MACT 4233 - Assignment 1

Code

Omar Moustafa 900222400

This is an R Markdown (http://rmarkdown.rstudio.com) Notebook. When you execute code within the notebook, the results appear beneath the code.

Hide

```
getwd()
```

```
[1] "/Users/omar/Desktop"
```

Question 1. Read any data set (of your choice) that consists of at least 5 quantitative variables into R

Hide

```
# Store the data set in an object and call it 'x'
x = read.csv("updated_version.csv", header = TRUE)
```

a. Print the first 5 rows of the data set using the R command

Hide

```
print(head(x,5))
```

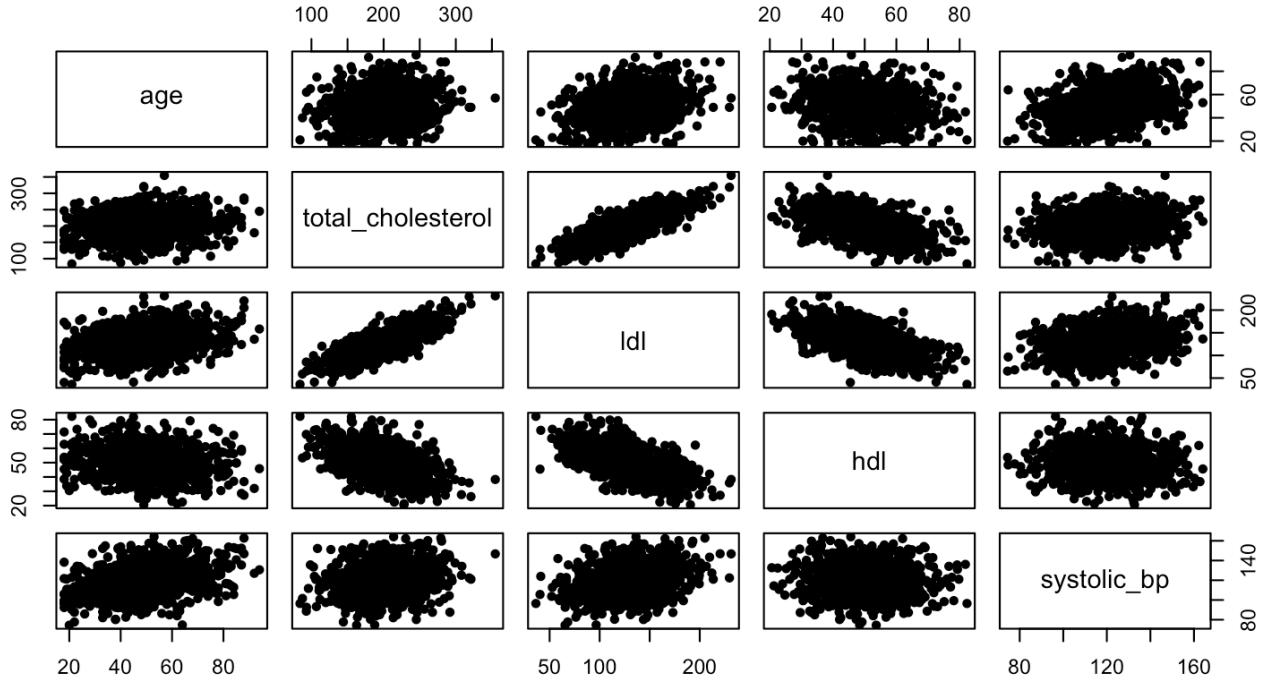
a..	s..	total_cholesterol	ldl	hdl	systolic_bp	diastolic_bp	smok...	diabet
<int>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<int>	<i
1 57	1	229.4636	175.8791	39.22569	124.07013	91.37878	0	
2 58	1	186.4641	128.9849	34.95097	95.49255	64.35504	1	
3 37	1	251.3007	152.3476	45.91329	99.51933	64.95315	0	
4 55	1	192.0589	116.8037	67.20893	122.46000	73.82138	0	
5 53	1	151.2034	107.0174	60.69384	123.02226	81.12195	0	

5 rows | 1-10 of 10 columns

b. Produce the scatter plot matrix of any five variables

Hide

```
# Selecting the first five out of the six numeric variables in 'df'
pairs(x[, c("age", "total_cholesterol", "ldl", "hdl", "systolic_bp")],
      main = "Scatter Plot Matrix of 5 Numeric Variables", pch = 16)
```



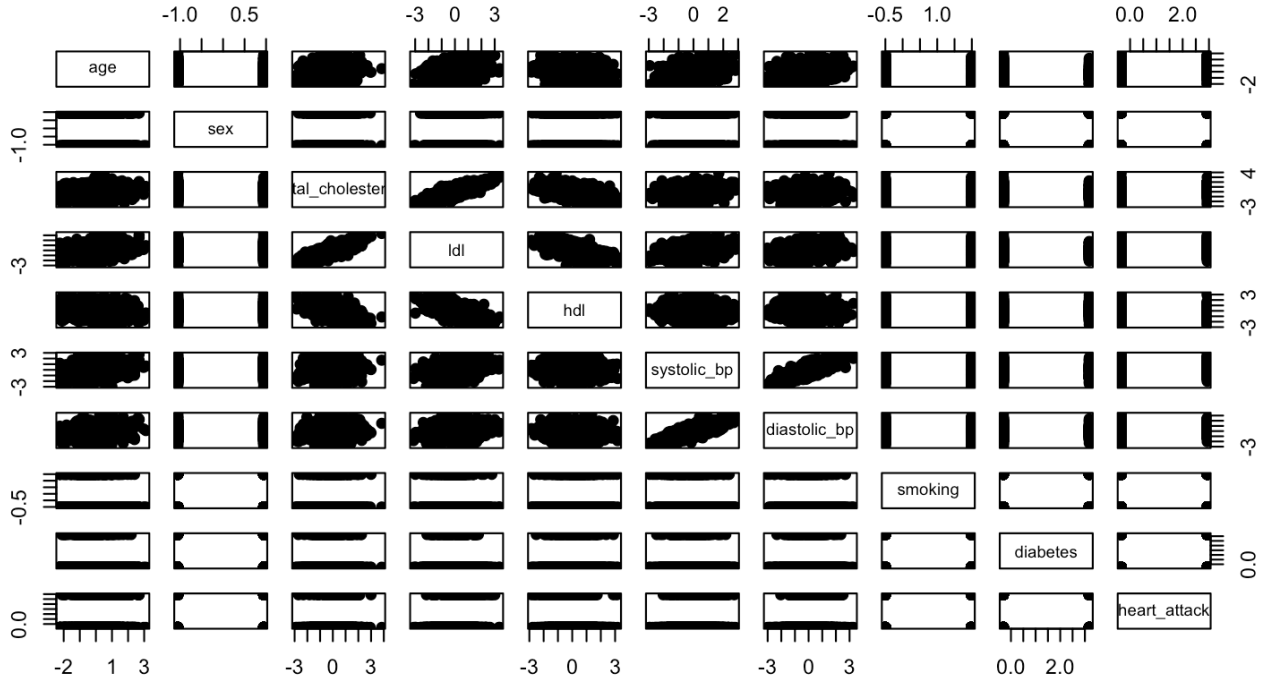
c. Comments on the above graphical displays:

1. There's a clear positive relationship between the variables "total_cholesterol" & "ldl" as the points seem to trend upwards, indicating that as total_cholesterol increases, ldl tends to do the same. This is consistent with the known physiological relationship which is that LDL is a main component of total cholesterol.
2. There's a clear negative relationship between the variables "ldl" & "hdl" as the points seem to trend downward, indicating that ldl increases, hdl tends to decreases.
3. It seems that the variable "hdl" has non-linear relationships with both "age" and "systolic_bp."
4. There are a handful of data points in the variables "total_cholesterol" & "ldl" which could be classified as outliers would could, therefore, represent high-risk patients suffering from cholesterol levels that are either abnormally high or abnormally low.

d. Standardize the variables in your data using z = scale(x), then construct the scatter plot matrix using the following command

Hide

```
z = scale(x)
pairs(z, pch = 19, main = "Scatter Plot Matrix")
```



e. What are the differences between this plot and the previous one?

1. One major difference is that in this plot, all variables were standardized using z = scale(x) which converts each variable to follow the criteria of the standard normal random variable, particularly to have a mean of 0 and standard deviation of 1. In the new plot, the axes are in standardized units (z-score values) instead of the original units like the previous plot.
2. Another key difference is that in this plot, correlations and outliers are easier to perceive and takeaway from it compared to the previous plot. This is because, the relationships existing in the new plot appear more centered around zero which helps clearly identify them.

f. Compute the mean vector x_bar and the covariance matrix, S, and the correlation matrix, R, of these variables.

Hide

```
# Compute the mean vector, 'x_bar'
x_bar = colMeans(x)
print("Mean Vector X-Bar:")
```

```
[1] "Mean Vector X-Bar:"
```

Hide

```
x_bar
```

```
             age              sex total_cholesterol              ldl
hdl       systolic_bp
         49.88600          0.52700        201.08749        130.04781         49.8
1124        120.31269
     diastolic_bp           smoking         diabetes      heart_attack
         80.23125          0.20200          0.09000          0.10400
```

Hide

```
# Compute the covariance matrix, 'S'
S = cov(x)
print("Covariance Matrix S:")
```

```
[1] "Covariance Matrix S:"
```

Hide

```
S
```

```
                         age          sex total_cholesterol           ldl
hdl systolic_bp diastolic_bp
age               201.9089129 -0.573495495        78.6981325  117.98363070  -19.01310
949  75.4141915   35.5018033
sex                -0.5734955  0.249520521         0.0863663   -0.08899521    0.14285
132   0.6408879    0.2216204
total_cholesterol  78.6981325  0.086366295      1603.4142094  953.64979963 -195.71252
679 119.6793779   39.2518584
ldl               117.9836307 -0.088995210       953.6497996  902.50129617 -178.86801
795 134.6746711   65.3015759
hdl               -19.0131095  0.142851325      -195.7125268 -178.86801795  105.00464
991 -10.2644589   -8.5822511
systolic_bp        75.4141915  0.640887934       119.6793779  134.67467108  -10.26445
888 240.4823397  127.7065860
diastolic_bp       35.5018033  0.221620434        39.2518584   65.30157589   -8.58225
114 127.7065860  104.7739932
smoking            -0.8878599  0.013559560        -0.9137725   -1.63408650    0.31759
241  -0.7617190   -0.1953443
diabetes           -0.3821221  0.006576577        -0.3283134   -0.28173921    0.04958
405  -0.3567948   -0.0193464
heart_attack        0.4733293  0.017209209         2.1634570    1.45851949   -0.45485
394   0.8258848    0.5452793
                        smoking     diabetes heart_attack
age               -0.887859860 -0.382122122   0.47332933
sex                0.013559560  0.006576577   0.01720921
total_cholesterol -0.913772469 -0.328313420   2.16345699
ldl               -1.634086497 -0.281739207   1.45851949
hdl                0.317592411  0.049584053  -0.45485394
systolic_bp       -0.761718995 -0.356794765   0.82588482
diastolic_bp      -0.195344259 -0.019346404   0.54527932
smoking            0.161357357  0.007827828   0.02601802
diabetes           0.007827828  0.081981982   0.02566567
heart_attack       0.026018018  0.025665666   0.09327728
```

Hide

```
# Compute the correlation matrix, 'R'
R = cor(x)
print("Correlation Matrix R:")
```

```
[1] "Correlation Matrix R:"
```

Hide

R

```
                          age          sex total_cholesterol          ldl         hdl
systolic_bp diastolic_bp
age                1.00000000 -0.080797725       0.138313260  0.276388580 -0.13057834
0.34224234  0.244087714
sex               -0.08079772  1.000000000       0.004317857 -0.005930476  0.02790788
0.08273464  0.043344093
total_cholesterol  0.13831326  0.004317857       1.000000000  0.792760752 -0.47697047
0.19273246  0.095765838
ldl                0.27638858 -0.005930476       0.792760752  1.000000000 -0.58103797
0.28908157  0.212360137
hdl               -0.13057834  0.027907882      -0.476970473 -0.581037967  1.00000000
-0.06459371 -0.081822022
systolic_bp        0.34224234  0.082734637       0.192732460  0.289081575 -0.06459371
1.00000000  0.804535019
diastolic_bp       0.24408771  0.043344093       0.095765838  0.212360137 -0.08182202
0.80453502  1.000000000
smoking           -0.15555081  0.067576869      -0.056809481 -0.135411871  0.07715631
-0.12228095 -0.047509397
diabetes          -0.09392155  0.045981967      -0.028635618 -0.032753986  0.01689968
-0.08035588 -0.006601069
heart_attack       0.10906810  0.112802792       0.176903926  0.158964588 -0.14533820
0.17437730  0.174423239
                       smoking     diabetes heart_attack
age                -0.15555081 -0.093921545    0.1090681
sex                 0.06757687  0.045981967    0.1128028
total_cholesterol -0.05680948 -0.028635618    0.1769039
ldl                -0.13541187 -0.032753986    0.1589646
hdl                 0.07715631  0.016899683   -0.1453382
systolic_bp        -0.12228095 -0.080355882    0.1743773
diastolic_bp       -0.04750940 -0.006601069    0.1744232
smoking             1.00000000  0.068059330    0.2120762
diabetes            0.06805933  1.000000000    0.2934982
heart_attack        0.21207618  0.293498168    1.0000000
```

g. Verify the relationship between the covariance and correlation matrices, that is, how R is obtained from S and vice versa.

Hide

```
if(!require("dplyr")) install.packages("dplyr")
library(dplyr)
```

Hide

```
# Compute D, which is the diagonal matrix of standard deviations
D = diag(S)^-0.5  # Extract diag and take sqrt
D = diag(D)  # Create a diagonal matrix

R1 = D %*% S %*% D  # Matrix Multiplication

D_inverse = diag(S)^0.5  # Now take the positive sqrt
D_inverse = diag(D_inverse)  # Create a diagonal matrix

# Compute S from R
S1 = D_inverse %*% R %*% D_inverse  # Transform correlation back to covariance

near(R, R1)  # All are true now
```

```
                  age  sex total_cholesterol  ldl  hdl systolic_bp diastolic_bp smok
ing diabetes heart_attack
age               TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
sex               TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
total_cholesterol TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
ldl               TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
hdl               TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
systolic_bp       TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
diastolic_bp      TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
smoking           TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
diabetes          TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
heart_attack      TRUE TRUE             TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
```

Hide

```
cat("\n") # Printing an empty line for legibility reasons
```

Hide

```
near(S, S1)  # All are true now
```

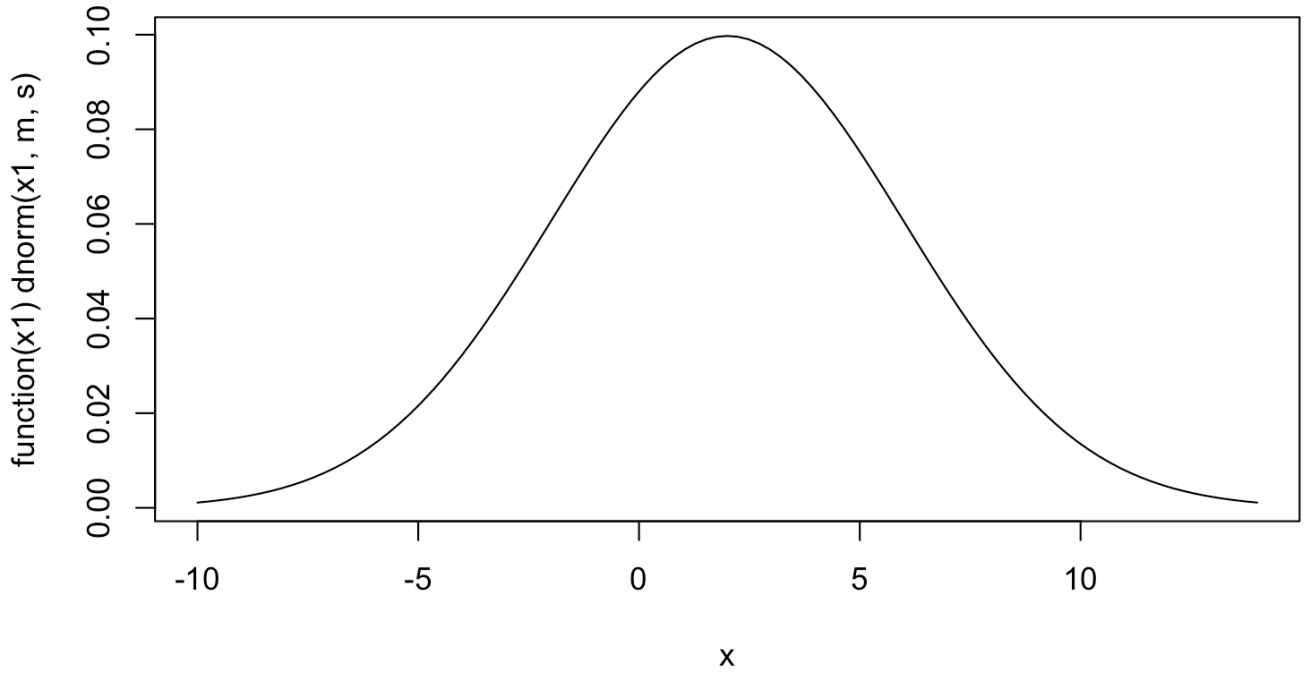
```
                  age  sex total_cholesterol  ldl  hdl systolic_bp diastolic_bp smok
ing diabetes heart_attack
age               TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
sex               TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
total_cholesterol TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
ldl               TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
hdl               TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
systolic_bp       TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
diastolic_bp      TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
smoking           TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
diabetes          TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
heart_attack      TRUE TRUE              TRUE TRUE TRUE        TRUE         TRUE    T
RUE     TRUE         TRUE
```

Question 2. Consider any univariate normal random variable $X \sim N(\mu, \sigma)$, other than the standard normal random variable ($\mu = 0$ and $\sigma = 1$), then using R:

a. Plot the density function, f(x), over its appropriate range

Hide

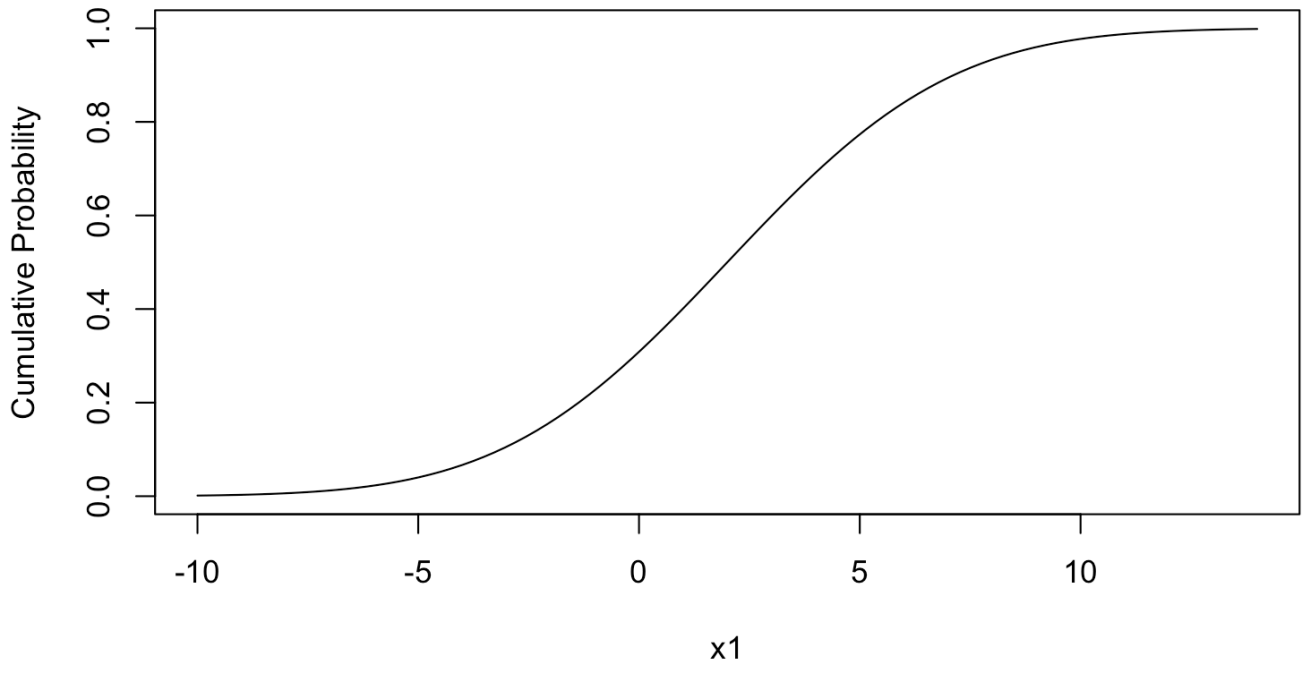
```
# Graph of Normal PDF
m = 2 # mean
s = 4 # sigma (standard deviation)
x1 = 0
plot(function(x1) dnorm(x1, m, s), m - 3 * s, m + 3 * s)
```



b. Plot cumulative distribution function, F(x), over its appropriate range.

Hide

```
# Graph of Normal CDF
# x1, m, and s were already defined in part (a)
plot(function(x1) pnorm(x1, m, s), m - 3 * s, m + 3 * s, xlab = "x1", ylab = "Cumulat
ive Probability")
```



c. Compute the height of the density function when x = μ - 1.5σ

Hide

```
# Normal PDF
# m and s were already defined in part (a)
new_x = m - 1.5*s
dnorm(new_x, m, s)
```

```
[1] 0.0323794
```

d. Compute the probability that the random variable X is less than μ - 1.5σ

Hide

```
# Normal CDF
# new_x, m, and s were already defined in part (c)
pnorm(new_x, mean = m, sd = s)
```

```
[1] 0.0668072
```

e. Compute the value of x such that the Pr(X >= x) = 0.17

Hide

```
# Quantiles of Normal
# m and s were already defined in part (a)
p = 0.17
qnorm(1 - p, m, s) # 1-p is what will help us find Pr(X >= x) = 0.17
```

```
[1] 5.816661
```

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Cmd+Option+I*.