HW1_shwasoo

Problem 1

R command

Syntax for package 'swirl'

```
install.packages('swirl')
library(swirl)
install_course('R_Programming_E')
swirl()
```

Problem 2

> *Things I want to learn in STAT 5014*

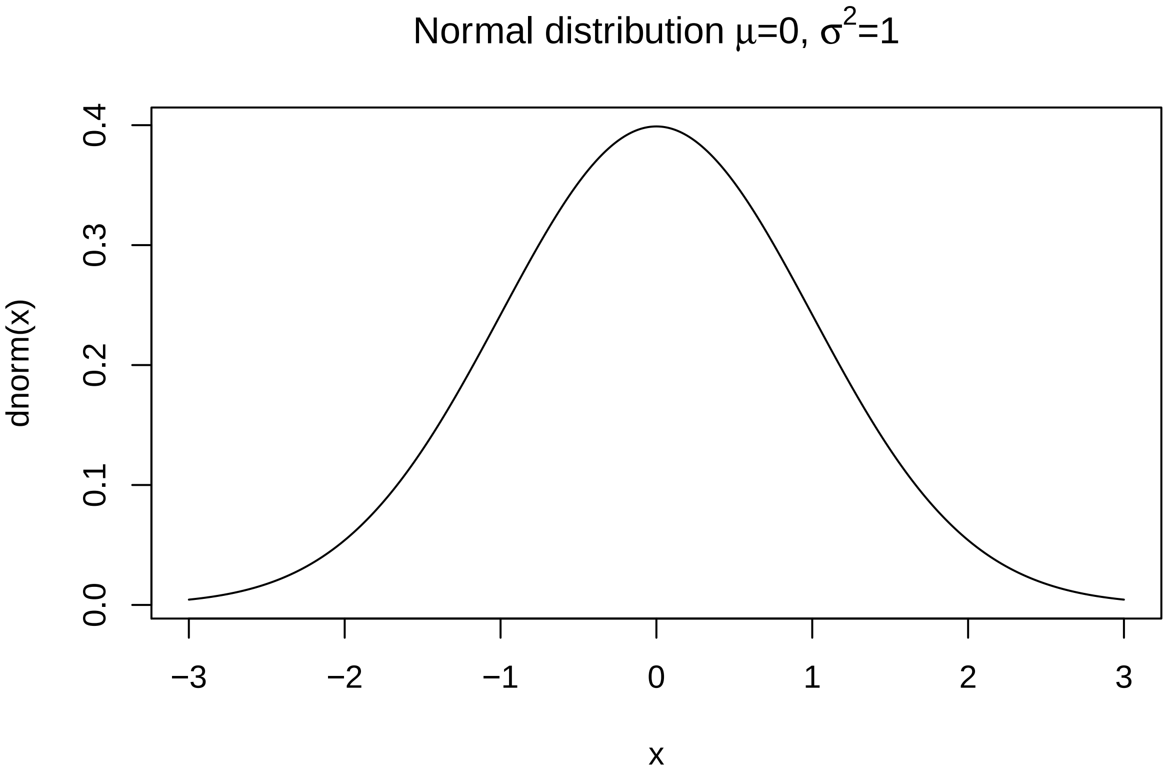
- Building algorithms in Python for statistical researches (Data mining, establishing statistcal models)
- Utilizing data visualization packages in R, which is one of the best functions of this programming language
- Acquiring basic programming skills for regression, ANOVA and applying to other fields

> *Some probability distributions*

$$\frac{1}{\sqrt{2\pi\sigma^2}}e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{1}$$

Above function is the normal distribution with mean μ and variance σ^2. To take a look how this distribution looks like, we will use R 3.6.0 to make a plot. To generate the function, we make a sequence of real numbers from -3 to 3, of 10000 numbers.

```
x<-seq(-3,3,length=10000) #Generating 10000 numbers from -3 to 3
plot(x,dnorm(x),type='l',main=expression(paste('Normal distribution ',mu,'=0, ',sigma^2,'=1')))
```

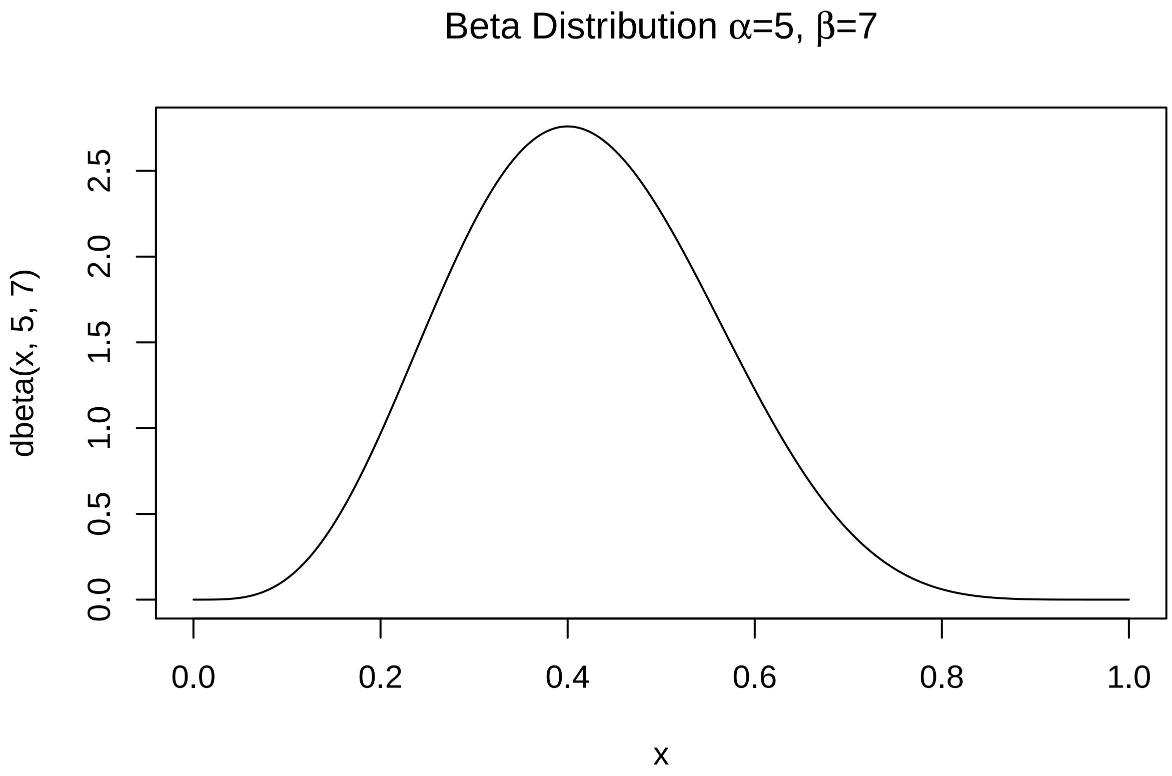


```
#Generate a normal distribution plot which mean is 0 and variance is 1
```

$$\beta(\alpha,\beta)x^{\alpha-1}(1-x)^{\beta-1} \tag{2}$$

Above function is the beta distribution with two parameters α and β. To take a look how this distribution looks like, we will use R 3.6.0 to make a plot. To generate the function, we make a sequence of real numbers from 0 to 1, which the intervals of sequent numbers are equally 0.0001. When generating plot, we will use parameters $\alpha = 5$ and $\beta = 7$.

```
x<-seq(0,1,0.0001) #Generate numbers from 0 to 1 with interval 0.0001
plot(x,dbeta(x,5,7),type='l',main=expression(paste('Beta Distribution ',alpha,'=5, ',beta,'=7')))
```



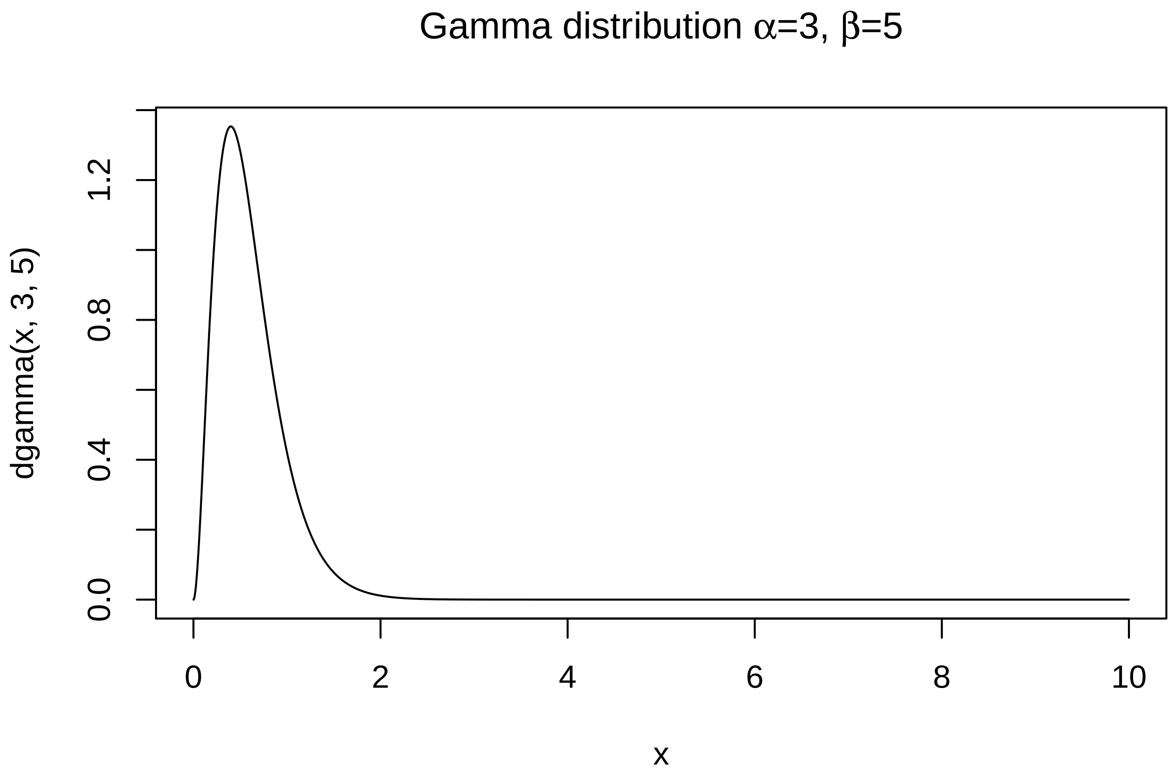
```
#Generate a beta distribution plot with alpha = 5 and beta = 7
```

Keep in mind that the shape of distribution can change when parameters change.

$$\frac{e^{-\frac{x}{\beta}}x^{\alpha-1}}{\gamma(\alpha)\beta^{\alpha}} \tag{3}$$

Above function is the gamma distribution with two parameters α and β. To take a look how this distribution looks like, we will use R 3.6.0 to make a plot. To generate the function, we make a sequence of 10000 real numbers from 0 to 10, which the intervals of sequent numbers are equal. When generating plot, we will use parameters $\alpha = 3$ and $\beta = 5$.

```
x<-seq(0,10,length=10000) #Generate 10000 numbers from 0 to 10
plot(x,dgamma(x,3,5),type='l',main=expression(paste('Gamma distribution ',alpha,'=3, ',beta,'=5')))
```



```
#Generate a gamma distribution plot with alpha = 3 and beta = 5
```

Keep in mind that the shape of distribution can change as parameters change.

Problem 3

In order to generate a scatterplot and a histogram, we will be using a basic dataset in R 3.6.0. The dataset we will be skimming is called 'mtcars', which is in 'datasets' package. The package will automatically be installed when you are downloading R. This data is originally from the *1974 Motor Trend US magazine* and has information of 32 automobiles (*rdocumentation.org*, 9.3.2019). It contains 11 variables and below is some information about those variables.

```
summary(mtcars)
```

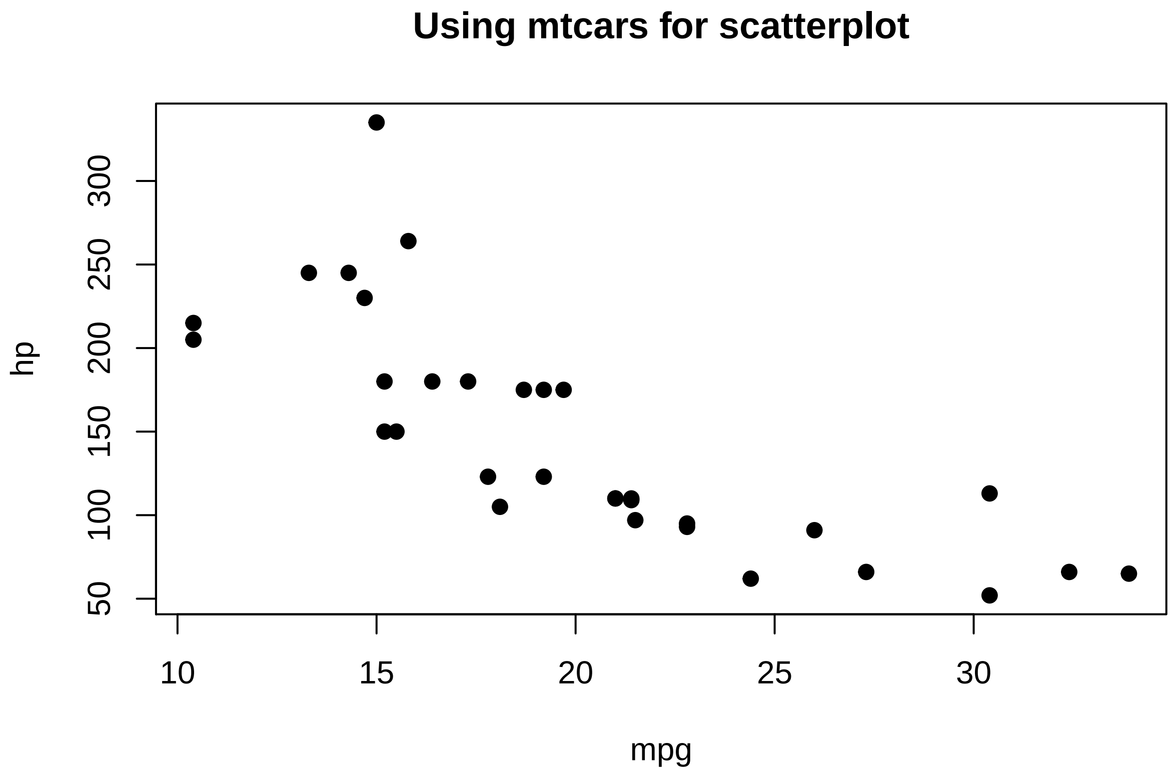
```
##       mpg             cyl             disp             hp       
##  Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0  
##  1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5  
##  Median :19.20   Median :6.000   Median :196.3   Median :123.0  
##  Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7  
##  3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0  
##  Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0  
##       drat             wt             qsec             vs        
##  Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000  
##  1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000  
##  Median :3.695   Median :3.325   Median :17.71   Median :0.0000
##  Mean   :3.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375
##  3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000
##  Max.   :4.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000
##        am              gear            carb
##  Min.   :0.0000   Min.   :3.000   Min.   :1.000
##  1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000
##  Median :0.0000   Median :4.000   Median :2.000
##  Mean   :0.4062   Mean   :3.688   Mean   :2.812
##  3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :1.0000   Max.   :5.000   Max.   :8.000
```

Each list shows minimum, 1st quantile, median, mean, 3rd quantile, and maximum value of the variable. From this data, we will use mpg variable and hp variable, which stands for miles per gallon and gross horsepower respectively, to make general plots.

Problem 4

To make a scatterplot, you can simply type ‘plot’ and enter the variables you will be using for x-axis and y-axis sequentially. These are the 2 arguments that is essential for ‘plot’ function. You can also modify your plot by using ‘pch’ argument to change the shape of points, ‘main’ to change the title of your plot, ‘xlab’ and ‘ylab’ to change the name of the axes labels.

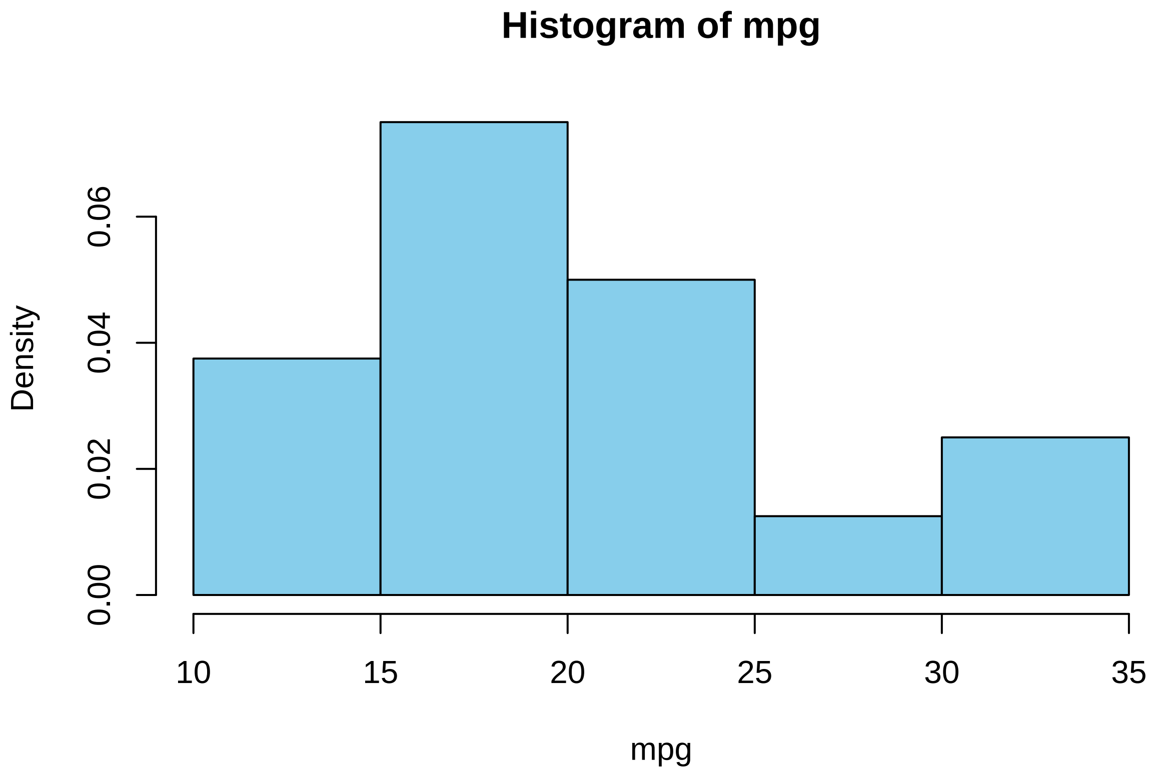
```
plot(mtcars$mpg,mtcars$hp,pch=19,main='Using mtcars for scatterplot',xlab='mpg',ylab='hp')
```



On above plot, we can see that the points are usually going decreasing direction. Which means, when mpg increases hp would usually decrease. Therefore we can assume that variable mpg and hp are in negative correlation.

To make a histogram, you can use 'hist' function. Only one argument is required for this function, which is the variable you want to use as x-axis. However, like the function above, you can put additional arguments to modify your plot. The 'col' argument is used to change the color of the bars, and 'prob' argument is used to change the y-axis. The default of y-axis will be count, but it can be changed to relative frequency.

```r
hist(mtcars$mpg,col='skyblue',xlab='mpg',main='Histogram of mpg',prob=T)
```



On above plot, we can see that the bar is the highest on 15-20 interval and gets lower as mpg increases, and slightly gets higher on 30-35 interval. From this histogram, we can assume that the data points are usually centered at value between 10 and 15, and there are also some more values between 30 and 35.