Homework Assignment 2

Jack, 2022/3/11

Due on March 13, 2022 at 11:59 pm

1. Consider the multivariate normal distribution vector $\mathrm{X} = (\mathrm{X}_1, \mathrm{X}_2, \mathrm{X}_3)^{\mathrm{T}}$ having mean vector $\mu = (0, 1, 2,)^{\mathrm{T}}$ and covariance matrix

$$\Sigma = \begin{bmatrix} 1 & -0.5 & 0.5 \\ -0.5 & 1 & -0.5 \\ 0.5 & -0.5 & 1 \end{bmatrix}$$

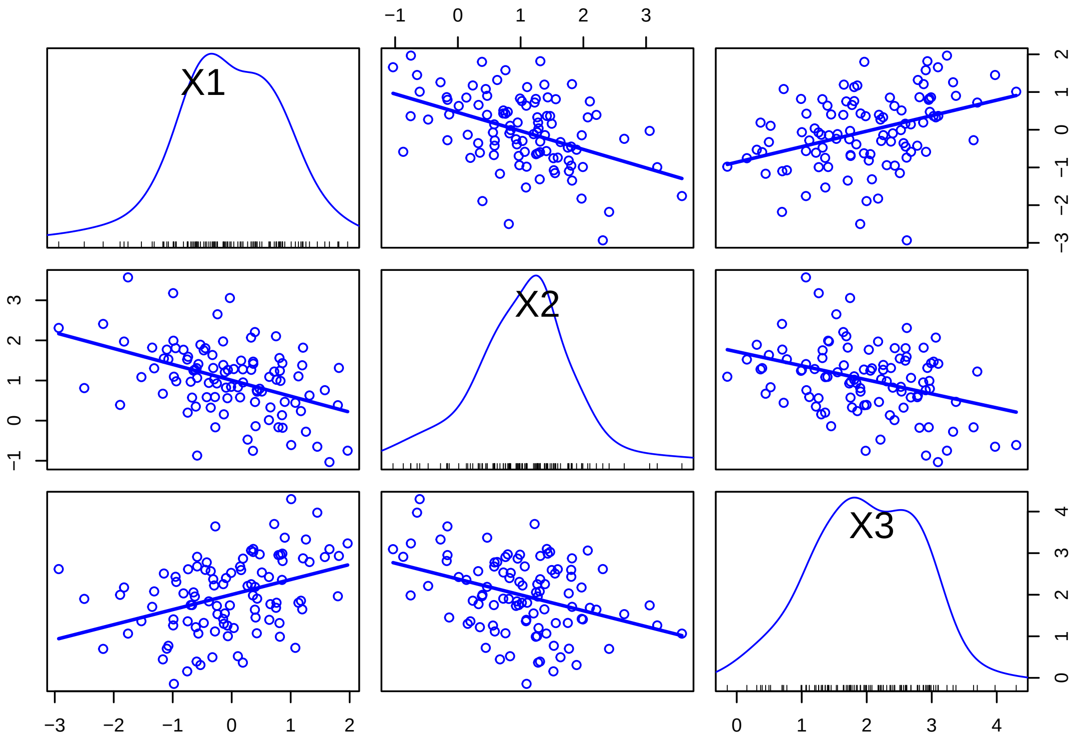
a) Generate 100 random observations from the multivariate normal distribution given above with *set.seed(12)*. (Hint: see *?mvrnorm*) You may need to use the package MASS.

```
library(MASS) # you may need to use this package
set.seed(12)
cov = matrix(c(1,-0.5,0.5, -0.5,1,-0.5, 0.5,-0.5,1), nrow=3,ncol=3)
mu = c(0, 1, 2)
X<-mvrnorm(100, mu, cov)
head(X)
```

```
##         [,1]    [,2]   [,3]
## [1,] -2.9331  2.3101 2.6166
## [2,]  1.2586 -0.2766 3.3281
## [3,] -0.0309  3.0566 1.7439
## [4,] -1.5314  1.0853 1.3632
## [5,] -2.1794  2.4104 0.6966
## [6,]  0.7511  2.1036 1.6855
```

b) Construct a scatterplot matrix for **X** and add a fitted smooth density curve on the diagonal panels for each $\mathrm{X}_1, \mathrm{X}_2, \mathrm{X}_3$ to verify that the location and correlation for each plot agrees with the parameters of the corresponding bivariate distributions.

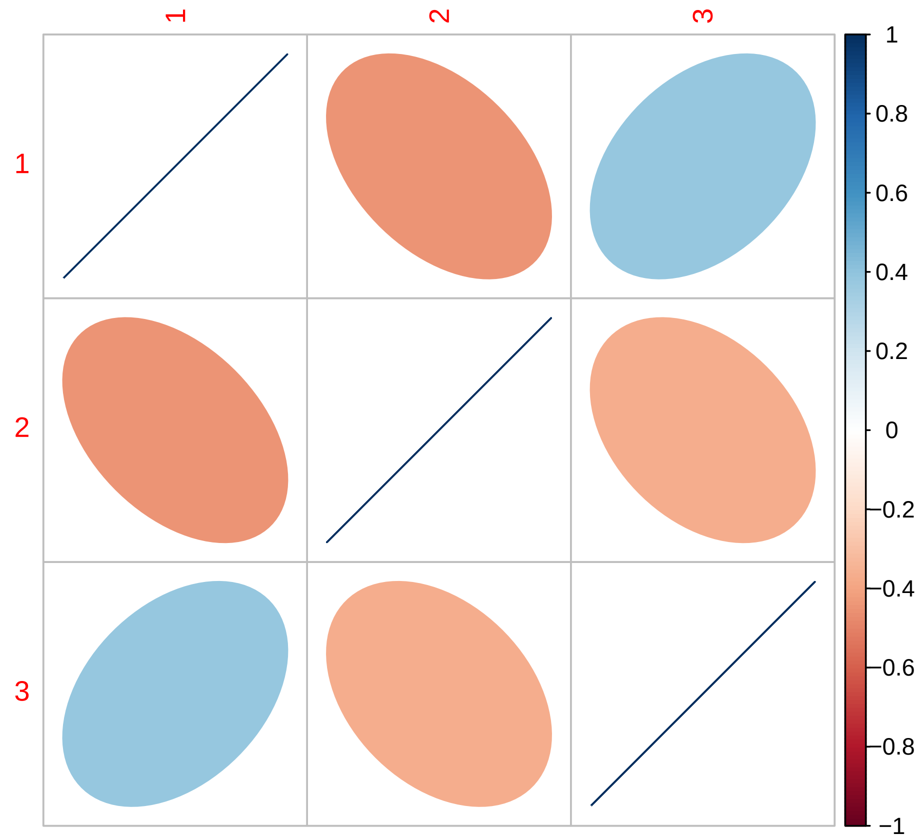
```
library("car")
scatterplotMatrix(X, smooth=F)
```




```
# regLine=F
```

c) Obtain the correlation plot for the generated sample **X**, where coefficients are added to the plot whose magnitude are presented by different colors. Let the visualization method of correlation matrix to be ellipse.

```
library(corrplot) # you may need to use this package
corr <- cor(X)
corrplot(corr, method="ellipse")
```



d) Given the covariance matrix Σ, find σ_{x_1}, σ_{x_2} and $\rho_{x_1 x_2}$. Consider the joint PDF of bivariate normal distribution

$$f_{XY}(x,y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho^2}}$$
$$\exp\left\{-\frac{1}{2\left(1-\rho^2\right)}\left[\left(\frac{x-\mu_X}{\sigma_X}\right)^2 + \left(\frac{y-\mu_Y}{\sigma_Y}\right)^2 - 2\rho\frac{(x-\mu_X)\,(y-\mu_Y)}{\sigma_X\sigma_Y}\right]\right\},$$

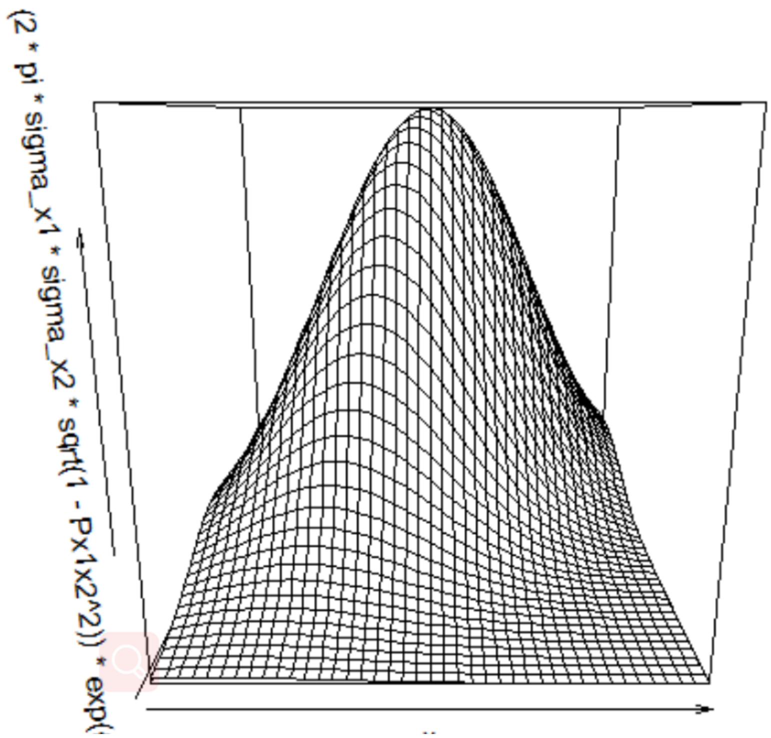
sketch a surface plot for X_1 and X_2, based on their bivariate probability density function. (Hint: if you want to use *curve3d*, please install and use the package *emdbook*)

Ans: Initially, according to the function and method of covariance matrix, we can solve each ρ and each σ

$$Cov = \begin{pmatrix} \sigma_{x_1}^2 & \rho_{x_1x_2}\sigma_{x_1}\sigma_{x_2} & \rho_{x_1x_3}\sigma_{x_1}\sigma_{x_3} \\ \rho_{x_1x_2}\sigma_{x_1}\sigma_{x_2} & \sigma_{x_2}^2 & \rho_{x_2x_3}\sigma_{x_2}\sigma_{x_3} \\ \rho_{x_1x_3}\sigma_{x_1}\sigma_{x_3} & \rho_{x_2x_3}\sigma_{x_2}\sigma_{x_3} & \sigma_{x_3}^2 \end{pmatrix} = \begin{bmatrix} 1 & -0.5 & 0.5 \\ -0.5 & 1 & -0.5 \\ 0.5 & -0.5 & 1 \end{bmatrix}$$

```
```r
library(emdbook) # you may need to use this package
x <- X[,1]
y <- X[,2]
sigma_x1 <- 1
sigma_x2 <- 1
Px1x2 <- 0.5
f_x1x2 <- curve3d(1/(2*pi*sigma_x1*sigma_x2*sqrt(1-Px1x2^2))*exp((-1/(2*(1-Px1x2^2)))*(((x-mu[1])/sigma_
```
```

! [] (homework2---Handout_files/figure-latex/unnamed-chunk-4-1.pdf)<!-- -->



Plug all the variable into this equation, we can simplify the function as follow:

$(1/(3*pi))*exp((2/3)*(x^2+(y-1)^2-x*(y-1)))$

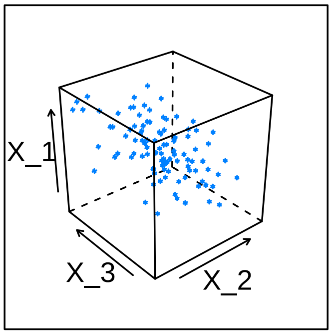
e) Sketch 3-D scatter plots for each of X_1, X_2 and X_3 as a z axis and rest two variables as x and y axes. Put these 3 plots in one picture.

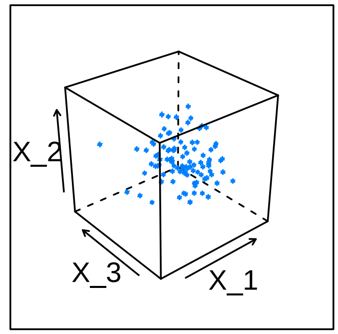
```
library(lattice) # you may need to use this package
X_1 <- X[,1]
X_2 <- X[,2]
X_3 <- X[,3]

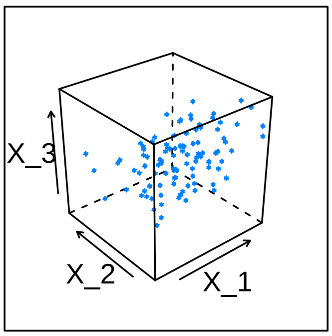
# X_1 as z axis
print(cloud(X_1 ~ X_2 * X_3), split = c(1, 1, 2, 2), more = TRUE)

# X_2 as z axis
print(cloud(X_2 ~ X_1 * X_3), split = c(2, 1, 2, 2), more = TRUE)

# X_3 as z axis
print(cloud(X_3 ~ X_1 * X_2), split = c(1, 2, 2, 2), more = TRUE)
```







2. A continuous random variable X has the probability density function

$$f_X(t) = \begin{cases} at + bt^2 & 0 < t < 1 \\ 0 & \text{otherwise} \end{cases}.$$

If $E[X] = 1/2$, find (a) a and b; (b) $P(X < 1/2)$; (c) $\text{Var}(X)$; (d) Generate the density plot of X

(a) We know that $E[X] = 1/2$ and the probability density function(pdf) of variable X. We can get a system of two equations to solve the two unknowns number a and b.

$$\begin{cases} \int_0^1 at + bt^2 dt = 1 \\ \int_0^1 t(at + bt^2) dt = \frac{1}{2} \end{cases}.$$

And then we can solve for definite integrals to solve the system. Finally, the value of a is 6 and b is -6.

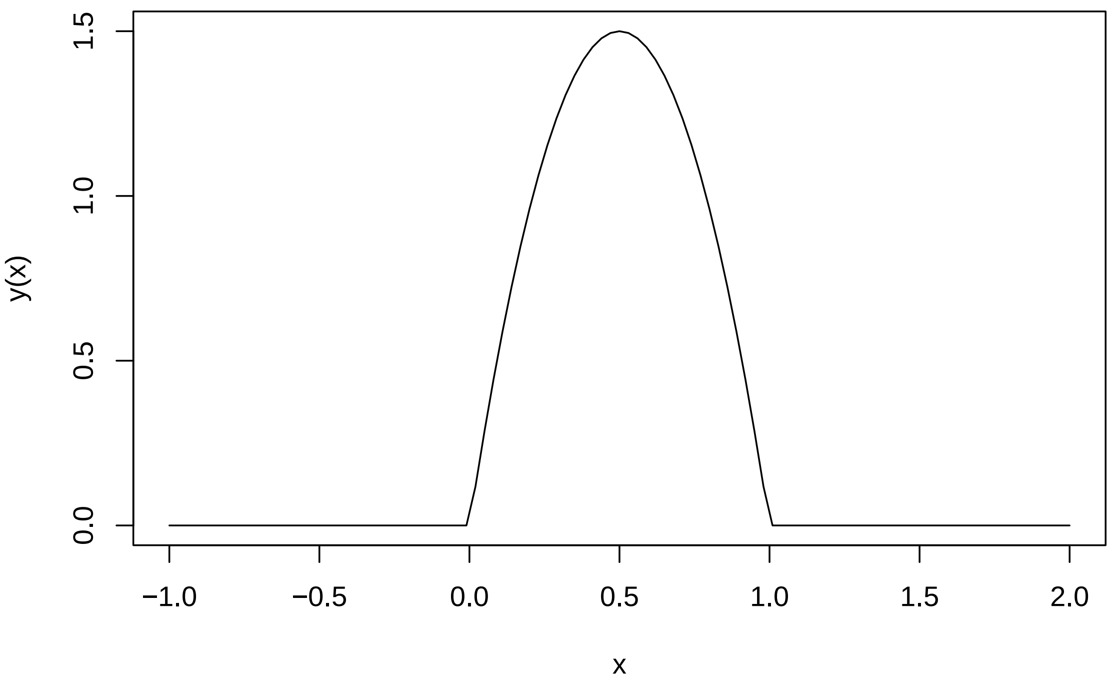
(b) $P(X < \frac{1}{2}) = Ft(X) = \int_0^{\frac{1}{2}} at + bt^2 dt = \int_0^{\frac{1}{2}} 6t - 6t^2 dt = 3t^2 - 2t^3|_{t=0}^{t=\frac{1}{2}} = \frac{3}{4} - \frac{1}{4} - 0 = \frac{1}{2}$

(c) According the function $Var(X) = E[X^2] - E^2[X]$, we can get the variance of X. $E[X^2] = \int_0^1 at^3 + bt^4 dt = \int_0^1 6t^3 - 6t^4 dt = \frac{3}{2}t^4 - \frac{6}{5}t^5|_{t=0}^{t=1} = \frac{3}{10}$

And we also know that $E(X) = \frac{1}{2}$, $Var(X) = E[X^2] - E^2[X] = \frac{3}{10} - (\frac{1}{2})^2 = \frac{1}{20}$

(d) The density plot of X is as follow:

```
fx<-function(t){
  if(t>0 & t<1){
    return (6*t-6*t^2)
  }
  else{
    return(0)
  }
}
y <- Vectorize(fx)
curve(y, -1, 2)
```



3. Consider a nonparametric regression model

$$y_i = g\left(x_i\right) + \epsilon_i, \quad 1 \le i \le n,$$

where y_i's are observations, g is an unknown function, and ϵ_i's are independent and identically distributed random errors with zero mean and variance σ^2. n is the number of observations. Usually one fits the mean function g first and then estimates the variance σ^2 from residual sum of squares $\hat{\sigma}^2 = \sum_{i=1}^n \hat{\epsilon}_i^2/(n-1)$ where $\hat{\epsilon}_i = y_i - \hat{g}(x_i)$. However this method requires an estimate of the unknown function g. Then some researchers proposed some difference-based estimators which does not require the estimation of g. Assume that x is univariate and $0 \le x_1 \le \cdots \le x_n \le 1$. Rice (1984) proposed the first order difference-based estimator

$$\hat{\sigma}_R^2 = \frac{1}{2(n-1)} \sum_{i=2}^{n} \left(y_i - y_{i-1}\right)^2 .$$

Gasser, Sroka and Jennen-Steinmetz (1986) proposed the second order difference based estimator and for equidistant design points (i.e. x_i and x_{i+1} have the same distance for all $i = 1, 2, \dots, n$), $\hat{\sigma}_{GSJ}^2$

reduces to

$$\hat{\sigma}^2_{GSJ} = \frac{2}{3(n-2)} \sum_{i=2}^{n-1} \left(\frac{1}{2} y_{i-1} - y_i + \frac{1}{2} y_{i+1} \right)^2 .$$

Consider the temperature anomaly dataset. Temperature anomalies in degrees Celsius are based on the new version HadCRUT4 land-sea dataset (Morice et al., 2012). We focus on the global median annual temperature anomalies from 1850 to 2019 relative to the 1961-1990 average. We try to build up the model between time and global median temperature y_i and year x_i.

(a) Use *read.csv* to read the temperature anomaly dataset. Let x be the vector of years from 1850-2019, y be the vector of corresponding global median annual temperature anomalies, and n be the number of observations

```
data <-read.csv("temperature-anomaly.csv")
data_global <- data[which(data$Entity=="Global"),]
x <- data_global['Year'][,]
y <- data_global['Median'][,]
n <- nrow(data_global)
n
```

```
## [1] 170
```

Show the result:

```
head(x)
```

```
## [1] 1850 1851 1852 1853 1854 1855
```

```
tail(x)
```

```
## [1] 2014 2015 2016 2017 2018 2019
```

```
head(y)
```

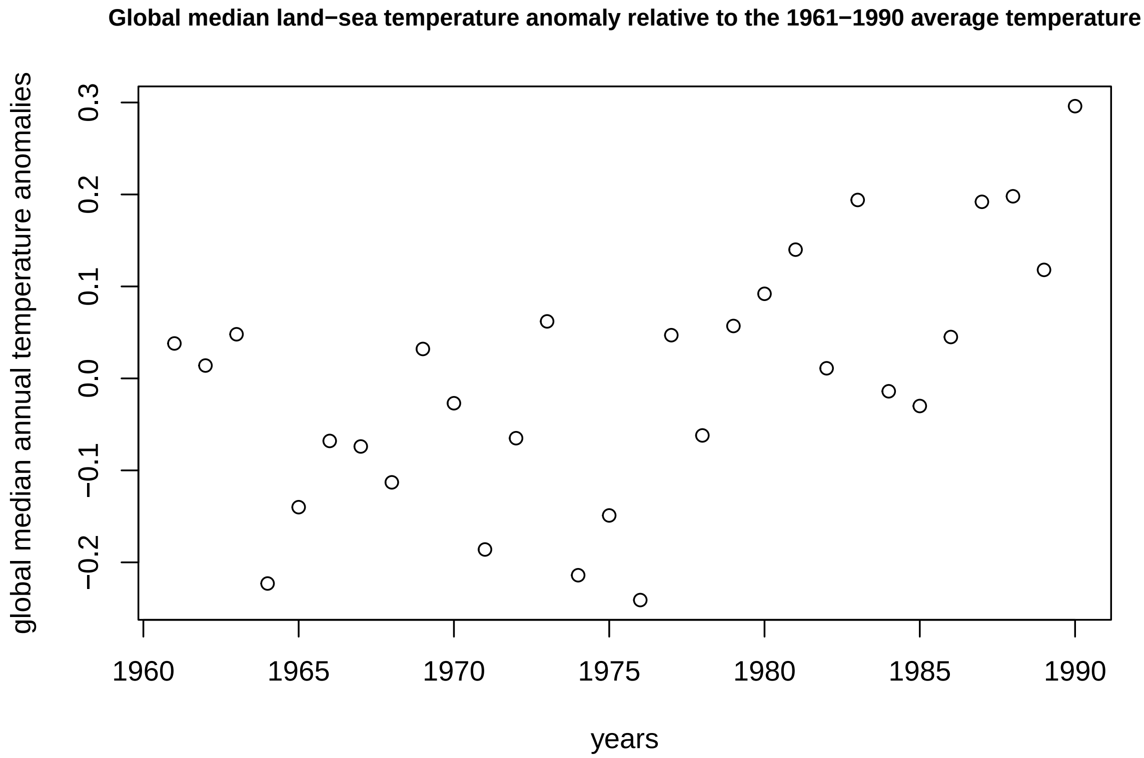
```
## [1] -0.373 -0.218 -0.228 -0.269 -0.248 -0.272
```

```
length(y)
```

```
## [1] 170
```

(b) Display a scatter plot between global median annual temperature anomalies and years with caption "Global median land-sea temperature anomaly relative to the 1961-1990 average temperature", $x-$label years and $y-$label temperature anomalies.

```
data_global <- data[which(data$Entity=="Global") ,]
data_year <- data_global[which(data_global$Year >= 1961), ]
data_year <- data_year[which(data_year$Year <= 1990), ]
x2 <- data_year['Year'][,]
y2 <- data_year['Median'][,]
plot(x2,y2,xlab = "years",ylab = "global median annual temperature anomalies",main = "Global median
```



(c) Change the years x to a new vector x such that $x_i = i/n$. Compute the first order difference-based estimator. (Note: the change of x or not will not affect the computation of the estimator) Using the function of the first order difference-based estimator, we can get the result as follow:

```
new_x <- x/n
# First order difference-based estimator
R <- 1/(2*(n-1))*sum((y[2:n] - y[1:n-1])^2)
R
```

```
## [1] 0.006658
```

(d) Compute the second order difference-based estimator. Using the function of the second order difference based estimator, we can get the result as follow:

```
# diff = 1/2*y[1:n-2] - y[2:n-1] + 1/2*y[3:n]
# Second order difference-based estimator
yi_1 <- n-2
yi_2 <- n-1
yi_3 <- n
GSJ <- 2/(3*(n-2))*sum((0.5*y[1:yi_1] - y[2:yi_2] + 0.5*y[3:yi_3])^2)
GSJ
```

```
## [1] 0.005406
```