GR5223_HW1

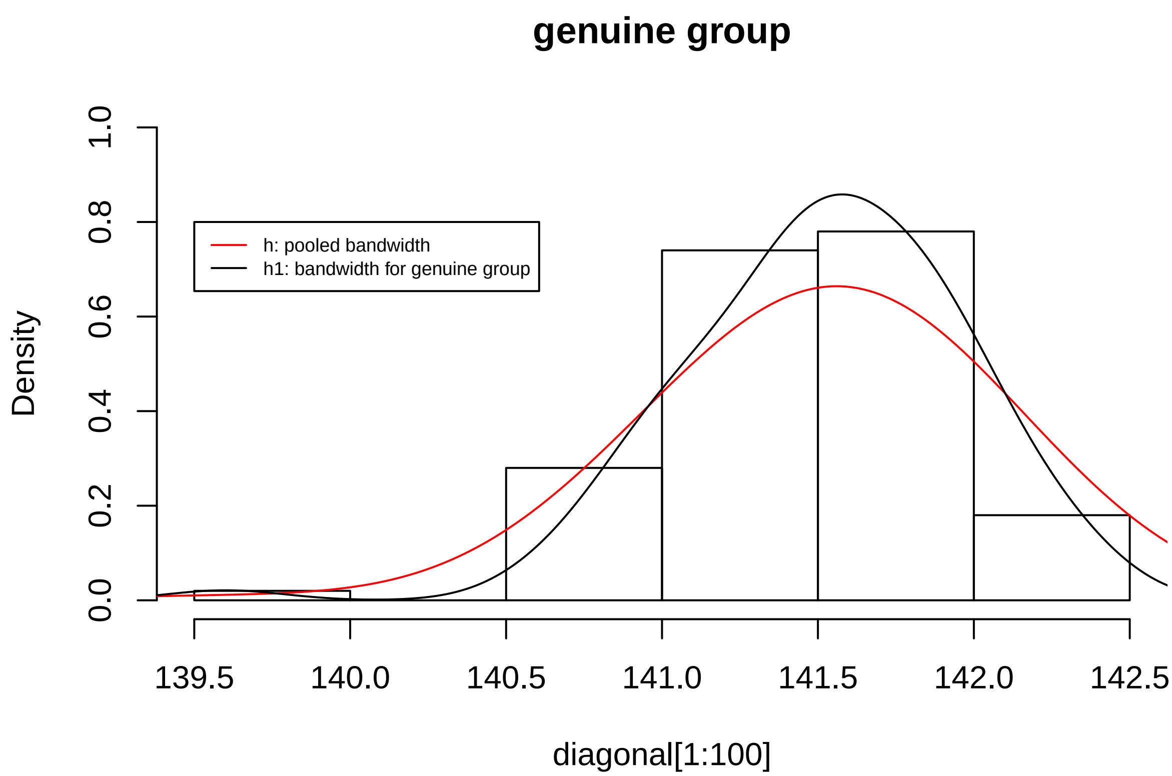
NAME: Yuhao Wang, UNI: yw3204

2/24/2019

Cha 1

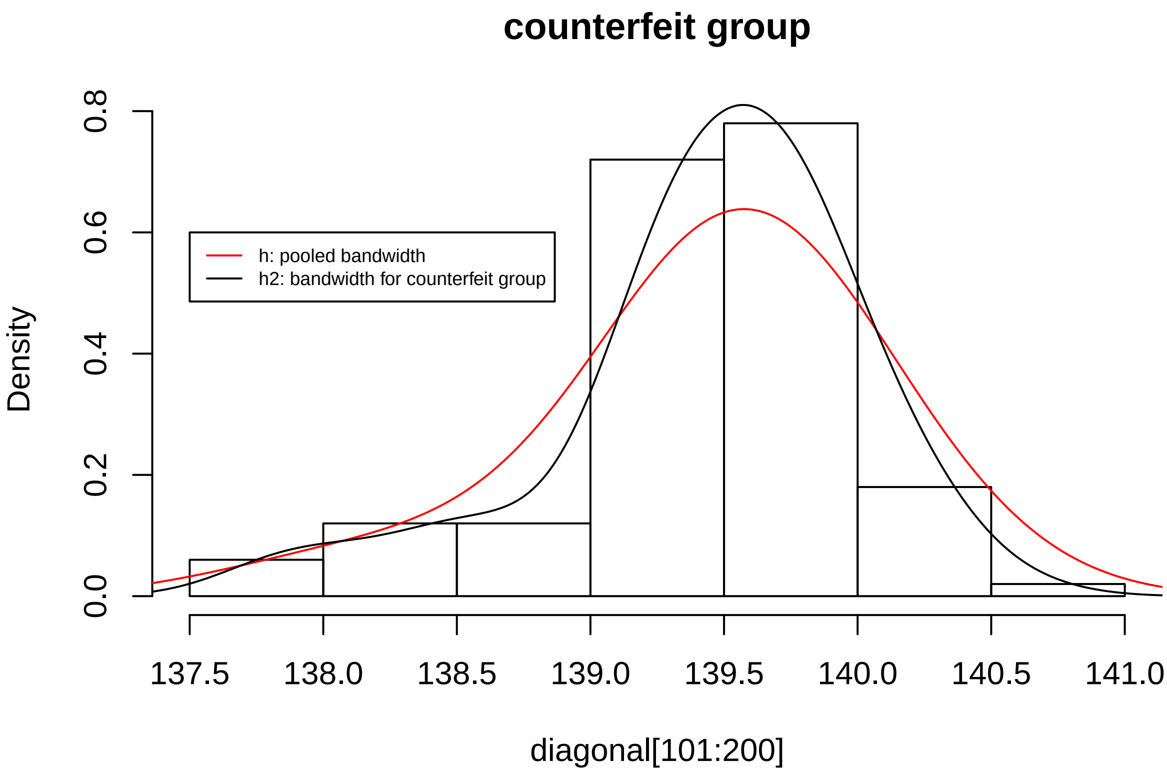
1.9

```
load("/Users/apple/Desktop/semester_2/2.Multi_Stat_Infe/data/bank2.rda")
diagonal <- bank2$Diagonal
h <- 1.06*sd(diagonal)*length(diagonal)^(-1/5)

# density for the genuine group
h1 <- 1.06*sd(diagonal[1:100])*length(diagonal[1:100])^(-1/5)
hist(diagonal[1:100], probability = T, ylim = c(0, 1), main = "genuine group")
lines(density(diagonal[1:100], bw = h, kernel = "gaussian"), col = "red")
lines(density(diagonal[1:100], bw = h1, kernel = "gaussian"))
legend(139.5, 0.8, legend=c("h: pooled bandwidth", "h1: bandwidth for genuine group"),
       col=c("red", "black"), lty = 1, cex = 0.6)
```




```
# density for the counterfeit group
h2 <- 1.06*sd(diagonal[101:200])*length(diagonal[101:200])^(-1/5)
hist(diagonal[101:200], probability = T, main = "counterfeit group")
lines(density(diagonal[101:200], bw = h, kernel = "gaussian"), col = "red")
lines(density(diagonal[101:200], bw = h2, kernel = "gaussian"))
```

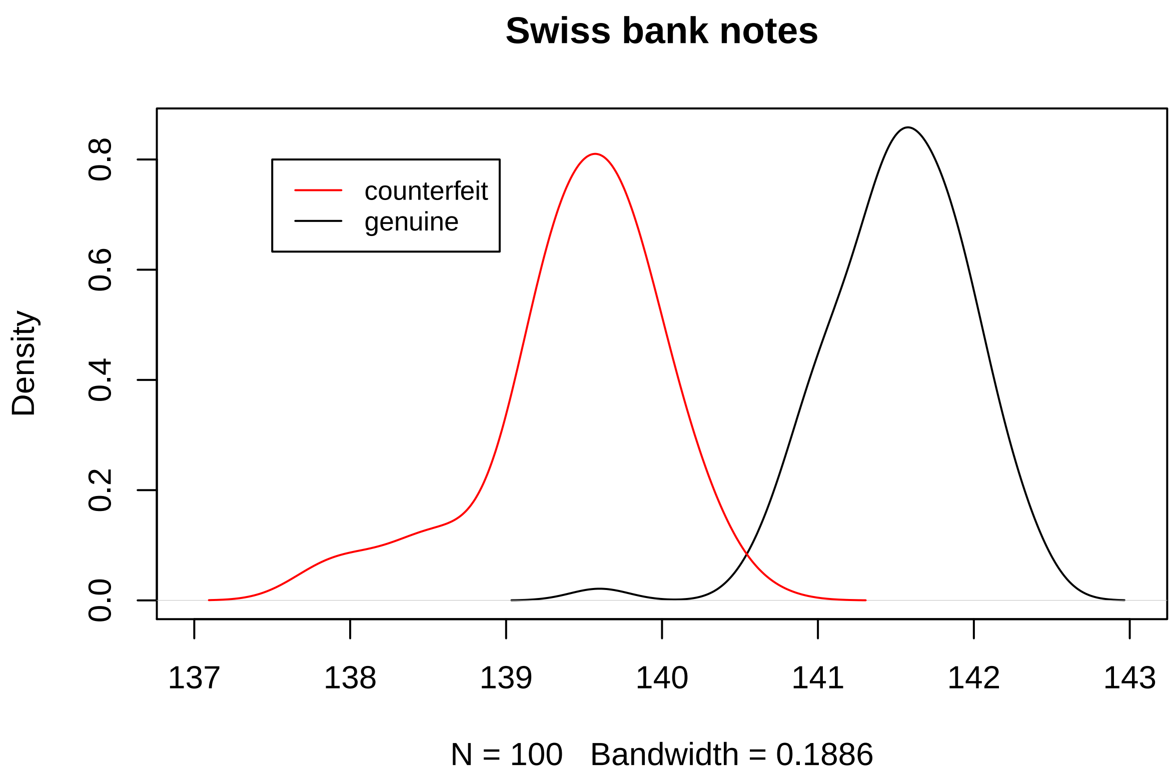
```r
legend(137.5, 0.6, legend=c("h: pooled bandwidth", "h2: bandwidth for counterfeit group"),
       col=c("red", "black"), lty = 1, cex = 0.6)
```



We are using the Gaussian kernel here for estimating density and thus the thumb of rule for choosing the bandwidth is $h = 1.06 * \hat{\sigma} * n^{-1/5}$. By observing the plots above, it is better to have different bandwidth for the two group.

1.10

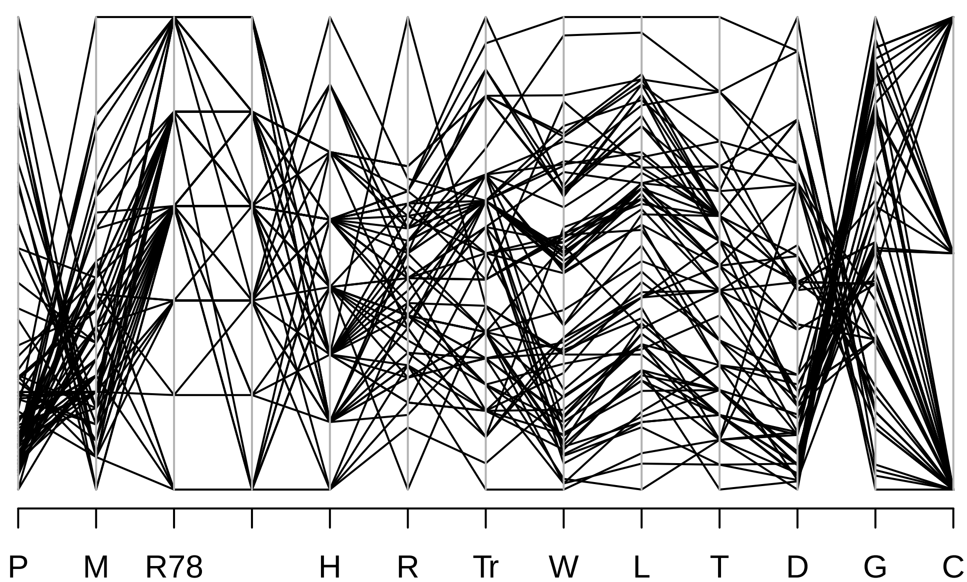
```r
plot(density(diagonal[1:100], bw = h1, kernel = "gaussian"), xlim=c(137, 143),
     main = "Swiss bank notes")
lines(density(diagonal[101:200], bw = h2, kernel = "gaussian"), col = "red")
legend(137.5, 0.8, legend=c("counterfeit", "genuine"),
       col=c("red", "black"), lty = 1, cex = 0.8)
```



It is not effective to seperate the two group simply based on the diagonal variable.

1.11

```
library("MASS")
load("/Users/apple/Desktop/semester_2/2.Multi_Stat_Infe/data/carc.rda")
car_dat <- sapply(carc[,1:13], as.numeric)
parcoord(car_dat)
```



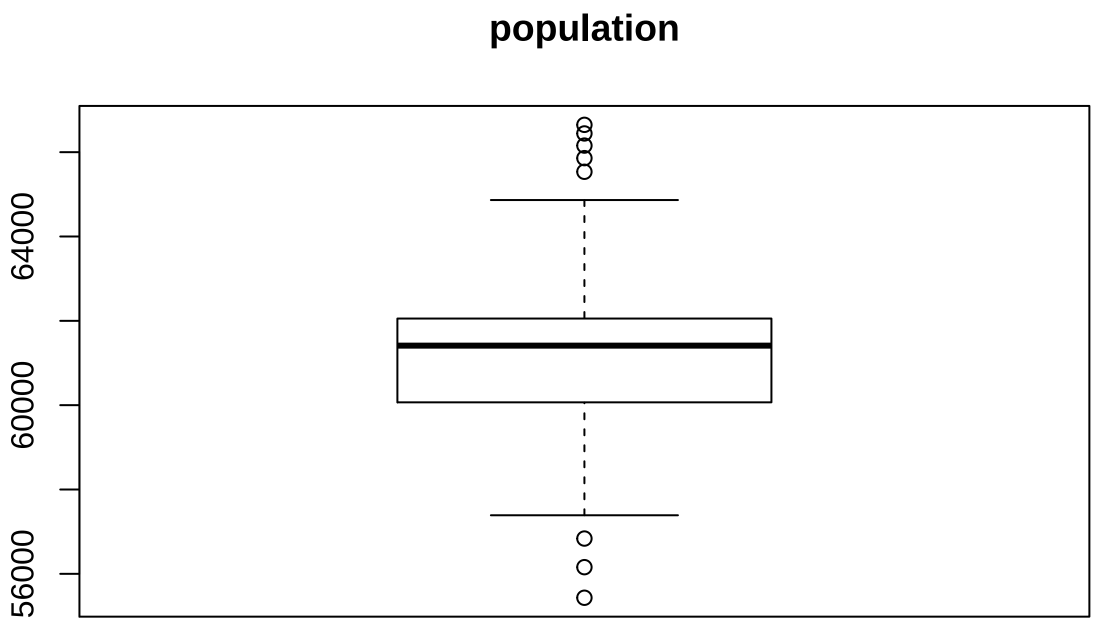
Observing the PCP above, we may find there is a negetive relationship between variable 12 and 13 and also a slight positive relationship between variable 9 and 10. One shortcoming of PCPs is: we cannot distinguish observations when two lines cross at one point. Another shortcoming is it only considers a subset of pairs when comparing variables mutually.

1.12

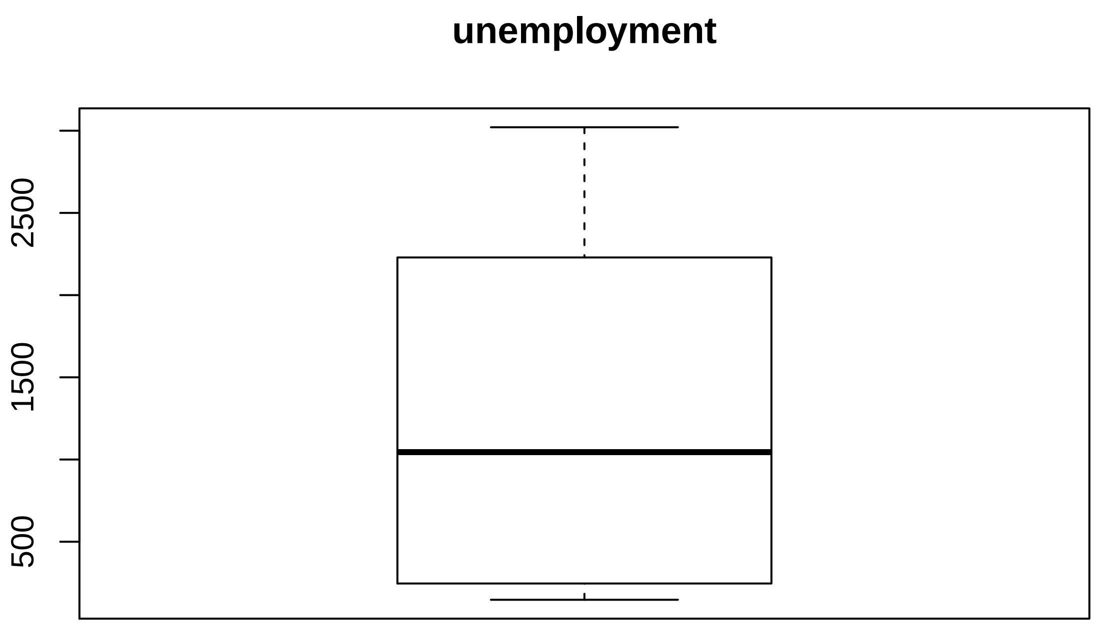
If there are only a few points equally located at the vertical line, it is probable that the variable is a discrete variable. Therefore, in question 1.11, the possible discrete variables are R78, R77, H and C.

1.17

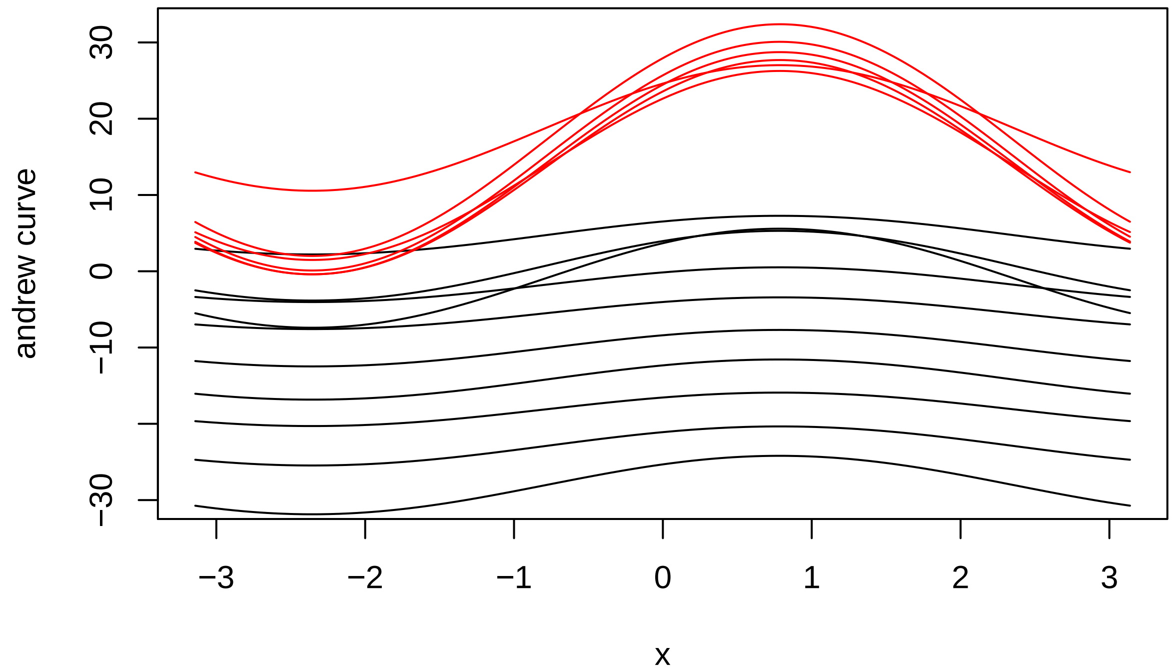
```
load("/Users/apple/Desktop/semester_2/2.Multi_Stat_Infe/data/annualpopu.rda")
# boxplot
boxplot(annualpopu$Inhabitants, main = "population")
```



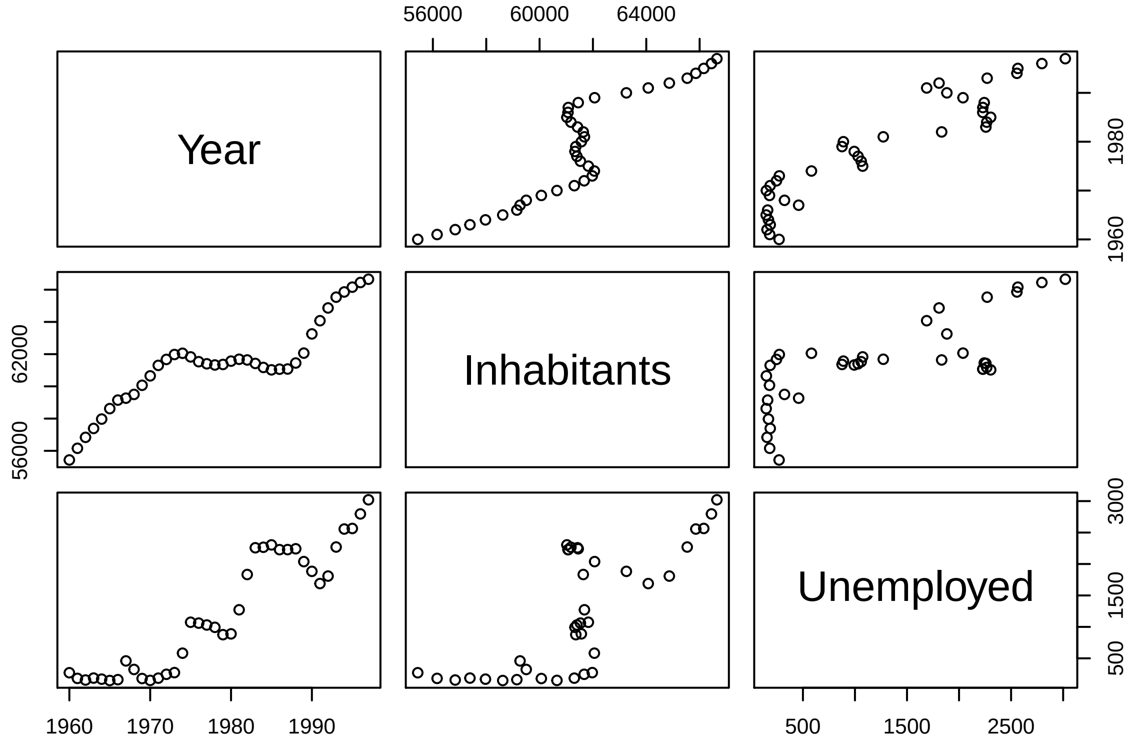

```
boxplot(annualpopu$Unemployed, main = "unemployment")
```



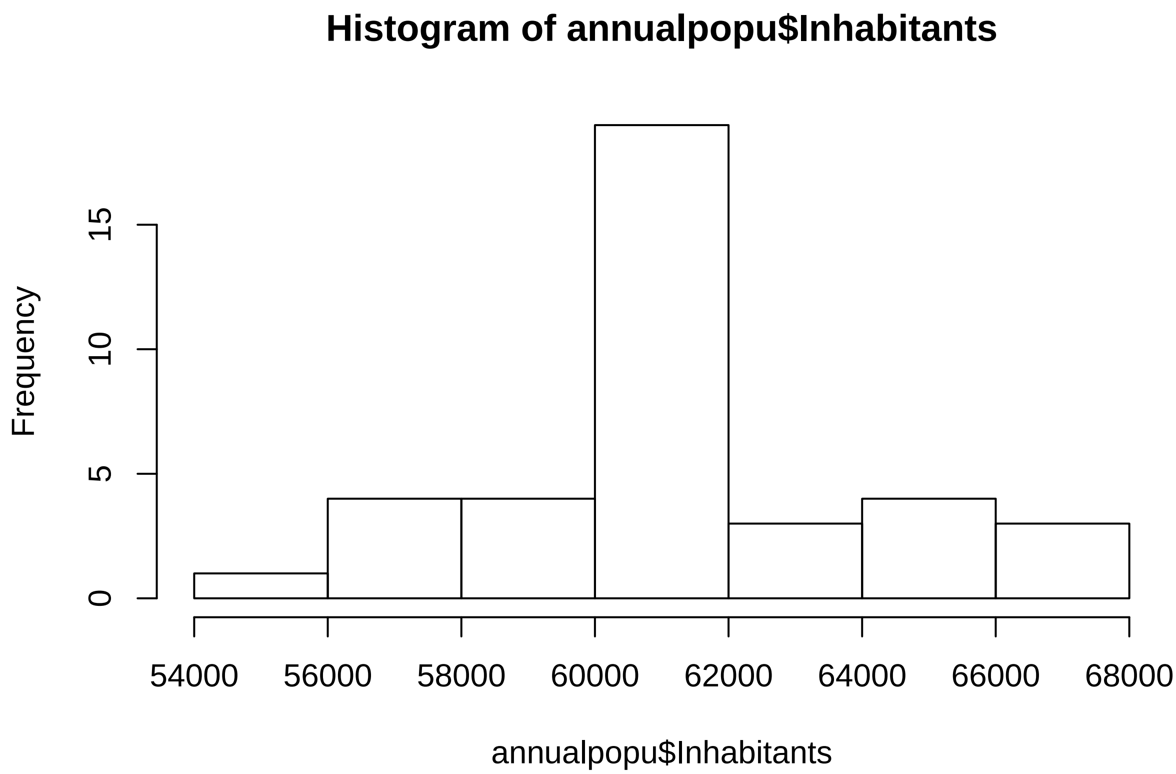

```r
# Andrew's curve
andcur <- function(x, t) {
  res <- c()
  for(i in t) {
    res <- c(res, x[2]/sqrt(2) + x[3]*sin(i) + x[3]*cos(i))
  }
  res <- unlist(res)
  res <- (res-42000)/100
  return(res)
}

obs <- annualpopu[1:20, ]
t_range <- seq(-pi, pi, 0.01)

plot(t_range, andcur(obs[1, ], t_range), type = "l", ylab = "andrew curve",
     ylim = c(-30, 32), xlab = "x")

for(i in 2:10) {
  lines(t_range, andcur(obs[i, ], t_range))
}

for(i in 15:20) {
  lines(t_range, andcur(obs[i, ], t_range), col = "red")
}
```




```r
# scatter plot
pairs(annualpopu)
```




```r
# histogram
hist(annualpopu$Inhabitants)
```



The five-number summaries are 55433, 60213, 61412, 62034 and 66648. The boxplot tells us that the median is around 62000 and there are several outliers. Andrew's curve tells us that we may cluster the red curves as a group and the black as another. The scatter plots shows that there's a increasing trend for both variable 2 and 3. The histogram gives us rough information about the distribution which centers around 61000.

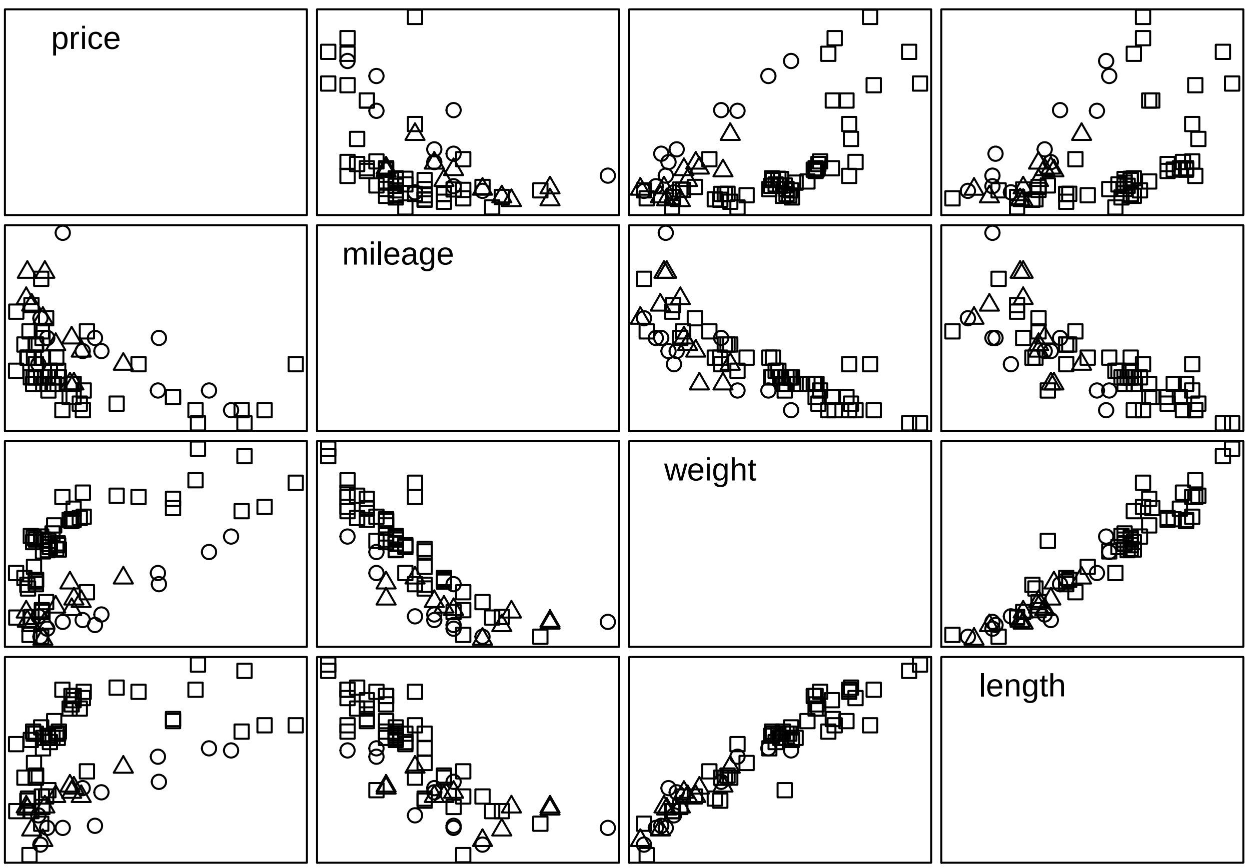
For the advantages and disadvantages, boxplot is simple and clear but gives little information related to time variation. Andrew's curve is complicated but may offer information about latent clusterings. Scatter plot is direct and will tell us the relationship between two variables. Last, the histgram tells us the distribution about the variable and like boxplot, gives little information related to time series.

1.18

```
# the folowing code is adopted from "https://github.com/QuantLet/SMS2/tree/master/SMSdrafcar"
x = cbind(carc[,1], carc[,2], carc[,8], carc[,9])
y = c("price", "mileage", "weight", "length")
p = dim(x)[2]

par(mfrow=c(p,p), mar = 0.2 + c(0,0,0,0))        # creates display pxp with margins=0.2

for (k in 0:15) {
  i = (k %/% 4) + 1                              # div, ith raw
  j = (k %% 4) + 1                               # mod, jth column
  if (i>j) {
    plot(x[,i]~x[,j], xlab = "", ylab = "", axes=FALSE, frame.plot=TRUE,
         pch=as.numeric(carc$C)-1-(carc$C=="Europe")+(carc$C=="Japan"), cex=1.5)
  }
  if (i<j) {
    plot(x[,i]~x[,j], xlab = "", ylab = "", axes=FALSE, frame.plot=TRUE,
```

```
        pch=as.numeric(carc$C)-1-(carc$C=="Europe")+(carc$C=="Japan"), cex=1.5)
  }
  if (i == j) {
    plot(0~0,xlab = "", ylab = "", axes=FALSE, xlim=c(1,5), ylim=c(1,5), frame.plot=TRUE)
    text(2,4.5, y[i], cex=1.5)                          # print text on diagonal graphs
  }
}
```

```
## Warning in plot.formula(0 ~ 0, xlab = "", ylab = "", axes = FALSE, xlim =
## c(1, : the formula '0 ~ 0' is treated as '0 ~ 1'

## Warning in plot.formula(0 ~ 0, xlab = "", ylab = "", axes = FALSE, xlim =
## c(1, : the formula '0 ~ 0' is treated as '0 ~ 1'

## Warning in plot.formula(0 ~ 0, xlab = "", ylab = "", axes = FALSE, xlim =
## c(1, : the formula '0 ~ 0' is treated as '0 ~ 1'

## Warning in plot.formula(0 ~ 0, xlab = "", ylab = "", axes = FALSE, xlim =
## c(1, : the formula '0 ~ 0' is treated as '0 ~ 1'
```



In the plot, the square marks U.S. car, the triangles mark Japanese car and the circles mark European car. In the region of heavy cars, the price is relatively higher, the mileage is relatively lower and the length is relatively longer. Most of them are U>S> cars. In the region of high fule economy, the price is relatively lower, the weight is relatively lower and the length is relatively shorter.

Cha 2

2.2

No. Because if we plug in 0 to the characteristic function of A, we find 0 is always a legitimate eigenvalue. Thus, it is impossible that all eigenvalue are positive.

2.3

Denote all eigenvalues by $\lambda_1, \ldots, \lambda_n$. According to formula, we have $|A| = \Pi_{i=1}^n \lambda_i$ and since $\lambda_i \neq 0$ for all i, then $|A| \neq 0$. Thus, matrix A is not a sigular matrix and its inverse exists.

2.4

```r
A <- matrix(c(1, 2, 3, 2, 1, 2, 3, 2, 1), 3, 3)
jd <- eigen(A)
lda <- diag(jd$values)
gma <- jd$vectors
# check the Jordan decomposition theorem
gma %*% lda %*% t(gma)
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    3
## [2,]    2    1    2
## [3,]    3    2    1
```

```r
# check orthogonal
gma %*% t(gma)
```

```
##               [,1]          [,2]         [,3]
## [1,]  1.000000e+00 -1.704785e-16 1.110223e-16
## [2,] -1.704785e-16  1.000000e+00 5.945621e-17
## [3,]  1.110223e-16  5.945621e-17 1.000000e+00
```

```r
# check determinant
prod(jd$values)
```

```
## [1] 8
```

```r
det(A)
```

```
## [1] 8
```

```r
# check trace
sum(jd$values)
```

```
## [1] 3
```

```r
sum(diag(A))
```

```
## [1] 3
```

```r
# compute inverse
gma %*% solve(lda) %*% t(gma)
```

```
##        [,1] [,2]  [,3]
## [1,] -0.375  0.5  0.125
```

```
## [2,]  0.500 -1.0  0.500
## [3,]  0.125  0.5 -0.375
```

```r
solve(A)
```

```
##        [,1] [,2]   [,3]
## [1,] -0.375  0.5  0.125
## [2,]  0.500 -1.0  0.500
## [3,]  0.125  0.5 -0.375
```

```r
# compute A^2
gma %*% lda**2 %*% t(gma)
```

```
##      [,1] [,2] [,3]
## [1,]   14   10   10
## [2,]   10    9   10
## [3,]   10   10   14
```

```r
A %*% A
```

```
##      [,1] [,2] [,3]
## [1,]   14   10   10
## [2,]   10    9   10
## [3,]   10   10   14
```

Hence, the Jordan decomposition is:

$$A = \Gamma\Lambda\Gamma$$

where,

$$\Gamma = \begin{pmatrix} -0.61 & 0.36 & 0.71 \\ -0.52 & -0.86 & 0 \\ -0.61 & 0.36 & -0.71 \end{pmatrix}$$

$$\Lambda = \begin{pmatrix} 5.7 & 0 & 0 \\ 0 & -0.7 & 0 \\ 0 & 0 & -2.0 \end{pmatrix}$$

2.5

Let $a = (a_1, \ldots, a_p)^T$ and $x = (x_1, \ldots, x_p)^T$.

Then, we have:

$a^T x = x^T a = \Sigma_{i=1}^p a_i x_i$ and $\frac{\partial a^T x}{\partial x_i} = \frac{\partial a^T x}{\partial x_i} = a_i$ for $i = 1, \ldots, p$.

Therefore, $\frac{\partial a^T x}{\partial x} = a$.

Let $A = (a_{ij})_{i,j=1}^p$, where $a_{ij} = a_{ji}$.

Then, $x^T A x = \Sigma_{i,j=1}^p a_{ij} x_i x_j$ and

$\frac{\partial x^T A x}{\partial x_i} = \Sigma_{j=1}^n a_{ij} x_j + \Sigma_{j=1}^n a_{ji} x_j = 2\Sigma_{j=1}^n a_{ji} x_j = 2A_i^T x$, where $A_i = (a_{i1}, \ldots, a_{ip})^T$ is the i^{th} row.

Thus, $\frac{\partial x^T A x}{\partial x} = 2Ax$.

Keep taking the second derivative, we have $\frac{\partial^2 x^T A x}{\partial x_i \partial x_j} = a_{ij} + a_{ji} = 2a_{ij}$.

Thus, $\frac{\partial^2 x^T A x}{\partial x \partial x^T} = 2A$.

Cha 3

3.1

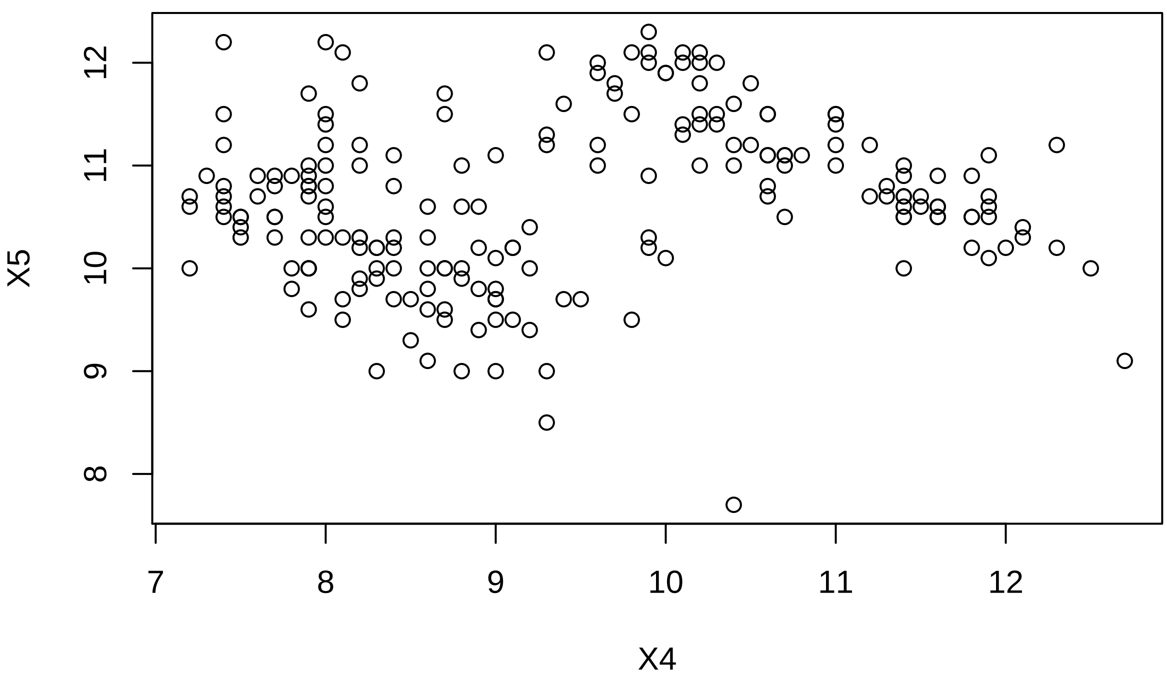
```
cov(bank2$`Inner Frame Lower`, bank2$`Inner Frame Upper`)
```

```
## [1] 0.1645389
```

The covariance $s_{X_4X_5}$ is about 0.16 and thus positive. The reason is due to Simpson's paradox.

3.2

```
plot(bank2$`Inner Frame Lower`, bank2$`Inner Frame Upper`, xlab = "X4", ylab = "X5")
```




```
cov(bank2[1:100, ]$`Inner Frame Lower`, bank2[1:100, ]$`Inner Frame Upper`)
```

```
## [1] -0.2634747
```

```
cov(bank2[101:200, ]$`Inner Frame Lower`, bank2[101:200, ]$`Inner Frame Upper`)
```

```
## [1] -0.4901919
```

By observing the plot, we will expect the covaraince for the subgroups to be negative. And by calculation, the covariance for the genuine bank notes is -0.26 and -0.49 for the counterfeit bank notes.

3.4

```
cov(carc[, 2], carc[, 8])
```

```
## [1] -3732.025
```

Intuitively, we will expect a negative sign. Because the heavier the car, the fewer miles per gallon it could run. Covaraince is not sufficient for judging a linear relationship while correlation is.

3.5

```
load("/Users/apple/Desktop/semester_2/2.Multi_Stat_Infe/data/pullover.rda")
n_pullover <- nrow(pullover)
cor(pullover)
```

```
##                    Sales      Price Advertisement      Hours
## Sales          1.0000000 -0.1675760     0.8672280  0.6328673
## Price         -0.1675760  1.0000000     0.1212619 -0.4637879
## Advertisement  0.8672280  0.1212619     1.0000000  0.3082688
## Hours          0.6328673 -0.4637879     0.3082688  1.0000000
```

```
# Fisher's Z-transformation
W <- 1/2 * log((1+cor(pullover)[1, 2]) / (1-cor(pullover)[1, 2]))
mu <- 0
var <- 1/(n_pullover-3)
z <- (W-mu)/sqrt(var)
z
```

```
## [1] -0.4475859
```

The sign is negative. And according to the test above, we accept the null since $-1.96 < -0.45 < 1.96$ under significance level 5%.

3.8

```
pullover_lm <- lm(Sales ~ Price, pullover)
predict(pullover_lm, data.frame(Price = 105))
```

```
##        1
## 172.5544
```

We regress sales on price, which gives us the estimated regression line: $y = -0.364x + 210.774$. Plug in $x = 105$, we get the predicted sales is around 173.

3.10

First, we decompose the total sum of squares:

$$\begin{aligned}
\Sigma_i(y_i - \bar{y})^2 &= \Sigma_i(y_i - \hat{y}_i + \hat{y}_i - \bar{y})^2 \\
&= \Sigma_i(y_i - \hat{y}_i)^2 + \Sigma_i(\hat{y}_i - \bar{y})^2 + 2\Sigma_i(y_i - \hat{y}_i)(\hat{y}_i - \bar{y}) \\
&= \Sigma_i(y_i - \hat{y}_i)^2 + \Sigma_i(\hat{y}_i - \bar{y})^2 + 2\Sigma_i(y_i - \hat{y}_i)\hat{y}_i - 2\bar{y}\Sigma_i(y_i - \hat{y}_i) \\
&= \Sigma_i(y_i - \hat{y}_i)^2 + \Sigma_i(\hat{y}_i - \bar{y})^2
\end{aligned}$$

This is becasue $y_i - \hat{y}_i$ is orthogonal with $\hat{y}_i$ and $\Sigma_i(y_i - \hat{y}_i) = 0$.

Then, we prove R^2 is the square of the correlation between X and Y:

$$\begin{aligned}
\frac{\Sigma_i(\hat{y}_i-\bar{y})^2}{\Sigma_i(y_i-\bar{y})^2} &= \frac{\Sigma_i((\hat{\beta}_1 x_i+\hat{\beta}_0)-(\hat{\beta}_1\bar{x}+\hat{\beta}_0))^2}{\Sigma_i(y_i-\bar{y})^2} \\
&= \frac{\Sigma_i((\hat{\beta}_1 x_i+\hat{\beta}_0)-(\hat{\beta}_1\bar{x}+\hat{\beta}_0))^2}{\Sigma_i(y_i-\bar{y})^2} \\
&= \frac{\Sigma_i(\hat{\beta}_1 x_i-\hat{\beta}_1\bar{x})^2}{\Sigma_i(y_i-\bar{y})^2} \\
&= \frac{\hat{\beta}_1^2\Sigma_i(x_i-\bar{x})^2}{\Sigma_i(y_i-\bar{y})^2} \\
&= \frac{\Sigma_i(x_i-\bar{x})^2}{\Sigma_i(y_i-\bar{y})^2} * (\frac{\Sigma_i(x_i-\bar{x})(y_i-\bar{y})}{\Sigma_i(x_i-\bar{x})^2})^2 \\
&= \frac{(\Sigma_i(x_i-\bar{x})(y_i-\bar{y})^2}{\Sigma_i(y_i-\bar{y})^2\Sigma_i(x_i-\bar{x})^2}
\end{aligned}$$

Thus, by definition, this is exactly the square of the correlation between X and Y.

3.15

```r
# Fisher's Z-transformation
W1 <- 1/2 * log((1+cor(pullover)[1, 4]) / (1-cor(pullover)[1, 4]))
var1 <- 1/(n_pullover-3)
tanh(W1 - 1.96*sqrt(var1))
```

```
## [1] 0.00537432
```

```r
tanh(W1 + 1.96*sqrt(var1))
```

```
## [1] 0.9027703
```

By theorem, we have $|\frac{W-\mathbb{E}(W)}{\sqrt{Var(W)}}| < 1.96$ with probability around 95%, where $\mathbb{E}(W) = tanh^{-1}(\rho_{X_1X_4})$.

Then, we have $W - 1.96\sqrt{Var(W)} < tanh^{-1}(\rho_{X_1X_4}) < W + 1.96\sqrt{Var(W)}$.

Plug in W and $Var(W)$ and solve the inequality, we get the 95% confidence interval for $\rho_{X_1X_4}$ is $(0.005, 0.903)$.

3.16

```r
pullover_yen <- pullover
pullover_yen[, 2] <- pullover_yen[, 2]*106
pullover_yen[, 3] <- pullover_yen[, 3]*106

cov(pullover)
```

```
##                    Sales      Price Advertisement     Hours
## Sales         1152.45556  -88.91111     1589.6667  301.6000
## Price          -88.91111  244.26667      102.3333 -101.7556
## Advertisement 1589.66667  102.33333     2915.5556  233.6667
## Hours          301.60000 -101.75556      233.6667  197.0667
```

```r
cov(pullover_yen)
```

```
##                      Sales       Price Advertisement       Hours
## Sales             1152.456   -9424.578     168504.67    301.6000
## Price            -9424.578 2744580.267    1149817.33 -10786.0889
## Advertisement 168504.667 1149817.333   32759182.22  24768.6667
## Hours              301.600  -10786.089      24768.67    197.0667
```

```r
# another way of computing the covariance between X1 and X2 using the old covariance
cov(pullover_yen[, 1], pullover_yen[, 2])
```

```
## [1] -9424.578
```

```r
cov(pullover[, 1], pullover[, 2]) * 106
```

```
## [1] -9424.578
```

```r
# another way of computing the covariance between X2 and X3 using the old covariance
cov(pullover_yen[, 2], pullover_yen[, 3])
```

```
## [1] 1149817
```

```r
cov(pullover[, 2], pullover[, 3]) * 106*106
```

```
## [1] 1149817
```

Comparing the two covariance matrises above, they differ significantly in some entries. To compute the new covaraince between X_1 and X_2, we multiply the old by the exchange rate while to compute the new covaraince between X_2 and X_3, we multiply the old by the square of exchange rate.

3.18

The trace is:

$$
\begin{aligned}
tr(\mathcal{H}) &= tr(I - \frac{1}{n}(1, \ldots, 1)^T(1, \ldots, 1)) \\
&= tr(I) - tr(\frac{1}{n}(1, \ldots, 1)^T(1, \ldots, 1)) \\
&= n - \frac{1}{n}n \\
&= n - 1
\end{aligned}
$$

.

To calculate the rank, we create the following matrix and do a series of row operations.

$$
\mathbf{X} = \begin{pmatrix} 1 & \mathbf{1} \\ \mathbf{0} & \mathcal{H} \end{pmatrix}
$$

Multiply the first row by $\frac{1}{n}$ and add it to the rest rows, we get:

$$
\mathbf{X} = \begin{pmatrix} 1 & \mathbf{1} \\ \mathbf{\frac{1}{n}} & I_n \end{pmatrix}
$$

And then subtract the sum of the second row to the last row from the first row, we get:

$$
\mathbf{X} = \begin{pmatrix} 0 & \mathbf{0} \\ \mathbf{\frac{1}{n}} & I_n \end{pmatrix}
$$

Apparently, this matrix has a rank of n. Since we have added an additional dimention to the original matrix, the original thus has a rank of $n - 1$. That's to say, $rank(\mathcal{H}) = n - 1$.

3.19

Note that $\mathcal{H} = I - \frac{1}{n}(1, \ldots, 1)^T(1, \ldots, 1)$ and $D = diag(Var(X_j))$, where X_j is the j^{th} column of X.

Then,

$\mathcal{H}X = X - (\frac{1}{n}, \ldots, \frac{1}{n})^T(1, \ldots, 1)X$

$= X - (\frac{1}{n}, \ldots, \frac{1}{n})^T(\Sigma_{i=1}^n x_{i1}, \ldots, \Sigma_{i=1}^n x_{ip})$

$= X - (\bar{X_1}, \ldots, \bar{X_p})$, where $\bar{X_j} = (\frac{\Sigma_{i=1}^n x_{ij}}{n}, \ldots, \frac{\Sigma_{i=1}^n x_{ij}}{n})^T = (\bar{x_j}, \ldots, \bar{x_j})^T$

$= (x_{ij} - \bar{x_j})_{i,j}$

By multiplying with $D^{-\frac{1}{2}}$, we have:

$\mathcal{H}XD^{-\frac{1}{2}} = (\frac{x_{ij} - \bar{x_j}}{\sqrt{Var(X_j)}})_{i,j}$

Then, we check the new mean and variance:

$(1, \ldots, 1)\mathcal{H}XD^{-\frac{1}{2}} = (\frac{\Sigma_i x_{i1} - n\bar{x_1}}{\sqrt{Var(X_1)}}, \ldots, \frac{\Sigma_i x_{ip} - n\bar{x_p}}{\sqrt{Var(X_p)}}) = (0, \ldots, 0)$

$S_{\mathcal{X}^*} = (\frac{\Sigma_k(x_{kj} - \bar{x_i})(x_{kj} - \bar{x_j})}{\sqrt{Var(X_i)}\sqrt{Var(X_j)}} = \mathcal{R}_{\mathcal{X}}.$

note that the effect of multiplying the centering matrix is setting the column mean to 0 and that of the multiplying $D^{-1/2}$ is setting column covariance to 1.