

Assignment 1

Tanjina Islam and Miguel Morales Expósito, group 12

14 February 2018

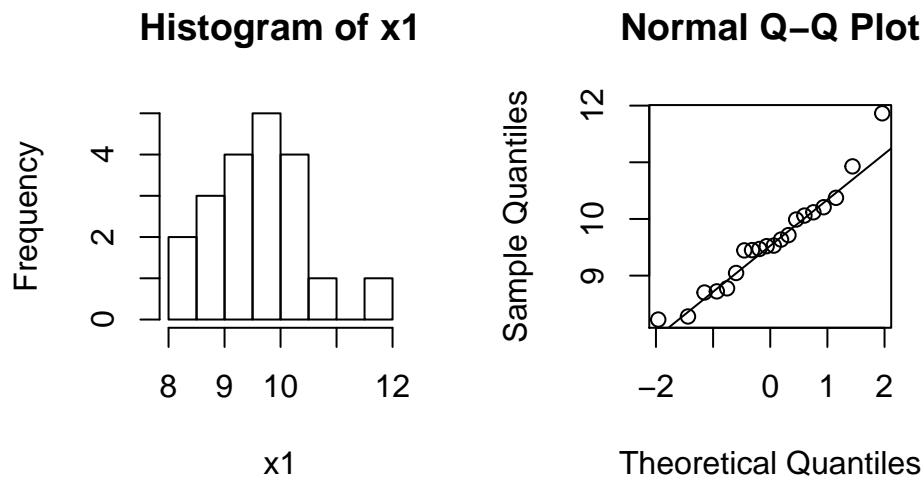
Exercise 1

We load the data from the data source.

```
load(file="assign1.RData")
```

Dataset x1

```
hist(x1);  
qqnorm(x1);qqline(x1);
```

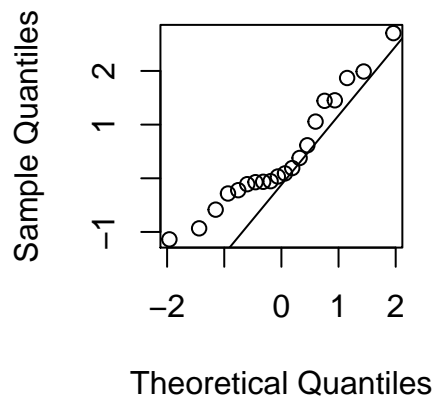


We can see that the histogram does not show a normal distribution but it is probably because the size of the sample is very small. However, the Q-Q-plot shows that the distribution is normal.

We can see the similarities with a Q-Q-plot of a normal distribution of the same size.

```
qqnorm(rnorm(length(x1)));qqline(rnorm(length(x1)))
```

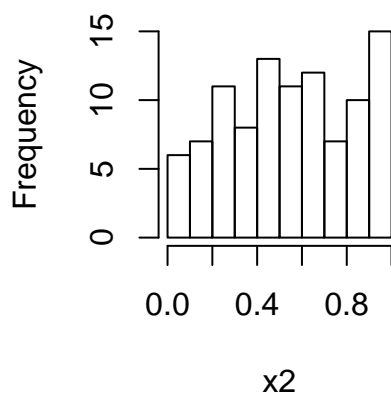
Normal Q–Q Plot



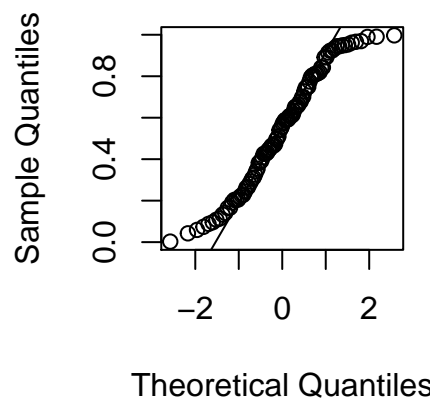
Dataset x2

```
hist(x2);  
qqnorm(x2);qqline(x2)
```

Histogram of x2



Normal Q–Q Plot

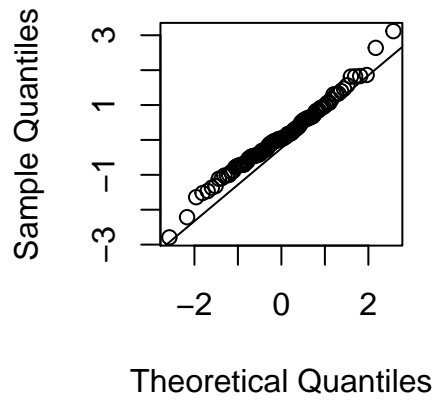


Both the histogram and QQ-plot do not show that the distribution is normal. The QQ-plot shows deviated tails.

We can compare the QQ-plot to one representing a sample with normal distribution of the same size.

```
qqnorm(rnorm(length(x2)));qqline(rnorm(length(x2)))
```

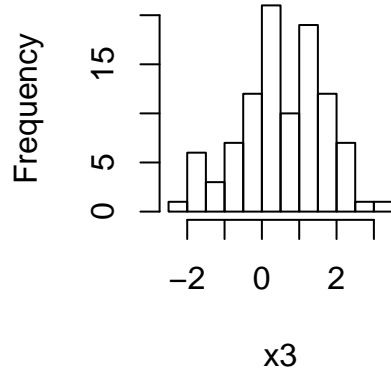
Normal Q–Q Plot



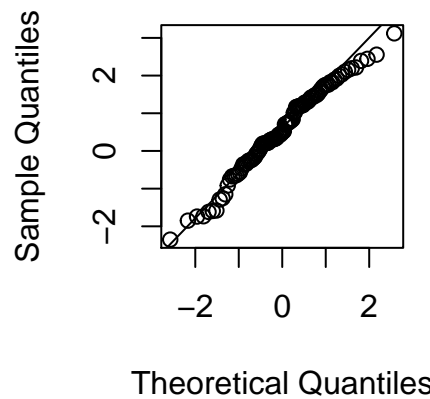
Dataset x3

```
hist(x3);  
qqnorm(x3);qqline(x3)
```

Histogram of x3



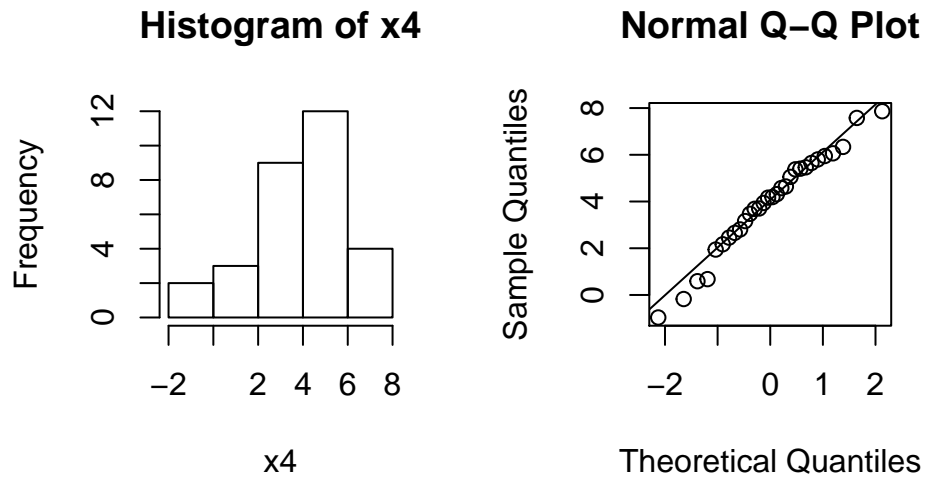
Normal Q–Q Plot



Both the histogram and QQ-plot show that the sample follows a normal distribution.

Dataset x4

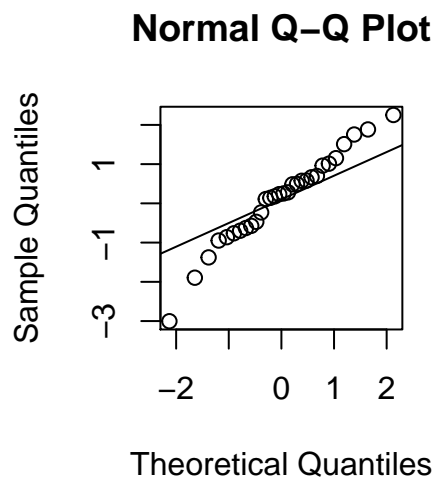
```
hist(x4);  
qqnorm(x4);qqline(x4)
```



The histogram does not show a normal distribution but the QQ-plot does.

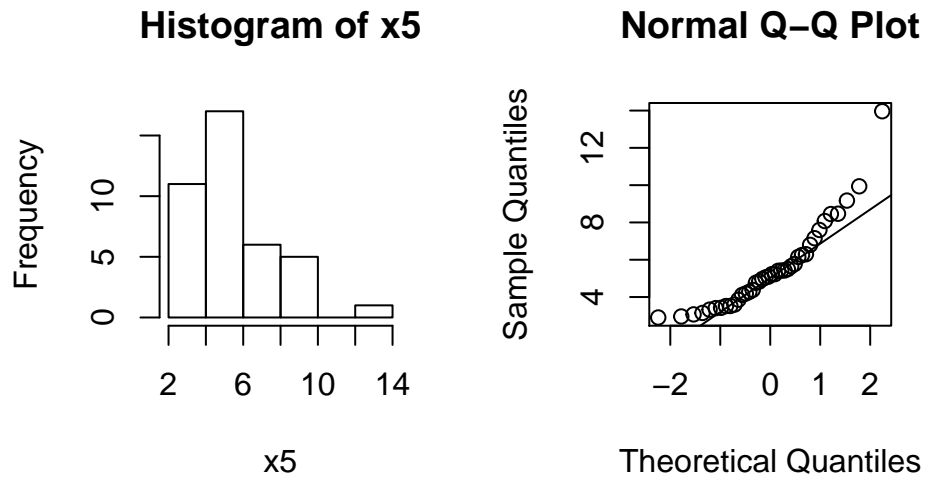
We can see the similarities with a QQ-plot of a normal distribution of the same size.

```
qqnorm(rnorm(length(x4)));qqline(rnorm(length(x4)))
```



Dataset x5

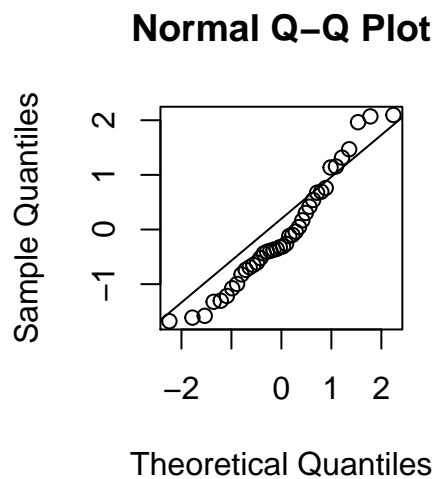
```
hist(x5);  
qqnorm(x5);qqline(x5)
```



Neither the histogram nor the QQ-plot depict a normal distribution.

The QQ-plot is not similar to one of a normal distribution of the same size as x5.

```
qqnorm(rnorm(length(x5)));qqline(rnorm(length(x5)))
```



Exercise 2

First we define a function to generate the list of p-values after running the tests 1000 times. The function takes by arguments m, n, μ, ν and sd .

```
get_p_values<-function(m,n,mu,nu,sd,iter=1000){  
  B=iter  
  p=numeric(B)  
  
  for(b in 1:B){  
    x=rnorm(m,mu,sd)  
    y=rnorm(n,nu,sd)  
    p[b]=t.test(x,y,var.equal = TRUE)[[3]]  
  }  
  return(p)  
}
```

(1) Test values: $\mu=\nu=180$, $m=n=30$ and $sd=10$.

We obtain the number of p-values smaller than 5%

```
p=get_p_values(30,30,180,180,10)  
p_amount=sum(p<0.05)  
power95=mean(p<0.05)
```

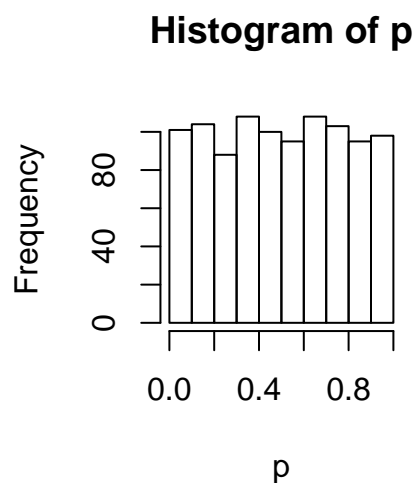
There are 46 values smaller than 5%. The power of the test is 0.046 with 95% confidence.

We obtain the number of p-values smaller than 10%

```
p_amount=sum(p<0.10)  
power90=mean(p<0.10)
```

There are 101 values smaller than 10%. The power of the test is 0.101 with 90% confidence.

```
hist(p)
```



Findings: The statistical power of 0.046 and 0.101 indicate that the probability of correctly rejecting $H_0(\mu == \nu)$ is very low. This makes sense because in this test μ and ν are equal. Checking the histogram we see that the p-values follow a uniform distribution which indicates that H_0 is true.

(2) Test values: $\mu=\nu=180$, $m=n=30$ and $sd=1$.

We obtain the number of p-values smaller than 5%

```
p=get_p_values(30,30,180,180,1)
p_amount=sum(p<0.05)
power95=mean(p<0.05)
```

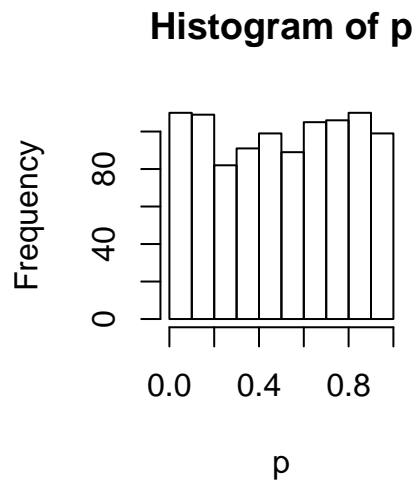
There are 51 values smaller than 5%. The power of the test is 0.051 with 95% confidence.

We obtain the number of p-values smaller than 10%

```
p_amount=sum(p<0.10)
power90=mean(p<0.10)
```

There are 110 values smaller than 10%. The power of the test is 0.11 with 90% confidence.

```
hist(p)
```



Findings: The statistical power of 0.051 and 0.11 indicate that the probability of correctly rejecting $H_0(\mu == \nu)$ is very low. This makes sense because in this test μ and ν are equal. Checking the histogram we see that the p-values follow a uniform distribution which indicates that H_0 is true.

(3) Test values: $\mu=180$, $\nu=175$, $m=n=30$ and $sd=6$.

We obtain the number of p-values smaller than 5%

```
p=get_p_values(30,30,180,175,6)
p_amount=sum(p<0.05)
power95=mean(p<0.05)
```

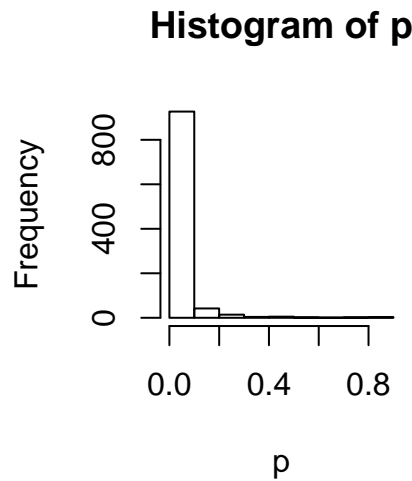
There are 890 values smaller than 5%. The power of the test is 0.89 with 95% confidence.

We obtain the number of p-values smaller than 10%

```
p_amount=sum(p<0.10)
power90=mean(p<0.10)
```

There are 927 values smaller than 10%. The power of the test is 0.927 with 90% confidence.

```
hist(p)
```



Findings: The statistical power of 0.89 and 0.927 indicate that the probability of correctly rejecting $H_0(\mu == \nu)$ is high. This makes sense because in this test μ and ν are different. Checking the histogram we see that the p-values do not follow a uniform distribution which they would, if H_0 was true.

Exercise 3

First we define a function to get the powers for the t-test for every value in *values* as ν .

```
get_powers<-function(m,n,mu,sd,values,iter=1000){
  powers = numeric(length(values))
  i=1
  for(nu in values){
    p=get_p_values(m,n,mu,nu,sd,iter=iter)
    powers[i]=mean(p<0.05)
    i=i+1
  }
  return(powers)
}
```

We define the values of ν .

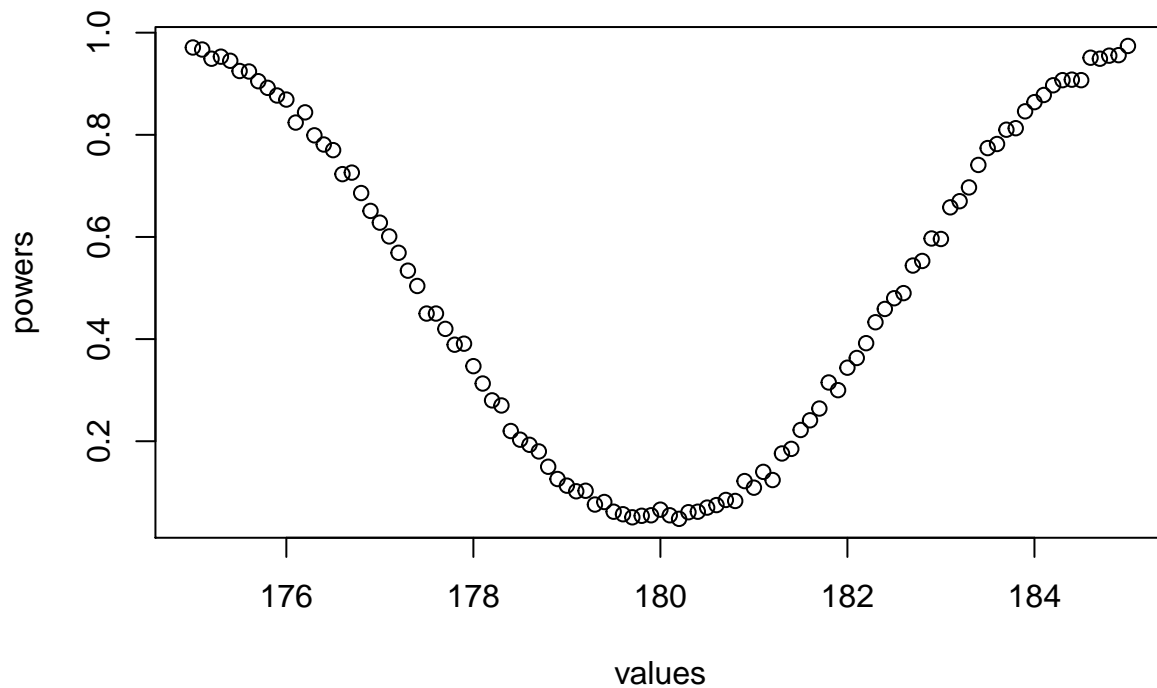
```
values = seq(175,185,by=0.1)
```

(1) **Test values:** $\mu=180$, $m=n=30$ and $sd=5$. We calculate the powers for each value of ν .


```
powers=get_powers(30,30,180,5,values)
```

We plot the power as a function of nu .

```
plot(values,powers)
```

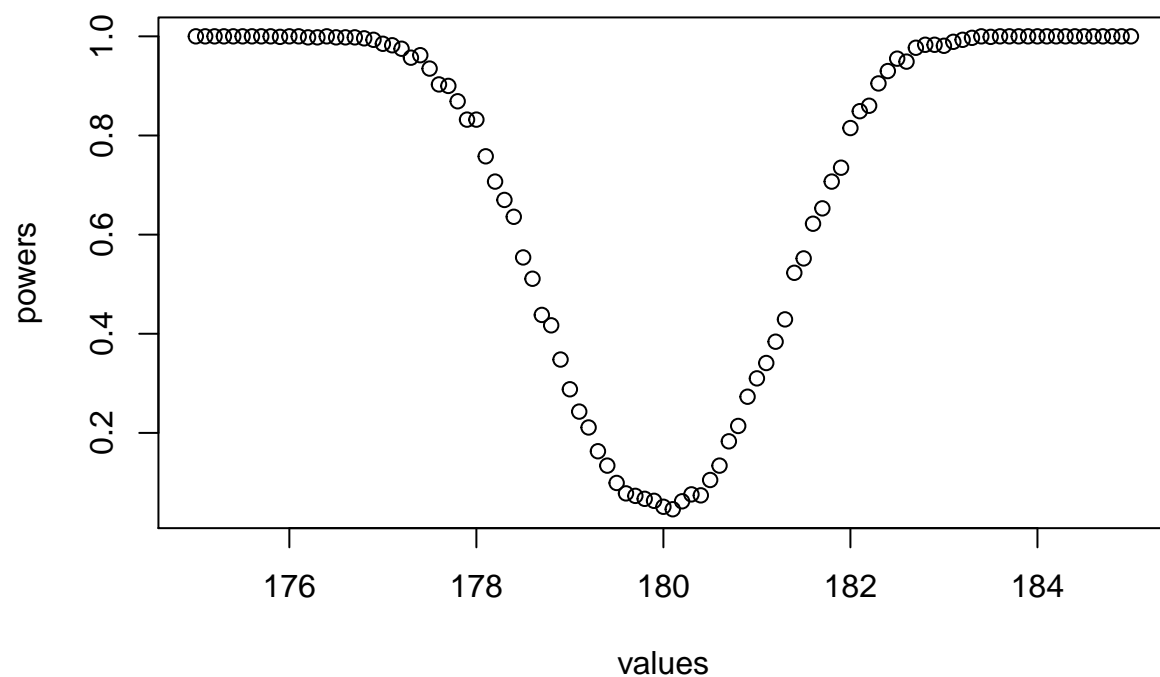


(2) Test values: $\mu=180$, $m=n=100$ and $sd=5$. We calculate the powers for each value of nu .

```
powers=get_powers(100,100,180,5,values)
```

We plot the power as a function of nu .

```
plot(values,powers)
```

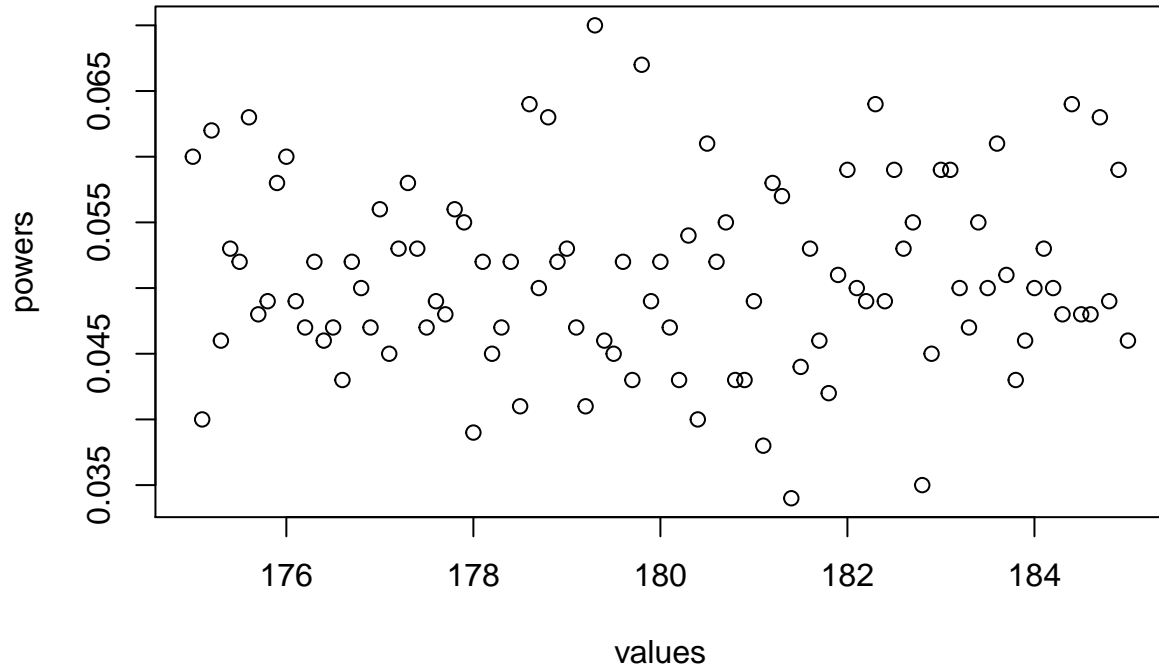


(3) Test values: $\mu=180$, $m=n=30$ and $sd=100$. We calculate the powers for each value of nu .

```
powers=get_powers(30,30,180,100,values)
```

We plot the power as a function of nu .

```
plot(values,powers)
```



(4) **Findings:** With the first two plots we can see how the power of the test decreases when the nu values get closer to mu and it increases when nu values differ from mu . This indicates that the probability of correctly rejecting H_0 ($mu == nu$) is higher when nu values differ more from mu and it gets lower when they get closer to mu .

Comparing the first and second plot, we see that when the population of the samples are bigger, the power obtained for the test are more precise. For example, in the first plot for $nu=178$ we get a power of 0.325 while in the second plot we get a power of 0.818 for the same value of nu .

In the last plot we see that, with the same population as in plot 1 but with a high standard deviation, we get low power values for any nu values while in the other plots we were getting high power values for the same nu values.