

# MAS Workshop

Department of Statistics - USJ

2024-03-28

```
# load the packages

library(readxl)    # to load the data set
library(dplyr)     # select operator
library(ggplot2)   # to create plots
library(magrittr)  # pipe operator
library(car)       # to obtain vif value
library(DescTools) # to obtain the mode
```

## Descriptive Statistics

### Load the data

```
data_descriptive <- read_xlsx("Descriptive Statistics - Data.xlsx")
```

### Glimpse on the dataset

```
glimpse(data_descriptive)
```

```
Rows: 1,111
Columns: 10
$ Date          <dtm> 2023-01-01, 2023-01-01, 2023-01-01, 2023-01-01, 2~
$ `Customer Group` <chr> "SLICK CHICKS", "SLICK CHICKS", "SLICK CHICKS", "S~
$ Plant         <chr> "D051", "D051", "D051", "D051", "D051", "D051", "D~
$ `Product hierarchy` <chr> "Swim Bottom", "Swim Bottom", "Swim Bottom", "Swim~
$ Gender        <chr> "WOMEN", "WOMEN", "WOMEN", "WOMEN", "WOMEN", "WOME~
$ SMV           <dbl> 4.880, 4.880, 4.734, 4.734, 4.734, 4.734, 4.880, 4~
$ `Shipping Type` <chr> "Courier", "Courier", "Courier", "Courier", "Couri~
$ `Order Qty`    <dbl> 200, 200, 200, 200, 200, 200, 3, 197, 3, 197, 3, 1~
$ Earnings      <dbl> 509.2671800, 836.6231200, 812.6663206, 812.6663206~
$ `Std Hrs`     <dbl> 16.2666667, 16.2666667, 15.7800000, 15.7800000, 15~
```

slide number: 41

### one way frequency table

```
table(data_descriptive$`Shipping Type`)
```

Air Collect      Courier Sea Collect

87

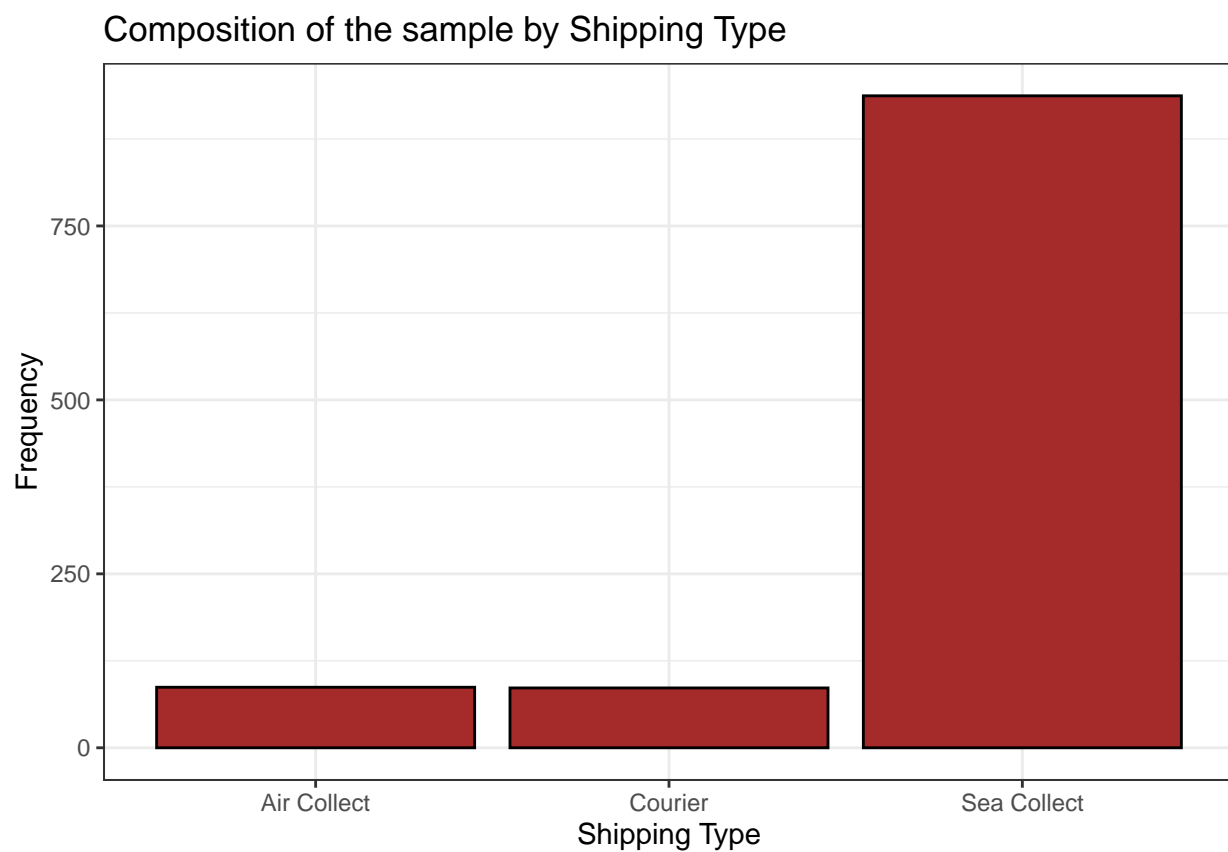
86

937

slide number: 42

## barchart

```
data_descriptive %>%  
  select(`Shipping Type`) %>%  
  na.omit() %>%  
  ggplot(aes(x = as.factor(`Shipping Type`))) +  
  geom_bar(color="black",  
           fill="brown" ) +  
  theme_bw() +  
  labs(title = "Composition of the sample by Shipping Type",  
       x = "Shipping Type",  
       y = "Frequency")
```

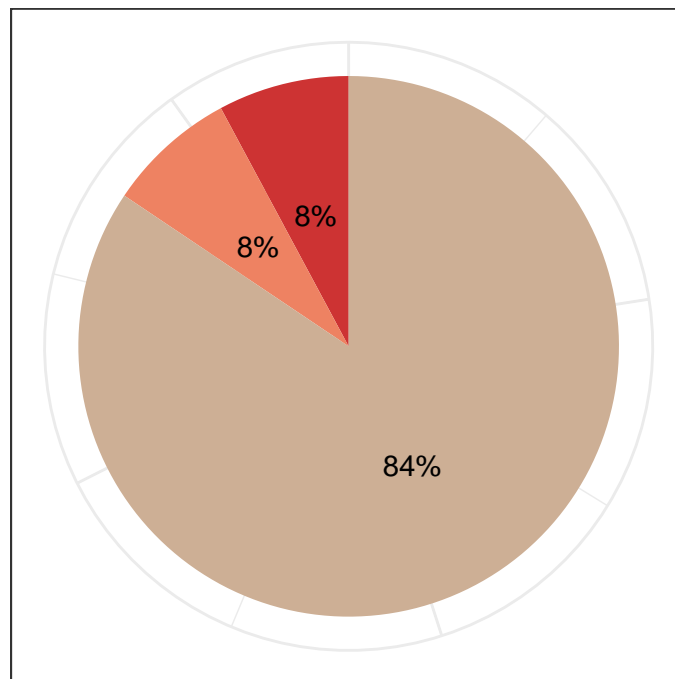


slide number: 43

## pie chart

```
data.frame(Shipping_Type = c("Air Collect", "Courier", "Sea Collect"),
  Frequency = c(87, 86, 937)) %>%
  ggplot(aes(x = "", y = Frequency,
    fill = Shipping_Type)) +
  geom_bar(stat="identity", width=1) +
  coord_polar("y", start=0) +
  geom_text(aes(label = paste0(
    round((Frequency/sum(Frequency))*100), "%"),
    position = position_stack(vjust = 0.5)) +
  theme_bw() +
  scale_fill_manual(values=c("brown3", "salmon2", "peachpuff3")) +
  labs(x = NULL, y = NULL,
    fill = "Shipping Type",
    title = "Composition of the sample by Shipping Type") +
  theme(axis.line = element_blank(),
    axis.text = element_blank(),
    axis.ticks = element_blank(),
    plot.title = element_text(hjust = 0.5)) +
  theme(legend.position = "bottom")
```

Composition of the sample by Shipping Type



Shipping Type ■ Air Collect ■ Courier ■ Sea Collect

slide number: 44

## summary measures

```
# mean  
mean(data_descriptive$SMV,  
      na.rm = TRUE)
```

```
[1] 10.39356
```

```
# median  
median(data_descriptive$SMV,  
        na.rm = TRUE)
```

```
[1] 9.741
```

```
# mode  
Mode(data_descriptive$SMV,  
      na.rm = TRUE)
```

```
[1] 14.088
```

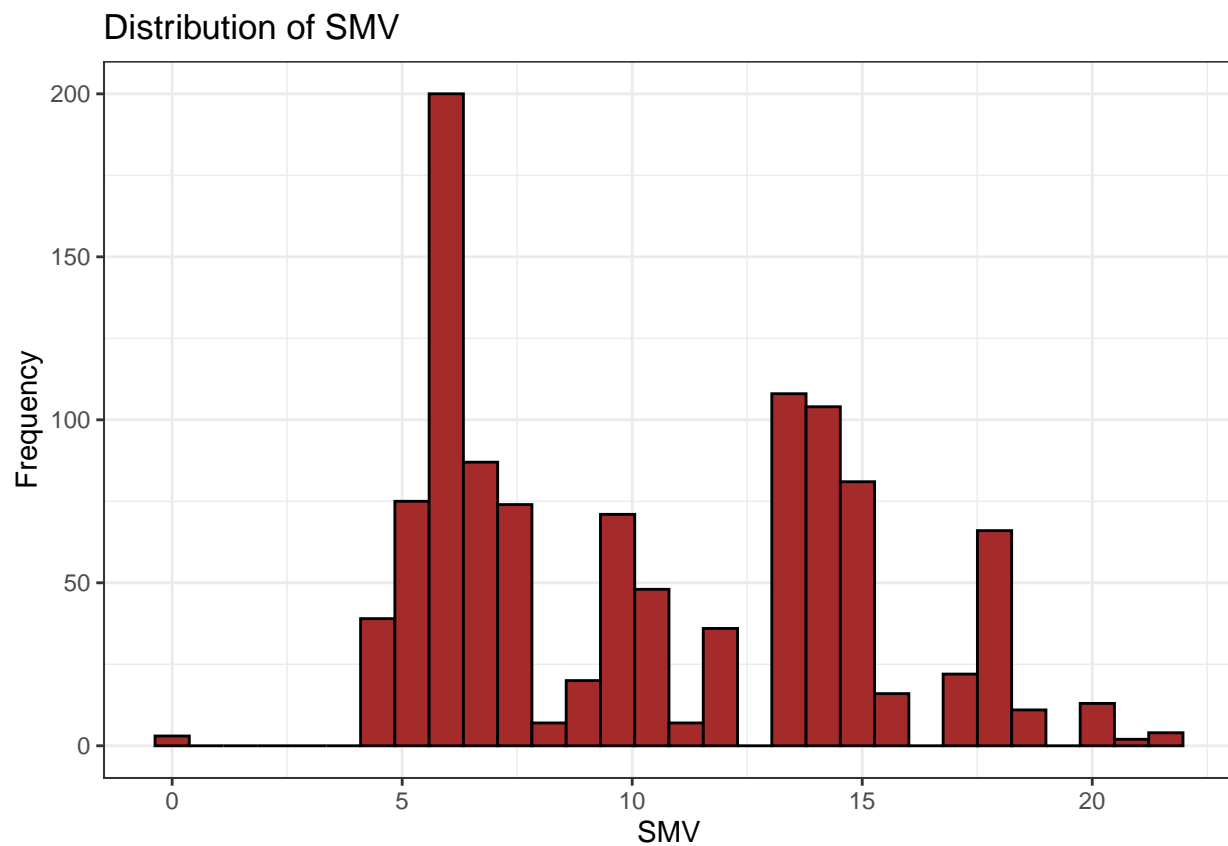
```
attr("freq")
```

```
[1] 71
```

slide number: 45

## histogram - asymmetric

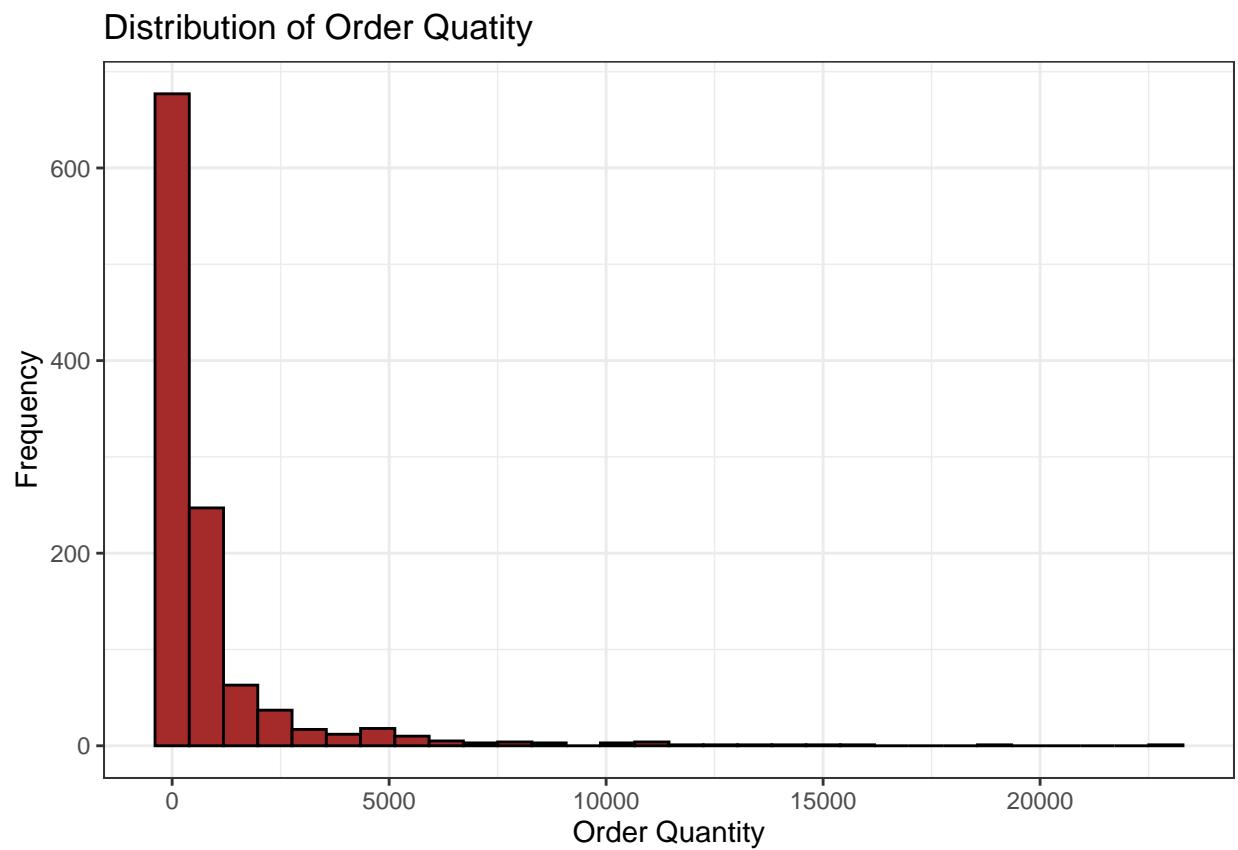
```
data_descriptive %>%  
  select(SMV) %>%  
  na.omit() %>%  
  ggplot(aes(x = SMV)) +  
    geom_histogram(color = "black",  
                  fill = "brown") +  
  theme_bw() +  
  labs(title = "Distribution of SMV",  
       x = "SMV",  
       y = "Frequency")
```



slide number: 46

## histogram - positively skewed

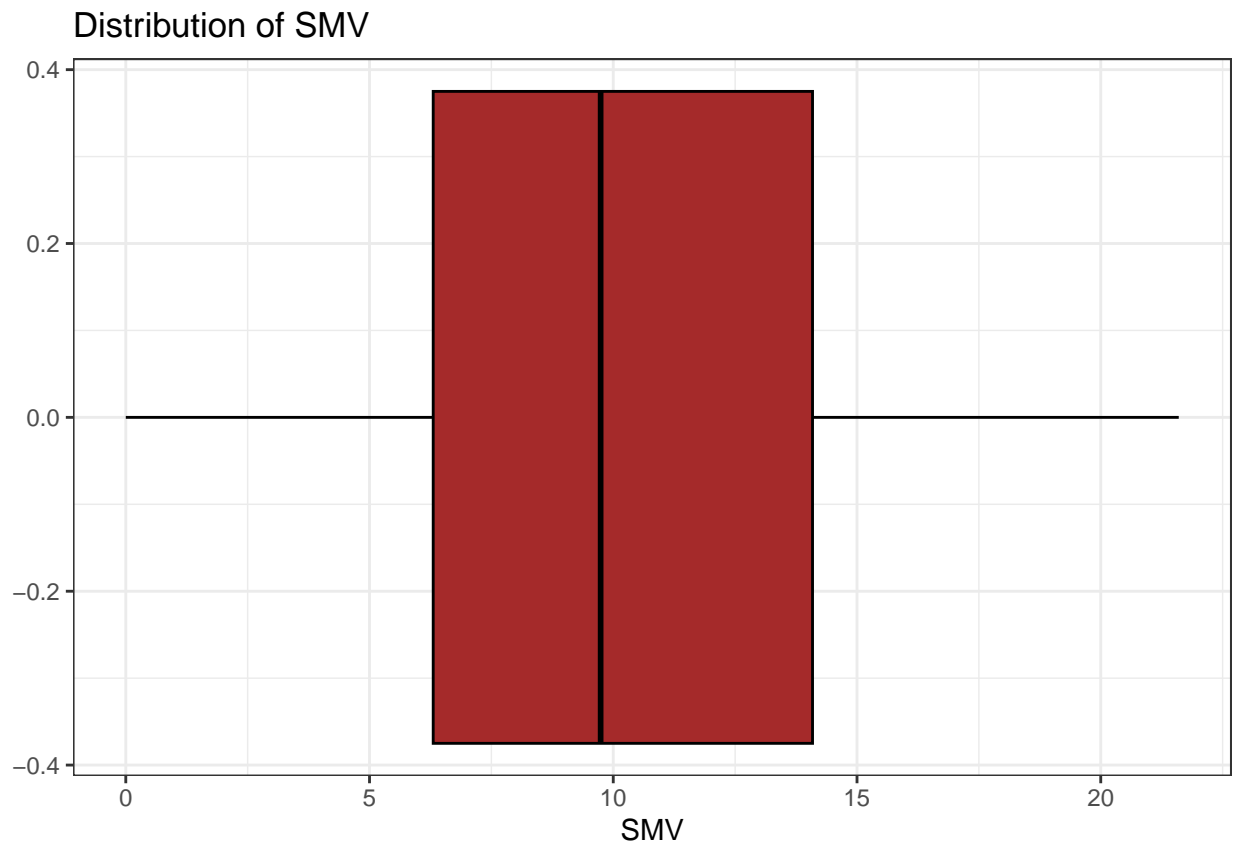
```
data_descriptive %>%  
  select(`Order Qty`) %>%  
  na.omit() %>%  
  ggplot(aes(x = `Order Qty`)) +  
    geom_histogram(color = "black",  
                  fill = "brown") +  
  theme_bw() +  
  labs(title = "Distribution of Order Quatity",  
       x = "Order Quantity",  
       y = "Frequency")
```



slide number: 47

## boxplot

```
data_descriptive %>%  
  select(SMV) %>%  
  na.omit() %>%  
  ggplot(aes(x = SMV)) +  
    geom_boxplot(color = "black",  
                 fill = "brown" ) +  
  theme_bw() +  
  labs(title = "Distribution of SMV")
```





slide number: 51

two way frequency table

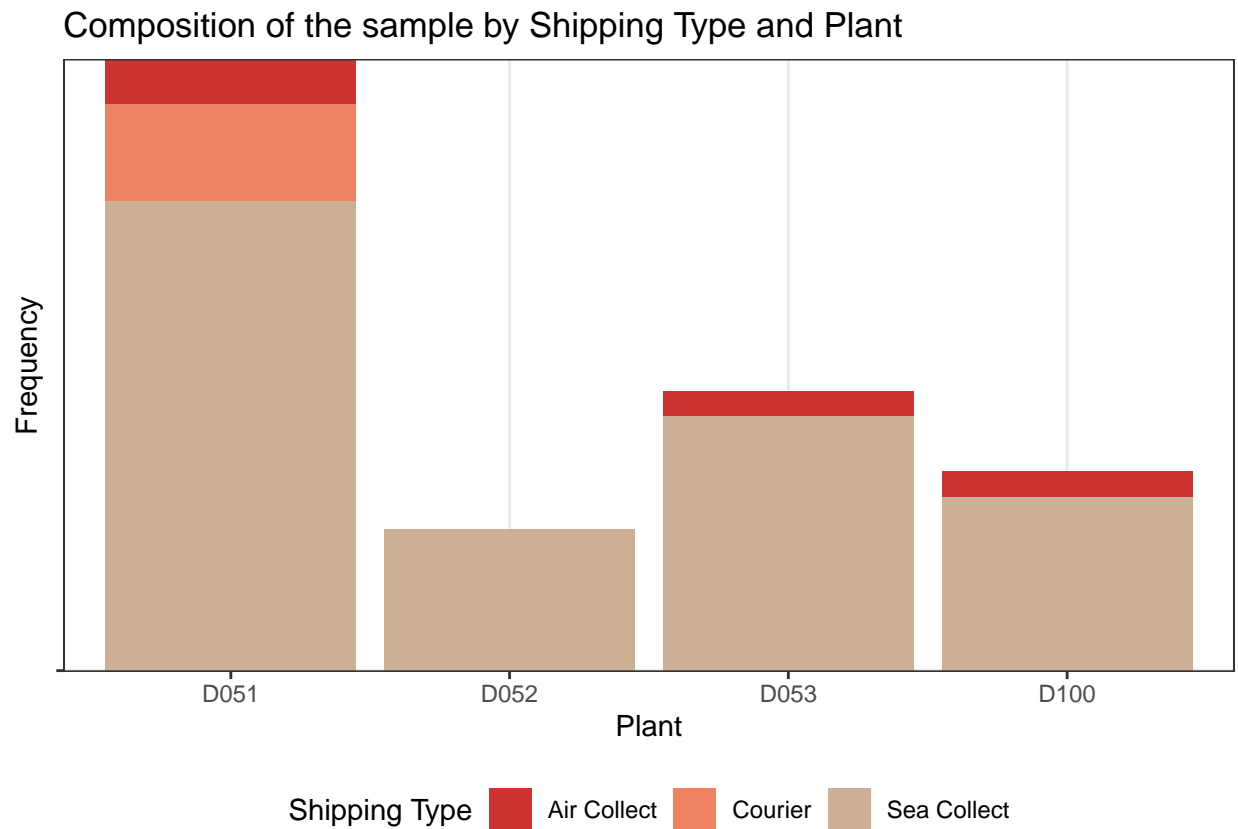
```
table(data_descriptive$`Shipping Type`,  
       data_descriptive$Plant)
```

	D051	D052	D053	D100
Air Collect	40	0	24	23
Courier	86	0	0	0
Sea Collect	422	126	235	154

slide number: 52

## stacked bar chart

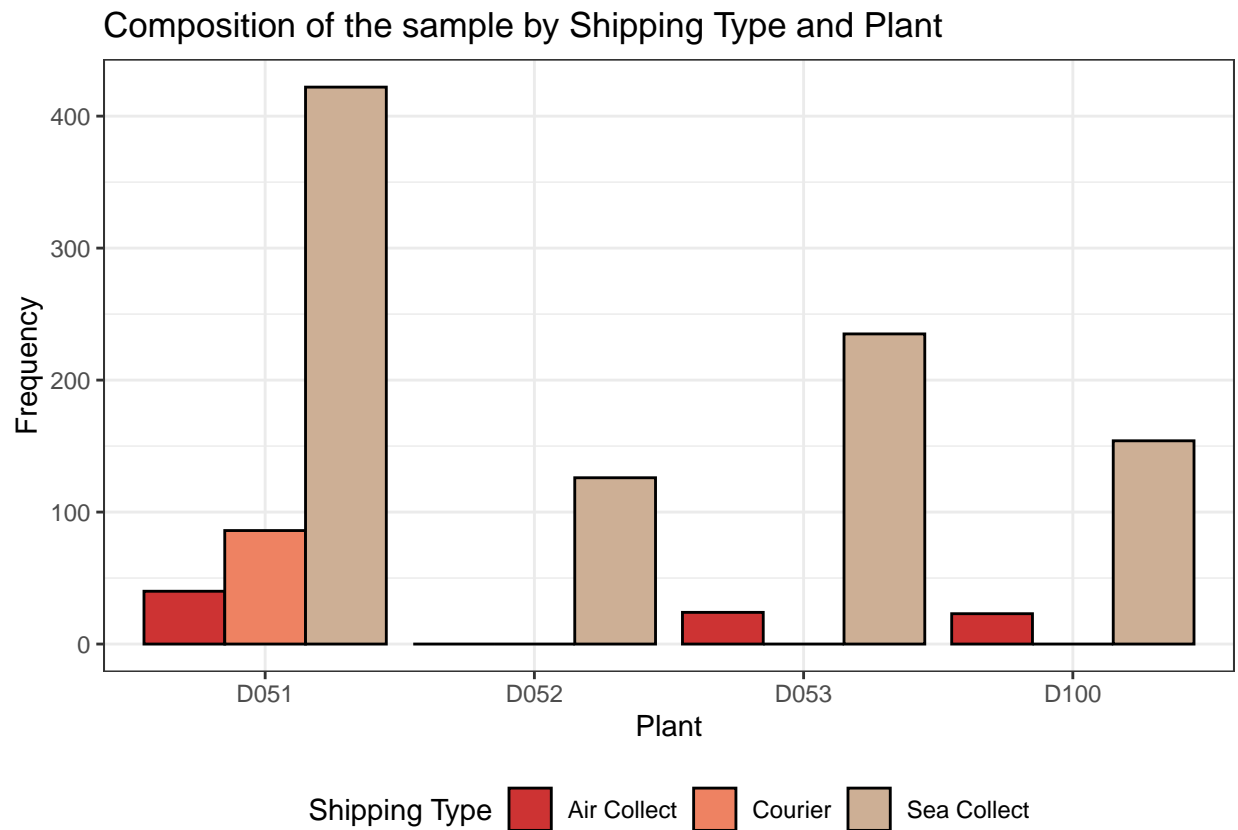
```
data_descriptive %>%  
  na.omit() %>%  
  ggplot(aes(x = Plant, y = "", fill = `Shipping Type`)) +  
  geom_bar(stat = "identity") +  
  theme_bw() +  
  scale_fill_manual(values = c("brown3", "salmon2", "peachpuff3")) +  
  labs(x = "Plant",  
       y = "Frequency",  
       fill = "Shipping Type",  
       title = "Composition of the sample by Shipping Type and Plant") +  
  theme(legend.position = "bottom")
```



slide number: 53

## cluster bar chart

```
table(data_descriptive$`Shipping Type`,
      data_descriptive$Plant) %>%
  as.data.frame() %>%
  ggplot(aes(x = Var2, y = Freq, fill = Var1)) +
  geom_bar(stat = "identity", position = position_dodge(), col = "black") +
  theme_bw() +
  scale_fill_manual(values=c("brown3", "salmon2", "peachpuff3")) +
  labs(x = "Plant",
       y = "Frequency",
       fill = "Shipping Type",
       title = "Composition of the sample by Shipping Type and Plant") +
  theme(legend.position = "bottom")
```



slide number: 54

## scatterplot - positive linear

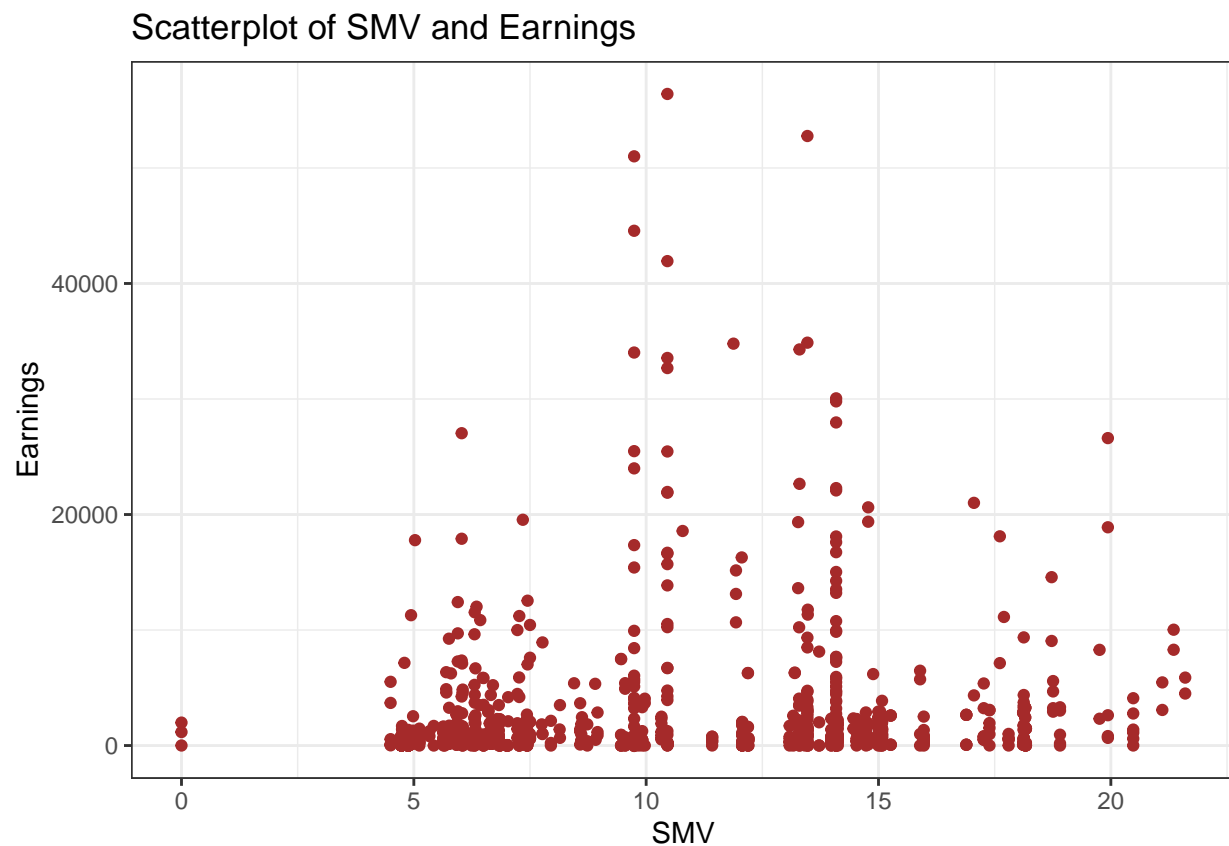
```
data_descriptive %>%  
  ggplot(aes(x = `Order Qty`,  
             y = Earnings)) +  
  geom_point(col = "brown") +  
  theme_bw() +  
  labs(title = "Scatterplot of Order Quantity and Earnings",  
       x = "Order Quantity",  
       y = "Earnings")
```



slide number: 55

scatterplot - no linear

```
data_descriptive %>%  
  na.omit() %>%  
  ggplot(aes(x = SMV,  
             y = Earnings)) +  
  geom_point(col = "brown") +  
  theme_bw() +  
  labs(title = "Scatterplot of SMV and Earnings",  
       x = "SMV",  
       y = "Earnings")
```



slide number: 56

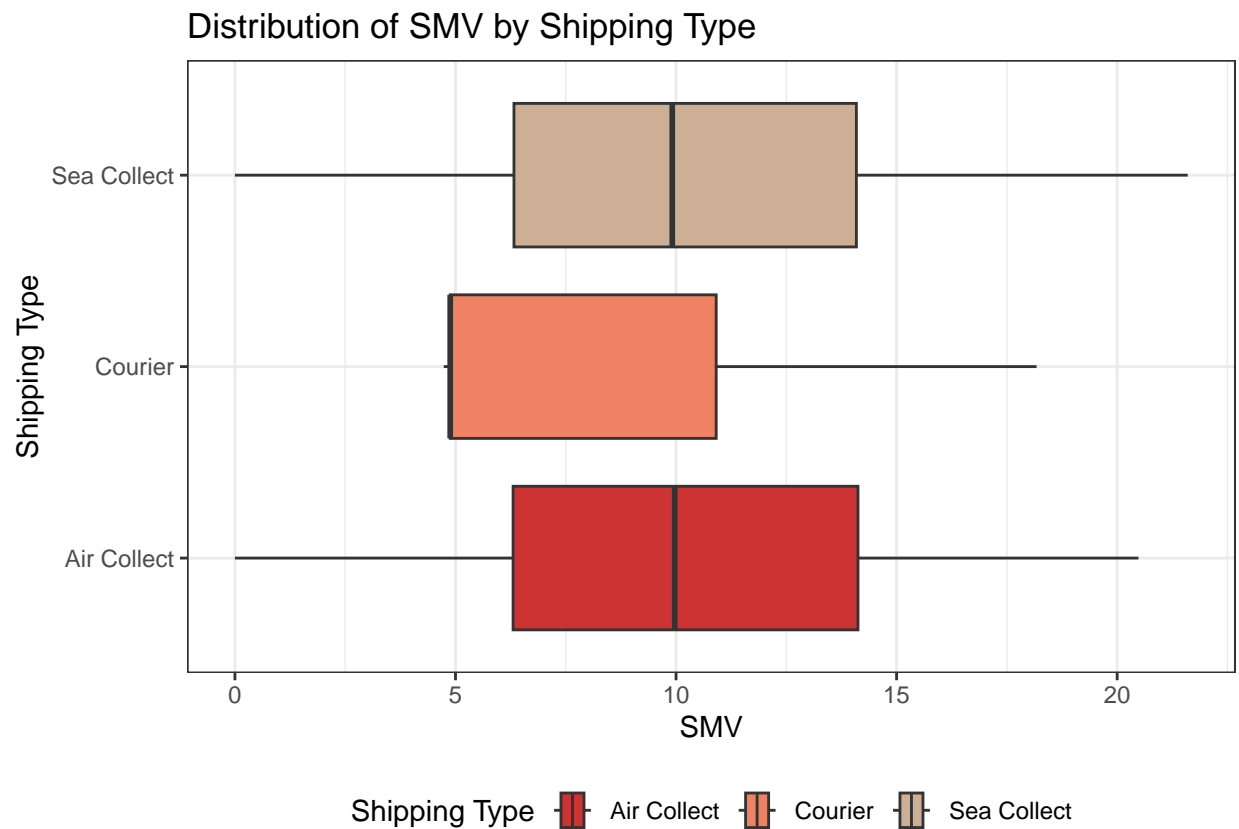
boxplot with groups

```
data_descriptive %>%  
  select(SMV,  
         `Shipping Type`) %>%  
  na.omit() %>%  
  ggplot(aes(x = SMV,  
             y = `Shipping Type`,
```

```

    fill = `Shipping Type`)) +
  geom_boxplot() +
  theme_bw() +
  scale_fill_manual(values=c("brown3", "salmon2", "peachpuff3")) +
  labs(x = "SMV",
       y = "Shipping Type",
       fill = "Shipping Type",
       title = "Distribution of SMV by Shipping Type") +
  theme(legend.position = "bottom")

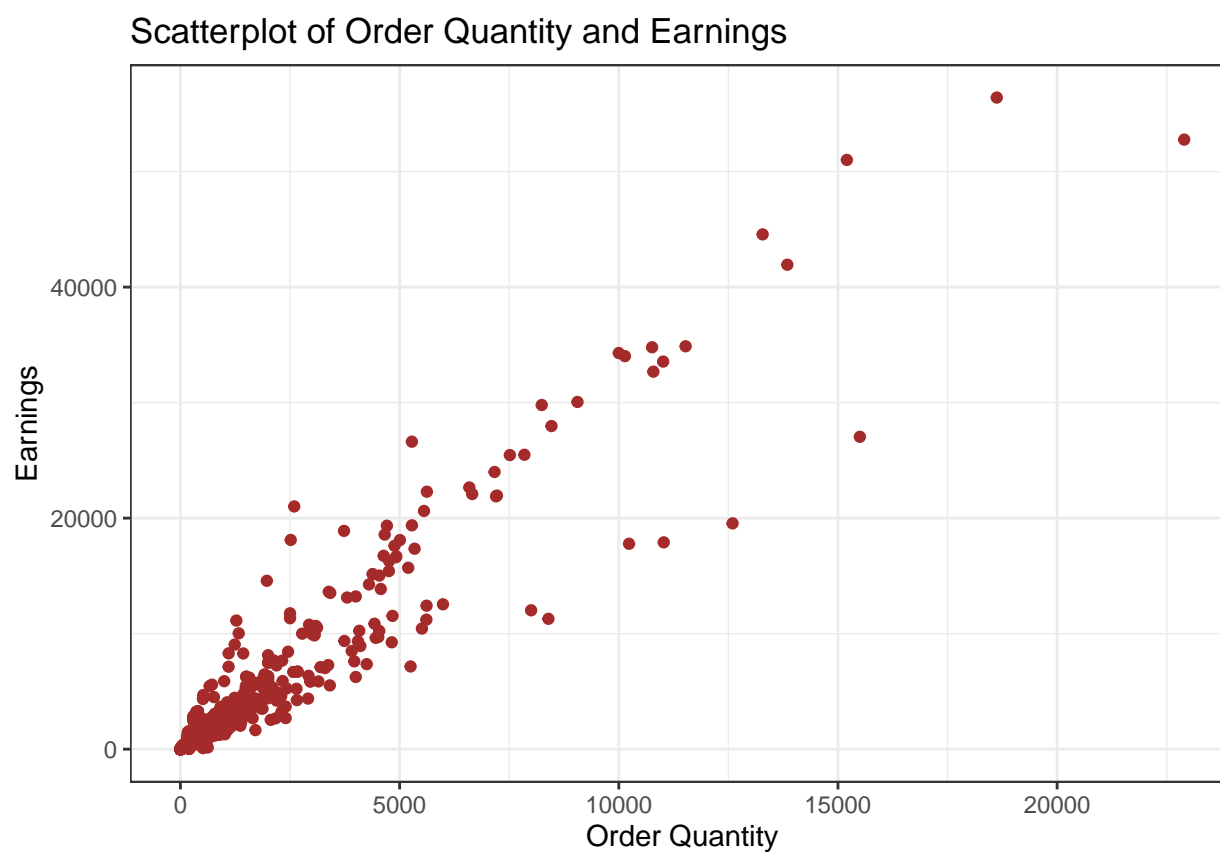
```



## Correlation Analysis

slide number: 63

```
# scatterplot - positive linear
data_descriptive %>%
  na.omit() %>%
  ggplot(aes(x = `Order Qty`,
             y = Earnings)) +
  geom_point(col = "brown") +
  theme_bw() +
  labs(title = "Scatterplot of Order Quantity and Earnings",
       x = "Order Quantity",
       y = "Earnings")
```



```
# positive linear relationship

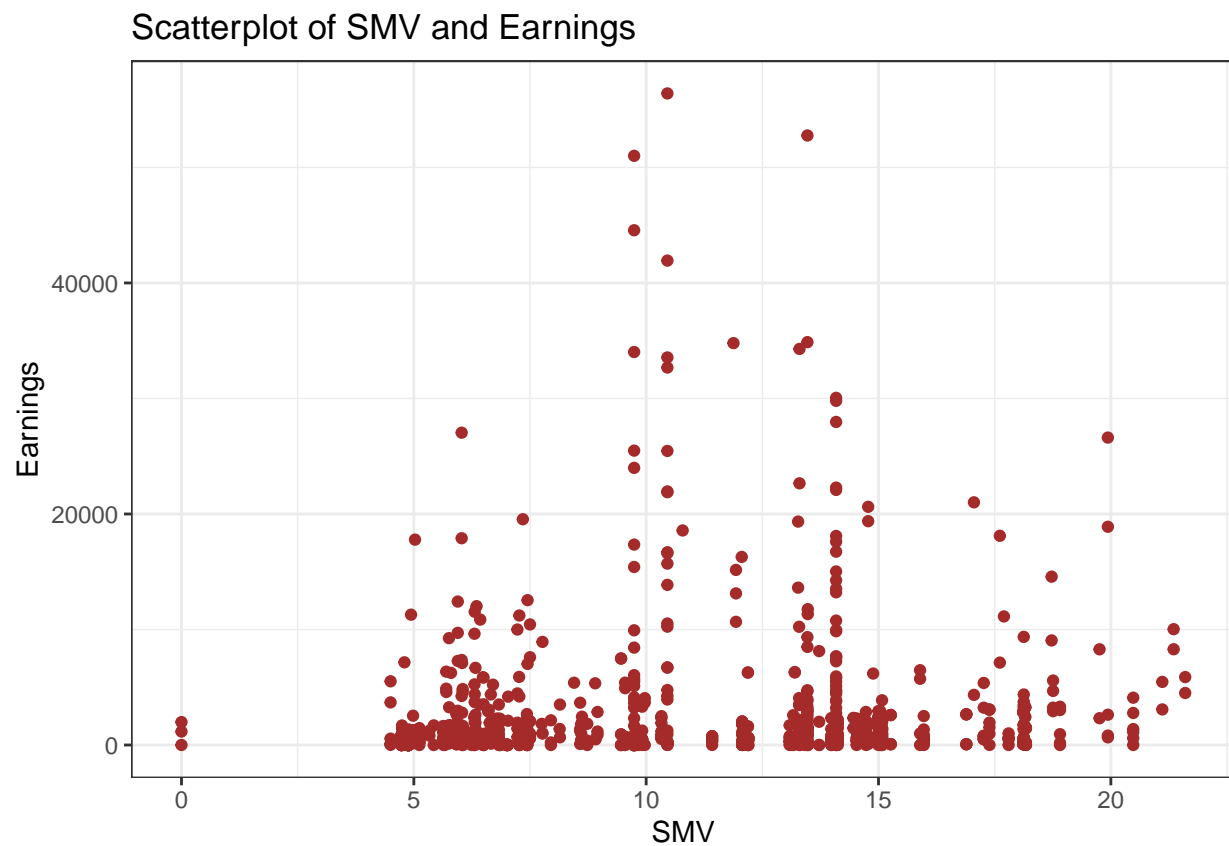
# correlation value
cor(x = data_descriptive$`Order Qty`,
    y = data_descriptive$Earnings)
```

```
[1] 0.9371457
```

```
#  $r = 0.9371457$ 
```

## slide number: 64

```
# scatterplot - no linear
data_descriptive %>%
  na.omit() %>%
  ggplot(aes(x = SMV,
             y = Earnings)) +
  geom_point(col = "brown") +
  theme_bw() +
  labs(title = "Scatterplot of SMV and Earnings",
       x = "SMV",
       y = "Earnings")
```



```
# no linear relationship

# correlation value
cor(data_descriptive$SMV,
    data_descriptive$Earnings,
    use = "complete.obs")
```

```
[1] 0.1142413
```

```
# 0.1142413
```



# Hypothesis Testing

## One sample test for mean - Slide no 61

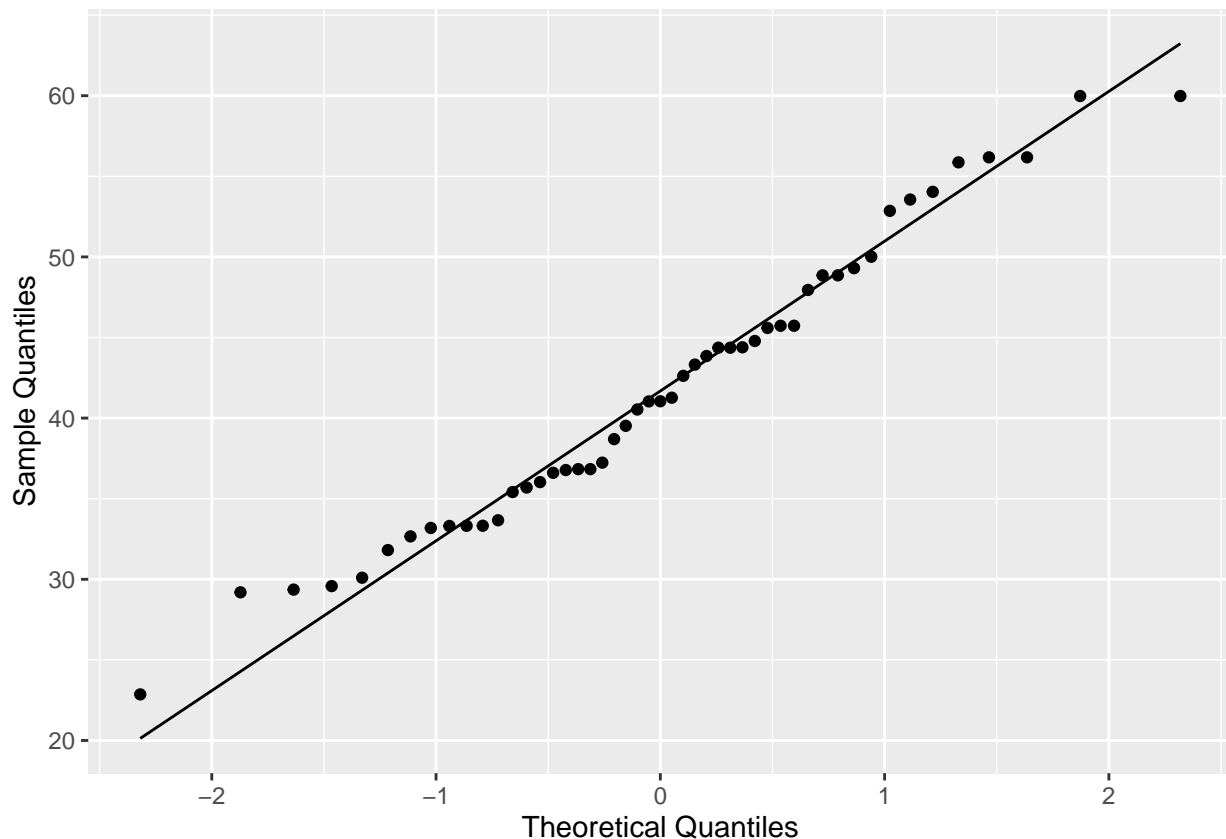
Example: Suppose we want to test whether the mean earnings per hour for Outer Known customer group is less than 50 at 5% significance level.

```
# loading dataset  
Hypothesis.data <- read_excel("Hypothesis Data.xlsx")
```

Step 1: Check whether Earnings per hour values are normally distributed

Normal probability plot

```
ggplot(Hypothesis.data, aes(sample = Earnings.per.hour)) + stat_qq() +  
  stat_qq_line() +  
  labs(x = "Theoretical Quantiles", y = "Sample Quantiles")
```



Normality test

```
shapiro.test(Hypothesis.data$Earnings.per.hour)
```

Shapiro-Wilk normality test

data: Hypothesis.data\$Earnings.per.hour  
W = 0.97508, p-value = 0.3806

Hypothesis to be tested:

H0: Data are normally distributed.

H1: Data are not normally distributed.

According to the Shapiro-Wilk normality test  $p\text{-value} = 0.3806 > 0.05$ .

Hence, We can conclude that Earnings per hour values are normally distributed.

## Step 2: Perform the t-test

```
t.test(Hypothesis.data$Earnings.per.hour, alternative = "less", mu = 50)
```

One Sample t-test

```
data: Hypothesis.data$Earnings.per.hour
t = -6.5365, df = 48, p-value = 1.89e-08
alternative hypothesis: true mean is less than 50
95 percent confidence interval:
 -Inf 43.84388
sample estimates:
mean of x
 41.71904
```

Since  $p\text{-value} = 1.89e-08 < 0.05$ , we reject null hypothesis.

Hence, there is sufficient evidence to suggest that the mean earnings per hour for the Outer Known customer group is less than 50.

## Two sample test for comparison between means - Slide no 63

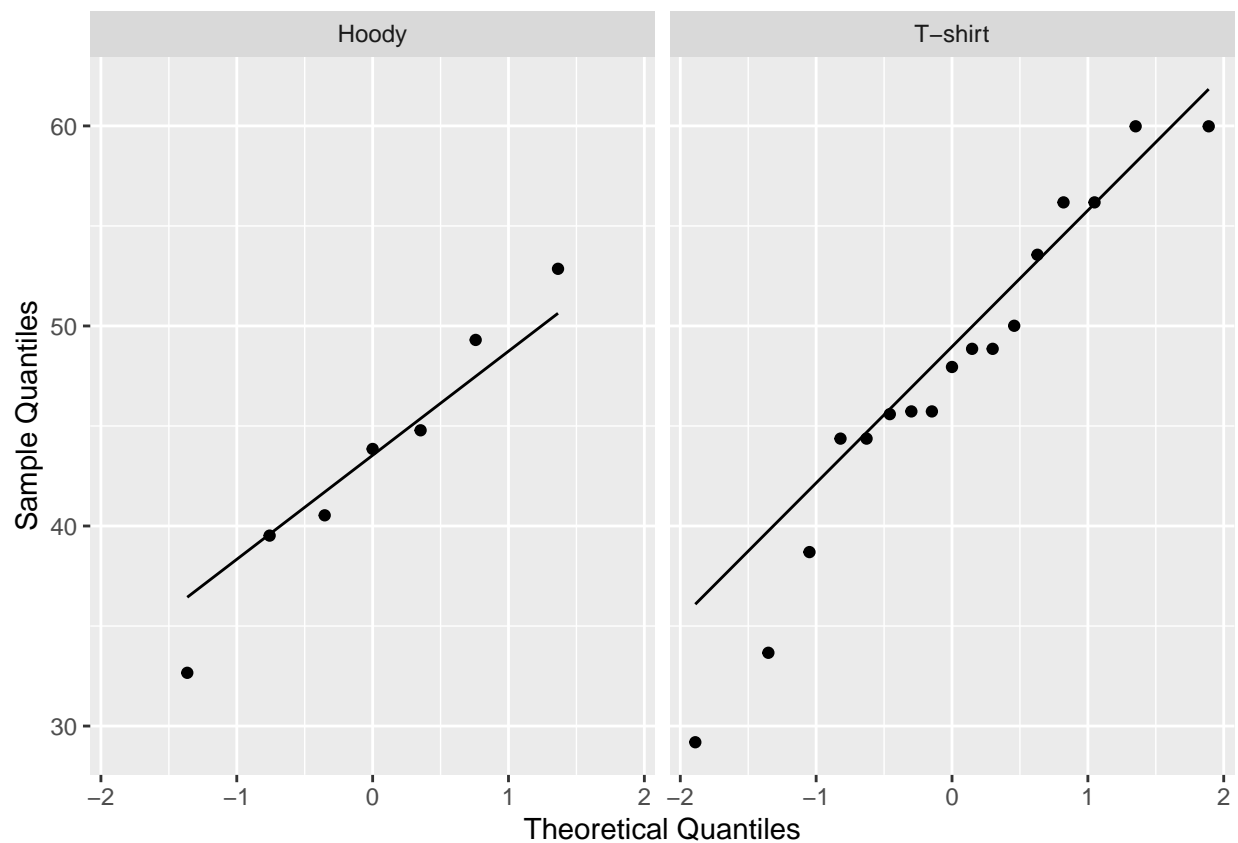
Example: Suppose we want test whether there is a significant difference in earnings per hour between Hoody products and T-shirt products of Outer Known customer group at 5% significance level.

```
# Loading relevant data
two.sample.data <- Hypothesis.data %>% filter(Product.name %in% c("Hoody", "T-shirt"))
```

### Step 1: Check whether Earnings per hour values are normally distributed

Normal probability plot

```
ggplot(two.sample.data, aes(sample = Earnings.per.hour)) + stat_qq() +
  stat_qq_line() + facet_grid(.~Product.name) +
  labs(x = "Theoretical Quantiles", y = "Sample Quantiles")
```



### Normality test

```
test1 <- two.sample.data %>% filter(Product.name == "Hoody")
shapiro.test(test1$Earnings.per.hour)
```

#### Shapiro-Wilk normality test

```
data: test1$Earnings.per.hour
W = 0.98413, p-value = 0.9771
```

```
test2 <- two.sample.data %>% filter(Product.name == "T-shirt")
shapiro.test(test2$Earnings.per.hour)
```

#### Shapiro-Wilk normality test

```
data: test2$Earnings.per.hour
W = 0.94884, p-value = 0.4384
```

Hypothesis to be tested:

H0: Data are normally distributed.

H1: Data are not normally distributed.

According to the Shapiro-Wilk normality test both p-values > 0.05.

Hence, We can conclude that Earnings per hour values of the two categories are normally distributed.

## Step 2: Check for equality of variance

```
var.test(Earnings.per.hour ~ Product.name, data = two.sample.data,  
         alternative = "two.sided")
```

F test to compare two variances

```
data: Earnings.per.hour by Product.name  
F = 0.61833, num df = 6, denom df = 16, p-value = 0.5739  
alternative hypothesis: true ratio of variances is not equal to 1  
95 percent confidence interval:  
 0.1850935 3.2424317  
sample estimates:  
ratio of variances  
 0.6183291
```

Hypothesis to be tested:

H0: Two population variances are equal.

H1: Two population variances are not equal.

According to the F test both p-values = 0.5739 > 0.05.

Hence, We can conclude that Two population variances are equal.

## Step 3: Perform the t-test

```
t.test(Earnings.per.hour ~ Product.name, data = two.sample.data,  
       alternative = "two.sided", var.equal = TRUE)
```

Two Sample t-test

```
data: Earnings.per.hour by Product.name  
t = -1.1755, df = 22, p-value = 0.2524  
alternative hypothesis: true difference in means between group Hoody and group T-shirt is not equal to 0  
95 percent confidence interval:  
 -11.672730  3.227215  
sample estimates:  
 mean in group Hoody mean in group T-shirt  
    43.35860         47.58136
```

Since p-value = 0.2524 > 0.05, we do not reject null hypothesis.

Hence, there is sufficient evidence to conclude that there is a significant difference in earnings per hour between the two product types.

## Multiple linear regression analysis

Example:

Suppose we aim to identify the factors affecting earnings from the product sales in the apparel industry. To examine the relationship between selected variables and earnings, we will conduct a multiple regression analysis. For this analysis, we will utilize the variables Earnings, MOH Value, and Std Hrs. The response variable is earnings whereas MOH Value and Std Hrs are the predictor variables.

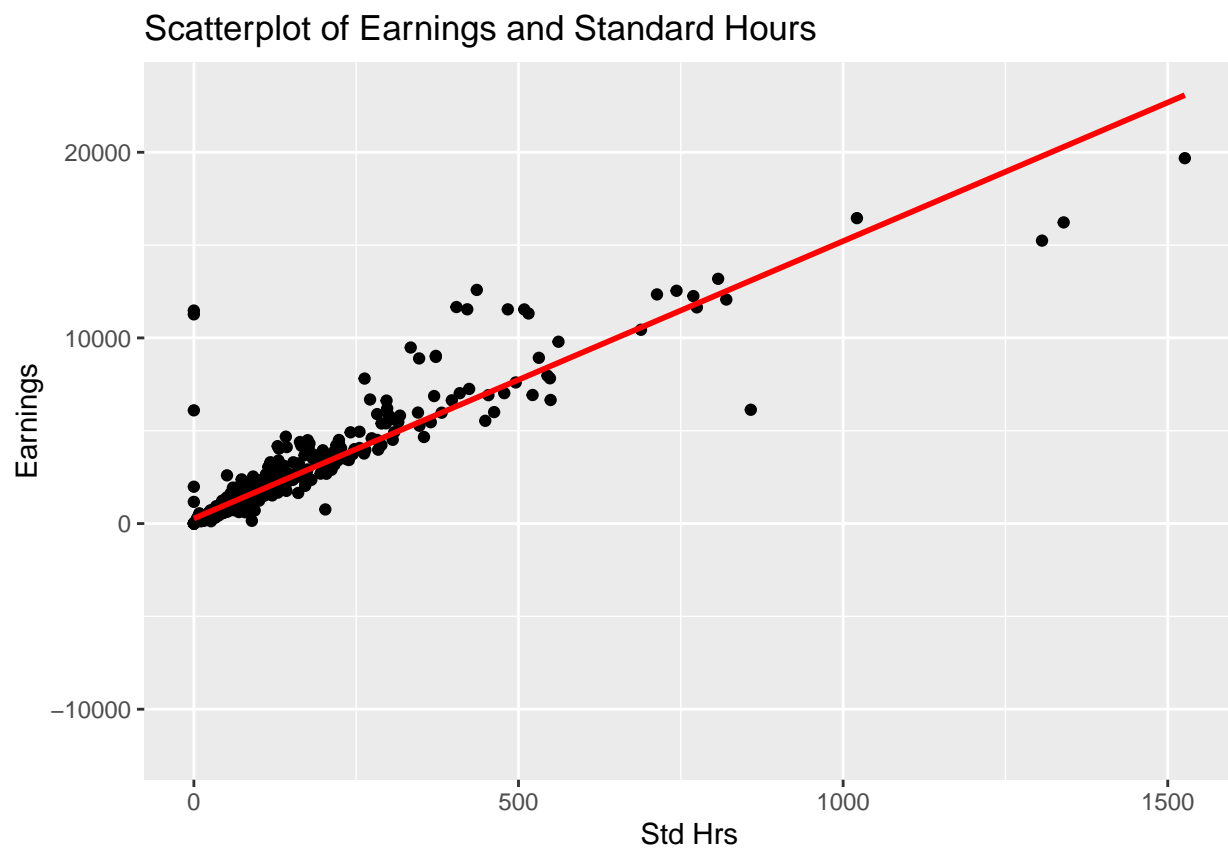
```
# load the data set
MAS_data_set <- read_excel("data set 2.xlsx")

# create a data set for regression analysis
Reg_data <- MAS_data_set %>%
  select(`MOH Value`, `Earnings`, `Std Hrs`)
```

Check the linearity assumption

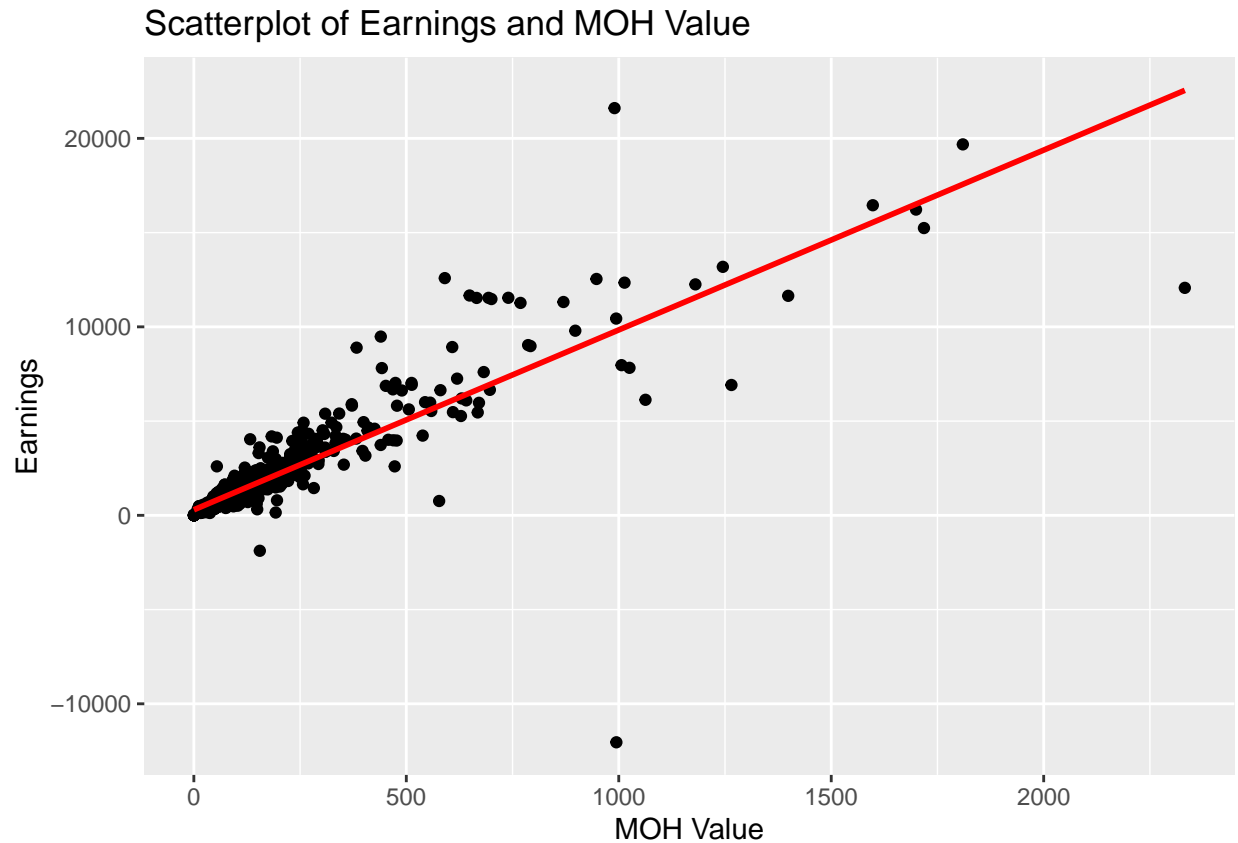
```
# scatter plot of Earnings and Standard Hours

ggplot(Reg_data, aes(x=`Std Hrs`, y= Earnings)) +
  geom_point() + geom_smooth(method = lm, se = FALSE, color = "red")+
  ggtitle("Scatterplot of Earnings and Standard Hours")
```



```
# scatter plot of Earnings and MOH Value

ggplot(Reg_data, aes(x=`MOH Value`, y= Earnings)) +
  geom_point() + geom_smooth(method = lm, se = FALSE, color = "red")+
  ggtitle("Scatterplot of Earnings and MOH Value")
```



## Fit the model

Call:  
`lm(formula = Earnings ~ `Std Hrs` + `MOH Value`, data = Reg_data)`

Residuals:

Min	1Q	Median	3Q	Max
-6847.3	-248.4	-157.2	60.3	7315.7

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	225.9173	44.2228	5.109	4.3e-07 ***
`Std Hrs`	6.8198	0.6583	10.359	< 2e-16 ***
`MOH Value`	5.6160	0.4265	13.167	< 2e-16 ***

---  
 Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 934.3 on 636 degrees of freedom  
 (8 observations deleted due to missingness)  
 Multiple R-squared: 0.8733, Adjusted R-squared: 0.8729  
 F-statistic: 2192 on 2 and 636 DF, p-value: < 2.2e-16

## Checking assumptions

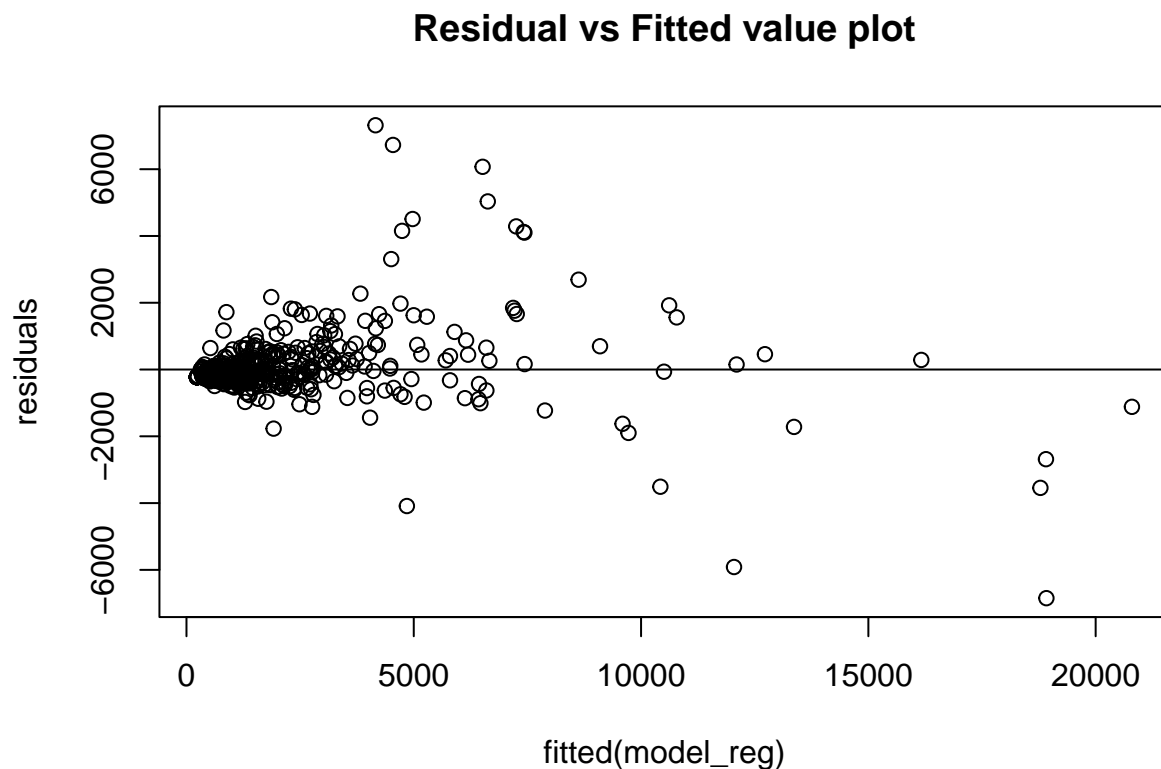
### Check the multicollinearity assumption

```
vif(model_reg) # Since VIF values are less than 10, we can avoid the multicollinearity
```

```
`Std Hrs`  `MOH Value`  
8.174816   8.174816
```

### Check the constant variance assumption of residuals

```
# obtain the residuals  
residuals <- resid(model_reg)  
  
# residual vs. fitted value plot  
plot(fitted(model_reg), residuals) + title("Residual vs Fitted value plot")  
  
integer(0)  
# add a horizontal line at 0  
abline(0,0)
```



### Check the normal assumption of residuals

```
# Q-Q plot for residuals  
qqnorm(residuals)
```

```
# add a straight diagonal line to the plot  
qqline(residuals)
```

