9 Principal Component Analysis

Immanuel Klein

```
library("tidyverse")
library("ggplot2")
```

This task is about analyzing the Iris dataset using PCA and K-means clustering. First, we create a reduced dataset excluding the species variable, and perform PCA using the empirical covariance matrix to calculate loadings and scores, interpreting the first two principal components and their explained variance. We then repeat the same analysis using the correlation matrix and compare the results. Next, we convert the petal length to millimeters, and look at the impact on PCA using both covariance and correlation matrices. Then we use the original dataset to plot the first two principal components with species marked by color, assessing whether this 2D plot represents the original 4D space. Finally, we perform K-means clustering on the first two principal components with K=3, and compare the results to the actual labels to evaluate classification accuracy and whether using the entire dataset improves this accuracy.

Exercise (a)

```
data(iris)
iris.reduced <- iris[, -5]

# Extract loadings and scores, perform PCA with cov matrix
pca.result.cov <- prcomp(iris.reduced, scale. = FALSE)

# Show loadings and scores
pca.result.cov$rotation
```

```
                    PC1         PC2         PC3        PC4
Sepal.Length  0.36138659 -0.65658877  0.58202985  0.3154872
Sepal.Width  -0.08452251 -0.73016143 -0.59791083 -0.3197231
```

```
Petal.Length  0.85667061  0.17337266 -0.07623608 -0.4798390
Petal.Width   0.35828920  0.07548102 -0.54583143  0.7536574
```

```r
head(pca.result.cov$x, 5)
```

```
           PC1        PC2         PC3          PC4
[1,] -2.684126 -0.3193972  0.02791483  0.002262437
[2,] -2.714142  0.1770012  0.21046427  0.099026550
[3,] -2.888991  0.1449494 -0.01790026  0.019968390
[4,] -2.745343  0.3182990 -0.03155937 -0.075575817
[5,] -2.728717 -0.3267545 -0.09007924 -0.061258593
```

```r
# Calculate (cumulative) explained variance.
variance.explained <-
  pca.result.cov$sdev^2 / sum(pca.result.cov$sdev^2)
cumulative.variance.explained <- sum(variance.explained[1:2])
paste('Cumulative Variance Explained by First Two PCs: ',
      cumulative.variance.explained)
```

```
[1] "Cumulative Variance Explained by First Two PCs:  0.977685206318795"
```

- PC1 seems to be a general factor, particularly related to petal characteristics. Flowers with larger petal length and width will have higher values on this component. Because `Sepal.Width` contributes negatively, this dimension might also separate flowers with wide sepals from those with narrow sepals.
- PC2 mainly represents the width of the sepal. This component could be interpreted as differentiating between flowers with narrow versus wide sepals.

Exercise (b)

```r
# Calculate the pcs with cor matrix
pca.result.cor <- prcomp(iris.reduced, scale. = TRUE)

# Show loadings and scores
pca.result.cor$rotation
```

```
                   PC1         PC2        PC3        PC4
Sepal.Length  0.5210659 -0.37741762  0.7195664  0.2612863
Sepal.Width  -0.2693474 -0.92329566 -0.2443818 -0.1235096
Petal.Length  0.5804131 -0.02449161 -0.1421264 -0.8014492
Petal.Width   0.5648565 -0.06694199 -0.6342727  0.5235971
```

```
head(pca.result.cor$x, 5)
```

```
           PC1        PC2         PC3         PC4
[1,] -2.257141 -0.4784238  0.12727962  0.02408751
[2,] -2.074013  0.6718827  0.23382552  0.10266284
[3,] -2.356335  0.3407664 -0.04405390  0.02828231
[4,] -2.291707  0.5953999 -0.09098530 -0.06573534
[5,] -2.381863 -0.6446757 -0.01568565 -0.03580287
```

```
# Calculate explained variance.
variance.explained <-
  pca.result.cor$sdev^2 / sum(pca.result.cor$sdev^2)
cumulative.variance.explained <- sum(variance.explained[1:2])
paste('Variance Explained by First Two PCs: ',
      cumulative.variance.explained)
```

```
[1] "Variance Explained by First Two PCs:  0.958132072000016"
```

The results of using the correlation matrix are somewhat similar to those of using the covariation matrix when looking at the sign only. However, regarding magnitude they differ quite a bit.

Exercise (c)

```
# Create a new dataset with Petal.Length in millimeters
iris.mm.reduced <- iris.reduced
iris.mm.reduced$Petal.Length <- iris.mm.reduced$Petal.Length * 10

# Perform PCA with cov matrix
pca.result.mm.cov <- prcomp(iris.mm.reduced, scale. = FALSE)
# Loadings and variance explained
loadings.mm.cov <- pca.result.mm.cov$rotation
variance.explained.mm.cov <-
  pca.result.mm.cov$sdev^2 / sum(pca.result.mm.cov$sdev^2)
cumulative.variance.explained.mm.cov <-
  sum(variance.explained.mm.cov[1:2])
print('Covariance Matrix')
```

```
[1] "Covariance Matrix"
```

```
pca.result.mm.cov$rotation
```

```
                    PC1          PC2         PC3         PC4
Sepal.Length  0.04083715 -0.723859293  0.60196083  0.33466881
Sepal.Width  -0.01055112 -0.689576596 -0.63020980 -0.35666286
Petal.Length  0.99824759  0.022442542 -0.01090193 -0.05365844
Petal.Width   0.04150602 -0.002859015 -0.49026515  0.87057979
```

```
paste('Variance Explained by First Two PCs: ',
      cumulative.variance.explained.mm.cov)
```

```
[1] "Variance Explained by First Two PCs:  0.999649906773551"
```

```
# Perform PCA with cor matrix
pca.result.mm.cor <- prcomp(iris.mm.reduced, scale. = TRUE)
# Loadings and variance explained
loadings.mm.cor <- pca.result.mm.cor$rotation
variance.explained.mm.cor <-
  pca.result.mm.cor$sdev^2 / sum(pca.result.mm.cor$sdev^2)
cumulative.variance.explained.mm.cor <-
  sum(variance.explained.mm.cor[1:2])
print('Correlation Matrix')
```

```
[1] "Correlation Matrix"
```

```
pca.result.mm.cor$rotation
```

```
                   PC1         PC2        PC3        PC4
Sepal.Length  0.5210659 -0.37741762  0.7195664  0.2612863
Sepal.Width  -0.2693474 -0.92329566 -0.2443818 -0.1235096
Petal.Length  0.5804131 -0.02449161 -0.1421264 -0.8014492
Petal.Width   0.5648565 -0.06694199 -0.6342727  0.5235971
```

```
paste('Variance Explained by First Two PCs: ',
      cumulative.variance.explained.mm.cor)
```

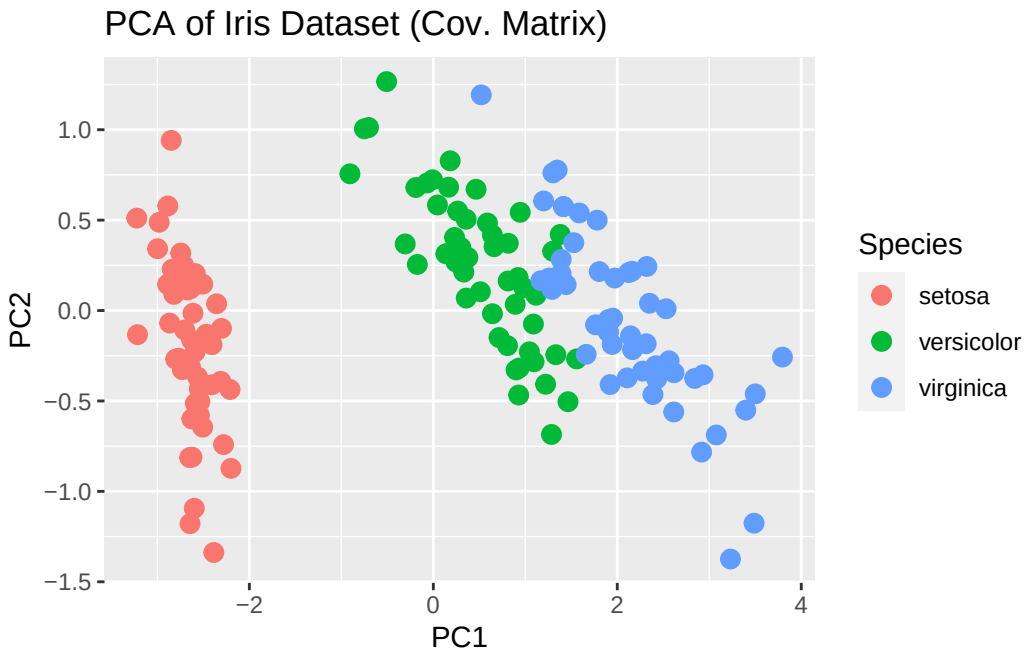
```
[1] "Variance Explained by First Two PCs:  0.958132072000016"
```

- For the covariance matrix, the principal components have definitely changed in their loadings. The explained variance has also increased.
- For the correlation matrix, the principal components have not changed in their loadings. The explained variance has also stayed the same.

Exercise (d)

```r
# Extract the first two principal components
pca.data <- data.frame(PC1 = pca.result.cov$x[,1],
                       PC2 = pca.result.cov$x[,2],
                       Species = iris$Species)

ggplot(pca.data, aes(x = PC1, y = PC2, color = Species)) +
  geom_point(size = 3) +
  labs(title = "PCA of Iris Dataset (Cov. Matrix)",
       x = "PC1",
       y = "PC2")
```



- The first principal component seems to capture the most variance among the species, with a clear separation from left to right.

- The first two principal components capture a significant amount of variance in the dataset, especially in distinguishing setosa from the other two species. This suggests that the two-dimensional plot provides a somewhat reasonable representation of the data's structure.
- However, the overlap between versicolor and virginica in this plot indicates that not all the information from the original space is captured by the first two principal components. Some important variations that distinguish these two species might be present in the third and fourth components.

Exercise (e)

```
pca.data <- data.frame(PC1 = pca.result.cov$x[,1],
                       PC2 = pca.result.cov$x[,2])

set.seed(47)

# Perform K-means clustering with K = 3
kmeans.result <- kmeans(pca.data, centers = 3, nstart = 20)

# Convert Species to integers
species.as.int <- as.integer(iris$Species)

# See from tables which cluster belongs to which species and adjust
table(species.as.int, iris$Species)
```

```
species.as.int setosa versicolor virginica
             1     50          0         0
             2      0         50         0
             3      0          0        50
```

```
cluster.adjusted <- kmeans.result$cluster
cluster.adjusted[kmeans.result$cluster == 1] <- 2 # versicolor
cluster.adjusted[kmeans.result$cluster == 2] <- 3 # virginica
cluster.adjusted[kmeans.result$cluster == 3] <- 1 # setosa
table(cluster.adjusted, iris$Species)
```

```
cluster.adjusted setosa versicolor virginica
```

```
  1     50          0          0
  2      0         47         14
  3      0          3         36
```

```
# Calculate classification error
classification.error <-
  mean(cluster.adjusted != species.as.int)
classification.error
```

```
[1] 0.1133333
```

```
# Now with the original dataset
set.seed(47)
kmeans.full <- kmeans(iris.reduced, centers = 3, nstart = 20)

# See from tables which cluster belongs to which species and adjust
table(species.as.int, iris$Species)
```

```
species.as.int setosa versicolor virginica
             1     50          0         0
             2      0         50         0
             3      0          0        50
```

```
cluster.adjusted.full <- kmeans.full$cluster
cluster.adjusted.full[kmeans.full$cluster == 1] <- 2 # versicolor
cluster.adjusted.full[kmeans.full$cluster == 2] <- 3 # virginica
cluster.adjusted.full[kmeans.full$cluster == 3] <- 1 # setosa
table(cluster.adjusted.full, iris$Species)
```

```
cluster.adjusted.full setosa versicolor virginica
                    1     50          0         0
                    2      0         48        14
                    3      0          2        36
```

```
# Calculate classification error
classification.error.full <-
  mean(cluster.adjusted.full != species.as.int)
classification.error.full
```

```
[1] 0.1066667
```

When using the entire dataset, one data point changes clusters, which leads to a small reduction in the classification error.