjustment to select signals and random forest (RF) (Breiman, 2001) regressor to predict the wine quality $\hat{Y}$.
- **FOCI-OPCA**: Using the selection and resampling scheme which selects the number of principal components via the highest precision (lowest FDP), and combine it with FOCI. Then using the RF to regress on training set (estimate of latent confounder incl.) and predict on test test.
- **FOCI-FPCA**: Including all the principle components in the CODEC conditioning set and use RF to regress on training set (estimate of latent confounder incl.) and predict.
- **FOCI-OVAE**: Using the selection and resampling scheme to select the dimension of latent space by minimizing the FDP and sample from that latent space to be the

estimate of $\mathbf{Z}$. Then use RF to regress on traning set (estimate of latent confounder incl.) and predict.

- **LASSO**: LASSO (Tibshirani, 1996) with BIC-like criteria (Gao and Song, 2010)

  $$\text{BICC} = n\log(\hat{\sigma}^2) + 3 \times s\log(n),$$

  where n is the sample size, $\hat{\sigma}^2$ is the MSE of the model, s is the optimal Markov blanket the lasso would choose based on different regularization constants λ. We use BICC to tune the λ. Then using the best BICC model and RF to regress and predict.

- **SCAD**: Smoothly clipped absolute deviation (SCAD) (Fan and Li, 2001) with BICC to tune the hyperparameter λ and use the best model and RF to regress and predict.

5.2 Results

The results is given in Table 5.1. From what we have seen, we notice that FOCI with latent confounder estimators attained competitive prediction errors as the other methods like LASSO, but with fewer number of variables. For SCAD feature selection, we noticed that even if it selects a small set of predictors, however, the training and test MSPE are worse than the FOCI family.

We can also see that using original FOCI selects all the predictors which leads to the huge performance difference between the train and the test set.

Next, speaking of including all the principal components to retrieve all the useful information of original latent confounder $\mathbf{Z}$. We did achieve the best training MSPE among all the methods, however, by adding too many additional estimates of latent confounders as predictors, we will have the reduced generalization ability. In general, we should balance between the information retrieved and the genaralization ability.

Finally, experiment with UCI Wine Quality dataset may not be a perfect in the sense of implementing our new methods since we do not know whether there are confounders or not. We expect to acquire DNA sequence datasets in the future, which often has the situation that the dimension p is larger than the sample size N as well as exists certain verified confounders. In that sense, it is necessary to select variables.

Methods	$\lvert\hat{S}\rvert$	Training MSPE	Test MSPE
FOCI	10	0.072	0.496
FOCI-OPCA	4	0.138	0.526
FOCI-FPCA	**2**	**0.068**	0.811
FOCI-OVAE	6	0.099	0.469
LASSO	9	0.073	**0.365**
SCAD	3	0.201	0.567

Table 5.1: Comparison table of FOCI and its extensions, LASSO, and smoothly clipped absolute deviation (SCAD) in UCI Wine Quality variable selection. $\lvert\hat{S}\rvert$ denotes the number of variables selected.

Chapter 6

Concluding Remarks

We conclude that this particular nonparametric variable selection method is computationally efficient and has the potential to extend to the case that there are hidden confoundings. Given sufficient samples, we propose to use FOCI with proper latent variable estimation method since it not only selects correct Markov blanket with certain control of false discovery proportion and give us a view of the empirical distributions of the latent confounders, but also remove the spurious association between non-signals $X \backslash \mathcal{S}$ and response Y.

To be more specific, if one have a prior knowledge that the relationship between $\mathbf{X}$ and $\mathbf{Z}$ is dense, it is recommend to use principal components analysis to estimate the latent confounders. We also gave out theoretical justifications of using principle components as estimator. If one assume that this relationship is not dense and the sample size is sufficient to train, it is recommend to use variational autoencoder to encode the latent space. If we are lacking enough samples, we generally recommend to use FOCI with PCA.

Besides, using the scheme of resampling and as the result discovering the frequency of each variable will make the variable selection decision process more distinguishable (e.g. with a heuristic threshold), especially giving aids to determine the dimension of estimated latent space. We propose to use this scheme given sufficient computation ability since it requires a lot time of running the whole loop.

In the context of magnitude mentioned in Section 4.2.8, we proposed that if the signals are not dominated by the confounders, FOCI will be a nice choice and perform well. We thus suggest to first investigate the magnitude of given predictors and gain some prior knowledge on the potential confounders.

What's more, even though the FOCI with latent variable estimators generally generate satisfying results, however, the computational complexity is relatively high compared to VAE which utlizes GPU. We then proposed to use some new PCA algorithms which utilize GPU parallel computations.

Finally, on the given real dataset, we can conclude that using FOCI family (with or without latent confounder estimation) generally has similar performance compared with the classic methods like LASSO and SCAD with respect to MSPE and generalization ability but has a smaller Markov blanket.

Chapter 7

Future Works

We note that we only used two kinds of latent confounder estimators. It is also worthwhile to try on other different latent confounder estimators, such as kernel principal component analysis, classical factor analysis model (Child, 1990), and deep neural networks to compare the variable selection capability and the prediction error. As for sparse relationship between signals and confounders, one may further dig into the construction of the variational autoencoder and other deep neureal networks to see if using changing architectures would lead to a better estimation result and prediction error. Besides, one may also based on different priors $p(Z)$ such as using the distributions from location-scale family and instead of sampling from posterior distribution, but to directly use the posterior mean to make a more stable estimation of latent confounders, which would provide us a clearer view of the empirical distribution of confounders.

After Experiment 4, we have theoretically justified the use of PCA as estimator under the scope of equal CODEC values. It is worth to discover the rate of convergence of the CODEC value and other norm preserving transformations with the random error term included. A formal proof remains to be given out in the future.

Under the nonlinear relationship between $\mathbf{X}$ and $\mathbf{Z}$, we observed that VAE estimator perform better than PCA in FOCI variable selection in the sense of precision and selection frequency. It is worth to experiment our methods in more real-life datasets like DNA data to see whether it selects the correct Markov blanket.

In addition, speaking of false discovery proportion control (or FDR, which the FDP taking expectation) of variable selection, Candes, Fan, Janson, and Lv (2018) proposed the Model-X knockoff to teases apart important from irrelevant variables (controlling FDR) while guaranteeing Type I error control. However, it is assumed based on the fact that there is no confounding variables that may affect the procedure of variable selection. It is left open to find a new threshold for controlling the FDR/FDP for the variable selection under latent confounding and hopefully one could define a new statistic using CODEC.

Finally, speaking of small sample size, Chatterjee (2020) mentioned that the ranking-based correlation seems to have less power than several popular tests of independence, which also affectting the CODEC's performance in testing. It is worth experimenting and provide avenues for boosting the power.

Bibliography

Abdi, H. (2007). The Kendall Rank Correlation Coefficient. pp. 7.

Andrecut, M. (2009). Parallel gpu implementation of iterative pca algorithms. *Journal of Computational Biology 16*(11), 1593–1599.

Azadkia, M. and S. Chatterjee (2021, January). A simple measure of conditional dependence. *arXiv:1910.12327 [cs, math, stat]*. arXiv: 1910.12327.

Azadkia, M., S. Chatterjee, and N. Matloff (2021, March). FOCI: Feature Ordering by Conditional Independence.

Benesty, J., J. Chen, Y. Huang, and I. Cohen (2009). Pearson correlation coefficient. In *Noise reduction in speech processing*, pp. 1–4. Springer.

Breiman, L. (2001). Random forests. *Machine learning 45*(1), 5–32.

Bro, R. and A. K. Smilde (2014). Principal component analysis. *Analytical Methods 6*(9), 2812–2831. Publisher: Royal Society of Chemistry.

Candes, E., Y. Fan, L. Janson, and J. Lv (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.

Chatterjee, S. (2020, April). A new coefficient of correlation. *arXiv:1909.10140 [math, stat]*. arXiv: 1909.10140.

Child, D. (1990). *The essentials of factor analysis.* Cassell Educational.

Clark, M. (2018). Graphical & Latent Variable Modeling.

Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association 74*(368), 829–836.

Dette, H., K. F. Siburg, and P. A. Stoimenov (2013). A Copula-Based Non-parametric Measure of Regression Dependence. *Scandinavian Journal of Statistics 40*(1), 21–41. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9469.2011.00767.x.

DeVos, M. (2018). Simon Fraser University Geometry and Symmetry, Lecture Notes 18: Isometries. URL: https://www.sfu.ca/~mdevos/notes/geom-sym/18_isometries.pdf. Last visited on 2021/10/10.

Dua, D. and C. Graff (2017). Uci machine learning repository: Wine quality data set. *URL: https://archive. ics. uci. edu/ml/datasets/wine+ quality*.

Fan, J. and R. Li (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association 96*(456), 1348–1360.

Friedman, J. H. (2017). *The elements of statistical learning: Data mining, inference, and prediction.* springer open.

Gao, X. and P. X.-K. Song (2010). Composite likelihood bayesian information criteria for model selection in high-dimensional data. *Journal of the American Statistical Association 105*(492), 1531–1540.

Geiger, B. C. and G. Kubin (2012). Relative information loss in the pca. In *2012 IEEE Information Theory Workshop*, pp. 562–566. IEEE.

Gillies, D. F. (2018). Lecture notes in data analysis and probabilistic inference.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition* (2 ed.). Springer Series in Statistics. New York: Springer-Verlag.

Janzing, D. and B. Schölkopf (2018). Detecting confounding in multivariate linear models via spectral analysis. *Journal of Causal Inference 6*(1).

Kingma, D. P. and J. Ba (2017, January). Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*. arXiv: 1412.6980.

Kingma, D. P. and M. Welling (2014, May). Auto-Encoding Variational Bayes. *arXiv:1312.6114 [cs, stat]*. arXiv: 1312.6114.

Kramer, O. (2013). *Dimensionality reduction with unsupervised nearest neighbors.* Springer.

Linton, O. and P. Gozalo (1996, December). Conditional Independence Restrictions: Testing and Estimation.

Massey, F. J. (1951). The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American Statistical Association 46*(253), 68–78. Publisher: [American Statistical Association, Taylor & Francis, Ltd.].

Morris, T. P., I. R. White, and M. J. Crowther (2019). Using simulation studies to evaluate statistical methods. *Statistics in medicine 38*(11), 2074–2102.

Myers, L. and M. J. Sirois (2006). Spearman Correlation Coefficients, Differences between. In *Encyclopedia of Statistical Sciences.* American Cancer Society. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/0471667196.ess5050.pub2.

Pastore, M. and A. Calcagnì (2019). Measuring Distribution Similarities Between Samples: A Distribution-Free Overlapping Index. *Frontiers in Psychology 10*, 1089.

Poczos, B. and J. Schneider (2012, March). Nonparametric Estimation of Conditional Information and Divergences. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, pp. 914–923. PMLR. ISSN: 1938-7228.

Rényi, A. (1959). On the dimension and entropy of probability distributions. *Acta Mathematica Academiae Scientiarum Hungarica 10*(1-2), 193–215.

Shanmugam, R. (2018, November). Elements of causal inference: foundations and learning algorithms. *Journal of Statistical Computation and Simulation 88*(16), 3248–3248.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal 27*(3), 379–423.

Sinitambirivoutin, E. (2020). An introduction to Variational Auto Encoders (VAEs) | by Emrick Sinitambirivoutin | Towards Data Science.

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

Yu, W., Y. Gu, J. Li, S. Liu, and Y. Li (2017). Single-pass pca of large high-dimensional data. *arXiv preprint arXiv:1704.07669*.

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

NONPARAMETRIC VARIABLE SELECTION UNDER LATENT CONFOUNDING

Authored by (in block letters):
For papers written by groups the names of all authors are required.

Name(s): GENG

First name(s): ZIPEI

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date
Zürich, 2021.10.01

Signature(s)
耿子沛

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.