

BIOSTAT 274 Spring 2021 Homework 3

Due 11:59 PM 5/23/2020

Zian ZHUANG

Theoretical Part

Q1.

a.

Answer:

Firstly, we can calculate the average value of each column,

$$\begin{aligned} col_1 &= \frac{1 - 1 + 2 + 2}{4} = 1 \\ col_2 &= \frac{2 + 2 + 1 - 1}{4} = 1 \end{aligned}$$

Then we subtract each column with average and get,

$$X = \begin{pmatrix} 0 & 1 \\ -2 & 1 \\ 1 & -2 \\ 1 & 0 \end{pmatrix}$$

b.

Answer:

To calculate the PCs (v_1, v_2) , we need to maximize the $v^T X^T X v$ with the constraint $v^T v = 1$, i.e. we need to maximize $v^T X^T X v - \lambda(v^T v - 1)$. Apply the Lagrangian method, we have,

$$\begin{aligned} \frac{\partial}{\partial v} &= (X^T X - \lambda I)v = 0 \implies X^T X v = \lambda v \\ \frac{\partial}{\partial \lambda} &= v^T v - 1 = 0 \end{aligned}$$

plug in the centered matrix X_0 , we have

$$\begin{aligned} X^T X v &= \lambda v \\ \begin{pmatrix} 6 & -4 \\ -4 & 6 \end{pmatrix} v &= \lambda v \end{aligned}$$

Then set $v = (a, b)$, we know,

$$\begin{cases} 6a - 4b = \lambda a \\ -4a + 6b = \lambda b \end{cases} \\ \implies (6 - \lambda) \frac{(6 - \lambda)a}{4} = 4a$$

Then we can solve for λ ,

$$\begin{aligned} (6 - \lambda) \frac{(6 - \lambda)a}{4} &= 4a \\ (6 - \lambda)^2 &= 16 \\ 36 - 12\lambda + \lambda^2 &= 16 \\ \lambda^2 - 12\lambda + 20 &= 0 \\ (\lambda - 2)(\lambda - 10) &= 0 \\ \lambda &= 2, 10 \end{aligned}$$

when $\lambda = 10$, given that $v^T v - 1 = 0$, we can obtain the corresponding v_{1*} ,

$$\begin{cases} 6a - 4b = 10a \\ -4a + 6b = 10b \\ a^2 + b^2 = 1 \end{cases} \implies \begin{cases} a = -\sqrt{0.5} \\ b = \sqrt{0.5} \end{cases} \text{ or } \begin{cases} a = \sqrt{0.5} \\ b = -\sqrt{0.5} \end{cases} \implies v = \begin{pmatrix} \sqrt{0.5} \\ -\sqrt{0.5} \end{pmatrix} \text{ or } \begin{pmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}$$

Since $\begin{pmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}$ and $\begin{pmatrix} \sqrt{0.5} \\ -\sqrt{0.5} \end{pmatrix}$ are same vector with opposite direction, Without loss of generality, we only keep $\begin{pmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}$.

when $\lambda = 2$, given that $v^T v - 1 = 0$, we can obtain the corresponding v_{2*} ,

$$\begin{cases} 6a - 4b = 2a \\ -4a + 6b = 2b \\ a^2 + b^2 = 1 \end{cases} \implies \begin{cases} a = \sqrt{0.5} \\ b = \sqrt{0.5} \end{cases} \text{ or } \begin{cases} a = -\sqrt{0.5} \\ b = -\sqrt{0.5} \end{cases} \implies v = \begin{pmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix} \text{ or } \begin{pmatrix} -\sqrt{0.5} \\ -\sqrt{0.5} \end{pmatrix}$$

Since $\begin{pmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}$ and $\begin{pmatrix} -\sqrt{0.5} \\ -\sqrt{0.5} \end{pmatrix}$ are same vector with opposite direction, Without loss of generality, we only keep $\begin{pmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}$.

plug v_{1*}, v_{2*} in the original equation $v^T X^T X v$, we found that, v_{1*} generate a larger value 10. Thus we know that X has 2 PCs and the first and second PC loading is $\begin{pmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}$ and $\begin{pmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}$, respectively.

We can have 2 PCs by calculating Xv ,

```
df <- matrix(c(0,-2,1,1,1,1,-2,0),ncol=2)
df %*% matrix(c(-1,1,1,1)*sqrt(0.5),ncol=2)
```

```
##           [,1]      [,2]
## [1,]  0.7071068  0.7071068
## [2,]  2.1213203 -0.7071068
## [3,] -2.1213203 -0.7071068
## [4,] -0.7071068  0.7071068
```

c.

We can calculate the eigenvalues and eigenvectors directly. Define eigenvalue λ for $X^T X$, we have,

$$\begin{aligned}
& |X^T X - \lambda I| = 0 \\
\Rightarrow & \left| \begin{pmatrix} 6-\lambda & -4 \\ -4 & 6-\lambda \end{pmatrix} \right| = 0 \\
& (6-\lambda)^2 - 16 = 0 \\
& \lambda^2 - 12\lambda + 20 = 0 \\
& (\lambda - 2)(\lambda - 10) = 0 \\
& \lambda = 2, 10
\end{aligned}$$

Since $10 > 2$, we know that $\lambda_1 = 10, \lambda_2 = 2$. Then we can calculate the corresponding eigenvectors v_1, v_2 , given the constraint $v^T v = 1$

$$\begin{aligned}
& \begin{pmatrix} 6-10 & -4 \\ -4 & 6-10 \end{pmatrix} v_1 = 0 \\
& \begin{pmatrix} -4 & -4 \\ -4 & -4 \end{pmatrix} v_1 = 0 \Rightarrow v_1 = \begin{pmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix} \text{ or } \begin{pmatrix} \sqrt{0.5} \\ -\sqrt{0.5} \end{pmatrix} \\
& \begin{pmatrix} 6-2 & -4 \\ -4 & 6-2 \end{pmatrix} v_2 = 0 \\
& \begin{pmatrix} 4 & -4 \\ -4 & 4 \end{pmatrix} v_2 = 0 \Rightarrow v_2 = \begin{pmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix} \text{ or } \begin{pmatrix} -\sqrt{0.5} \\ -\sqrt{0.5} \end{pmatrix}
\end{aligned}$$

Without loss of generality, we only keep $v_1 = \begin{pmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}$ and $v_2 = \begin{pmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}$ as two PC loading. This results is consistent with (b).

Then we can see,

$$v_1^T v_2 = \begin{pmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix}^T \begin{pmatrix} \sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix} = -0.5 + 0.5 = 0$$

which means that eigenvectors (corresponding to two different eigenvalue) are orthogonal.

Then we can calculate,

$$\begin{aligned}
d_1 &= \sqrt{\lambda_1} = \sqrt{10} \\
u_1 &= \frac{1}{d_1} X v_1 = \frac{\sqrt{20}}{20} \begin{pmatrix} 1 \\ 3 \\ -3 \\ -1 \end{pmatrix} \\
d_2 &= \sqrt{\lambda_2} = \sqrt{2} \\
u_2 &= \frac{1}{d_2} X v_2 = \frac{1}{2} \begin{pmatrix} 1 \\ -1 \\ -1 \\ 1 \end{pmatrix}
\end{aligned}$$

Then we can get PCs,

$$PC_1 = u_1 d_1 = \frac{\sqrt{2}}{2} \begin{pmatrix} 1 \\ 3 \\ -3 \\ -1 \end{pmatrix}$$

$$PC_2 = u_2 d_2 = \frac{\sqrt{2}}{2} \begin{pmatrix} 1 \\ -1 \\ -1 \\ 1 \end{pmatrix}$$

Finally we can get the SVD for X ,

$$X = UDV^T = \begin{pmatrix} \frac{\sqrt{20}}{20} & \frac{1}{2} \\ \frac{3\sqrt{20}}{20} & -\frac{1}{2} \\ -\frac{3\sqrt{20}}{20} & -\frac{1}{2} \\ -\frac{\sqrt{20}}{20} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} \sqrt{10} & 0 \\ 0 & \sqrt{2} \end{pmatrix} \begin{pmatrix} -\sqrt{0.5} & \sqrt{0.5} \\ \sqrt{0.5} & \sqrt{0.5} \end{pmatrix}$$

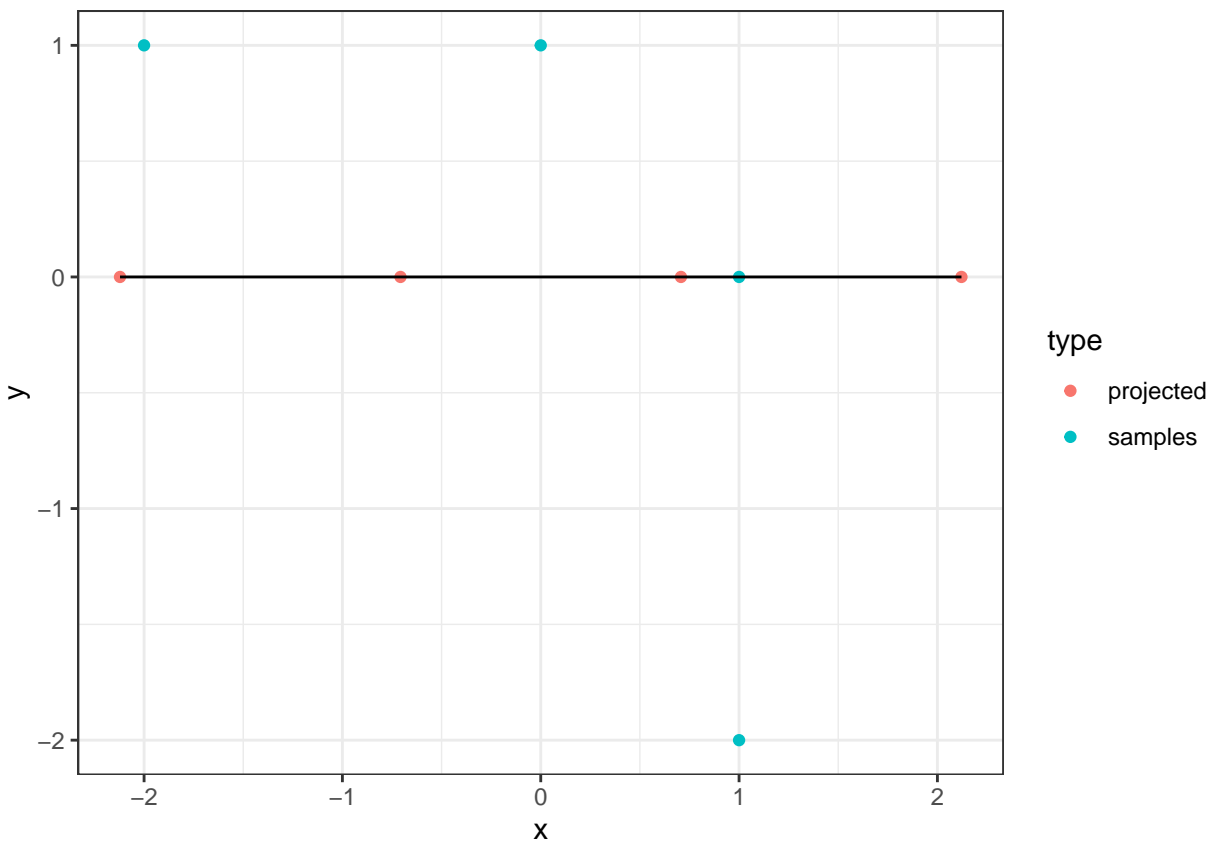
d.

To find the best 1-dimensional subspace (line) for approximating the sample points, we need to make $X\alpha$ has the largest variance. We will choose the first column of V as α . Then we get,

$$X\alpha = \begin{pmatrix} 0 & 1 \\ -2 & 1 \\ 1 & -2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} -\sqrt{0.5} \\ \sqrt{0.5} \end{pmatrix} = \sqrt{0.5} \begin{pmatrix} 1 \\ 3 \\ -3 \\ -1 \end{pmatrix}$$

Plot the samples and draw the projected line.

```
library(ggplot2)
df <- rbind(data.frame(x=c(0,-2,1,1),y=c(1,1,-2,0),type="samples"),
            data.frame(x=c(1,3,-3,-1)*sqrt(0.5),y=c(0,0,0,0),type="projected"))
ggplot()+
  geom_point(data=df, aes(x=x,y=y,color=type))+
  geom_line(data = df%>%filter(type=="projected"), aes(x=x,y=y))+theme_bw()
```



e.

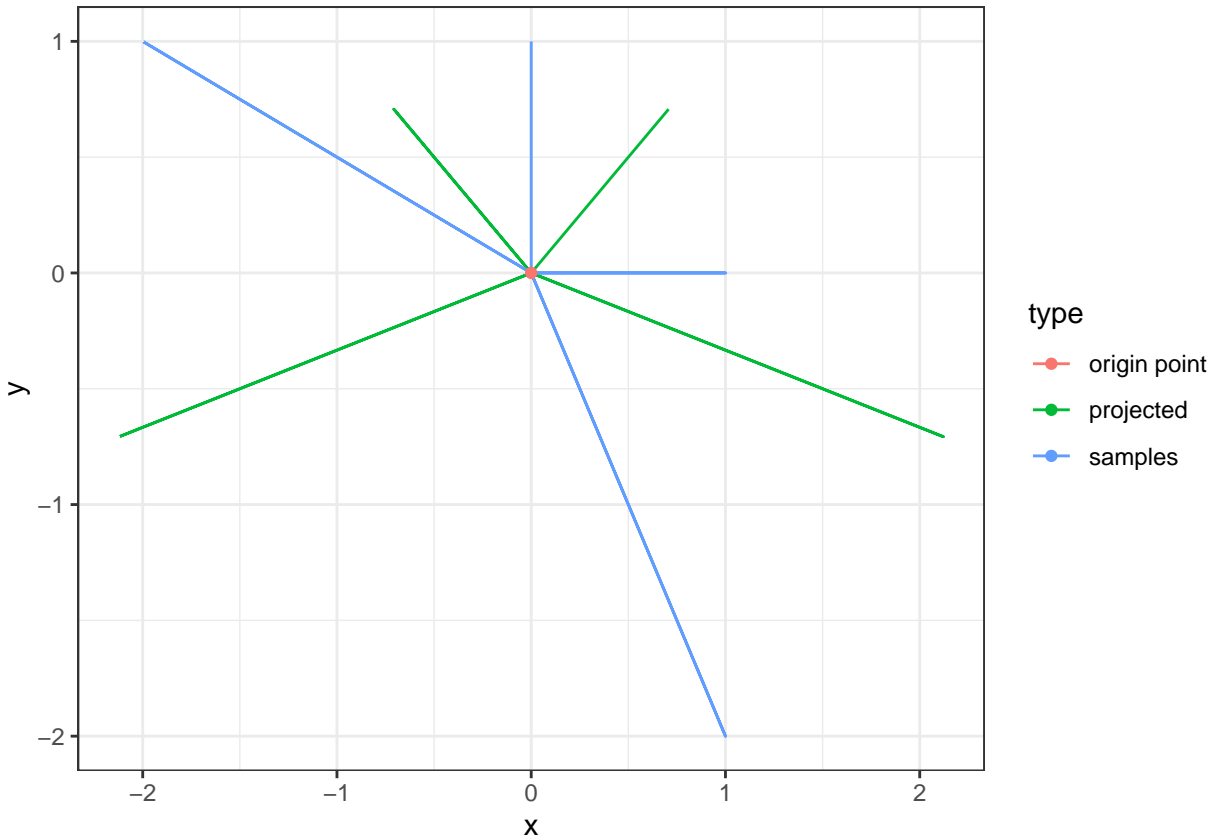
After projection we have,

$$X\alpha = \begin{pmatrix} 0 & 1 \\ -2 & 1 \\ 1 & -2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} -\sqrt{0.5} & \sqrt{0.5} \\ \sqrt{0.5} & \sqrt{0.5} \end{pmatrix} = \sqrt{0.5} \begin{pmatrix} 1 & 1 \\ 3 & -1 \\ -3 & -1 \\ -1 & 1 \end{pmatrix}$$

Create the PC scatter plots and argue that the PCs are the rotation of original data.

```
df <- rbind(data.frame(x=c(0,0,-2,0,1,0,1,0),
                        y=c(1,0,1,0,-2,0,0,0),
                        type="samples"),
            data.frame(x=c(1,0,3,0,-3,0,-1,0)*sqrt(0.5),
                        y=c(1,0,-1,0,-1,0,1,0)*sqrt(0.5),
                        type="projected"))

ggplot()+
  geom_path(data=df, aes(x=x,y=y,color=type))+
  geom_point(aes(x=0,y=0,color="origin point"))+
  theme_bw()
```



After connecting all sample points and projected points with the original point, it is easy to see that PCs are the rotation of original data.

f.

Since we know that the matrix after projection has independent columns, we know the rank-1 approximation is the matrix with one column. According to the (e) we know that,

$$\hat{X} = \sqrt{0.5} \begin{pmatrix} 1 \\ 3 \\ -3 \\ -1 \end{pmatrix}$$

```
df <- data.frame(dim1=c(1,3,-3,-1)*sqrt(0.5),dim2=c(1,-1,-1,1)*sqrt(0.5))
(vars <- apply(df, 2, var))
```

```
##      dim1      dim2
## 3.333333 0.666667
```

```
vars/sum(vars)
```

```
##      dim1      dim2
## 0.833333 0.166667
```

It accounts for 83.3% variance among the original data

g.

```
df_ns <- data.frame(dim1=c(1,-1,2,2),dim2=c(20,20,-10,10)) %>% scale(scale = FALSE)
df_s <- data.frame(dim1=c(1,-1,2,2),dim2=c(20,20,-10,10)) %>% scale(scale = TRUE)
```

Calculate the first PC of X'_0 ,

```
# un-scaled
svd(df_ns)$u[,1]*svd(df_ns)$d[1]
```

```
## [1] -9.97760536 -10.11137999 20.02209803 0.06688731
```

```
# scaled
svd(df_s)$u[,1]*svd(df_ns)$d[1]
```

```
## [1] 5.489451 16.468354 -16.468354 -5.489451
```

Calculate the first PC loading of X'_0 ,

```
# un-scaled
svd(df_ns)$v[,1]
```

```
## [1] 0.06688731 -0.99776054
```

```
# scaled
svd(df_s)$v[,1]
```

```
## [1] -0.7071068 0.7071068
```

As we can tell from the results, after scaling, the first PC and PC loading seems more balanced.

Test the effect of scale,

```
# un-scaled data
res.pca <- prcomp(df_ns)
vars <- apply(res.pca$x, 2, var)
# first PC accounts for 99.5% variance
vars/sum(vars)
```

```
##          PC1          PC2
## 0.994523923 0.005476077
```

```
# scaled data
res.pca <- prcomp(df_s)
vars <- apply(res.pca$x, 2, var)
# first PC accounts for 83.3% variance
vars/sum(vars)
```

```
##          PC1          PC2
## 0.83333333 0.1666667
```

When performing Principal Component Analysis(PCA), scaling is important. PCA tries to get the features with maximum variance and the variance is high for high magnitude features. If we do not scale the data, PCA will skew towards high magnitude features.