José Antonio Álvarez Ocete
917933752.

Homework 6

Problem 1

$$\mathbb{X} = \begin{pmatrix} 2 & 2 \\ 2 & 4 \\ 2 & 6 \\ 0 & 0 \\ -1 & -4 \\ -2 & -4 \\ -3 & -4 \end{pmatrix}$$

ⓐ Give the formula for the first and second PC:

$$\hat{Y}_1 = \vec{u}_1^T \cdot \vec{X} = u_{11} \cdot X_1 + u_{12} \cdot X_2$$
$$\hat{Y}_2 = \vec{u}_2^T \cdot \vec{X} = u_{21} \cdot X_1 + u_{22} \cdot X_2$$

where $\hat{Cov}(\mathbb{X}) = S = V \cdot \Lambda \cdot V^T$ is a spectral decomposition and $V = (\vec{u}_1 \ \vec{u}_2)$

Let's compute S:

$$\bar{\vec{x}} = \frac{1}{n} \mathbb{X}^T \cdot \mathbb{1}_n = \frac{1}{n} \begin{pmatrix} 2 & 2 & 2 & 0 & -1 & -2 & -3 \\ 2 & 4 & 6 & 0 & -4 & -4 & -4 \end{pmatrix} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} = \underline{\begin{pmatrix} 0 \\ 0 \end{pmatrix}}.$$

$$S = \frac{1}{n-1} (\mathbb{X} - \underbrace{\mathbb{1}_n \bar{\vec{x}}^T}_{= \vec{0}})^T (\mathbb{X} - \underbrace{\mathbb{1}_n \cdot \bar{\vec{x}}^T}_{= \vec{0}}) = \frac{1}{n-1} \mathbb{X}^T \mathbb{X} \underset{n=7}{=} \frac{1}{6} \begin{pmatrix} 26 & 48 \\ 48 & 104 \end{pmatrix} = \underline{\frac{1}{3} \begin{pmatrix} 13 & 24 \\ 24 & 52 \end{pmatrix}}$$

Find a spectral decomposition:

$$|S - \lambda I| = \begin{vmatrix} \frac{13}{3} - \lambda & 8 \\ 8 & \frac{52}{3} - \lambda \end{vmatrix} = \left(\frac{13}{3} - \lambda\right)\left(\frac{52}{3} - \lambda\right) - 64 = \frac{676}{9} - \lambda \frac{65}{3} + \lambda^2 - 64$$

$$= \lambda^2 - \lambda \frac{65}{3} + \frac{100}{9} = 0 \rightsquigarrow \lambda = \frac{65/3 \pm \sqrt{(65/3)^2 - 4 \cdot \frac{100}{9}}}{2} = \frac{1}{2}\left(\frac{65}{3} \pm \left(\frac{4225}{9} - \frac{400}{9}\right)^{1/2}\right) =$$

$$= \frac{1}{2}\left(\frac{65}{3} \pm \frac{1}{3}(\underset{3^2 \cdot 5^2 \cdot 17}{3825})^{1/2}\right) = \frac{1}{2}\left(\frac{65}{3} \pm \frac{15}{3}\sqrt{17}\right)$$

$$\rightarrow \begin{cases} \lambda_1 = \dfrac{65 + 15\sqrt{17}}{6} \approx \underline{21'1411} \\ \lambda_2 = \dfrac{65 - 15\sqrt{17}}{6} \approx \underline{0'5256} \end{cases} \quad (\lambda_1 > \lambda_2).$$

Obtain the eigen-vectors: $\begin{pmatrix} u_{11} \\ u_{12} \end{pmatrix}$

$|S-\lambda_1 I|=0 \to$ the rows are dependent.

(V1) $(S-\lambda_1 I)\vec{u}_1 = 0 \iff \begin{pmatrix} 13/3-\lambda_1 & 8 \\ 8 & 52/3-\lambda_1 \end{pmatrix} \vec{u}_1 = 0 \Rightarrow$

Give $u_{11}=1$ and normalize later

$\iff \left(\frac{13}{3}-\lambda_1\right)u_{11} + 8u_{12} = 0 \iff \left(\frac{13}{3} - \frac{65+15\sqrt{17}}{6}\right)u_{11} = -8u_{12} \iff$

$\iff u_{12} = \frac{-1}{8}\cdot\left(\frac{26-65+15\sqrt{17}}{6}\right) = \frac{39+15\sqrt{17}}{48} \Rightarrow \vec{u}_1 = \begin{pmatrix} 1 \\ \frac{39+15\sqrt{17}}{48} \end{pmatrix} \xrightarrow{\text{normalize}}$

$\vec{u}_1 \approx \begin{pmatrix} 0'4298 \\ 0'9029 \end{pmatrix}$, (V2) Compute $\vec{u}_2$ the same way: $\vec{u}_2 \approx \begin{pmatrix} -0'9029 \\ 0'4298 \end{pmatrix}$.

Then: $S = V\cdot\Lambda\cdot V^T$ where $V = (\vec{u}_1\ \vec{u}_2) = \begin{pmatrix} 0'4298 & -0'9029 \\ 0'9029 & 0'4298 \end{pmatrix}$

and $\Lambda = \begin{pmatrix} \frac{65+15\sqrt{17}}{6} & 0 \\ 0 & \frac{65-15\sqrt{17}}{6} \end{pmatrix}$.

Finally, the first and second PCs are:

$\hat{Y}_1 = \vec{u}_1^T\cdot\vec{X} = 0'4298\cdot X_1 + 0'9029\cdot X_2$

$\hat{Y}_2 = \vec{u}_2^T\cdot\vec{X} = -0'9029\cdot X_1 + 0'4298\cdot X_2$

(b) Determine the proportion of total variance due to the first sample PC.

That is: $\frac{\hat{\lambda}_1}{S_{11}+S_{22}}$, where $\hat{\lambda}_1 = \frac{65+15\sqrt{17}}{6}$, $S_{11}=\frac{13}{3}$, $S_{22}=\frac{52}{3}$.

$$\frac{\hat{\lambda}_1}{S_{11}+S_{22}} \approx \frac{21'1411}{21'6} \approx 0'9757$$

Scanned with CamScanner

(c) Compare the contributions of the two variates to the determination of the first PC based on loadings:

As seen in (a): $\hat{Y}_1 = \underbrace{0'4298}_{u_{11}} \cdot X_1 + \underbrace{0'9029}_{u_{12}} \cdot X_2$

Where we can see using the loadings that both variates contribute positevely and the contribution of X_2 is considerably greater than the first's.

(d) Compare the contribution of the two variates based to the determination of the first PC based on sample correlations.

Let's compute the sample correlations:

$Corr(Y_1, X_1) = r_{11} = u_{11} \cdot \sqrt{\frac{\lambda_1}{S_{11}}} \simeq \underline{0'9493}$

$Corr(Y_1, X_2) = r_{12} = u_{12} \cdot \sqrt{\frac{\lambda_1}{S_{22}}} \simeq \underline{0'9972}$

Obtaining the same result: Both variates contribute positevely and X_2 contribution is greater than X_1's.

(e) Repeat (a) to (d) with standarized data.

Using that $Cov(\vec{Z}) = Corr(\vec{X})$; we can study $Corr(\vec{X})$ without computing $\vec{Z}$. Let's obtain the spectral decomposition of $Corr(\vec{X})$.

Compute $Corr(\vec{X}) = D^{-1/2} \cdot S \cdot D^{-1/2}$, $D^{-1/2} = \begin{pmatrix} \sqrt{3/13} & 0 \\ 0 & \sqrt{3/52} \end{pmatrix} = \begin{pmatrix} 1/\sqrt{s_{11}} & 0 \\ 0 & \frac{1}{\sqrt{s_{22}}} \end{pmatrix}$

$$\left\{ \frac{s_{12}}{\sqrt{s_{11}}\sqrt{s_{22}}} = \frac{8}{\sqrt{\frac{13 \cdot 52}{3 \cdot 3}}} = \frac{8 \cdot 3}{13 \cdot 2} \approx 0'9231 \right\} \Rightarrow$$

$$\Rightarrow Corr(\vec{X}) = \begin{pmatrix} 1 & 0'9231 \\ 0'9231 & 1 \end{pmatrix}$$

, compute its spectral decomposition:

$Corr(\vec{X}) = V_2 \circ \Lambda_2 \cdot V_2$, $P_2 = \begin{pmatrix} \sqrt{2}/2 & -\sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$ and

$$\Lambda_2 = \begin{pmatrix} 1'9231 & 0 \\ 0 & 0'0769 \end{pmatrix}.$$

Finally:

$$\hat{Y}_{2,1} = \frac{1}{\sqrt{2}} Z_1 + \frac{1}{\sqrt{2}} Z_2$$

$$\hat{Y}_{2,2} = \frac{-1}{\sqrt{2}} Z_1 + \frac{1}{\sqrt{2}} Z_2.$$

Proportion of total sample variance due to the 1^{st} sample PC:

$$\frac{\hat{\lambda}_{2,1}}{\underbrace{s_{11}}_{1} + \underbrace{s_{22}}_{1}} \approx 0'9615$$

Using loadings we see that both variates contribute the same to the first PC ($u_{11} = u_{12} = \frac{1}{\sqrt{2}}$), and both positively. We obtain the same result comparing sample correlations:

$$Corr(Y_1, Z_1) = u_{11}\sqrt{\lambda_1} = u_{11}\sqrt{\lambda_2} = Corr(Y_1, Z_2) = 0'9806.$$

Problem 2

If the first PC of X_1, X_2 is

$Y_1 = \frac{1}{\sqrt{2}} X_1 + \frac{1}{\sqrt{2}} X_2$, is it possible that $Corr(X_1, X_2) < 0$?

We know that $Y_1 = \vec{v}_1^T \vec{X} \Rightarrow \underline{\vec{v}_1 = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix}}$ the first eigen-vector

$Corr(X_1, X_2) = \frac{Cov(X_1, X_2)}{\sqrt{Var X_1}\sqrt{Var X_2}}$, $Corr(X_1, X_2) < 0 \Leftrightarrow Cov(X_1, X_2) < 0$.

Let $V \cdot \Lambda \cdot V^T = \Sigma$ be a spectral decomposition of $\Sigma = Cov(\vec{X})$.

Then $V = (\vec{v}_1 \ \vec{v}_2)$, $\Lambda = \begin{pmatrix}\lambda_1 & \\ & \lambda_2\end{pmatrix}$ with $\lambda_1 \geq \lambda_2 > 0$.

Since $\vec{v}_1 \perp \vec{v}_2$ and $\vec{v}_1 = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix} \Rightarrow \vec{v}_2 = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\-1\end{pmatrix} \Rightarrow \underline{V = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\1 & -1\end{pmatrix}}$.

$$\Sigma = \begin{pmatrix}\sigma_{11} & \sigma_{12}\\ \sigma_{21} & \sigma_{22}\end{pmatrix} = \frac{1}{2}\begin{pmatrix}1 & 1\\1 & -1\end{pmatrix}\begin{pmatrix}\lambda_1 & 0\\0 & \lambda_2\end{pmatrix}\begin{pmatrix}1 & 1\\1 & -1\end{pmatrix} = \frac{1}{2}\begin{pmatrix}\lambda_1 & \lambda_2\\ \lambda_1 & -\lambda_2\end{pmatrix}\begin{pmatrix}1 & 1\\1 & -1\end{pmatrix} =$$

$$= \frac{1}{2}\begin{pmatrix}\lambda_1+\lambda_2 & \lambda_1-\lambda_2\\ \lambda_1-\lambda_2 & \lambda_1+\lambda_2\end{pmatrix} \Rightarrow \left.\begin{array}{l}\sigma_{12} = Cov(X_1, X_2) = \frac{\lambda_1-\lambda_2}{2}\\ \lambda_1 \geq \lambda_2 > 0\end{array}\right\} \sigma_{21} \geq 0 \Rightarrow$$

$$\Rightarrow \underline{Corr(X_1, X_2) \geq 0}$$

Problem 3

Problem 8.12 on page 474: Using table 1.5, summarize the data in fewer than $p=7$ dimensions. Conduct PCA using both $S = \widehat{Cov}(\vec{X})$ and $R = \widehat{Corr}(\vec{X})$. What have you learned? Does it make any difference which matrix is used for the analysis? Can the data be summarize in 3 or less dim.? Can you interpret the PC's?

Using the R code attached we conduct PCA. Starting with $Cov(\vec{X}) = S$:

$$\bar{\vec{x}} = \begin{pmatrix} 7'5 \\ 73'8571 \\ 4'5476 \\ 2'1905 \\ 10'0476 \\ 9'4048 \\ 3'0952 \end{pmatrix},$$

Compute S and its spectral decomposition. What it is really relevant about S is that

$S_{ii} < 10 \quad \forall i \in \{1, 3, 4, 7\}$

$S_{22} = 300'52$

$S_{55} = 11'36$, $S_{66} = 30'98$.

$(S_{22} >>> S_{ii} \quad \forall i \neq 2)$.

The eigenvalues of S are:

$\lambda_1 = 304'26$, $\lambda_2 = 28'28$, $\lambda_3 = 11'46$, $\lambda_4 = 2'52$

$\lambda_5 = 1'27$, $\lambda_6 = 0'5287$, $\lambda_7 = 0'2066$.

Again, $\lambda_1 >> \lambda_2 > \lambda_3 >> \lambda_4$ —

The proportion of total sample variance due to the first 3 PCs is:

$$\frac{\lambda_1 + \lambda_2 + \lambda_3}{S_{11} + \dots + S_{77}} \approx 0'9870 = \frac{343'999}{348'541}$$

This could tell us that indeed we can summarize the data using the first 3 PCs. However, S_{22} being so high (and so close to $Var\, Y_1 = \lambda_1 = 304'26$) makes us think that the 1^{st} PC is basically X_2, and due to it's huge variance we are obtaining a unbalanced PC. We can check the loadings to confirm our hypothesis:

$$\vec{u}_1 = \begin{pmatrix} 0'010 \\ -0'99 \\ -0'014 \\ 0'004 \\ 0'024 \\ -0'11 \\ 0'00234 \end{pmatrix}$$

where we can see that X_2 is unbalancing our analysis. To dodge this kind of effects we should study R instead of S.

We procede to the analysis of R; It's eigen values are:

$\lambda_1' = 2'3368$, $\lambda_2' = 1'386$, $\lambda_3' = 1'204$, $\lambda_4' = 0'7271$

$\lambda_5' = 0'6535$, $\lambda_6' = 0'5367$, $\lambda_7' = 0'1559$.

We can see they are more balanced. The first 3 eigen vectors are:

$$\vec{u}_1 = \begin{pmatrix} 0'2368 \\ -0'2055 \\ -0'5511 \\ -0'3776 \\ -0'4980 \\ -0'3246 \\ -0'3194 \end{pmatrix} \quad \vec{u}_2 = \begin{pmatrix} 0'2754 \\ -0'5266 \\ -0'0068 \\ 0'4346 \\ 0'1997 \\ -0'5670 \\ 0'3079 \end{pmatrix} \quad \vec{u}_3 = \begin{pmatrix} 0'6434 \\ 0'2245 \\ -0'1136 \\ -0'4071 \\ 0'1466 \\ 0'1598 \\ 0'5410 \end{pmatrix}$$

We can see a huge different with the $\vec{u}_i$ previously calculated for S: there is a balance between the contributions of every variate, and X_2 doesn't appeat more than the others. This confirms our hypothesis even more (it could also happen that X_2 was a relevant variate when the data is standarized, but this teachs us that that doesn't always happen, so we should always standarized our data for PCA).

Finally, let's compute the proportion of total sample variance due to the first 3 PCs:

$$\frac{\lambda_1+\lambda_2+\lambda_3}{S_{11}+\_\_\_+S_{77}} = \frac{\lambda_1+\lambda_2+\lambda_3}{7} \approx \underline{0'7038}$$

We are summarizing a lot of our data using only 3 PCs, but not quite enough to say that we can just use this 3 PCs instead of $\mathbb{X}$. Again, this is very different from the 0'99 obtained studying S.

Problem 4 Consider two samples of equal sizes $n_1 = n_2$:

$\vec{x}_{11}, \dots, \vec{x}_{1n_1}$
$\vec{x}_{21}, \dots, \vec{x}_{2n_2}$ with summary statistics: $\bar{\vec{x}}_1 = \begin{pmatrix} 6 \\ 0 \end{pmatrix}, \bar{\vec{x}}_2 = \begin{pmatrix} 0 \\ 6 \end{pmatrix}, S_1 = \begin{pmatrix} 4 & 2 \\ 2 & 5 \end{pmatrix}, S_2 \begin{pmatrix} 4 & 2 \\ 2 & 3 \end{pmatrix}$

For a new observation $\vec{x}_0 = \begin{pmatrix} x_{01} \\ x_{02} \end{pmatrix}$, consider:

1. Classifier 1: Fisher's rule if only x_{01} is observed.
2. Classifier 2: Fisher's rule if only x_{02} is observed.
3. Classifier 3: Fisher's rule based on $\vec{x}_0$.

Does there exist a $\vec{x}_0$, such that Classifiers 1 and 2 agree while disagreeing with Classifier 3?

Let's compute the different classifiers using Fisher's rule:

$$(\bar{\vec{x}}_1 - \bar{\vec{x}}_2)^T S_{pooled}^{-1} \left(\vec{x}_0 - \frac{1}{2}(\bar{\vec{x}}_1 + \bar{\vec{x}}_2)\right) \geq 0. \quad ; S_{pooled} = \begin{pmatrix} 4 & 2 \\ 2 & 4 \end{pmatrix}.$$

① $(\overline{x_{11}} - \overline{x_{12}})^T \cdot \sigma_{11}^{-1} \left(x_{01} - \frac{1}{2}(\overline{x_{11}} + \overline{x_{12}})\right) = (6-0) \cdot \frac{1}{4} \cdot \left(x_{01} - \frac{1}{2}(6+0)\right) =$

$= \frac{6}{4}(x_{01} - 3) \geq 0 \iff \underline{x_{01} \geq 3}$

② $(\overline{x_{21}} - \overline{x_{22}})^T \cdot \sigma_{22}^{-1} \left(x_{02} - \frac{1}{2}(\overline{x_{21}} + \overline{x_{22}})\right) = \frac{-6}{4}(x_{02} - 3) \geq 0 \iff \underline{x_{02} \leq 3}$

③ $S_{pooled}^{-1} = \frac{1}{16-4}\begin{pmatrix} 4 & -2 \\ -2 & 4 \end{pmatrix} = \underline{\frac{1}{6}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}}$

$(\bar{\vec{x}}_1 - \bar{\vec{x}}_2)^T S_{pooled}^{-1}\left(\vec{x}_0 - \frac{1}{2}(\bar{\vec{x}}_1 + \bar{\vec{x}}_2)\right) = \begin{pmatrix} 6 \\ -6 \end{pmatrix}^T \cdot \frac{1}{6}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}\left(\vec{x}_0 - \begin{pmatrix} 3 \\ 3 \end{pmatrix}\right) =$

$= (1\ ;\ -1)\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}\begin{pmatrix} x_{01} - 3 \\ x_{02} - 3 \end{pmatrix} = (3\quad -3)\begin{pmatrix} x_{01} - 3 \\ x_{02} - 3 \end{pmatrix} = 3(x_{01} - 3) - 3(x_{02} - 3) =$

$= 3x_{01} - 3x_{02} \geq 0 \iff \underline{x_{01} \geq x_{02}}$

Suppose ① and ② agree that $\vec{x}_0$ is Class 1, then:

$\begin{cases} x_{01} \geq 3 \\ x_{02} \leq 3 \end{cases} \Rightarrow x_{01} \geq 3 \geq x_{02} \rightarrow$

$\rightarrow x_{01} \geq x_{02} \Rightarrow$ Using ③ we obtain the same answer.

Suppose ① and ② agree that $\vec{x}_0$ is Class 2, then:

$\begin{cases} x_{01} < 3 \\ x_{02} > 3 \end{cases} \rightarrow x_{02} > 3 > x_{01} \rightarrow x_{02} > x_{01} \rightarrow$

$\rightarrow$ Using ③ we obtain the same answer.

Solution: No, there isn't a $\vec{x}_0$ such that both ① and ② agree, and at the same time disagree with ③.

Problem 5 Repeat the previous problem using $\bar{\vec{x}}_1 = \begin{pmatrix} 6 \\ 6 \end{pmatrix}$, $\bar{\vec{x}}_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$. (same S_1, S_2).

① $(\bar{x}_{11} - \bar{x}_{12})^T \cdot \sigma_{11}^{-1} \cdot \left(x_{01} - \frac{1}{2}(\bar{x}_{11} + \bar{x}_{12})\right) = 6/4 \cdot (x_{01} - 3) \geq 0$

$\iff \underline{x_{01} \geq 3}$

$$\left\{\begin{array}{l} x_{21} = x_{11} \\ x_{22} = x_{12} \\ \sigma_{11} = \sigma_{22} \end{array}\right.$$

② $(x_{21} - x_{22})^T \cdot \sigma_{22}^{-1} \cdot \left(x_{02} - \frac{1}{2}(x_{21} + x_{22})\right) =$

$= (x_{11} - x_{12})^T \cdot \sigma_{11}^{-1} \cdot \left(x_{02} - \frac{1}{2}(x_{11} + x_{12})\right) \geq 0 \iff$

$\iff \underline{x_{02} \geq 3}$

③ $(\bar{\vec{x}}_1 - \bar{\vec{x}}_2)^T \cdot S_{pooled}^{-1} \left(\vec{x}_0 - \frac{1}{2}(\bar{\vec{x}}_1 + \bar{\vec{x}}_2)\right) = (6 \;\; 6) \frac{1}{6} \cdot \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \cdot \left(\vec{x}_0 - \begin{pmatrix} 3 \\ 3 \end{pmatrix}\right) =$

$= (1 \;\; 1)\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}\begin{pmatrix} x_{01} - 3 \\ x_{02} - 3 \end{pmatrix} = (1 \;\; 1)\begin{pmatrix} x_{01} - 3 \\ x_{02} - 3 \end{pmatrix} = x_{01} - 3 + x_{02} - 3 =$

$= x_{01} + x_{02} - 6 \geq 0 \iff \underline{x_{01} + x_{02} \geq 6}$.

Suppose both ① and ② assign Class 1, then:

$\left.\begin{array}{l} x_{01} \geq 3 \\ x_{02} \geq 3 \end{array}\right\} x_{01} + x_{02} \geq 6 \rightarrow$ ③ assigns class 1.

Suppose both ① and ② assign Class 2, then:

$\left.\begin{array}{l} x_{01} < 3 \\ x_{02} < 3 \end{array}\right\} x_{01} + x_{02} < 6 \rightarrow$ ③ assigns class 2.

Solution: no, it doesn't exist $\vec{x}_0$ such that ① and ② agree and at the same time disagree with ③.

Problem 6 Considering three independent distribution:

$\pi_1: N_p(\vec{\mu}_1, \Sigma), n_1$

$\pi_2: N_p(\vec{\mu}_2, \Sigma), n_2$, use $S_{pooled} = \frac{1}{(n_1 + n_2 + n_3 - 3)} \cdot \left((n_1 - 1)S_1 + (n_2 - 1)S_2 + (n_3 - 1)S_3\right)$.

$\pi_3: N_p(\vec{\mu}_3, \Sigma), n_3$

Suppose that, upon classifying $\vec{x_0}$ using Fisher's rule, between π_1 and π_2 $\vec{x_0}$ is allocated to π_2 and between π_2 and π_3, $\vec{x_0}$ is allocated in π_3. Show that in the comparison between π_1 and π_3, $\vec{x_0}$ is allocated to π_3.

This is simply by transitivity of the Mahalanobis distance:

$$\left.\begin{array}{l} d_M(\vec{x_0}, \vec{\mu_2}) \leq d_M(\vec{x_0}, \vec{\mu_1}) \\ d_M(\vec{x_0}, \vec{\mu_3}) \leq d_M(\vec{x_0}, \vec{\mu_2}) \end{array}\right\} \Rightarrow d_M(\vec{x_0}, \vec{\mu_3}) \leq d_M(\vec{x_0}, \vec{\mu_1})$$

And knowing that upon comparing π_i, π_j $(i \neq j)$, Fisher's rule classifies $\vec{x_0}$ in π_i if and only if $d_M(\vec{x_0}, \vec{\mu_i}) \leq d_M(\vec{x_0}, \vec{\mu_j})$. This is only possible because the matrix S_{pooled} is common to the three Fisher's rules.

$\boxed{\text{Problem 7}}$ For the dataset on Table 1.6, construct Fisher's Rule. Moreover, calculate the apparent error rate (AER), as well as the expected actual error rate (EAER) using Lachenbruch's holdout.

Using the code attached we obtained the following Fisher's rule:

$$\vec{w} = \begin{pmatrix} 0'02341 \\ -0'03447 \\ 0'21027 \\ -0'08393 \\ -0'25345 \end{pmatrix}, \quad \vec{w_0} = \begin{pmatrix} -40'02724 \\ 162'7794 \\ 6'91909 \\ 216'267 \\ 7'35152 \end{pmatrix}, \quad \vec{w}^T \cdot (\vec{x_0} - \vec{w_0}) \geq 0$$

Using this classifier we can compute the AER by classifying each element of our population and seeing how many we classify incorrectly. We obtain 10 errors out of $n_1+n_2=98$ elements:

$$AER = \frac{10}{98} = 0'1020408$$

In order to compute the EAER we use Lachenbruch's holdout: For each element in our populations, recompute the Fisher's rule without using that element $(n_1+n_2=97)$ and classify it using the classifier obtained. By following this procedure we obtain a total of 13 errors:

$$EAER = \frac{13}{98} = 0'1327$$

As expected, EAER > AER. This doesn't <u>always</u> happen, but it is the expected result.

hw6 p1

For fractions:

```
library(MASS)
```

R Markdown

```
# Com
X <- cbind(c(2,2,2,0,-1,-2,-3),c(2,4,6,0,-4,-4,-4))
colnames(X) <- NULL
n <- dim(X)[1]
r <- dim(X)[2]
```

(1.b) Determine the proportion of total sample variance due to the first sample principale component.

```
# Compute sample mean and S
Ones <- rep(1,n)
x_sample_mean <- 1/n * t(X)%*%Ones
S <- 1/(n-1) * t(X - Ones%*%t(x_sample_mean))%*%(X - Ones%*%t(x_sample_mean))

# eigens
ev <- eigen(S)
eigen_values <- ev$values
V <- ev$vectors

prop <- eigen_values[1]/(S[1,1] + S[2,2])
cat( fractions(prop), '=', eigen_values[1], '/', (S[1,1] + S[2,2]))
```

```
## 0.975743 = 21.1411 / 21.66667
```

(1.c) Compare the contributions of the two variates to the determination of the first sample principal com-ponent based on loadings.

```
# We can see the loadings in V
V
```

```
##           [,1]       [,2]
## [1,] 0.4297717 -0.9029376
## [2,] 0.9029376  0.4297717
```

(1.d) Compare the contributions of the two variates to the determination of the first sample principal com-ponent based on sample correlations.

```
# Compute the correlations
cat('Corr(Y_1, X_1) =', V[1,1]*sqrt(eigen_values[1]/S[1,1]))
```

```
## Corr(Y_1, X_1) = 0.9492716
```

```
cat(' --- Corr(Y_1, X_2) =', V[2,1]*sqrt(eigen_values[1]/S[2,2]))
```

```
##  --- Corr(Y_1, X_2) = 0.9971958
```

(1.e) Repeat (a-d) with the data standarized.

```
# Com
D <- cbind(c(1/sqrt(S[1,1]),0),c(0,1/sqrt(S[2,2])))
Z <- (X - Ones%*%t(x_sample_mean))%*%t(D)
colnames(Z) <- NULL

# Compute sample mean and S
z_sample_mean <- 1/n * t(Z)%*%Ones
S_z <- 1/(n-1) * t(Z - Ones%*%t(z_sample_mean))%*%(Z - Ones%*%t(z_sample_mean))

# eigens
ev_z <- eigen(S_z)
eigen_values_z <- ev_z$values
V_z <- ev_z$vectors

prop_z <- eigen_values_z[1]/(S_z[1,1] + S_z[2,2])
cat( fractions(prop_z), '=', eigen_values_z[1], '/', (S_z[1,1] + S_z[2,2]))
```

```
## 0.9615385 = 1.923077 / 2
```

```
# We can see the Loadings in V
V_z
```

```
##           [,1]       [,2]
## [1,] 0.7071068 -0.7071068
## [2,] 0.7071068  0.7071068
```

```
# Compute the correlations
cat('Corr(Y_1, Z_1) =', V_z[1,1]*sqrt(eigen_values_z[1]))
```

```
## Corr(Y_1, Z_1) = 0.9805807
```

```
cat(' --- Corr(Y_1, Z_2) =', V_z[2,1]*sqrt(eigen_values_z[1]))
```

```
##  --- Corr(Y_1, Z_2) = 0.9805807
```

hw6 p3

For fractions:

```
library(MASS)
```

R Markdown

```
#import data
data <- read.table("T1-5.DAT")

#set up X
X <- as.matrix(data)[,1:7]
colnames(X) <- c('Wind','radiation','CO','NO','N02','0_3','HC')

n <- length(X[,1])
r <- length(X[1,])
```

(1.b) Determine the proportion of total sample variance due to the first sample principale component.

```
# Compute sample mean and S
Ones <- rep(1,n)
x_sample_mean <- 1/n * t(X)%*%Ones
S <- 1/(n-1) * t(X - Ones%*%t(x_sample_mean))%*%(X - Ones%*%t(x_sample_mean))

# eigens
ev <- eigen(S)
eigen_values <- ev$values
V <- ev$vectors

k <- 3
SS <- sum(diag((S)))
LL <- sum(eigen_values[1:k])
cat( fractions(LL/SS), '=', LL, '/', SS)
```

```
## 0.986968 = 343.9985 / 348.5407
```

```
# We can see the Loadings in V
V
```

```
##              [,1]        [,2]        [,3]          [,4]          [,5]
## [1,]  0.010039244  0.07622439  0.03087761  0.9203045748  0.3423859285
## [2,] -0.993199405  0.11615518  0.00659069 -0.0002118679  0.0022391022
## [3,] -0.014062314 -0.09956775 -0.18282641 -0.1382922410  0.6500776063
## [4,]  0.004710175  0.01320423 -0.13021553 -0.3277842624  0.6431560485
## [5,] -0.024255644 -0.15038113 -0.95526318  0.1023719020 -0.2065840405
## [6,] -0.112429558 -0.97335904  0.16981025  0.0632480276 -0.0002935726
## [7,] -0.002340785 -0.02382046 -0.08519558  0.1095073458  0.0619613872
##              [,6]         [,7]
## [1,]  0.011779079 -0.169729925
## [2,]  0.003353218 -0.001781987
## [3,] -0.563893916  0.443577538
## [4,]  0.497513370 -0.462855916
## [5,] -0.009009299 -0.105029951
## [6,]  0.051067254 -0.066992404
## [7,]  0.657012233  0.738019426
```

(1.e) Repeat with the data standarized. Aka, use R instead of S for the analysis.

```
# Com
R <- cor(X)

# eigens
ev_z <- eigen(R)
eigen_values_z <- ev_z$values
V_z <- ev_z$vectors

k <- 3
RR <- sum(diag((R)))
LL_r <- sum(eigen_values_z[1:k])
cat( fractions(LL_r/RR), '=', LL_r, '/', RR, fill=TRUE)
```

```
## 0.7038356 = 4.926849 / 7
```

```
# We can see the Loadings in V
eigen_values_z
```

```
## [1] 2.3367826 1.3860007 1.2040659 0.7270865 0.6534765 0.5366888 0.1558989
```

```
V_z
```

```
##            [,1]         [,2]       [,3]         [,4]        [,5]         [,6]
## [1,]  0.2368211  0.278445138  0.6434744  0.172719491  0.56053441 -0.223579220
## [2,] -0.2055665 -0.526613869  0.2244690  0.778136601 -0.15613432 -0.005700851
## [3,] -0.5510839 -0.006819502 -0.1136089  0.005301798  0.57342221 -0.109538907
## [4,] -0.3776151  0.434674253 -0.4070978  0.290503052 -0.05669070 -0.450234781
## [5,] -0.4980161  0.199767367  0.1965567 -0.042428178  0.05021430  0.744968707
## [6,] -0.3245506 -0.566973655  0.1598465 -0.507915905  0.08024349 -0.330583071
## [7,] -0.3194032  0.307882771  0.5410484 -0.143082348 -0.56607057 -0.266469812
##             [,7]
## [1,] -0.24146701
## [2,] -0.01126548
## [3,]  0.58524622
## [4,] -0.46088973
## [5,] -0.33784371
## [6,] -0.41707805
## [7,]  0.31391372
```

hw6 p7

For fractions:

```
library(MASS)
```

R Markdown

```
#import data
data <- read.table("T1-6.DAT")

#set up X
X_1<- as.matrix(data)[1:69,1:5]
X_2<- as.matrix(data)[70:98,1:5]

names <- c('(Age)','(SlL + SIR)','|S1L - S1R|','(S2L + S2R)','|S2L - S2R|')
colnames(X_1)<- names
colnames(X_2) <- names

n1 <- length(X_1[,1])
n2 <- length(X_2[,1])
r <- length(names)
```

Classifier function

```r
sampleMean<-function(X, n) {
  Ones <- rep(1,n)
  return (1/n * t(X)%*%Ones)
}
sampleCovariance<-function(X, n, sample_mean) {
  Ones <- rep(1,n)
  return (1/(n-1) * t(X - Ones%*%t(sample_mean))%*%(X - Ones%*%t(sample_mean)))
}

# If x_0 is NULL, the classifier is printed and nothing else is done
fisher<-function(X_1, X_2, x_0=NULL) {
  n1 <- length(X_1[,1])
  n2 <- length(X_2[,1])

  # Compute sample mean and S
  x1_sample_mean <- sampleMean(X_1, n1)
  x2_sample_mean <- sampleMean(X_2, n2)
  S1 <- sampleCovariance(X_1, n1, x1_sample_mean)
  S2 <- sampleCovariance(X_2, n2, x2_sample_mean)
  Spooled <- (n1-1)/(n1+n2-2) * S1 + (n2-1)/(n1+n2-2) * S2

  # Compute the classification
  w <- t(x1_sample_mean - x2_sample_mean) %*% solve(Spooled)
  mid <- 1/2*(x1_sample_mean + x2_sample_mean)

  # Print our classifier or classify the value
  if (is.null(x_0))
    cat(w, '*(x_0 -', mid, ') >= 0')
  else {
    result <- w%*%(x_0 - mid)
    if (result >= 0) {
      return (1)
    }
    return (2)
  }
}
```

Print the classiffier obtained

```r
# Use the classifier with our whole population, without predicting anything
fisher(X_1, X_2)
```

```
## 0.02340633 -0.03446657 0.2102708 -0.08393327 -0.2534507 *(x_0 - 40.02724 162.779
4 6.91909 216.267 7.351524 ) >= 0
```

Compute the apparent error rate (AER):

```r
# Returns 0 if then prediction is correct, 1 otherwise.
classify<-function(X_1, X_2, x_0, expeted_class) {
  if ( fisher(X_1, X_2, x_0) != expeted_class ) {
    return (1)
  }
  return (0)
}

# Use the classifier with our whole population, without predicting anything
errors <- 0
for (i in 1:n1) {
  errors <- errors + classify(X_1, X_2, X_1[i,], 1)
}
for (i in 1:n2) {
  errors <- errors + classify(X_1, X_2, X_2[i,], 2)
}
AER <- errors / (n1+n2)
cat( fractions(AER), '=', errors, '/', (n1+n2))
```

```
## 0.1020408 = 10 / 98
```

Compute the expected actual error rate (EAER):

```r
# Use the classifier with our whole population, without predicting anything
errors <- 0
for (i in 1:n1) {
  errors <- errors + classify(X_1[-i,], X_2, X_1[i,], 1)
}
for (i in 1:n2) {
  errors <- errors + classify(X_1, X_2[-i,], X_2[i,], 2)
}
EAER <- errors / (n1+n2)
cat( fractions(EAER), '=', errors, '/', (n1+n2))
```

```
## 0.1326531 = 13 / 98
```