

MTH 3240 R Notes 9

9 One-Sample t Test and Confidence Interval for a Population Mean μ

9.1 Carrying Out the Test and Computing the Confidence Interval

- To carry out a one-sample t test and compute a one-sample t confidence interval for μ , use the function:

```
t.test()      # one-sample t test and confidence interval, with
              # arguments x, mu, alternative, and conf.level
```

- The `t.test()` function takes the following arguments:

<code>x</code>	a data vector.
<code>alternative</code>	the direction for the alternative hypothesis, one of "two.sided", "less", or "greater". The default is "two.sided".
<code>mu</code>	the null hypothesized value (μ_0) for the unknown population mean, with default value 0.
<code>conf.level</code>	(optionally) the confidence level for a confidence interval for the unknown population mean, with default value 0.95.

- Be aware that R reports a *one-sided confidence interval* for μ when "less" or "greater" is specified for `alternative`.
- Suppose for example we have the following data set:

```
x.data <- c(58.7, 54.3, 52.1, 55.5, 54.0)
```

and we want to carry out the one-sample t test of

$$\begin{aligned}H_0 : \mu &= 50 \\ H_a : \mu &\neq 50\end{aligned}$$

and compute a 95% confidence interval for μ . We type:

```
t.test(x = x.data, mu = 50)
```

```
##  
## One Sample t-test  
##  
## data: x.data  
## t = 4.5094, df = 4, p-value = 0.01074  
## alternative hypothesis: true mean is not equal to 50  
## 95 percent confidence interval:  
## 51.89075 57.94925  
## sample estimates:  
## mean of x  
## 54.92
```

From the output:

- The observed value of the one-sample t test statistic is

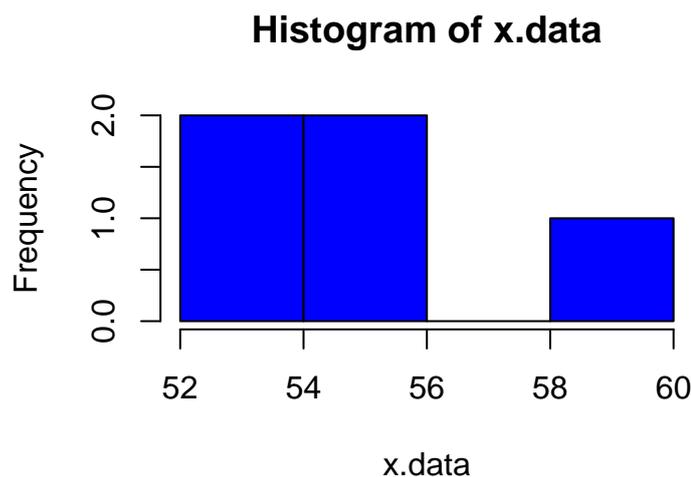
$$t = \frac{\bar{X} - 50}{S/\sqrt{n}} = 4.5094.$$

- The degrees of freedom for the t distribution of the test statistic under H_0 is $n - 1 = 4$.
- The p-value is 0.01074.
- A 95% confidence interval for μ is (51.891, 57.949).
- The sample mean is $\bar{X} = 54.92$.

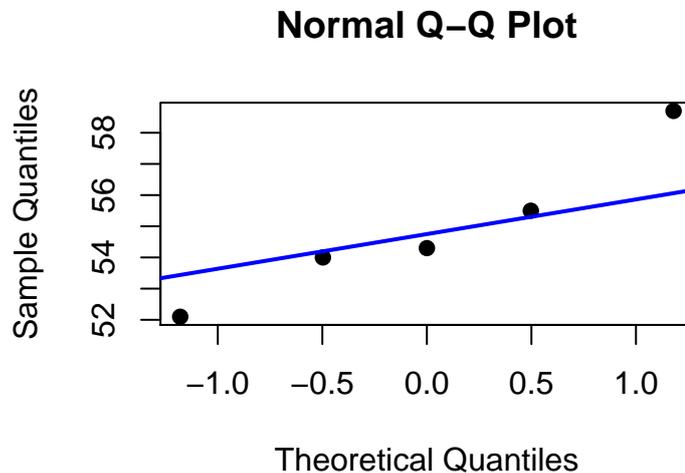
9.2 Checking the Normality Assumption

- To check the assumption that the sample is from a $N(\mu, \sigma)$ population, we plot the data in either a histogram or a normal probability plot.
- For example:

```
hist(x.data, col = "blue")
```



```
qqnorm(x.data, pch = 19)
qqline(x.data, lwd = 2, col = "blue")
```



Although it's difficult to tell based on such a small sample size, there don't appear to be any strong indications of non-normality, so the results of the one-sample t test and confidence interval appear to be valid.

Section 9.2 Exercises

Exercise 1 Here's a vector containing a random sample from a population whose true (unknown) mean is μ .

```
x.data <- c(53, 50, 41, 49, 44, 43, 55, 49, 42, 46, 52, 48, 46, 46, 48)
```

a) Use `t.test()` to carry out a **one-sample t test** of

$$H_0 : \mu = 50$$

$$H_a : \mu < 50$$

(and to compute a **one-sided t confidence interval** for μ). **Don't forget** to specify `mu = 50` and `alternative = "less"`. **Report the following values and answer the questions.**

Test statistic: $t =$ _____

P-value: $p =$ _____

One-sided upper confidence limit: $UCL =$ _____

Based on the t test, is there any statistically significant evidence that μ is less than 50 (Yes/No)?

Based on the one-sided confidence interval, is it plausible that μ is as large as 51 (Yes/No)?

- b) Check the normality assumption required for the t test and one-sided confidence interval:

```
hist(x.data, col = "blue")
```

```
qqnorm(x.data, pch = 19)  
qqline(x.data, col = "blue")
```

Based on the plots, does the normality assumption appear to be met (Yes/No)?

10 (Optional for Spring 2020) One-Sample Sign Test for a Population Median $\tilde{\mu}$

- To carry out a one-sample sign test for a population median $\tilde{\mu}$, we use the function:

```
binom.test() # one-sample sign test (or binomial test) with argu-  
             # ments x, n, alternative, and conf.level
```

- To use `binom.test()` to carry out a one-sample sign test, we must pass the value of the sign test statistic S^+ to `binom.test()` via the argument `x`. The arguments to be passed to `binom.test()` are:

<code>x</code>	the observed value of the sign test statistic S^+ via the argument.
<code>n</code>	the sample size.
<code>alternative</code>	the direction for the alternative hypothesis, one of "two.sided", "less", or "greater". The default is "two.sided".

- The observed value of the sign test statistic S^+ via the argument `x`.
- The sample size `n`.
- Optionally, the direction `alternative` of the alternative hypothesis H_a (either "less" and "greater" or "two.sided"). The default is "two.sided". The other options, "less" and "greater", carry out lower-tailed and upper-tailed tests, respectively.

- For example, suppose we want to use the data set

```
x.data <- c(58.7, 49.1, 54.3, 52.1, 55.5, 54.0)
```

to carry out a sign test of

$$\begin{aligned} H_0 : \tilde{\mu} &= 50 \\ H_a : \tilde{\mu} &\neq 50 \end{aligned}$$

Because 5 of the $n = 6$ observations are greater than 50, the test statistic is

$$S^+ = 5$$

and we pass this value to `binom.test()` to carry out the test:

```
binom.test(x = 5, n = 6, alternative = "two.sided")

##
## Exact binomial test
##
## data: 5 and 6
## number of successes = 5, number of trials = 6, p-value = 0.2188
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.3587654 0.9957893
## sample estimates:
## probability of success
## 0.8333333
```

From the output, we see that the p-value for the test is 0.2188.

- Note that:
 - The `binom.test()` function actually carries out a so-called *binomial test* for whether the true (unknown) success probability p in a binomial distribution with n trials is 0.5, but it turns out that the p-value for this test is equivalent to the p-value for the one-sample sign test for the true median $\tilde{\mu}$ of a population. In the output, we can consider the "true probability of success" in the output to be the true probability (p) that an observation sampled from the population will be greater than 50, and this probability will be different from 0.5 if (and only if) the population median $\tilde{\mu}$ is different from 50, i.e. if and only if $H_a : \tilde{\mu} \neq 50$ is true.
 - The confidence interval that `binom.test()` reports is *not* a confidence interval for the population median $\tilde{\mu}$. It's a confidence interval for the true (unknown) success probability p in a binomial distribution, which, in the context of the sign test performed above, is the true probability that an observation sampled from the population will be greater than 50.
 - The "probability of success" on the last line of the output is just x/n , i.e. the proportion of sample values that are greater than 50, or $5/6 = 0.8333333$.

11 Two-Sample t Test and Confidence Interval for $\mu_x - \mu_y$

11.1 Carrying Out the Test and Computing the Confidence Interval

- To carry out a two-sample t test and compute a two-sample t confidence interval for $\mu_X - \mu_Y$, we use:

```
t.test()      # two-sample t test and confidence interval, with argu-
              # ments x, y, mu, alternative, and conf.level
```

- For a two-sample t test, the arguments passed to `t.test()` are:

<code>x</code>	a data vector.
<code>y</code>	another data vector.
<code>alternative</code>	the direction for the alternative hypothesis, one of "two.sided", "less", or "greater". The default is "two.sided".
<code>mu</code>	the null hypothesized value for the <i>difference</i> between the unknown population means, with default value 0.
<code>conf.level</code>	(optionally) the confidence level for a confidence interval for the <i>difference</i> between unknown population means, with default value 0.95.

- Be aware that R reports a *one-sided confidence interval* for $\mu_x - \mu_y$ when "less" or "greater" are specified for `alternative`.
- Suppose for example that we have two independent random samples as the vectors:

```
x.data <- c(38.7, 44.3, 32.1, 37.5, 45.0)
y.data <- c(48.7, 47.6, 52.1, 39.4, 49.9)
```

To carry out a two-sample t test of

$$H_0 : \mu_x - \mu_y = 0$$

$$H_a : \mu_x - \mu_y \neq 0$$

and compute a 95% confidence interval for the difference $\mu_x - \mu_y$, type:

```
t.test(x = x.data, y = y.data, alternative = "two.sided", mu = 0,
       conf.level = 0.95)

##
## Welch Two Sample t-test
##
## data: x.data and y.data
## t = -2.4949, df = 7.9349, p-value = 0.03746
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -15.4432426 -0.5967574
## sample estimates:
## mean of x mean of y
## 39.52 47.54
```

From the output,

- The observed value of the t test statistic is

$$t = \frac{\bar{X} - \bar{Y}}{\sqrt{\frac{S_X^2}{n_x} + \frac{S_Y^2}{n_y}}} = -2.495.$$

- The degrees of freedom are

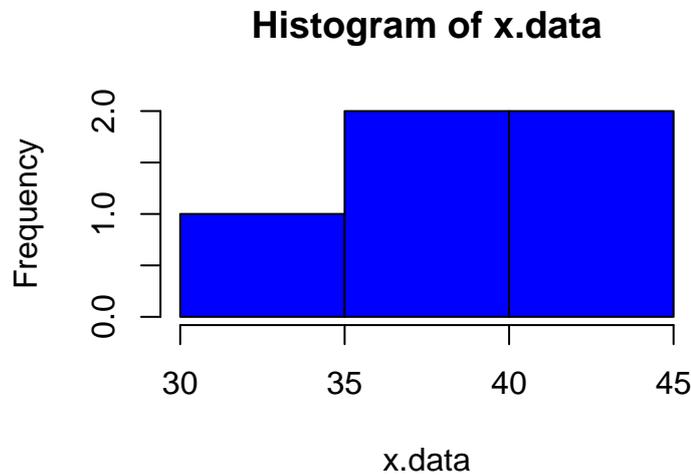
$$df = \frac{\left(\frac{S_X^2}{n_x} + \frac{S_Y^2}{n_y}\right)^2}{\frac{(S_X^2/n_x)^2}{n_x-1} + \frac{(S_Y^2/n_y)^2}{n_y-1}} = 7.935.$$

- The p-value is 0.03746.
- The 95% confidence interval for the true difference $\mu_x - \mu_y$ is (-15.443, -0.597).
- The sample means are $\bar{X} = 39.52$ and $\bar{Y} = 47.54$,

11.2 Checking the Normality Assumption

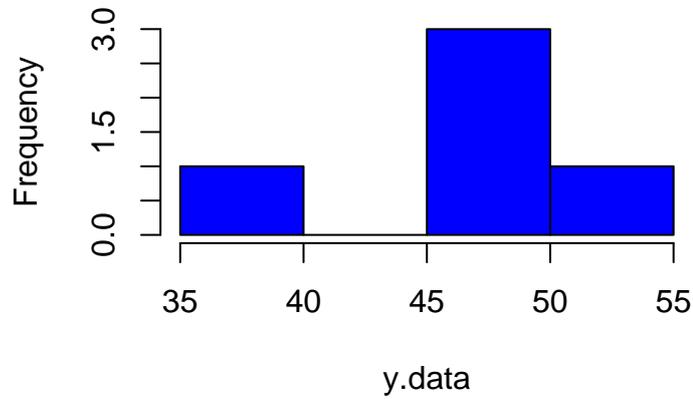
- To check the assumption that the two samples were drawn from $N(\mu_x, \sigma_x)$ and $N(\mu_y, \sigma_y)$ populations, we can make histograms or normal probability plots of the two samples.
- For example:

```
hist(x.data, col = "blue")
```



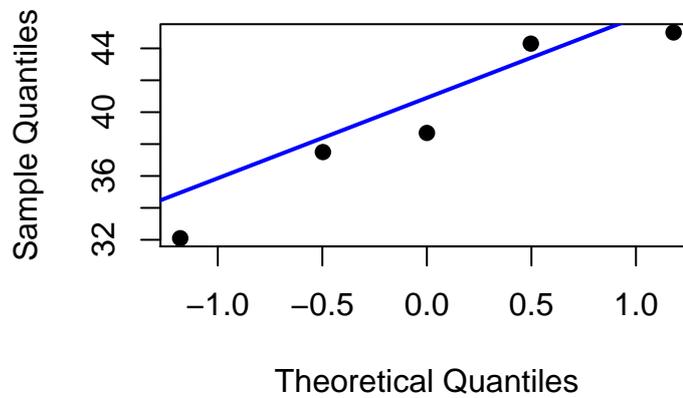
```
hist(y.data, col = "blue")
```

Histogram of y.data

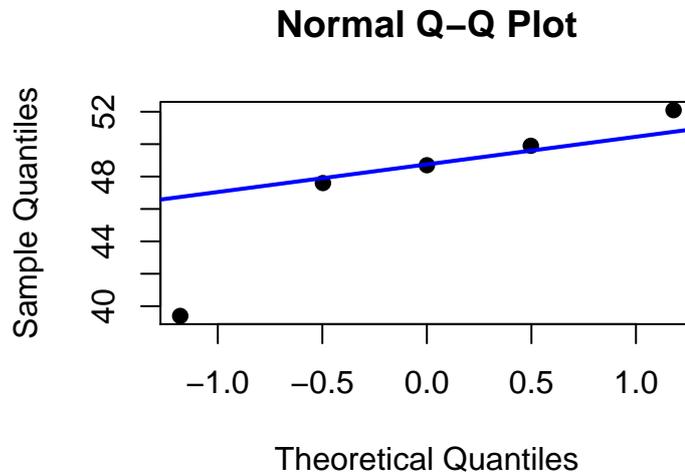


```
qqnorm(x.data, pch = 19)  
qqline(x.data, lwd = 2, col = "blue")
```

Normal Q-Q Plot



```
qqnorm(y.data, pch = 19)  
qqline(y.data, lwd = 2, col = "blue")
```



Although it's difficult to tell based on such small sample sizes, there don't appear to be any strong indications of non-normality, so the t test and confidence interval results appear to be valid.

Section 11.2 Exercises

Exercise 2 Here are two vectors containing random samples from two populations whose true (unknown) means are μ_x and μ_y .

```
x.data <- c(39, 57, 65, 49, 45, 53, 57, 50, 37, 58, 46, 62, 56, 67, 54)
y.data <- c(50, 55, 65, 48, 60, 41, 49, 46, 50, 49, 29, 48, 43, 55, 59)
```

a) Use `t.test()` to carry out a **two-sample t test** of

$$H_0 : \mu_x - \mu_y = 0$$

$$H_a : \mu_x - \mu_y \neq 0$$

and to compute a **t confidence interval** for the *effect size* $\mu_x - \mu_y$. **Report the following values and answer the questions.**

Test statistic: $t =$ _____

P-value: $p =$ _____

Confidence interval: $(LCL, UCL) =$ _____

Based on the t test, is there any statistically significant evidence for a difference between μ_x and μ_y (Yes/No)?

Based on the confidence interval, is it plausible that the effect size $\mu_x - \mu_y$ is 0 (Yes/No)?

- b) Check the normality assumption for each sample, as required for the t test and confidence interval:

```
hist(x.data, col = "blue")

qqnorm(x.data, pch = 19)
qqline(x.data, col = "blue")
```

```
hist(y.data, col = "blue")

qqnorm(y.data, pch = 19)
qqline(y.data, col = "blue")
```

Based on the plots, does the normality assumption appear to be met for both samples (Yes/No)?

12 Wilcoxon Rank Sum Test

- To carry out a Wilcoxon rank sum test for $\mu_x - \mu_y$, use the function:

```
wilcox.test()      # Wilcoxon rank sum test and confidence interval,
                  # with arguments x, y, mu, and alternative
```

- The arguments passed to `wilcox.test()` are:

<code>x</code>	a data vector.
<code>y</code>	another data vector.
<code>alternative</code>	the direction for the alternative hypothesis, one of "two.sided", "less", or "greater". The default is "two.sided".
<code>mu</code>	the null hypothesized value for the <i>difference</i> between the unknown population means, with default value 0.
<code>conf.int</code>	(optionally) a TRUE or FALSE indicator of whether a (nonparametric) confidence interval for the <i>difference</i> between unknown population means should be computed, with default FALSE.
<code>conf.level</code>	(optionally) the confidence level for a confidence interval for the <i>difference</i> between unknown population means, with default value 0.95.

- Be aware that the test statistic that R reports is *not* W_{rs} (the sum of the ranks of the x sample), but rather *it's the sum of the ranks of the x sample with the minimum possible value $(n_x(n_x + 1)/2)$ subtracted.* See the example below.
- Suppose, for example, that we have two independent random samples and we save them as the vectors:

```
x.data <- c(15.4, 12.2, 13.1, 16.8, 13.3)
y.data <- c(16.1, 17.5, 17.3, 20.2, 19.1, 17.8)
```

Suppose also that we want to test the hypotheses

$$H_0 : \mu_X - \mu_Y = 0$$

$$H_a : \mu_X - \mu_Y < 0$$

The Wilcoxon rank sum test is carried out by typing:

```
wilcox.test(x = x.data, y = y.data, alternative = "less")

##
## Wilcoxon rank sum test
##
## data: x.data and y.data
## W = 1, p-value = 0.004329
## alternative hypothesis: true location shift is less than 0
```

The value that R reports as the test statistic is not W_{rs} (the sum of the ranks of the x sample), but rather W_{rs} with its minimum possible value $n_x(n_x + 1)/2$ subtracted. In other words, R reports

$$W_{rs} - n_x(n_x + 1)/2$$

where n_x is the size of the smaller sample.

From the output above,

- The reported observed test statistic value is $W_{rs} - n_x(n_x + 1)/2 = 1$ (where $n_x = 5$), so $W_{rs} = 1 + n_x(n_x + 1)/2 = 16$.
- The p-value is 0.00433.

Thus we conclude that there's statistically significant evidence that the X population mean μ_X is less than the Y population mean μ_Y .

Section 12.0 Exercises

Exercise 3 Here are two vectors containing random samples from two populations whose true (unknown) means are μ_x and μ_y .

```
x.data <- c(0.6, 12.0, 2.7, 1.3, 0.8, 6.4, 0.9, 0.8, 0.8, 1.4, 0.5)
y.data <- c(3.0, 1.8, 1.8, 10.6, 2.9, 5.2, 2.5, 4.4, 2.1, 2.1, 2.1)
```

Use `wilcox.test()` to carry out a **rank sum test** of

$$H_0 : \mu_x - \mu_y = 0$$

$$H_a : \mu_x - \mu_y \neq 0$$

Report the following values and answer the question.

Test statistic: $W_{rs} = \underline{\hspace{2cm}}$

P-value: $p = \underline{\hspace{2cm}}$

Based on the rank sum test, is there any statistically significant evidence for a difference between μ_x and μ_y (Yes/No)?

13 Paired t Test and Confidence Interval for μ_d

13.1 Carrying Out the Test and Computing the Confidence Interval

- The `t.test()` function will carry out a paired t test and compute a paired t confidence interval for μ_D if we specify `paired = TRUE`:

```
t.test()      # one-sample t test and confidence interval, with
              # arguments x, mu, paired, alternative, and conf.level
```

<code>x</code>	a data vector.
<code>y</code>	another data vector.
<code>alternative</code>	the direction for the alternative hypothesis, one of "two.sided", "less", or "greater". The default is "two.sided".
<code>mu</code>	the null hypothesized value for the <i>difference</i> between the unknown population means, with default value 0.
<code>paired</code>	a TRUE or FALSE value indicating whether the samples are <i>paired</i> .
<code>conf.level</code>	(optionally) the confidence level for a confidence interval for the <i>difference</i> between unknown population means, with default value 0.95.

- Suppose for example that we have the following data from a *matched pairs* study, where each element of `x.data` is paired with the corresponding element of `y.data`:

```
x.data <- c(38.7, 44.3, 32.1, 37.5, 45.0)
y.data <- c(48.7, 47.6, 52.1, 39.4, 49.9)
```

We want to carry out a paired t test of

$$H_0 : \mu_d = 0$$

$$H_a : \mu_d \neq 0$$

or equivalently

$$H_0 : \mu_X - \mu_Y = 0$$

$$H_a : \mu_X - \mu_Y \neq 0$$

To carry out the test, we type:

```
t.test(x = x.data, y = y.data, paired = TRUE, mu = 0, alternative = "two.sided",
       conf.level = 0.95)

##
## Paired t-test
##
## data: x.data and y.data
## t = -2.4353, df = 4, p-value = 0.07157
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -17.163488  1.123488
## sample estimates:
## mean of the differences
## -8.02
```

From the output, we see that

- The paired t test statistic is

$$t = \frac{\bar{D} - 0}{S_d/\sqrt{n}} = -2.435$$

- The p-value is (from the t distribution with 4 degrees of freedom) is 0.0716.
- The 95% confidence interval for μ_d (or $\mu_X - \mu_Y$) is

$$(-17.1635, 1.1235)$$

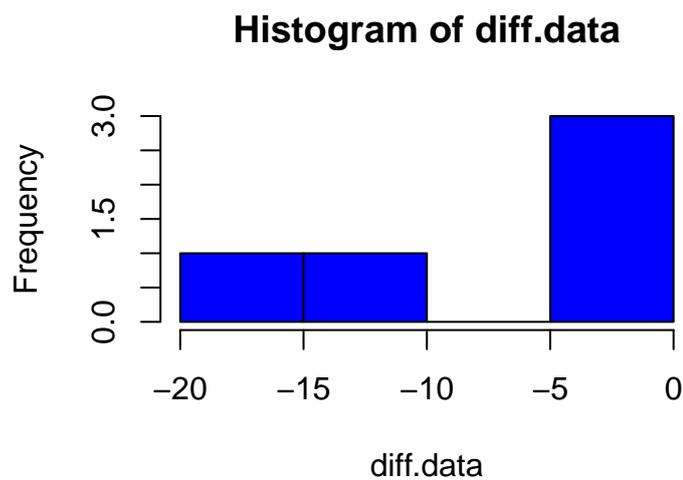
- The sample mean difference is $\bar{D} = -8.02$.

- Remember: When "less" or "greater" are specified for `alternative`, R reports a one-sided confidence interval for μ_d (or $\mu_X - \mu_Y$).

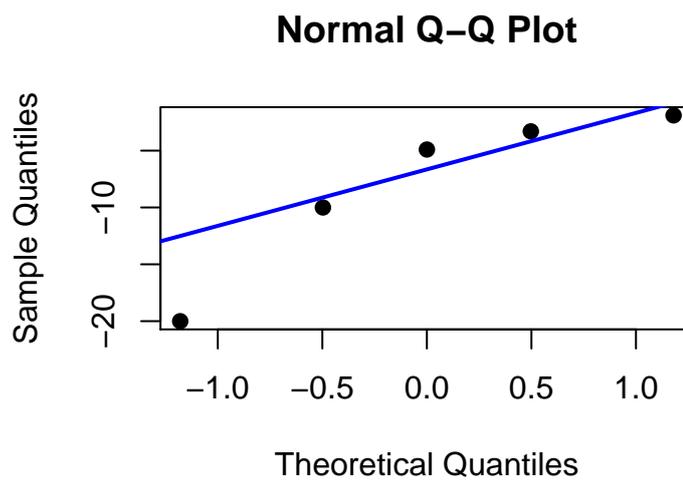
13.2 Checking the Normality Assumption

- To check the assumption that the *differences* came from a $N(\mu_d, \sigma_d)$ distribution, we make a histogram or normal probability plot of the *differences*.
- For example:

```
diff.data <- x.data - y.data  
hist(diff.data, col = "blue")
```



```
qqnorm(diff.data, pch = 19)  
qqline(diff.data, lwd = 2, col = "blue")
```



Although it's difficult to tell based on such a small sample size, there don't appear to be any strong indications of non-normality, so the results of the paired t test and confidence interval appear to be valid.

Section 13.2 Exercises

Exercise 4 Here are two vectors containing data from a **matched pairs study** involving **paired** samples from two populations whose true (unknown) means are μ_x and μ_y .

```
x.data <- c(22, 31, 30, 31, 21, 25, 29, 33, 27, 32, 23, 33, 33, 29, 33)
y.data <- c(23, 30, 28, 30, 24, 28, 26, 31, 27, 29, 23, 29, 29, 27, 26)
```

a) Use `t.test()`, specifying `paired=TRUE`, to carry out a **paired t test** of

$$H_0 : \mu_d = 0$$

$$H_a : \mu_d > 0$$

or equivalently

$$H_0 : \mu_X - \mu_Y = 0$$

$$H_a : \mu_X - \mu_Y > 0$$

and to compute a **one-sided paired t confidence interval** for the **effect size** μ_d , or equivalently $\mu_x - \mu_y$. **Don't forget** to specify `alternative="greater"`. **Report the following values and answer the questions.**

Test statistic: $t =$ _____

P-value: $p =$ _____

One-sided lower
confidence limit: $LCL =$ _____

Based on the paired t test, is there any statistically significant evidence that the effect size μ_d (or $\mu_x - \mu_y$) is greater than 0 (Yes/No)?

Based on the one-sided confidence interval, is it plausible that the effect size μ_d (or $\mu_x - \mu_y$) is as small as 0.2? (Yes/No)?

b) Compute the pairwise *differences* and use them to check the normality assumption required for the paired t test and confidence interval:

```
diff.data <- x.data - y.data
```

```
hist(diff.data, col = "blue")
qqnorm(diff.data, pch = 19)
qqline(diff.data, lwd = 2, col = "blue")
```

Based on the plots, does the normality assumption appear to be met for the *differences* (Yes/No)?

14 Wilcoxon Signed Ranks Test for Paired Samples

- The `wilcox.test()` function will carry out a Wilcoxon signed ranks test if we specify `paired = TRUE`:

```
wilcox.test()      # Wilcoxon signed ranks test and confidence interval,
                  # with arguments x, y, mu, paired, and alternative
```

The arguments passed to `wilcox.test()` are:

<code>x</code>	a data vector.
<code>y</code>	another data vector.
<code>alternative</code>	the direction for the alternative hypothesis, one of "two.sided", "less", or "greater". The default is "two.sided".
<code>mu</code>	the null hypothesized value for the <i>difference</i> between the unknown population means, with default value 0.
<code>paired</code>	a TRUE or FALSE value indicating whether the samples are <i>paired</i> .
<code>conf.int</code>	(optionally) a TRUE or FALSE indicator of whether a (nonparametric) confidence interval for the <i>difference</i> between unknown population means should be computed, with default FALSE.
<code>conf.level</code>	(optionally) the confidence level for a confidence interval for the <i>difference</i> between unknown population means, with default value 0.95.

- Consider again the paired samples

```
x.data
## [1] 38.7 44.3 32.1 37.5 45.0

y.data
## [1] 48.7 47.6 52.1 39.4 49.9
```

Suppose we want to carry out a Wilcoxon signed ranks test of the hypotheses

$$\begin{aligned}H_0 : \mu_d &= 0 \\ H_a : \mu_d &\neq 0\end{aligned}$$

or equivalently

$$\begin{aligned}H_0 : \mu_X - \mu_Y &= 0 \\ H_a : \mu_X - \mu_Y &\neq 0\end{aligned}$$

We carry out the test using the command:

```
wilcox.test(x = x.data, y = y.data, mu = 0, paired = TRUE, alternative = "two.sided")

##
## Wilcoxon signed rank test
##
## data: x.data and y.data
## V = 0, p-value = 0.0625
## alternative hypothesis: true location shift is not equal to 0
```

From the output,

- The signed ranks test statistic (denoted V in the output) is $W^+ = 0$.
- The p-value is 0.0625.

so we fail to reject the null hypothesis.

Section 14.0 Exercises

Exercise 5 Here are two vectors containing data from a **matched pairs study** involving **paired** samples from two populations whose true (unknown) means are μ_x and μ_y .

```
x.data <- c(49, 5, 31, 45, 64, 56, 149, 61, 52, 60, 59, 57, 49, 54, 59)
y.data <- c(61, 38, 57, 58, 72, 43, 121, 77, 60, 67, 71, 68, 65, 65, 67)
```

Use `wilcox.test()`, specifying `paired=TRUE`, to carry out a **signed rank test for paired samples** of the hypotheses

$$\begin{aligned}H_0 : \mu_d &= 0 \\ H_a : \mu_d &< 0\end{aligned}$$

or equivalently

$$\begin{aligned}H_0 : \mu_X - \mu_Y &= 0 \\ H_a : \mu_X - \mu_Y &< 0\end{aligned}$$

for the *effect size* μ_d , or equivalently $\mu_x - \mu_y$. **Don't forget** to specify `alternative="less"`. **Report the following values and answer the question.**

Test statistic: $W^+ =$ _____

P-value: $p =$ _____

Based on the signed rank test, is there any statistically significant evidence that the effect size μ_d (or $\mu_x - \mu_y$) is less than 0 (Yes/No)?

15 (Optional for Spring 2020) Sign Test for Paired Samples

- To carry out a sign test for $\tilde{\mu}_D$ using paired samples, we perform a one-sample sign test on the differences from the matched pairs study using the function:

```
binom.test() # paired-samples sign test (or binomial test) with
             # arguments x, n, alternative, and conf.level
```

The arguments to be passed to `binom.test()` are:

`x` the observed value of the paired samples sign test statistic S^+ via the argument.
`n` the sample size.
`alternative` the direction for the alternative hypothesis, one of "two.sided", "less", or "greater". The default is "two.sided".

- Recall that to use `binom.test()` to carry out a sign test, we must pass the value of the sign test statistic S^+ to `binom.test()` via the argument `x`.
- Suppose, for example, we want to carry out a paired samples sign test of the hypotheses

$$H_0 : \tilde{\mu}_D = 0$$

$$H_a : \tilde{\mu}_D \neq 0$$

using the data

```
x.data
## [1] 38.7 44.3 32.1 37.5 45.0

y.data
## [1] 48.7 47.6 52.1 39.4 49.9
```

The differences are

```
xy.diffs <- x.data - y.data
xy.diffs

## [1] -10.0 -3.3 -20.0 -1.9 -4.9
```

Since none of the differences are positive, the test statistic is $S^+ = 0$ and the sample size is $n = 5$ so we type:

```
binom.test(x = 0, n = 5, alternative = "two.sided")

##
## Exact binomial test
##
## data: 0 and 5
## number of successes = 0, number of trials = 5, p-value = 0.0625
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.0000000 0.5218238
## sample estimates:
## probability of success
## 0
```

From the output, we see that the p-value is 0.0625.

- Recall that:
 - `binom.test()` carries out a so-called *binomial test* for the true (unknown) success probability p in a binomial distribution, but it turns out that the p-value for this test is equivalent to the p-value for the sign test for $\tilde{\mu}_D$.
 - The confidence interval that `binom.test()` reports is *not* a confidence interval for the population median difference $\tilde{\mu}_D$. It's a confidence interval for the true (unknown) success probability p in a binomial distribution.