Simulation: Sparse Principal Component Analysis

Mohammed Yusuf Shaikh

Student No.: 1006695783

March 20, 2025

University of Toronto Mississauga

STA315H5S – Advanced Statistical Learning

Instructor: Prof. Dehan Kong

1 Introduction

Principal Component Analysis (PCA) a method we know is used as statistical technique to reduce the dimensionality of data. PCA Methods strategically aims to find a "good" linear combination of $X1, ..., X_p$ such that the linear combination explains the most variability or randomness present in the dataset.

Mathematically, each i-th principal component is a linear combination of the original features:

$$Y_i = \ell_i^\top \mathbf{X}$$

Where,

- Y_i is the i-th principal component
- $\mathbf{X} = (X_1, X_2, ..., X_p)^\top$ is the vector of original features
- $\ell_i = (\ell_{i1}, \ell_{i2}, ..., \ell_{ip})^\top$ is the loading vector (weights)

The paper discusses that PCA method uses Singular Value Decomposition (SVD) to compute principal components, where X is a standardized data matrix. The X data matrix can be broken down to $Z = UD$ which represnet principal component and matrix V are the loadings that explain how much each original variable contributes to each principal component.

$$X = UDV^T$$

- U contains the Eigenvectors
- D contains the Eigenvalues

Although in PCA method the components capture maximum variance and compresses data while retaining important information despite that we have a larger drawback that is PCA uses all the variables when creating each principal components as a result the PC ends up being a mixture of every variable making it harder to interpret therefore lack of sparsity implies PCA doesn't do variable selection. Moreover, the loadings are usually nonzero, making it hard to interpret which variables that are truly important and contribute to principal component.

To overcome the limitation of the PCA, a paper was presented by Hui Zou, Trevor Hastie, and Robert Tibshirani(2006). The paper (Zou, Hastie, and Tibshirani 2006) introduces a new method called Sparse Principal Component Analysis (SPCA). The idea focused on to make PCA a more interpretable method by forcing some of the loadings to be exactly zero. This paper presents a detailed review of SPCA, explains its methodology, compares it with existing alternatives, and reproduces a key simulation study from the original paper.

2 Why is there a need for Sparse PCA

According to PCA Method the loading vector contains non-zero values for all p variables meaning every variable contributes to each component. This makes interpretation difficult. This is a major problem when p variables are large in number. For example in gene expression data with 10,000 genes features, the principal component are influenced by all genes which hinders the ability to extract meaningful interpretation. Sparsity aims at to having many zero entries in a vector. In terms of PCA, sparse loadings implies that only a few variables significantly contribute to each component. The Sparse Principal component Analysis produces principal components with sparse loading which implies many coefficients are set to zero and making PC more interpretable.

The authors discusses earlier attempts to use sparsity in PCA method like SCotLASS (Sparse Component Technique using LASSO) method that used tuning parameter t, and when sufficiently t is small it forces some loading to be exactly 0. But one of the major drawback was that there was no clear method to determine an optimal **t** and it failed to produce sufficiently sparse loadings while maintaining a high percentage of explained variance.

3 Motivation & Algorithm for SPCA

The limitations encouraged the authors to produce an optimal algorithm, which involved recasting PCA as a ridge regression problem and they added lasso penalty to produce sparse loading. The equation is the Elastic Net Regression where we fit the sparse loadings which is given by:

$$\arg\min_{A,B} \sum_{i=1}^{n} \left\|\mathbf{x}_i - AB^T\mathbf{x}_i\right\|^2 + \lambda \sum_{j=1}^{k} \left\|\beta_j\right\|^2 + \sum_{j=1}^{k} \lambda_{1,j} \left\|\beta_j\right\|_1$$

This is the alternating minimization algorithm is to minimize the SPCA criterion which is available from **elasticnet** package as function **spca()**. To begin with we find principal component using PCA method and we take only few PC which explain majority of variances. We call the loading vectors as matrix A. With fixed matrix A we solve the elastic net problem to get matrix B to find sparse loading matrix. Nextly we fix matrix B and solve for matrix A and we repeat the steps until the net elastic regression converges. Hence in SPCA result in column matrix B which tells what variables are important for each principal component.

4 Simulation Procedure

We simulate a Synthetic dataset based on paper Sparse Principal Component Analysis (SPCA) give by Hui Zou, Trevor Hastie, and Robert Tibshirani(2006). The aim of the simulation is to evaluate how well classic PCA and Sparse Principal Component Analysis (SPCA) perform

on synthetic example that has three hidden factors. We will reproduce the same result as presented in paper by authors. We use R Package for fitting the SPCA model (and elastic net models) which is available under **elasticnet** package created by the authors.

The simulated data reproduced we used the programming language R (R Core Team 2022) with support of additional packages in R: 'tidyverse' (Wickham et al. 2019), 'dplyr' (Wickham et al. 2022), 'elasticnet' (Zou and Hastie 2020), 'knitr' (Xie 2014), 'PCA' (Verzani 2022), 'scale function' (R-bloggers 2021). We also utilized 'Applications in R' (Alexander 2023)

4.1 Generating Data

For generating data we used 10 observable variables(p) $X_1, ...X_{10}$. Number of variables are used as standard as per given in the paper procedures and we simulate 100 observation for each variable. For the next part we define 3 hidden variables - latent factors based on authors paper which imples the factors that influence observed variables.

$$V_1 \sim \mathcal{N}(0, 290)$$

$$V_2 \sim \mathcal{N}(0, 300)$$

The variables V_1, V_2 and V_3 are 3 underlying *hidden* factors that govern th observed variables. The factors V_1 & V_2 are independent variables with high variances, whereas V_3 is a linear combination of factors V_1 & V_2 and equation is given by:

$$V_3 = -0.3V_1 + 0.925V_2 + \epsilon, \quad \text{where } \epsilon \sim \mathcal{N}(0, 1)$$

$$V_3 \sim \mathcal{N}(0, 283.8)$$

It is given that V_1 & V_2 and ϵ are mutually independent of each other.V_3 being the linear combination of V_1 & V_2 we find the mean and variance of V_3. The calculation is provided in the Appendix. We constructed an exact covariance matrix of $X_1, ..., X_{10}$ variables and each variables from X_1 to X_4 is constructed as:

$$\begin{aligned} X_i &= V_1 + \epsilon_i, \quad \text{where } \epsilon_{ij} \sim \mathcal{N}(0, 1) \quad \text{for } i = 1, 2, 3, 4 \\ X_i &= V_2 + \epsilon_i, \quad \text{where } \epsilon_{ij} \sim \mathcal{N}(0, 1) \quad \text{for } i = 5, 6, 7, 8 \\ X_i &= V_3 + \epsilon_i, \quad \text{where } \epsilon_{ij} \sim \mathcal{N}(0, 1) \quad \text{for } i = 9, 10 \end{aligned}$$

where ϵ_{ij} is an independent Gaussian noise and $j \in \{1, 2, 3\}$ that indicates which latent factor group the variable belongs to. So, similarly for variables from X_5 to X_8 depend on V_2 and variables X_9 & X_{10} depends on V_3. In our simulation, we get $n = 100$ observations for each of the 10 variables, resulting in a data matrix of size 10×100.

Before applying dimension reduction methods, on our simulated data matrix X, we standardized it so that each variable had mean zero and unit variance using **scale** which is a generic function whose default method centers the columns of a matrix. Standardization ensures that all variables contribute equally when performing principal component analysis (PCA) or Sparse PCA. Note that we do not proceed with SCoTLASS or Simple Thresholding method under simulation as our key interest only remains in PCA and SPCA method.

4.2 Principal Component Analysis

We first applied Principal Component Analysis using the **'prcomp()'** function in R on our standardized dataset. It computes the principal components from the data matrix using Singular Value Decomposition (SVD) method as discussed in above methodology. This is generally the preferred method for numerical accuracy for obtaining stable principal components. We extracted the loading for the first three principal components, and we computed the proportion of variance explained by the first, second and third PCs using the standard PCA output. These loading and variances summarize how much of the variability in the data is captured by the principal components from Table 1.

4.3 Sparse Principal Component Analysis

Moving on from Principal component analysis, we apply Sparse PCA using **spca()** function from **elasticnet** package. The formula takes the input of standardized data matrix, with K = 2 that computes first 2 components i.e. PC1 and PC2, we impose sparsity by specifying that each component should have exactly four non-zero loading (sparse = "varnum", para = c(4,4)) in the simulation design with lambda set to standard default value of 1e-6.

```
You may wish to restart and use a more efficient way
let the argument x be the sample covariance/correlation matrix and set type=Gram
```

Table 1: The results from PCA and SPCA Loadings

Variable	PCA PC1	PCA PC2	PCA PC3	SPCA PC1	SPCA PC2
X1	0.118	-0.478	0.085	0.0	-0.5
X2	0.118	-0.478	0.092	0.0	-0.5
X3	0.117	-0.478	0.095	0.0	-0.5

Table 1: The results from PCA and SPCA Loadings

Variable	PCA PC1	PCA PC2	PCA PC3	SPCA PC1	SPCA PC2
X4	0.118	-0.478	0.078	0.0	-0.5
X5	-0.391	-0.146	-0.266	-0.5	0.0
X6	-0.391	-0.146	-0.263	-0.5	0.0
X7	-0.391	-0.146	-0.274	-0.5	0.0
X8	-0.391	-0.147	-0.298	-0.4	0.0
X9	-0.409	0.005	0.573	0.0	0.0
X10	-0.409	0.005	0.580	0.0	0.0
Explained Variance (%)	59.700	40.000	0.080	39.6	39.8

Later on, we obtain percentage of variance explained by each principal component that is obtained using the **summary()** function on the PCA object and it is extracted for PC1, PC2 & PC3. Similarly we compute percentage of variance explained for SPCA; given by *pev* output from spca function. The result are presented in Table 1 with first 10 rows showing the PCA and SPCA loadings for each variable. The last row of table summarizes the percentage of explained variances for the first three principal components under PCA method and the first two sparse principal components under SPCA method.

4.4 Results

The simulation result align closely with the results from thr paper presented by authors Hui Zou, Trevor Hastie, and Robert Tibshirani (2006). The result from first to PCs from PCA method catpture maximum total variance of 99.7% (PC1 explains 59.7% and PC2 explains 40%) which ideally shed the light upon the fact the simulated dataset is dominated by variables V_1 and V_2. Also, the loading values in Table 1 all behave similarly with Synthetic Example. The magnitude showcases that PC1 captures variability related to latent factors V_2 using X_5, X_6, X_7, X_8 variables as they have large loading values of -0.391. On the other hand, variables X_1, X_2, X_3, X_4 show loadings of -0.478 on PC2 corresponding to latent factor V_1.

For SPCA loading values observe from Table 1 that the method selects only important variables and forces other variable to be set to zero. Hence in the simulation we produce ideal sparse PCs. Note we did not use adjusted variance. The SPCA loadings match th simulation setup almost exactly The magnitude showcases that PC1 captures variability related to latent factor V2 using X_5, X_6, X_7, X_8 variables as they have large loading values of -0.5 On the other hand, variables X_1, X_2, X_3, X_4 show strong loadings of -0.5 on PC2 corresponding to latent factor V_1. To conclude with SPCA primarily focus on most important variables while still explaining high variance of 79.4% and also notice it is easy to interpret the key variables being highlighted as important.

5 Conclusion

To conclude with, in this paper we explored principal component analysis method and its limitation. To mitigate the limitation we show SPCA which showcased sparsity by forcing loading variables to be set to zero. We provide SCPA methodology explanation of the algorithm followed by reproducing a simulation example presented by the paper (Zou, Hastie, and Tibshirani 2006). The Table results compares PCA and SPCA principal components as well as explained variances. In conclusion SPCA provides a powerful alternative to PCA for analyzing high dimensional dataset.

6 Appendix

```r
# Readme
# - Make sure you load necessary packages
# - store reference.bib file in same folder to render pdf with references

# Load necessary package
# install.packages("elasticnet")
# install.packages("janitor")
# install.packages("dplyr")
# install.packages("knitr")
# install.packages("tidyverse")

# Load Libraries
library(elasticnet)
library(dplyr)
library(janitor)
library(knitr)
library(tidyverse)

# Using seed to get same reproducible result
set.seed(123)

# Generating Data
# no. of observations we went through multiple trial & errors for value of n for high dimens
n <- 10000
# no. of variable/features (predictors) we use for synthetic example
p <- 10

# Specifing Three latent variables - hidden factors as per paper Synthetic example
V1 <- rnorm(n, mean = 0, sd = sqrt(290))
V2 <- rnorm(n, mean = 0, sd = sqrt(300))
V3 <- -0.3 * V1 + 0.925 * V2 + rnorm(n)


# Generating Matrix as per synthetic example
X <- matrix(0, nrow = n, ncol = p)
X[, 1:4]  <- V1 + matrix(rnorm(n * 4), nrow = n)
X[, 5:8]  <- V2 + matrix(rnorm(n * 4), nrow = n)
X[, 9:10] <- V3 + matrix(rnorm(n * 2), nrow = n)
```

```r
# We standardize the data matrix as it is required for PCA and SPCA method
X_standardized <- scale(X, center = TRUE, scale = TRUE)

# PCA Method

# Apply PCA Method to standardized data prcomp() is used
pca_loading_value <- prcomp(X_standardized)
# We round the loading values for PC1, PC2 & PC3 but we extract it before using $rotation
print(round(pca_loading_value$rotation[, 1:3], 3))
```

```
         PC1    PC2    PC3
 [1,]  0.118 -0.478  0.085
 [2,]  0.118 -0.478  0.092
 [3,]  0.117 -0.478  0.095
 [4,]  0.118 -0.478  0.078
 [5,] -0.391 -0.146 -0.266
 [6,] -0.391 -0.146 -0.263
 [7,] -0.391 -0.146 -0.274
 [8,] -0.391 -0.147 -0.298
 [9,] -0.409  0.005  0.573
[10,] -0.409  0.005  0.580
```

```r
# Total variance is explained by PCA
total_explained_variance <- summary(pca_loading_value)$importance["Proportion of Variance", 1

# Sparse PCA Method

# Apply Sparse PCA Method to standardized data spca() is used from elastic net package
spca_result <- spca(X_standardized, K = 2, type = "predictor", sparse = "varnum", para = c(4
```

```
You may wish to restart and use a more efficient way
let the argument x be the sample covariance/correlation matrix and set type=Gram
```

```r
# We round the loading values for PC1 & PC2 but we extract it before using $loadings
print(round(spca_result$loadings, 1))
```

```
      PC1  PC2
 [1,]  0.0 -0.5
 [2,]  0.0 -0.5
 [3,]  0.0 -0.5
```

```
 [4,]  0.0 -0.5
 [5,] -0.5  0.0
 [6,] -0.5  0.0
 [7,] -0.5  0.0
 [8,] -0.4  0.0
 [9,]  0.0  0.0
[10,]  0.0  0.0
```

```r
# Total variance is explained by SPCA using $pev
spca_variance_explained <- spca_result$pev * 100

# We construct a table using knitr

# We use PCA and SPCA loadings table using tibble
loadings_table <- tibble::tibble(
  Variable = paste0("X", 1:10),
  'PCA PC1' = round(pca_loading_value$rotation[, 1], 3),
  'PCA PC2' = round(pca_loading_value$rotation[, 2], 3),
  'PCA PC3' = round(pca_loading_value$rotation[, 3], 3),
  'SPCA PC1' = round(spca_result$loadings[, 1], 1),
  'SPCA PC2' = round(spca_result$loadings[, 2], 1)
)

# 2. Explained variance table using tibble
pca_spca_explained_variance <- tibble::tibble(
  Variable = "Explained Variance (%)",
  'PCA PC1' = round(total_explained_variance[1], 1),
  'PCA PC2' = round(total_explained_variance[2], 1),
  'PCA PC3' = round(summary(pca_loading_value)$importance["Proportion of Variance", 3] * 100
  'SPCA PC1' = round(spca_result$pev[1] * 100, 1),
  'SPCA PC2' = round(spca_result$pev[2] * 100, 1)
)

# we combine two table using bind_rows
result_table <- bind_rows(loadings_table, pca_spca_explained_variance)

# We print the Table for paper.qmd
kable(result_table, caption = "Table 1: Results of the Simulation Example - Loadings and Vari
```

Table 2: Table 1: Results of the Simulation Example — Loadings and Variance

Variable	PCA PC1	PCA PC2	PCA PC3	SPCA PC1	SPCA PC2
X1	0.118	-0.478	0.085	0.0	-0.5
X2	0.118	-0.478	0.092	0.0	-0.5
X3	0.117	-0.478	0.095	0.0	-0.5
X4	0.118	-0.478	0.078	0.0	-0.5
X5	-0.391	-0.146	-0.266	-0.5	0.0
X6	-0.391	-0.146	-0.263	-0.5	0.0
X7	-0.391	-0.146	-0.274	-0.5	0.0
X8	-0.391	-0.147	-0.298	-0.4	0.0
X9	-0.409	0.005	0.573	0.0	0.0
X10	-0.409	0.005	0.580	0.0	0.0
Explained Variance (%)	59.700	40.000	0.080	39.6	39.8

References

Alexander, Rohan. 2023. *Telling Stories with Data: With Applications in r*. Boca Raton, FL: Chapman & Hall/CRC Press.

R Core Team. 2022. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

R-bloggers. 2021. "How to Use the Scale() Function in r." https://www.r-bloggers.com/2021/12/how-to-use-the-scale-function-in-r/.

Verzani, John. 2022. "Functions for Principal Component Analysis." https://cran.r-project.org/web/packages/LearnPCA/vignettes/Vig_07_Functions_PCA.pdf.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Wickham, Hadley, Romain François, Lionel Henry, and Kirill Müller. 2022. *Dplyr: A Grammar of Data Manipulation*. https://CRAN.R-project.org/package=dplyr.

Xie, Yihui. 2014. "Knitr: A Comprehensive Tool for Reproducible Research in R." In *Implementing Reproducible Computational Research*, edited by Victoria Stodden, Friedrich Leisch, and Roger D. Peng. Chapman; Hall/CRC. http://www.crcpress.com/product/isbn/9781466561595.

Zou, Hui, and Trevor Hastie. 2020. *Elasticnet: Elastic-Net for Sparse Estimation and Sparse PCA*. https://CRAN.R-project.org/package=elasticnet.

Zou, Hui, Trevor Hastie, and Robert Tibshirani. 2006. "Sparse Principal Component Analysis." *Journal of Computational and Graphical Statistics*. https://doi.org/10.1198/106186006X113430.