# Diabetes classification with dimension reduction techniques

**STATS/CSE 790 Assignment 4**

**2024-02-29**

**Pao Zhu Vivian Hsu (400547994)**

## Introduction

In this study, we utilize principal component analysis (PCA) and factor analysis (FA) when predicting the risk of diabetes. The data used in this analysis comes from an open source website called Kaggle (Islam et al., 2020; Larxel, 2024). It contains 520 rows of patient data collected from a hospital in Bangladesh and 17 variables including an indicator for positive or negative diabetes risk, age, and binary variables denoting the presence or absence of common diabetes symptoms (Larxel, 2024).

## Methods

We start the study by checking the shape of the data and ensuring there are no missing values. We then visualized the data in a pairs plot to check for any patterns in the data. Figure 1 shows a subset of this plot.

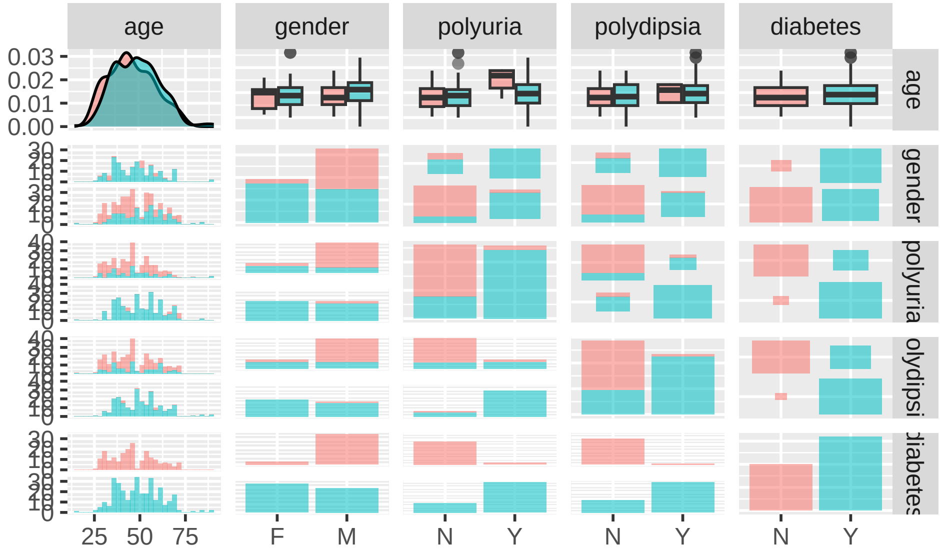

Figure 1: Pairs plot of a selection of the variables

Next, we then applied two different dimension reduction techniques to the data: PCA and FA.

The first method was PCA, which involves finding the direction with the most variation in the data by components. The model can be written as $W_i = v_i'(X - \mu)$ for $i = 1, ..., p$ principal components. The proportion of the total variation explained by the $i$th principal component can be expressed as $\lambda_i/tr\Sigma$ while the total variation explained by the first r principal components is $\frac{\Sigma^{r}_{i=1}}{tr\Sigma}$ (King-Yu, 2024). For our PCA, we selected 11 components since the scree plot in Figure 2 shows that approximately 85% of the variation is explained with this component size.

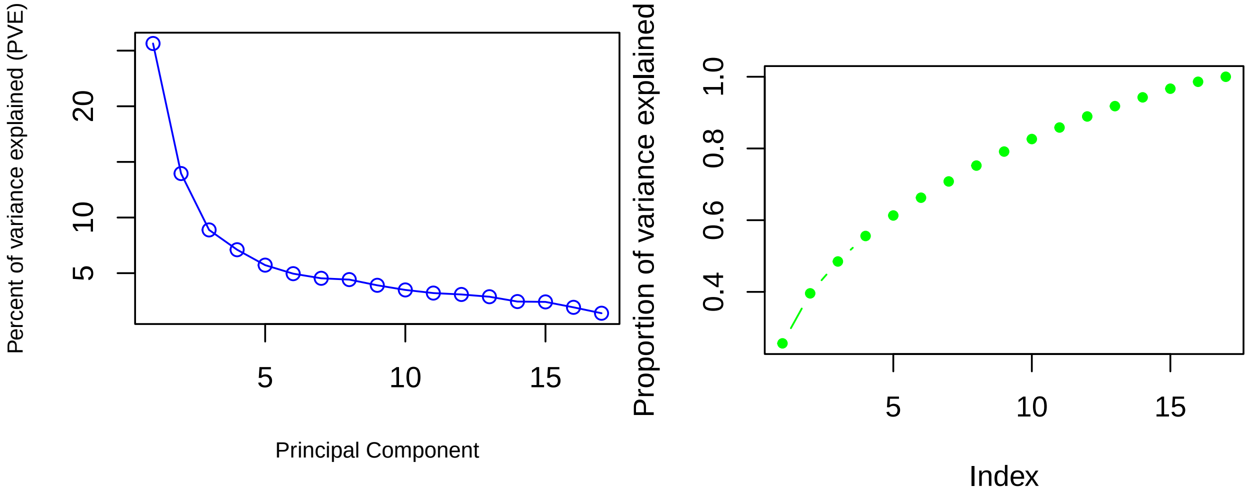

Figure 2: Scree plot for PCA

After applying PCA, a pairs plot was created to explore the dimension reduction in the data. Because there are a large number of components, the following pairs plot in Figure 3 illustrates the first few principal components.

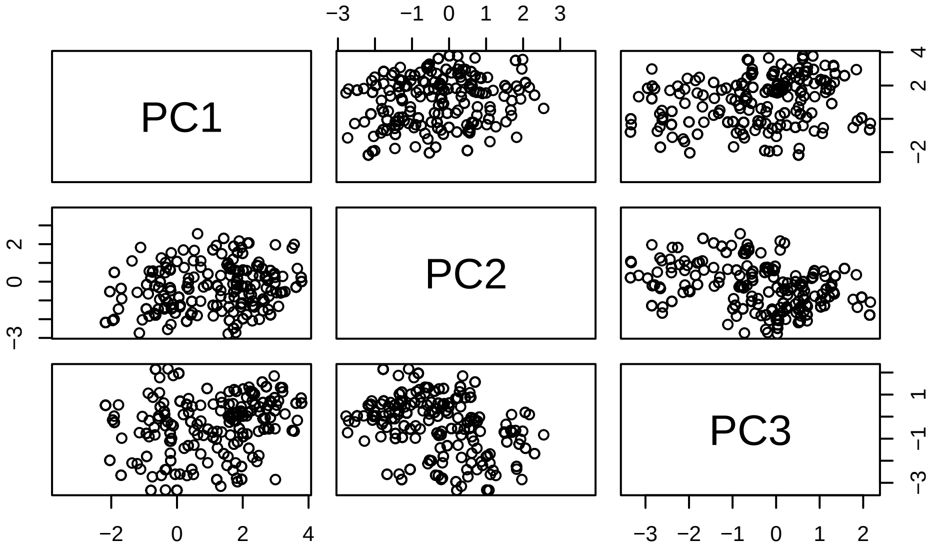

Figure 3: Pairs plot after PCA

The second method was FA, which involves replacing $p$ observed variables by $q < p$ latent factors. The model can be written as $X_i = \mu + \Lambda U_i + \epsilon$ for $i = 1, ..., n$ factors (King-Yu, 2024). For each FA, a hypothesis test is performed where the null hypothesis states that the number of factors in the model is sufficient. We performed FA using 1 to 9 factors and selected the optimal factor size as 9 because it failed to reject the null hypothesis (i.e. had a p-value larger than a significance level of $\alpha = 0.05$) while the other factor sizes rejected the null hypothesis.

After applying PCA and FA to the data, we then performed a clustering analysis. First, we performed clustering on the full data set. Then we applied clustering using only the PCs from our PCA.

## Results

Table 1 below summarizes the results of clustering prior to and after PCA. The adjusted Rand index (ARI) is lower after PCA was applied compared to using the full set of variables. We see a similar pattern for the misclassification error rate where the performance is better for clustering prior to PCA compared to after.

Table 1: Clustering performance comparison

| Method | Adjusted Rand Index | Misclassification Error Rate |
|---|---:|---:|
| Clustering prior to PCA | 0.193 | 0.315 |
| Clustering after PCA | 0.035 | 0.869 |

## Conclusion

Overall, our study shows that applying PCA prior to clustering for diabetes classification is not very effective. A few ways to improve the accuracy of these models include feature engineering, performing more extensive data cleansing prior to modelling, and incorporating subject-matter expertise when interpreting results.

## References

Islam, M. M. F., Ferdousi, R., Rahman, S., & Bushra, H. Y. (2020). Likelihood prediction of diabetes at early stage using data mining techniques. In M. Gupta, D. Konar, S. Bhattacharyya, & S. Biswas (Eds.), *Computer vision and machine intelligence in medical image analysis* (pp. 113–125). Springer Singapore. https://doi.org/10.1007/978-981-13-8798-2_12

King-Yu, S. (2024). *Statistical learning lecture 11: Principal component analysis and factor analysis.*

Larxel. (2024). *Early classification of diabetes.* https://www.kaggle.com/datasets/andrewmvd/early-diabetes-classification

## Appendix

```r
# ----- PACKAGE & DATA SETUP ----- #
library(tidyverse)
library(GGally)
library(mixture)
library(e1071)
library(kableExtra)

# Load data
diabs_raw <- read.csv("/Users/Vivian/Documents/Workspaces/R_Workspace/stats790/A1/diabetes_d

# ----- DATA TRANSFORMATION ----- #
# Check the data shape and for missing data
head(diabs_raw)
sum(sapply(diabs_raw, function(col){ifelse(is.na(col), 1, 0)}))
sum(sapply(diabs_raw, function(col){ifelse(sum(col == ""), 1, 0)}))

# Data transformation for analysis
diabs_int <- diabs_raw %>%
  mutate(gender = as.integer(ifelse(gender=="Male",1, ifelse(gender=="Female",0,
                            NA))))

# Data transformation for visualization
diabs_pairs <- diabs_raw
diabs_pairs[diabs_pairs==1]<-"Y"
diabs_pairs[diabs_pairs==0]<-"N"
diabs_pairs <- diabs_pairs %>%
  mutate(diabetes = class, gender = substring(gender, 1, 1)) %>%
  select(-c(class))

# ----- DATA VISUALIZATION ----- #
# Full pairs plot
ggpairs(data=diabs_pairs)

# Smaller pairs plot
ggpairs(data=diabs_pairs[,c(1,2,3,4,17)], aes(colour=diabetes, alpha=0.4))

# Principal component analysis (PCA)
pcDiabs<-prcomp(diabs_int, scale=TRUE)
summary(pcDiabs)

# Generate a scree plots for PCA
pve <- 100 * pcDiabs$sdev^2 / sum(pcDiabs$sdev^2)
plot(pve, type = "o", cex.lab=0.75,
  xlab = "Principal Component", col = "blue",
  ylab = "Percent of variance explained (PVE)")
plot(summary(pcDiabs)$importance[3,],
     type="b", pch=20, col="green",
     ylab="Proportion of variance explained")

pairs(pcDiabs$x, col=diabs_int[,17])
pairs(pcDiabs$x[,1:3], col=diabs_int[,17])

factanal(diabs_int, 1, scale=TRUE)
factanal(diabs_int, 2, scale=TRUE)
factanal(diabs_int, 3, scale=TRUE)
factanal(diabs_int, 4, scale=TRUE)
factanal(diabs_int, 5, scale=TRUE)
factanal(diabs_int, 6, scale=TRUE)
factanal(diabs_int, 7, scale=TRUE)
factanal(diabs_int, 8, scale=TRUE)
factanal(diabs_int, 9, scale=TRUE)
```

```r
# Clustering prior to PCA
x <- scale(diabs_int[,-c(17)])
gpcmDiabs <- gpcm(x, G=1:6, start=0, atol=1e-2)
gpcmDiabs

# Results
tabDiabs<-table(diabs_int[,5], gpcmDiabs$map)
tabDiabs

# Clustering after PCA
gpcmDiabsPCA <- gpcm(pcDiabs$x[,1:11], G=1:6, start=0, atol=1e-2)
gpcmDiabsPCA

# Results
tabDiabsPCA<-table(diabs_int[,5], gpcmDiabsPCA$map)
tabDiabsPCA

# RESULTS
method <- c("Clustering prior to PCA", "Clustering after PCA")
crand <- c(classAgreement(tabDiabs)$crand,
           classAgreement(tabDiabsPCA)$crand)
misclass <- c(1-classAgreement(tabDiabs)$diag,
              1-classAgreement(tabDiabsPCA)$diag)
summary <- data.frame("Method" = method,
                      "Adjusted Rand Index" = round(crand,3),
                      "Misclassification Error Rate" = round(misclass,3),
                      check.names = FALSE)
kable(summary)
```