title: “R_Project_Applied Statistics, and Probabilties” author: “Jason Park/116333818” output: html_document:https://github.com/jasonpark9001/R_Project.git (https://github.com/jasonpark9001/R_Project.git) number_sections: yes toc: yes pdf_document: toc: yes — This project is to explore the data set; indtify the type of variables; perform hypothesis tests; and and perform linear regression analysis on the quantitative variables.

In this project we will be working with the dataset `mtcars’ from the datasets library in R 1.1 Loading the Dataset

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.0.3
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(datasets)
data(mtcars)
```

1.2 Use `head' and` names’ to explore the first few rows of the dataset and to #explore the different rows.

```
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
names(mtcars)
```

```
##  [1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "vs"   "am"   "gear"
## [11] "carb"
```

1.3 Classify each of the variables (ie is the columns) into quantitative or qualitative vari-ables.

```
cyl_4 <- filter(mtcars, cyl == '4')
cyl_8 <- filter(mtcars, cyl == '8')
automatic <-  filter(mtcars, am == '0')
manual <- filter(mtcars, am == '1')
```

2 Hypothesis Testing In this section we will perform two Hypothesis Tests.

2.1 Test if there is any significant difference between the mean horsepower for cars with automatic or manual transmission.

a. filter command to get two subsets of dataset

```
x_horse_auto <- automatic$hp  # automatic cars' hp subset
y_horse_manual <- manual$hp   # manual cars' hp subset
#x_bar and y_ bar
x_1 <- mean(x_horse_auto)
y_1 <- mean(y_horse_manual)
x_bar <- mean(x_horse_auto)
y_bar <- mean(y_horse_manual)
null_0 <- 0
m <- 19
n <- 13
sd1_bar <- sd(x_horse_auto)
sd2_bar <- sd(y_horse_manual)

#b)assumption: null hypothesis, hp0 ;mu1 - mu2 = 0
#c) alternative hypothesis, hpa ; mu1 - mu2 =/= 0
#d)test statistics & calculate df
t_statistic_two_sample <- function(x_bar, y_bar, null_0, sd1_bar, sd2_bar, m, n){
  (x_bar - y_bar - null_0)/ sqrt((sd1_bar^2)/m + (sd2_bar^2/n))

}

df_up_part <-  ((sd1_bar)^2/m + (sd2_bar)^2/n)^2
df_down_part <- ((sd1_bar)^2/m)^2/(m-1) + ((sd2_bar)^2/n)^2/(n-1)
df_pre <- df_up_part / df_down_part
df_pre
```

```
## [1] 18.71541
```

```
df_real <- floor(df_pre)
df_real
```

```
## [1] 18
```

```
#e) calculate the t_stat
t_horse_pwr<- t_statistic_two_sample(x_bar, y_bar, null_0, sd1_bar, sd2_bar, m, n)

#f) p-value with significan level = 0.05
alpha <- 0.05
Pvalue_t_twoside <- function(t, df){
  2*pt(abs(t), df, lower.tail = FALSE)
}

p_val_hor_power <- Pvalue_t_twoside(t_horse_pwr, df_real)
p_val_hor_power
```

```
## [1] 0.2215876
```

```
# g) conclusion
result <- p_val_hor_power > alpha
result
```

```
## [1] TRUE
```

```
#since p_val two-smple t test is greater than significant value(alpha= 0.05),
#the null hypothesis is fail to reject.
# we have enough envidence that the mean value of horse power from manual and automatic cars
# are not different.
```

2.2 Test if there is any significan difference between the average miles per gallon for cars equipped with 4 or 8 cylinders.

a. filter command to get two subsets of dataset

```
cyl_4 <- filter(mtcars, cyl == '4')
cyl_8 <- filter(mtcars, cyl == '8')
x_cyl_4 <- cyl_4$mpg # 4 cylinders cars' mpg subset
y_cyl_8 <- cyl_8$mpg   # 4 cylinders cars' mpg subset
```

x_bar and y_ bar and other necessary information sd1_bar & sd2_bar, m,n

```
x_bar_cyl <- mean(x_cyl_4)
y_bar_cyl <- mean(y_cyl_8)
null_0 <- 0
m_cyl <- 11
n_cyl <- 14
sd1_cyl_bar <- sd(x_cyl_4)
sd2_cyl_bar <- sd(y_cyl_8)
sd1_cyl_bar
```

```
## [1] 4.509828
```

```
sd2_cyl_bar
```

```
## [1] 2.560048
```

b)assumption: null hypothesis, hp0 ;mu1 - mu2 = 0 c) alternative hypothesis, hpa ; mu1 - mu2 =/= 0 d)test statistics & calculate df

```
t_statistic_two_sample <- function(x_bar, y_bar, null_0, sd1_bar, sd2_bar, m, n){
  (x_bar - y_bar - null_0)/ sqrt((sd1_bar^2)/m + (sd2_bar^2/n))

}

df_up_part_cyl <-  ((sd1_cyl_bar)^2/m_cyl + (sd2_cyl_bar)^2/n_cyl)^2
df_down_part_cyl <- ((sd1_cyl_bar)^2/m_cyl)^2/(m_cyl-1) + ((sd2_cyl_bar)^2/n_cyl)^2/(n_cyl-1)
df_pre_cyl <- df_up_part_cyl / df_down_part_cyl
df_pre_cyl
```

```
## [1] 14.96675
```

```
df_real_cyl <- floor(df_pre_cyl)
df_real_cyl
```

```
## [1] 14
```

e. calculate the t_stat

```
t_cyl_mpg<- t_statistic_two_sample(x_bar_cyl, y_bar_cyl, null_0, sd1_cyl_bar, sd2_cyl_bar, m_
cyl, n_cyl)
t_cyl_mpg
```

```
## [1] 7.596664
```

f. p-value with significan level = 0.05

```
alpha <- 0.05
Pvalue_t_twoside <- function(t, df){
  2*pt(abs(t), df, lower.tail = FALSE)
}

p_val_cyl_mpg <- Pvalue_t_twoside(t_cyl_mpg, df_real_cyl)
p_val_cyl_mpg
```

```
## [1] 2.487594e-06
```

g. conclusion

```
result_mpg <- p_val_cyl_mpg > alpha
result_mpg
```

```
## [1] FALSE
```

```
diff_alpha_0.01 <- 0.01
result_mpg_0.01 <- p_val_cyl_mpg > diff_alpha_0.01
result_mpg_0.01
```

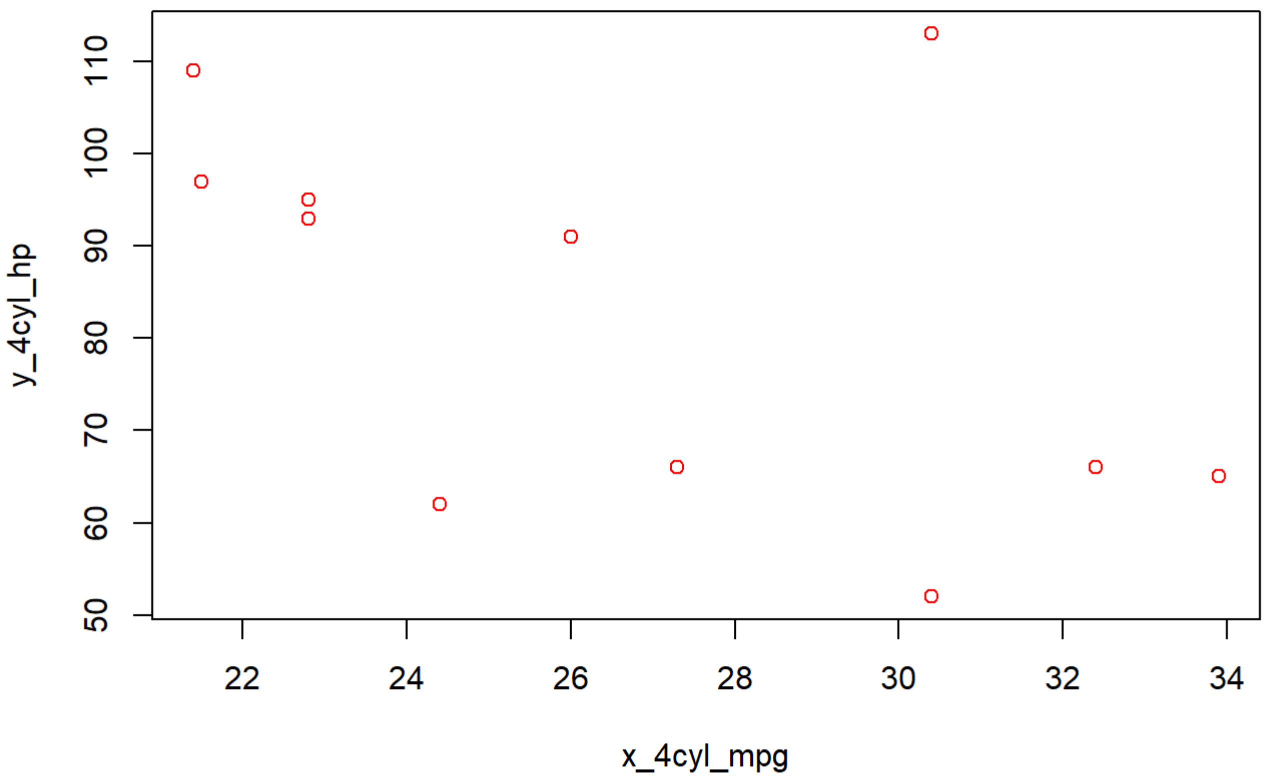
```
## [1] FALSE
```

The p-value = 0 is less than significant value 0.05 nor 0.01 null hypothesis h0 is rejected. We can conclude that the true average mpg from 4 cylinder cars and 8 cylinder cars are different.

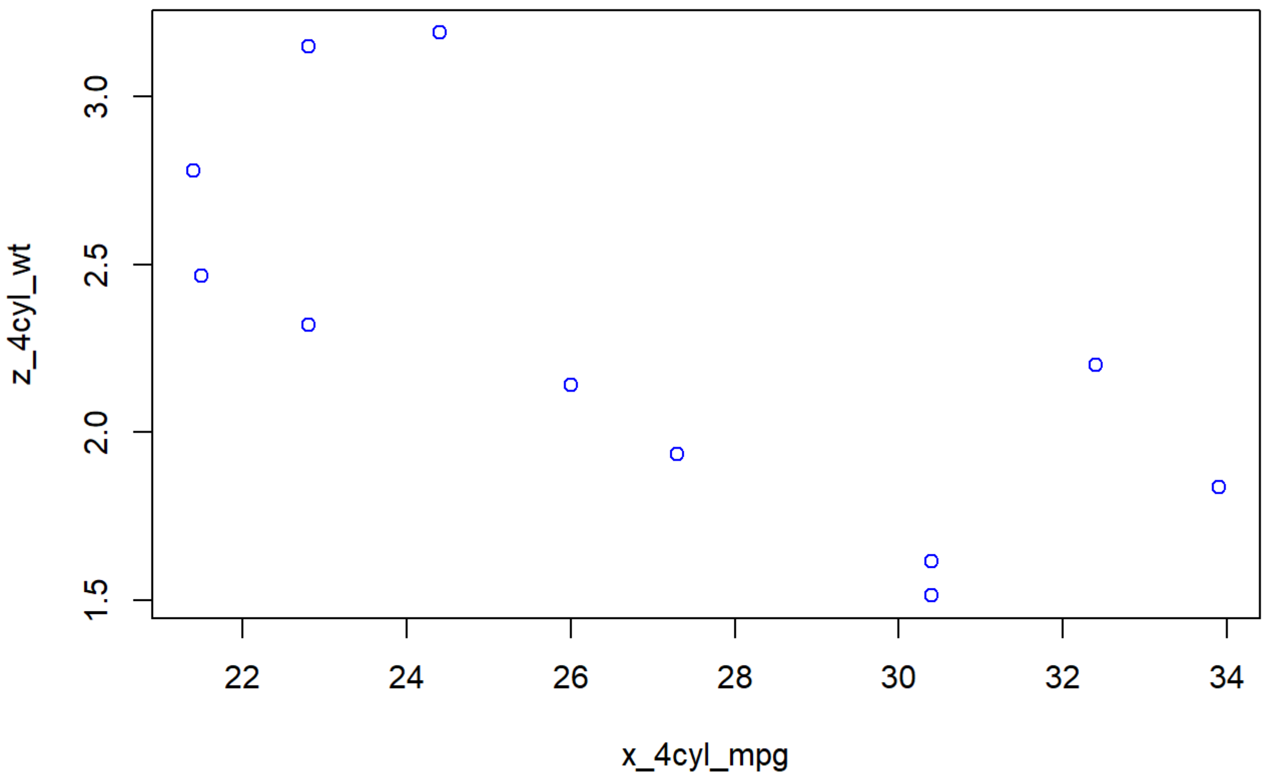
3 Linear Regression Form a new dataset, whose colums are exactly the quantitative columns of `mtcars’.

3.1 Use the `plot’ function to view simultaneous scatterplots. Then use the correlation functionto identify all pairs of variable that suggest a possiblity of a strong linear relationship.

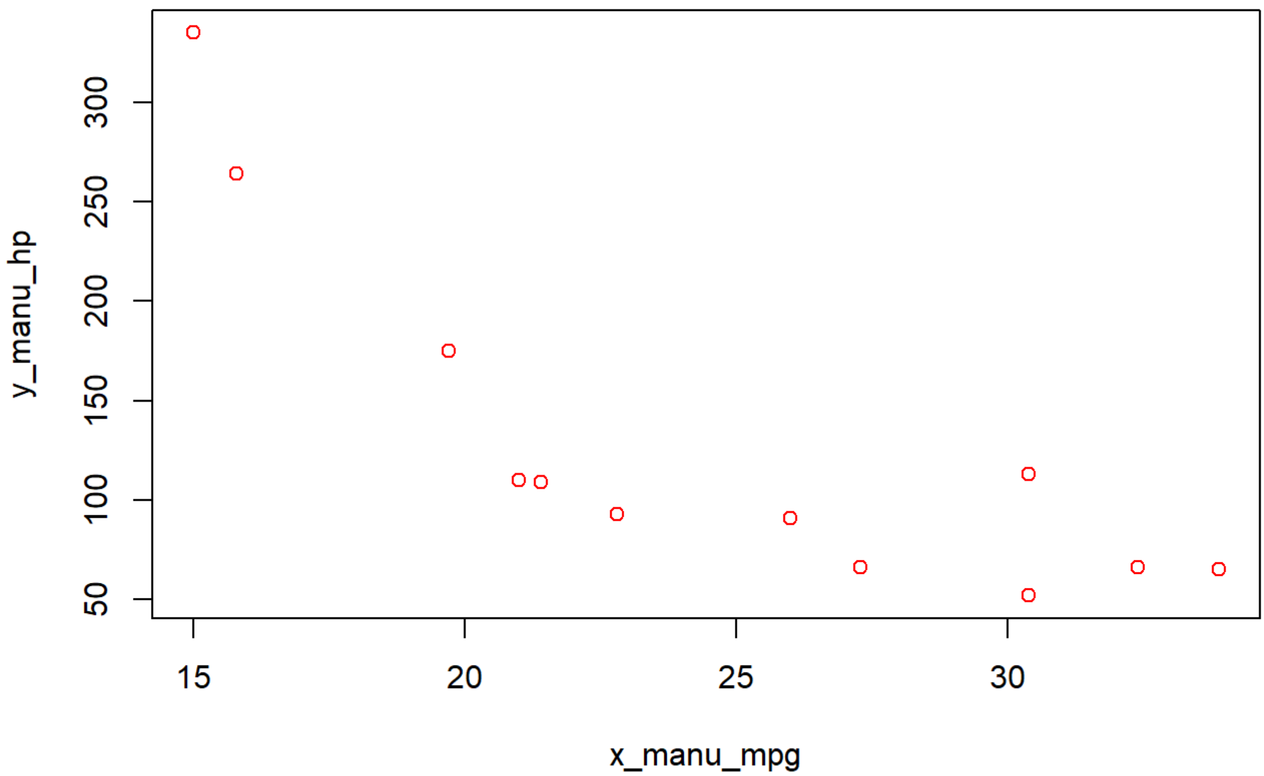
```
x_4cyl_mpg <- cyl_4$mpg
y_4cyl_hp <- cyl_4$hp
z_4cyl_wt <- cyl_4$wt
plot(x_4cyl_mpg, y_4cyl_hp, col = 'red')# no significant relationship
```



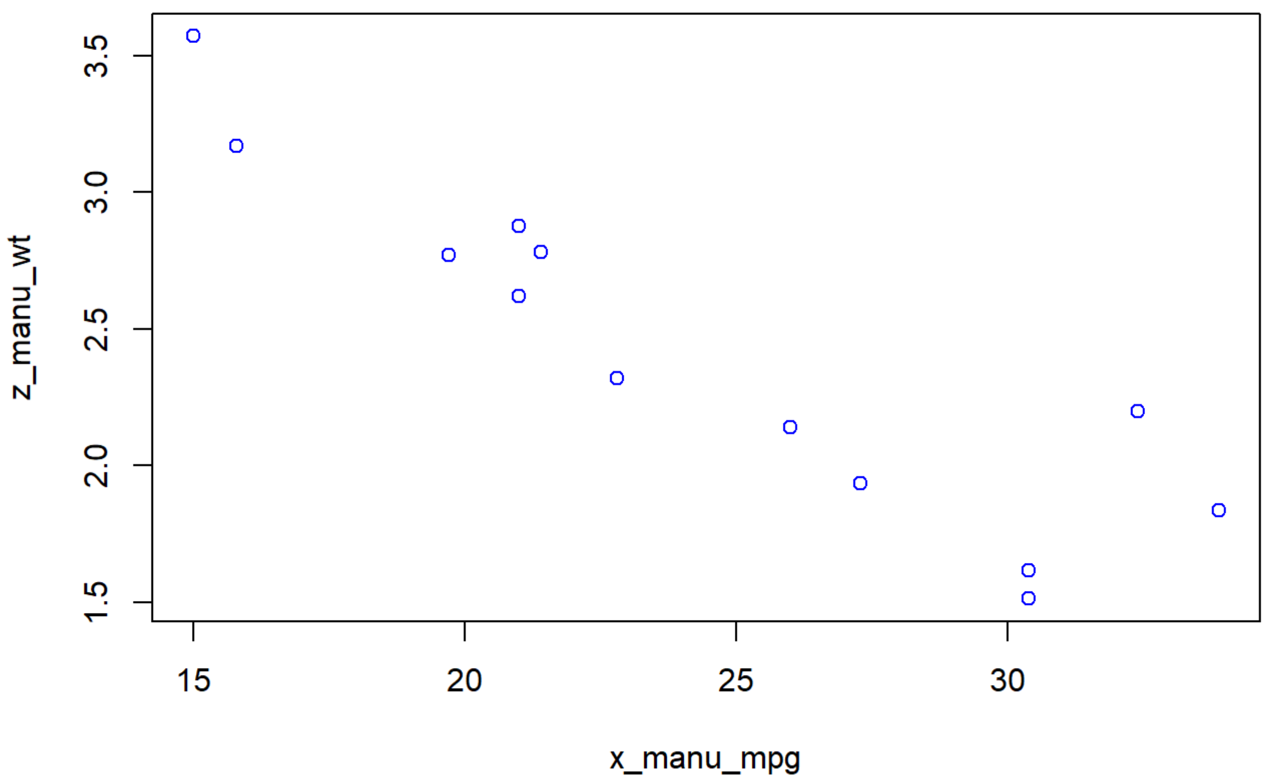

```
plot(x_4cyl_mpg, z_4cyl_wt, col = 'blue')#...
```



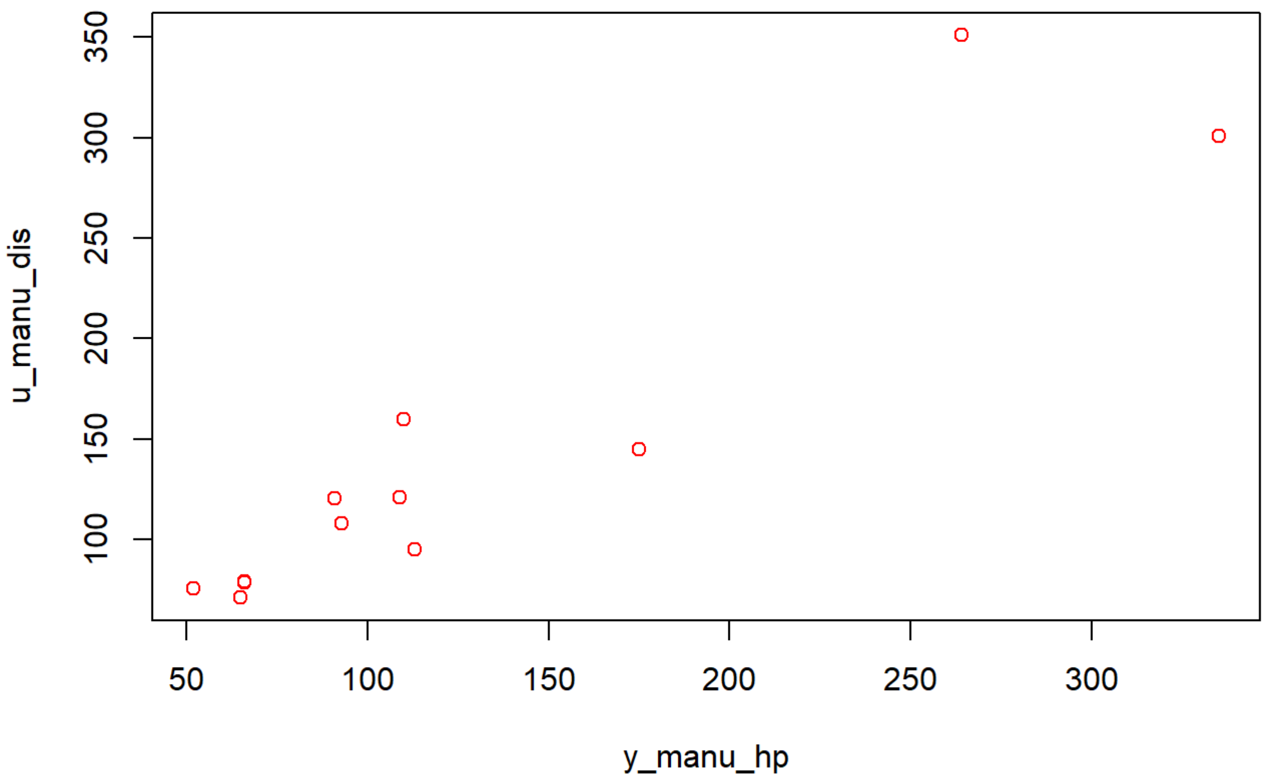

```
x_manu_mpg <- manual$mpg
y_manu_hp <- manual$hp
plot(x_manu_mpg, y_manu_hp, col = 'red')#somewhat negative relationship
```




```
z_manu_wt <- manual$wt
plot(x_manu_mpg, z_manu_wt, col = 'blue')# good negative relationship
```



```
u_manu_dis <- manual$disp
plot(y_manu_hp, u_manu_dis, col = 'red')
```



Correlation function

```
s_xx <- function(x){
  sum(x*x) - ((sum(x))^2) / length(x)
}

s_xy <-function(x,y){
  sum(x*y) - (sum(x)*sum(y))/length(x)
}
s_yy <- function(y){
  sum(y*y) - ((sum(y))^2) / length(y)
}

corr <- function(x,y){
  s_xy(x,y)/ (sqrt(s_xx(x))*sqrt(s_xx(y)))
}
corr_manual <- corr(x_manu_mpg, z_manu_wt)
corr_manual
```

```
## [1] -0.9089148
```

```
corr_horpwr_disp <- corr(y_manu_hp, u_manu_dis)
corr_horpwr_disp # since corr_horpwr_disp is 0.9240353 thus there is strong relationship
```

```
## [1] 0.9240353
```

Positive relationship between horse power and displacement Since the correlation coefficient, r is -.9089, it shows strong negative relationship between mpg and horser power of automobile among manual cars.

3.2 Choose two pairs of your choice that suggest strong linear relationship, for each of these pairs identify what the explanatory and response variables will be.

Either using the `lm' function or the functions, estimate the model parameters for the simple linear regression for both of these pairs.

```
linearModel_mpg_hp <- lm(mpg~hp, mtcars)
linearModel_mpg_hp
```

```
##
## Call:
## lm(formula = mpg ~ hp, data = mtcars)
##
## Coefficients:
## (Intercept)           hp
##    30.09886     -0.06823
```

```
linearModel_horpwr_disp <- lm(hp~disp, mtcars)
```

3.3 interpret the slope parameter estimates that you have computed appropriately. b_0 = 30.09886 and b_1 = -0.06823

```
linearModel_mpg_hp
```

```
##
## Call:
## lm(formula = mpg ~ hp, data = mtcars)
##
## Coefficients:
## (Intercept)           hp
##    30.09886     -0.06823
```

```
linearModel_horpwr_disp
```

```
##
## Call:
## lm(formula = hp ~ disp, data = mtcars)
##
## Coefficients:
## (Intercept)         disp
##     45.7345       0.4376
```

b_0 45.7345 and b_1 0.4376