```
library(tidyverse)
library(lubridate)
library(ggplot2)
library(readxl)
library(openxlsx)
library(dplyr)
library(caTools)
library(pROC)
library(repr)
library(car)
library(olsrr)

df = read_excel("bike.xlsx")
head(df,n=3)

sapply(df, class)

sapply(df, function(x) sum(is.na(x)))

colnames(df)

names(df)[2] <- "date"
names(df)[9] <- "weather"
names(df)[12] <- "humidity"
names(df)[16] <- "count"
colnames(df)

options(repr.plot.width=4, repr.plot.height=3)

ggplot(data=df)+geom_bar(mapping=aes(x=workingday), fill = 'red')

df_box <- df[,c(10,11,12,13)]
head(df_box,n=3)

options(repr.plot.width=6, repr.plot.height=6)

boxplot(df_box , main="Different boxplots",
        col="blue",border="black")

library("gridExtra")

options(repr.plot.width=14, repr.plot.height=6)
g1 <- ggplot(df, aes(x = casual)) + geom_histogram (fill="#69b3a2", color="#e9ecef" , binwidth = 400)
g2 <- ggplot(df, aes(x = registered)) + geom_histogram (fill = "pink", color="#e9ecef" , binwidth = 400)
g3 <- ggplot(df, aes(x = count)) + geom_histogram (fill = "blue", color="#e9ecef" , binwidth = 400)

grid.arrange(g1, g2,g3, ncol = 3)
```

```
p <- ggplot(df, aes(x=date, y=count)) +
        geom_line(color = "#00AFBB", size = 1)
p

df_line = df
head(df_line,3)

df_line$holiday[df_line$holiday == 0] <- "No Holiday"
df_line$holiday[df_line$holiday == 1] <- "Holiday"
head(df_line,3)

p1 <- ggplot(df_line, aes(x=date, y=count)) + geom_line(aes(color = holiday), size = 1) + scale_color_manual(values = c("#00AFBB",
"#E7B800"))+theme_minimal()
p1

df1 = df
head(df1,n=7)

df1$instant <- NULL
df1$date <- NULL
head(df1,n=2)

colnames(df1)

options(repr.plot.width=14, repr.plot.height=10)

library(reshape2)

# creating correlation matrix
corr_mat <- round(cor(df1),2)

# reduce the size of correlation matrix
melted_corr_mat <- melt(corr_mat)
head(melted_corr_mat)

# plotting the correlation heatmap
library(ggplot2)
ggplot(data = melted_corr_mat, aes(x=Var1, y=Var2,
                                    fill=value)) +
geom_tile() +
geom_text(aes(Var2, Var1, label = value),
          color = "white", size = 4)

df1 <- df1 %>% mutate_at(c('season' ,'yr', 'mnth', 'holiday', 'weekday', 'workingday', 'weather', 'temp', 'atemp', 'humidity', 'windspeed'
,'casual', 'registered'), ~(scale(.) %>% as.vector))
head(df1,n=7)
```

```
set.seed(123)

sample <- sample.split(df1$count, SplitRatio = 0.75)
train  <- subset(df1, sample == TRUE)
test   <- subset(df1, sample == FALSE)

head(train , 3)

head(test , 3)

dim(train)
dim(test)

#df2 <- train %>% mutate_at(c('season' ,'yr', 'mnth', 'holiday', 'weekday', 'workingday', 'weather', 'temp', 'atemp', 'humidity', 'windspeed'
,'casual', 'registered'), ~(scale(.) %>% as.vector))
#head(df2,n=7)
#dim(df2)

model <- lm(count ~., data = train)
summary(model)

model2 <- lm(count ~ casual + registered , data = train)
summary(model2)

pred <- predict(model2,test)

df_test = test
df_test$New_count = pred
head(df_test,n=7)

g1 = ggplot(df_test, aes(x = count, y = New_count)) +
    geom_point() + stat_smooth(method = "lm",
        col = "#C42126",
        se = FALSE,
        size = 1)
g1

plot(df$casual , df$count)
#lines(lowess(df$casual,df$count), col = "blue")
abline(lm(count~casual,data=df),col='red')

plot(df$registered , df$count)
#lines(lowess(df$casual,df$count), col = "blue")
abline(lm(count~registered,data=df),col='green' , lwd = 5)

vif(model2)

cor(df$registered, df$casual ,  method = "pearson", use = "complete.obs")
```

```
shapiro.test(model2$residuals)

hist(model2$residuals)

durbinWatsonTest(model2)

ols_test_score(model2)
```