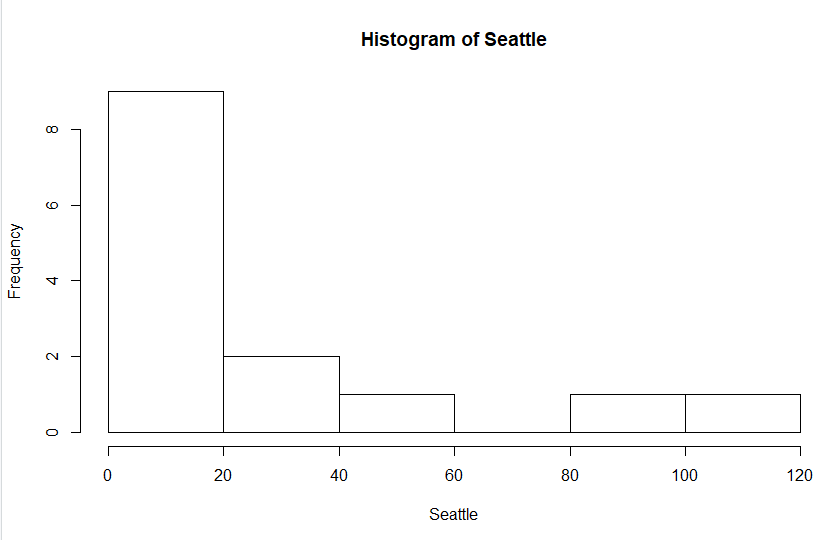
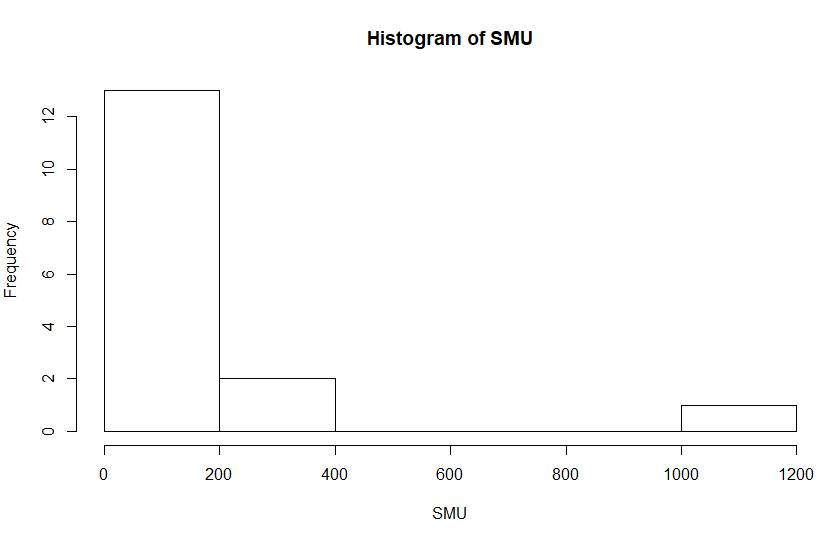
**Create histogram :**

*SMU = c(34, 1200, 23, 50, 60, 50, 0, 0, 30, 89, 0, 300, 400, 20, 10, 0)*

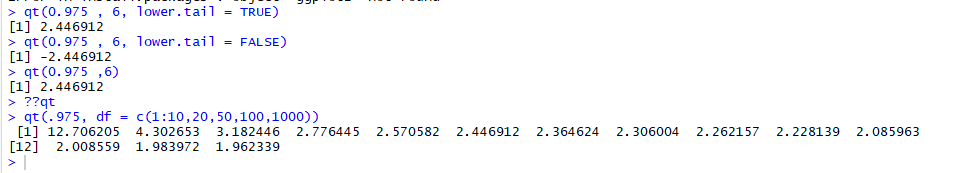
*Seattle = c(20, 10, 5, 0, 30, 50, 0, 100, 110, 0, 40, 10, 3, 0)*

*hist(SMU)*

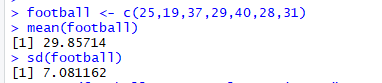
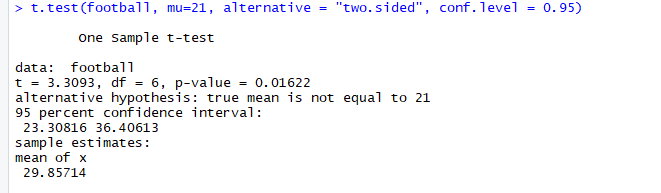
*hist(Seattle)*

**t-value for α = 0.05 and d.f. = 6**



**1 sample t-test ( 2-sided with 2 sided CI )**

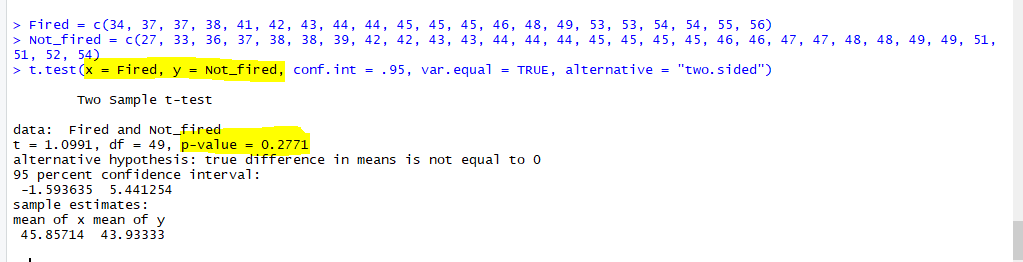
 

2 sided p-value = 0.0162

CI = [23.30816, 36.40613]

t-stat = 3.3093

**2 sample 2 sided t-test**



**t-test for data from a dataframe –**

|  |  |
| --- | --- |
| library(Sleuth2)  creativity <- case0101  qt(0.995, 45, lower.tail=T)  t.test(Score ~ Treatment, data=creativity , alternative='two.sided', conf.level=0.99, mu=0, var.equal=T) |  |

**arrays –**

|  |  |
| --- | --- |
| weather <- rep('Sunny',8) |  |
| happy <- c(80,71,92,93,78,82,12,20)  sunny <- data.frame(weather,happy) |  |
| names(cloudy) <- c("weather","happy")  *--we need to change header/column names before doing a rbind )* |  |
| weather.final <- rbind(sunny,cloudy) |  |
| qqnorm is a generic function the default method of which produces a normal QQ plot of the values in y. qqlineadds a line to a “theoretical”, by default normal, quantile-quantile plot which passes through the probs quantiles, by default the first and third quartiles. | |
| qqnorm(weather.final$happy[weather.final$weather=="Sunny"], main="Sunny")  qqline(weather.final$happy[weather.final$weather=="Sunny"])  alternative approach :  qqnorm(subset(cloud,Treatment=="Seeded")$Rainfall) |  |
| hist(weather.final$happy[weather.final$weather=="Sunny"] , xlab='Happiness' , main="Sunny")  box() |  |
| Side by side box plot | |
| boxplot(happy ~ weather , data=weather.final , horizontal=T, main = 'boxplots of happiness score ,by weather') |  |
| boxplot(happy ~ weather , data=weather.final , horizontal=F, main = 'boxplots of happiness score ,by weather') |  |
| If you want to add related diagrams to the same screen | |
| par(mfrow=c(2,2))  > hist(subset(cloud, Treatment=='Unseeded')$Rainfall, main='Unseeded', xlab='Rainfall')  > hist(subset(cloud, Treatment=='Seeded')$Rainfall, main='Seeded', xlab='Rainfall')  > qqnorm(subset(cloud, Treatment=='Unseeded')$Rainfall)  > qqnorm(subset(cloud, Treatment=='Seeded')$Rainfall) |  |
|  |  |

**Rank sum test**

|  |  |
| --- | --- |
| method <- c(rep('New',4),rep('Trad',3))  scores <- c(37,49,55,77,23,31,46)  teafching <- data.frame(method,scores) |  |
| wilcox.test(scores ~ method , data=teafching, alternative='greater') |  |
| wilcox.test(scores ~ method , data=teafching, alternative='greater',exact=F)  default is exact=T |  |
| wilcox.test(scores ~ method , data=teafching, alternative='greater',conf.int=T , conf.level=0.9) |  |

**Stem and leaf plots – Rank Sum test**

|  |  |
| --- | --- |
| > library(Sleuth2)  > cognitive <- case0402  str(cognitive) |  |
| stem(subset(cognitive, Treatmt=="Conventional", Time, drop=TRUE)) |  |
| stem(subset(cognitive, Treatmt=="Modified", Time, drop=TRUE)) |  |
| wilcox.test(Time ~ Treatmt, cognitive, alternative='less') |  |

**Exact binomial test**

|  |  |
| --- | --- |
| > site1 <- c(51, 39, 35, 17, 11, 14, 24, 37, 35)  > site2 <- c(38, 19, 23, 19, 7, 16, 14, 38, 24)  > binom.test(6, 9, alternative='greater') |  |

**Signed rank test**

|  |  |
| --- | --- |
| wilcox.test(site1, site2, paired=T, alternative='greater') |  |

**Paired test**

|  |  |
| --- | --- |
| t.test(col1 , col2 , paired=T,alternative=’greater’) |  |

**Scatter plot**

plot(baseball$Wins, baseball$Payroll, xlab = 'Number of Wins' , ylab = 'Payroll')