Lecture 2: Review

Pratheepa Jeganathan

09/25/2019

Recall

- What is a regression model?

Today (more review)

- Descriptive statistics – graphical
- Descriptive statistics – numerical
- Inference about a population mean
- Difference between two population means

Descriptive statistics – graphical

Right-to-work example

- This example from the text considers the effect of right-to-work legislation (which varies by state) on various factors. A description of the data can be found here.
- The variables are:
  - Income: income for a four-person family
  - COL: cost of living for a four-person family
  - PD: Population density
  - URate: rate of unionization in 1978
  - Pop: Population
  - Taxes: Property taxes in 1972
  - RTWL: right-to-work indicator

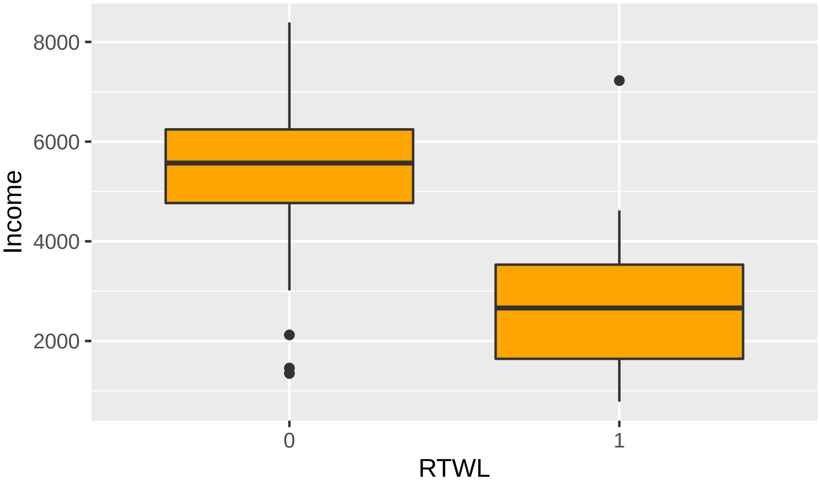
- In a study like this, there are many possible questions of interest.
- Our focus will be on the relationship between `RTWL` and `Income`.
- However, we recognize that other variables have an effect on `Income`.
- Let's look at some of these relationships.

```r
url = "http://www1.aucegypt.edu/faculty/hadi/RABE4/Data4/P
rtw.table = read.table(url, header=TRUE, sep='\t')
head(rtw.table)
```

```
##           City COL   PD URate     Pop Taxes Income RTWL
## 1      Atlanta 169  414  13.6 1790128  5128   2961    1
## 2       Austin 143  239  11.0  396891  4303   1711    1
## 3 Bakersfield 339   43  23.7  349874  4166   2122    0
## 4   Baltimore 173  951  21.0 2147850  5001   4654    0
## 5 Baton Rouge  99  255  16.0  411725  3965   1620    1
## 6       Boston 363 1257  24.4 3914071  4928   5634    0
```

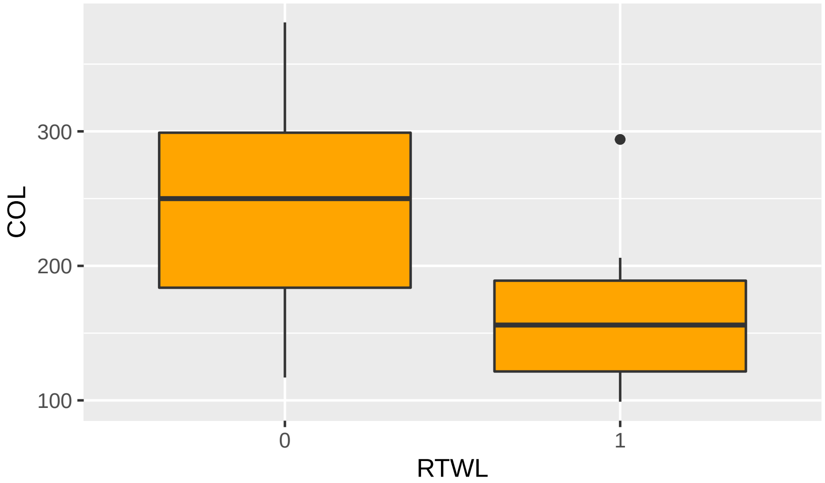
- `RTWL` is a binary variable.
- `Income` is a continuous variable.
- *Boxplot*: a graphical way to visualize the relationship between `Income` and `RTWL`.

```r
library(ggplot2)
rtw.table$RTWL = factor(rtw.table$RTWL)
p = ggplot(data = rtw.table,
  aes(x = RTWL, y = Income)) +
  geom_boxplot(fill = "orange")
```


Income
8000
6000
4000
2000
0
1
RTWL

- One variable that may have an important effect on the relationship between RTWL is the cost of living `COL`.
- It also varies between right-to-work states.

```r
p2 = ggplot(data = rtw.table,
  aes(x = RTWL, y = COL)) +
  geom_boxplot(fill = "orange")
```

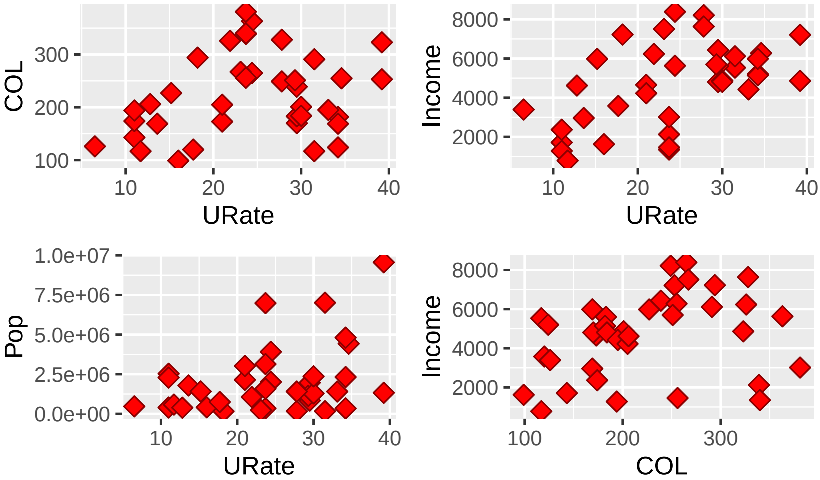


300
200
100
COL
0
1
RTWL

- We may want to include more than one plot in a given display.

```
p1 = ggplot(data = rtw.table) +
  geom_point(aes(x = URate, y = COL),
    shape=23, fill="red", color="darkred", size=3)
p2 = ggplot(data = rtw.table) +
  geom_point(aes(x = URate, y = Income),
    shape=23, fill="red", color="darkred", size=3)
p3 = ggplot(data = rtw.table) +
  geom_point(aes(x = URate, y = Pop),
    shape=23, fill="red", color="darkred", size=3)
p4 = ggplot(data = rtw.table) +
  geom_point(aes(x = COL, y = Income),
    shape=23, fill="red", color="darkred", size=3)
```

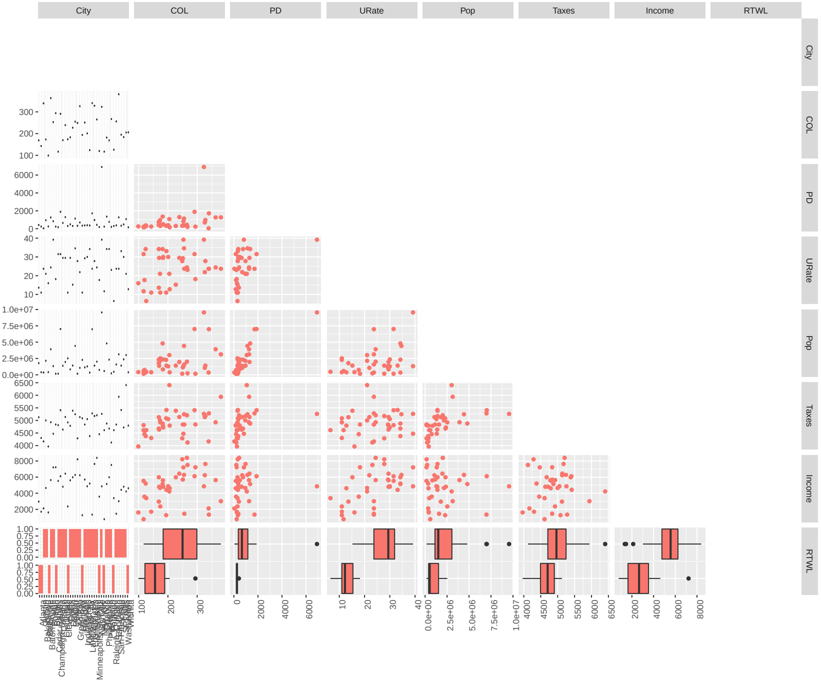
```r
library(gridExtra)
gridExtra::grid.arrange(p1, p2, p3, p4, nrow = 2)
```


COL
300
200
100
10
20
30
40
URate
Income
8000
6000
4000
2000
10
20
30
40
URate
Pop
1.0e+07
7.5e+06
5.0e+06
2.5e+06
0.0e+00
10
20
30
40
URate
Income
8000
6000
4000
2000
100
200
300
COL

- To learn more about laying out multiple ggplots layout

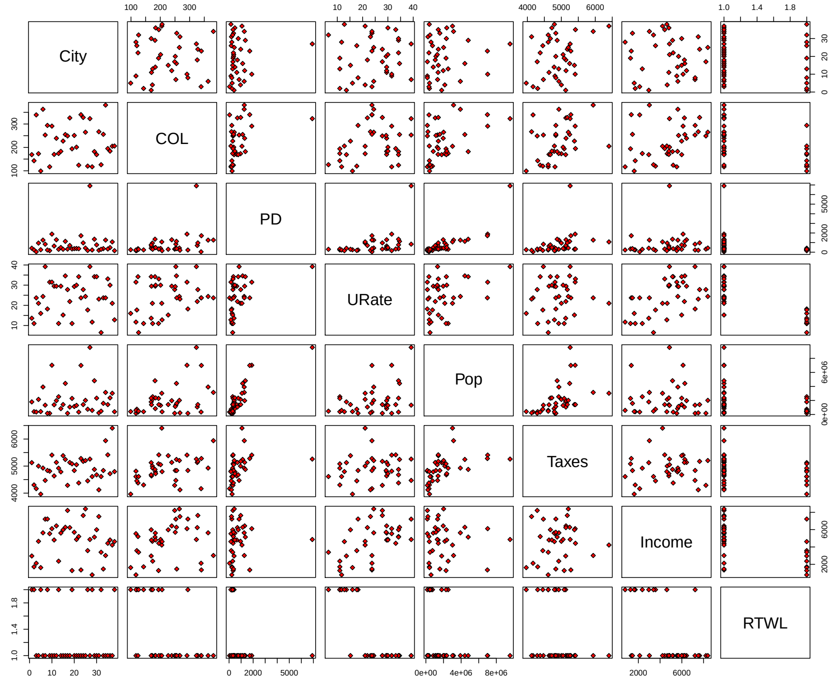
- Alternatively, to display all pairwise relationships in a given data set

```
library(GGally)
ggpairs(rtw.table,
  mapping= aes(color="darkred"),
  cardinality_threshold = 38,
  upper = "blank", diag = "blank",
  lower = list(continuous = "points",
    combo = "box",
    discrete = "facetbar",
    na = "na")) +
  theme(axis.text.x = element_text(angle = 90,
    hjust = 1))
```



City
COL
PD
URate
Pop
Taxes
Income
RTWL
300
200
100
6000
4000
2000
0
40
30
20
10
1.0e+07
7.5e+06
5.0e+06
2.5e+06
0.0e+00
6500
6000
5500
5000
4500
4000
8000
6000
4000
2000
1.00
0.75
0.50
0.25
0.00
Atlanta
Baton
Champaign
Cedar
Cle
Gre
Minneapolis
Raleigh
San
Was
Wichita
100
200
300
0
2000
4000
6000
10
20
30
40
0.0e+00
2.5e+06
5.0e+06
7.5e+06
1.0e+07
4000
4500
5000
5500
6000
6500
2000
4000
6000
8000

```r
pairs(rtw.table, pch=23, bg='red')
```

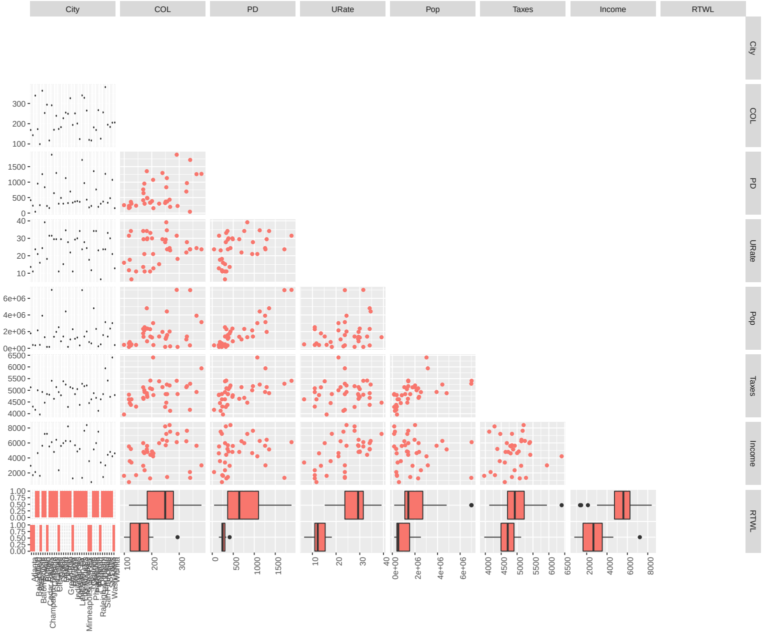

City
COL
PD
URate
Pop
Taxes
Income
RTWL
100
200
300
10
20
30
40
4000
5000
6000
1.0
1.4
1.8
0
10
20
30
0
2000
5000
0e+00
6e+06
2000
6000
100
200
300
10
20
30
40
4000
5000
6000
1.0
1.4
1.8
0
10
20
30
0
2000
5000
0e+00
4e+06
8e+06
2000
6000

- Observations
    - Look at the pairwise relationships
    - PD versus any other variable, there is a point that stands out from all the rest
    - This data point is New York City, the 27th row of the table.
- Let's look at the 27th row

```
print(rtw.table[27,])
```

```
##          City COL   PD URate     Pop Taxes Income RTWL
## 27 New York 323 6908  39.2 9561089  5260   4862    0
```

- Pairwise plots without 27th row



Building a model for right-to-work example

- Some of the main goals of this course:
  - Build a statistical model describing the *effect* of `RTWL` on `Income`.
  - This model should recognize that other variables also affect `Income`.
  - What sort of *statistical confidence* do we have in our conclusion about `RTWL` and `Income`?
  - Is the model adequate do describe this data set?
  - Are there other (simpler, more complicated) better models?

Numerical descriptive statistics

Mean of a sample

- Given a sample of numbers $X = (X_1, \ldots, X_n)$ the sample mean, $\overline{X}$ is
$$\overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i.$$

- There are many ways to compute this in `R`.

```
X = c(1,3,5,7,8,12,19)
print(X)
```

```
## [1]  1  3  5  7  8 12 19
```

```
print(mean(X))
```

```
## [1] 7.857143
```

```
print((X[1]+X[2]+X[3]+
    X[4]+X[5]+X[6]+X[7])/7)
```

```
## [1] 7.857143
```

```
print(sum(X)/length(X))
```

```
## [1] 7.857143
```

Example

- We'll also illustrate these calculations with part of an example we consider below, on differences in blood pressure between two groups.
- Reference: Moore, David S., and George P. McCabe (1989). Introduction to the Practice of Statistics. Original source: Lyle, Roseann M., et al., "Blood pressure and metabolic effects of calcium supplementation in normotensive white and black men," JAMA, 257(1987), pp. 1772-1776.

- Description: Results of a randomized comparative experiment to investigate the effect of calcium on blood pressure in African-American men. A treatment group of 10 men received a calcium supplement for 12 weeks, and a control group of 11 men received a placebo during the same period. All subjects had their blood pressure tested before and after the 12-week period.
- Number of cases: 21
- Variable Names:
    - Treatment: Whether subject received calcium or placebo
    - Begin: seated systolic blood pressure before treatment
    - End: seated systolic blood pressure after treatment
    - Decrease: Decrease in blood pressure (Begin - End)

```
url = 'http://www.stanford.edu/class/stats191/data/Calcium.
calcium.table = read.table(url,
  header=TRUE, skip=26, nrow=21)
head(calcium.table)
```

```
##   Treatment Begin End Decrease
## 1   Calcium   107 100        7
## 2   Calcium   110 114       -4
## 3   Calcium   123 105       18
## 4   Calcium   129 112       17
## 5   Calcium   112 115       -3
## 6   Calcium   111 116       -5
```

- Number of observations in Calcium and Palcebo groups

```
library(dplyr)
library(magrittr)
class(calcium.table$Treatment)
```

```
## [1] "factor"
```

```
calcium.table %>%
  group_by(Treatment) %>%
  summarize(n())
```

```
## # A tibble: 2 x 2
##   Treatment `n()`
##   <fct>     <int>
## 1 Calcium      10
## 2 Placebo      11
```

- Mean blood pressure in `Calcium` and `Placebo` samples before and after treatment

```
calcium.table %>%
  group_by(Treatment) %>%
  summarize(mean.bp.before = mean(Begin),
    mean.bp.after = mean(End),
    mean.bp.decrease = mean(Decrease))
```

```
## # A tibble: 2 x 4
##   Treatment mean.bp.before mean.bp.after mean.bp.decreas
##   <fct>              <dbl>         <dbl>            <dbl
## 1 Calcium             115.          110.             5
## 2 Placebo             113.          114.            -0.27
```

Standard deviation of a sample

Given a sample of numbers $X = (X_1, \ldots, X_n)$ the sample standard deviation S_X is

$$S_X^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2.$$

```
calcium.table %>%
  group_by(Treatment) %>%
  summarize(sd.bp.before = sd(Begin),
    sd.bp.after = sd(End),
    sd.bp.decrease = sd(Decrease))
```

```
## # A tibble: 2 x 4
##   Treatment sd.bp.before sd.bp.after sd.bp.decrease
##   <fct>            <dbl>       <dbl>          <dbl>
## 1 Calcium          10.8         7.80           8.74
## 2 Placebo           9.02       11.3            5.90
```

Median of a sample

- Given a sample of numbers $X = (X_1, \ldots, X_n)$ the sample median is the `middle` of the sample:
  - if n is even, it is the average of the middle two points.
  - If n is odd, it is the midpoint.

```
calcium.table %>%
  group_by(Treatment) %>%
  summarize(median.bp.before = median(Begin),
    median.bp.after = median(End),
    median.bp.decrease = median(Decrease))
```

```
## # A tibble: 2 x 4
##   Treatment median.bp.before median.bp.after median.bp.d
##   <fct>                <dbl>           <dbl>
## 1 Calcium               112.             109
## 2 Placebo               112              114
```

Quantiles of a sample

- Given a sample of numbers $X = (X_1, \ldots, X_n)$ the q-th quantile is a point x_q in the data such that $q \cdot 100\%$ of the data lie to the left of x_q.
    - The 0.5-quantile is the median: half of the data lie to the right of the median.

```
calcium.table %>%
  group_by(Treatment) %>%
  summarize(thirdquar.bp.before =
      quantile(Begin, probs = .75),
    thirdquar.bp.after =
      quantile(End, probs = .75),
    thirdquar.bp.decrease =
      quantile(Decrease, probs = .75))
```

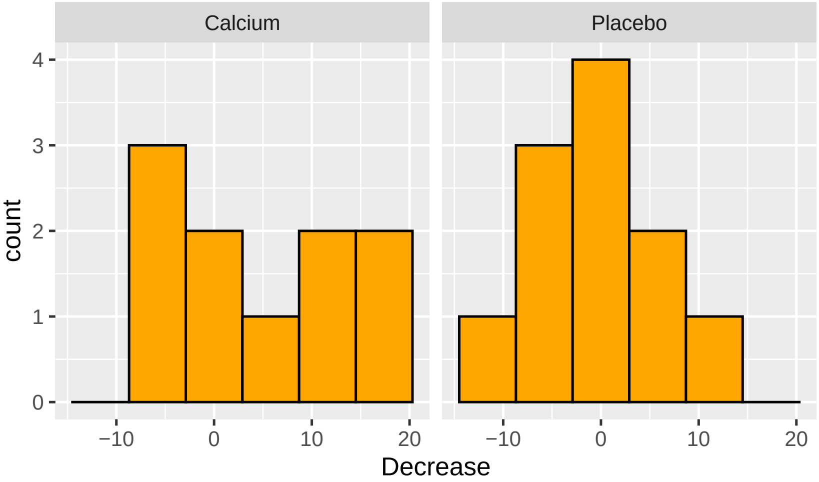
```
## # A tibble: 2 x 4
##   Treatment thirdquar.bp.before thirdquar.bp.after thirc
##   <fct>                   <dbl>              <dbl>
## 1 Calcium                  120.               115.
## 2 Placebo                  118                120
```

More graphical statistical summaries

- We've already seen a boxplot. Another common statistical summary is a histogram.

```r
p = ggplot(calcium.table,
  aes(x = Decrease,
    group = Treatment)) +
  geom_histogram(color = "black",
    fill = "orange", bins = 6) +
  facet_wrap(.~Treatment)
```

```
p
```



Inference about a population mean

A testing scenario

- Suppose we want to determine the efficacy of a new drug on blood pressure.
- Our study design is: we will treat a large patient population (maybe not so large: budget constraints limit it $n = 20$) with the drug and measure their blood pressure before and after taking the drug.
- We conclude that the drug is effective if the blood pressure has decreased on average. That is, if the average difference between before and after is positive.

Setting up the test

- The *null hypothesis*, H_0 is: *the average difference is less than or equal zero.*
- The *alternative hypothesis*, H_a , is: *the average difference is greater than zero.*
- Sometimes (actually, often), people will test the alternative, H_a : *the average difference is not zero* vs. H_0 : *the average difference is zero.*
- The test is performed by estimating the average difference and converting to standardized units.

Drawing from a box

- Formally, could set up the above test as drawing from a box of *differences in blood pressure*.
- A box model is a useful theoretical device that describes the experiment under consideration. In our example, we can think of the sample of decreases drawn 20 patients at random from a large population (box) containing all the possible decreases in blood pressure.

A simulated box model

- In our box model, we will assume that the decrease is an integer drawn at random from -3 to 6.
- We will draw 20 random integers from -3 to 6 with replacement and test whether the mean of our "box" is 0 or not.

```r
mysample = sample(-3:6, 20, replace=TRUE)
mysample
```

```
##  [1]  5  0  3 -3 -2  3 -2 -1 -3  1  1  6  2  6  3  5  1
```

- The test is usually a T test that uses the statistic

$$T = \frac{\overline{Y} - 0}{S_Y/\sqrt{n}}$$

- The formula can be read in three parts:
  - estimating the mean: $\overline{Y}$;
  - comparing to 0: subtracting 0 in the numerator;
  - converting difference to standardized units: dividing by $S_Y/\sqrt{n}$ our estimate of the variability of $\overline{Y}$.

```
T = (mean(mysample) - 0) / (sd(mysample) / sqrt(20))
T
```

```
## [1] 2.704154
```

- This T value is often compared to a table for the appropriate T distribution (in this case there are 19 *degrees of freedom*) and the 5% cutoff is

```
cutoff = qt(0.975, 19)
cutoff
```

```
## [1] 2.093024
```

- Strictly speaking the T distribution should be used when the values in the box are spread similarly to a normal curve.
- This is not the case here, but if n is large enough, there is not a huge difference.

```
qnorm(0.975)
```

```
## [1] 1.959964
```

- The result of the two-sided test is

```r
reject = (abs(T) > cutoff)
reject
```

```
## [1] TRUE
```

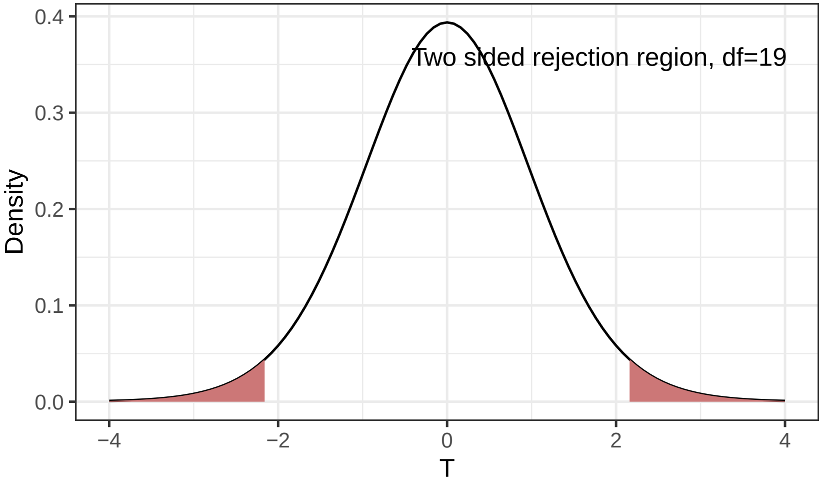
- If `reject` is `TRUE`, then we reject H_0 the mean is 0 at a level of 5%, while if it is `FALSE` we do not reject.
- Of course, in this example we know the mean in our “box” is not 0, it is 1.5.

- This rule can be visualized with the T density. The total grey area is 0.05=5%, and the cutoff is chosen to be symmetric around zero and such that this area is exactly 5%.
- For a test of size α we write this cutoff $t_{n-1,1-\alpha/2}$.

```r
library(ggplot2)
alpha = 0.05
df = 19
xval = seq(-4,4,length=101)
q = qt(1-alpha/2, df)

rejection_region = function(dens,
  q_lower, q_upper, xval) {
    fig = (ggplot(data.frame(x=xval), aes(x)) +
        stat_function(fun=dens, geom='line') +
        stat_function(fun=function(x) {
          ifelse(x > q_upper | x < q_lower,
            dens(x), NA)
          }, geom='area', fill='#CC7777') +
        labs(y='Density', x='T') +
        theme_bw())
    return(fig)
}
```

```r
T19_fig = rejection_region(function(x){dt(x, df)},
  -q, q, xval) +
  annotate('text',
    x=1.8, y = dt(2,df) + 0.3,
    label='Two sided rejection region, df=19')
```


Two sided rejection region, df=19
Density
0.4
0.3
0.2
0.1
0.0
−4
−2
0
2
4
T

Reasoning behind the test

- Suppose H_0 was true – say the mean of the box was zero.
- For example, we might assume the difference is drawn at random from integers -5 to 5 inclusive.

```r
# Generate a sample from a
# box for which the null is true
null_sample = function(n) {
    return(sample(-5:5, n, replace=TRUE))
}

# Compute the T statistic
null_T = function(n) {
    cur_sample = null_sample(n)
    return((mean(cur_sample) - 0) /
        (sd(cur_sample) / sqrt(n)))
}
```

Type I error

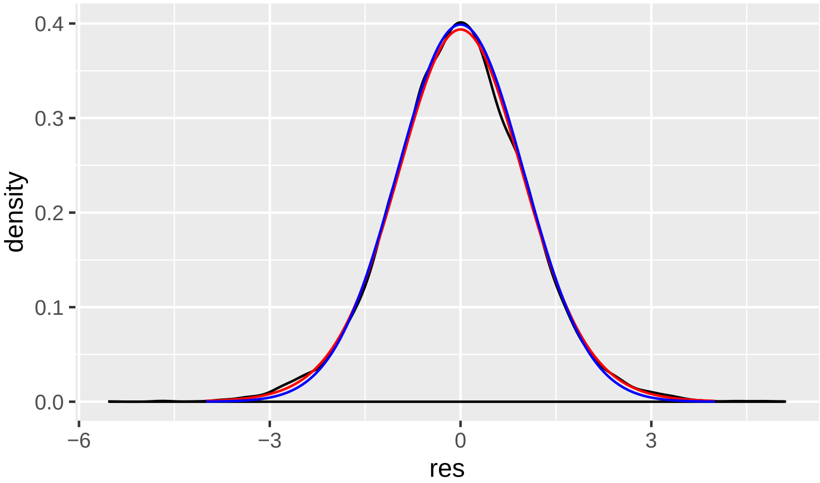
- When the null hypothesis is true, like in our simulation, we expect that the T statistic will exceed the cutoff only about 5% of the time.
- If we use the cutoff $t_{19,0.975}$ to decide in favor or against H_0, rejecting H_0 when the absolute value is larger than this value, then we have a test whose **Type I error** is about 5%.
- It is exactly 5% if the sample were drawn from a box whose values follow a normal curve...

```r
results = numeric(10000)
for (i in 1:10000) {
    results[i] = null_T(20)
}
mean(abs(results) >= qt(0.975, 19))
```

```
## [1] 0.052
```

- We use the T curve (close to the normal curve) because when H_0 is true, the distribution of the T statistic is close to the T curve.

```r
xval = seq(-4, 4, length=201)
df.temp = data.frame(xval = xval,
  dt.val = dt(xval, 19),
  dnorm.val = dnorm(xval))
p.den = ggplot() +
  geom_density(data = data.frame(res = results),
    aes(x = res, y = ..density..)) +
  geom_line(data = df.temp,
    aes(x = xval, y = dt.val),
    col = "red") + # T_19 density
  geom_line(data = df.temp,
    aes(x = xval, y = dnorm.val),
    col = "blue") # Normal(0,1) density
```


0.4
0.3
0.2
0.1
0.0
density
−6
−3
0
3
res

- R will compute this T statistic for you, and many other things.
- R will use the T distribution.

```r
t.test(mysample)
```

```
##
##  One Sample t-test
##
## data:  mysample
## t = 2.6975, df = 19, p-value = 0.01427
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  0.403366 3.196634
## sample estimates:
## mean of x
##       1.8
```

```r
T
```

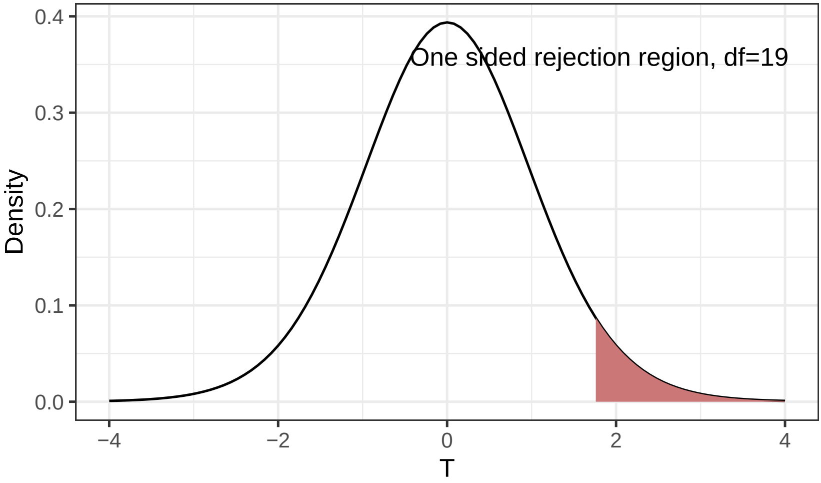
```
## [1] 2.704154
```

```r
2 * pt(abs(T), 19, lower = FALSE)
```

```
## [1] 0.01406267
```

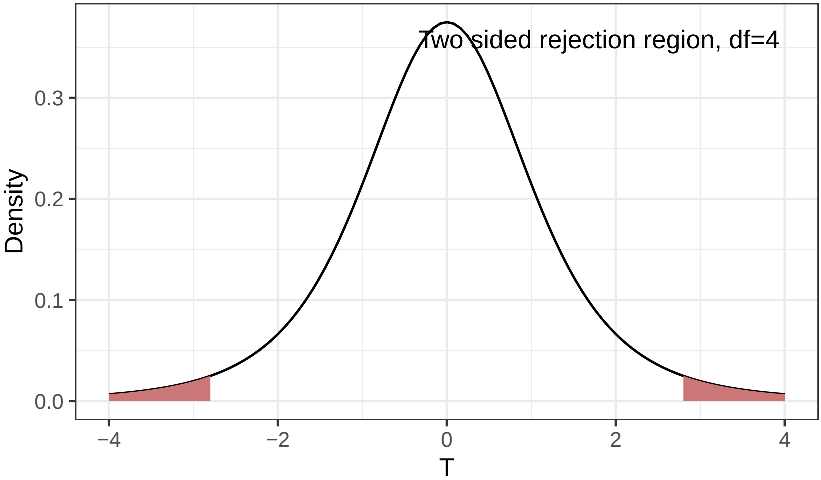
- As mentioned above, sometimes tests are one-sided.
- If the null hypothesis we tested was that the **mean is less than 0**, then we would reject this hypothesis if our observed mean was much larger than 0.
- This corresponds to a positive T value.

```r
cutoff = qt(0.95, 19)
T19_pos = rejection_region(function(x){dt(x, df)},
  -Inf, cutoff, xval) +
  annotate('text',
    x=1.8, y=dt(2,df)+0.3,
    label='One sided rejection region, df=19')
```


One sided rejection region, df=19
Density
0.4
0.3
0.2
0.1
0.0
−4
−2
0
2
4
T

- The rejection rules are affected by the degrees of freedom.
- Here is the rejection region when we only have 5 samples from our “box”.

```r
df = 4
cutoff = qt(0.975, df)
T4_fig = rejection_region(function(x) {dt(x, df)},
  -cutoff, cutoff, xval) +
  annotate('text',
    x=1.8,
    y=dt(2,19)+0.3,
    label='Two sided rejection region, df=4')
```


Two sided rejection region, df=4
Density
0.3
0.2
0.1
0.0
−4
−2
0
2
4
T

Confidence intervals

- Instead of testing a particular hypothesis, we might be interested in coming up with a reasonable range for the mean of our “box”.
- Statistically, this is done via a *confidence interval*.
- If the 5% cutoff is q for our test, then the 95% confidence interval is
$$[\bar{Y} - qS_Y/\sqrt{n}, \bar{Y} + qS_Y/\sqrt{n}],$$
where we recall $q = t_{n-1,0.975}$ with $n = 20$.
- If we wanted 90% confidence interval, we would use $q = t_{19,0.95}$. Why?

```r
cutoff = qt(0.975, 19)
L = mean(mysample) -
  cutoff*sd(mysample)/sqrt(20)
U = mean(mysample) +
  cutoff*sd(mysample)/sqrt(20)
data.frame(L, U)
```

```
##          L        U
## 1 0.403366 3.196634
```

```r
t.test(mysample)$conf.int
```

```
## [1] 0.403366 3.196634
## attr(,"conf.level")
## [1] 0.95
```

- There is at least 95% probability that the random interval $(0.40, 3.20)$ will contain the population mean.

OR

- If we keep repeating the experiment, we expect that approximately 95% of the times the confidence interval contains the population mean.

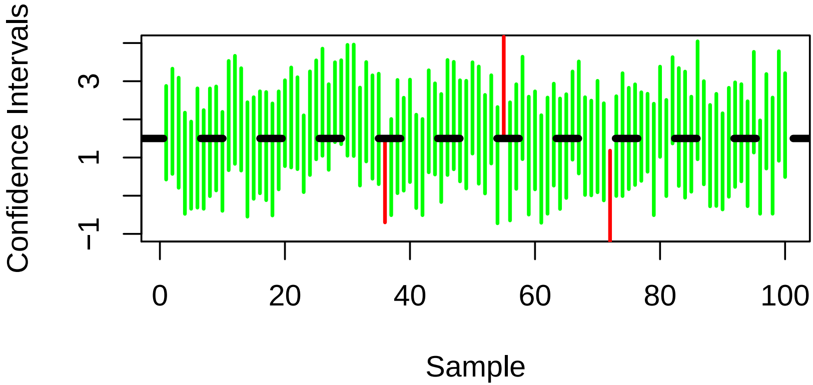
- Note that the endpoints above depend on the data.
    - Confidence intervals are random
- Not every interval will cover the true mean of our “box” which is 1.5.
- Let’s take a look at 100 intervals of size 90%. We would expect that roughly 90 of them cover 1.5.

```r
cutoff = qt(0.975, 19)
L = c()
U = c()
covered = c()
box = -3:6
for (i in 1:100) {
    mysample = sample(box, 20, replace=TRUE)
    l = mean(mysample) -
      cutoff*sd(mysample)/sqrt(20)
    u = mean(mysample) +
      cutoff*sd(mysample)/sqrt(20)
    L = c(L, l)
    U = c(U, u)
    covered = c(covered,
      (l < mean(box)) * (u > mean(box)))
}
sum(covered)
```

```
## [1] 97
```

- A useful picture is to plot all these intervals so we can see the randomness in the intervals, while the true mean of the box is unchanged.

```r
mu = 1.5
plot(c(1, 100), c(-2.5+mu, 2.5+mu),
  type='n',
  ylab='Confidence Intervals',
  xlab='Sample')
for (i in 1:100) {
   if (covered[i] == TRUE) {
       lines(c(i,i),
         c(L[i],U[i]), col='green', lwd=2)
   }
   else {
      lines(c(i,i),
        c(L[i],U[i]), col='red', lwd=2)
   }
}
abline(h=mu, lty=2, lwd=4)
```


Confidence Intervals
3
1
−1
0
20
40
60
80
100
Sample

Blood pressure example

- A study was conducted to study the effect of calcium supplements on blood pressure.
- We had loaded the data above (calcium.table).
- The two samples in the variables `treated` and `placebo`.

```
treated = calcium.table %>%
  filter(Treatment == "Calcium") %>%
  .$Decrease %>%
  as.numeric()
placebo = calcium.table %>%
  filter(Treatment == "Placebo") %>%
  .$Decrease %>%
  as.numeric()
```

- Some questions might be:
    - What is the mean decrease in BP in the treated group? placebo group?
    - What is the median decrease in BP in the treated group? placebo group?
    - What is the standard deviation of decrease in BP in the treated group? placebo group?
    - Is there a difference between the two groups? Did BP decrease more in the treated group?

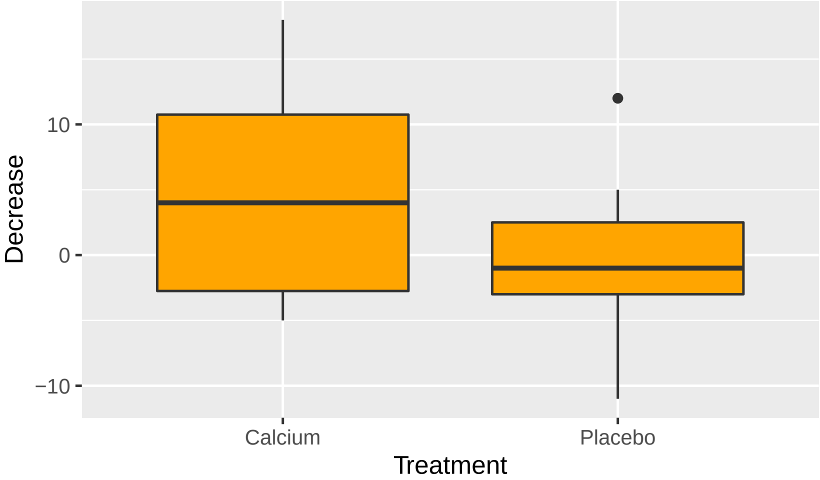
```r
summary(treated)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   -5.00   -2.75    4.00    5.00   10.75   18.00
```

```r
summary(placebo)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -11.0000  -3.0000  -1.0000  -0.2727   2.5000  12.0000
```

```r
p.cal = ggplot(data = calcium.table) +
  geom_boxplot(aes(x = Treatment,
    y = Decrease),
    fill = "orange")
```


10
0
−10
Decrease
Calcium
Placebo
Treatment

A hypothesis test

- In our setting, we have two groups that we have reason to believe are different.
- We have two samples:
  - $(X_1, \ldots, X_{10})$ (`treated`)
  - $(Z_1, \ldots, Z_{11})$ (`placebo`)
- We can answer this statistically by testing the null hypothesis

$$H_0 : \mu_X = \mu_Z.$$

- If variances are equal, the *pooled t-test* is appropriate.

Pooled t test

- The test statistic is

$$T = \frac{\overline{X} - \overline{Z} - 0}{S_P \sqrt{\frac{1}{10} + \frac{1}{11}}}, \qquad S_P^2 = \frac{9 \cdot S_X^2 + 10 \cdot S_Z^2}{19}.$$

- For two-sided test at level $\alpha = 0.05$, reject if $|T| > t_{19,0.975}$.
- Confidence interval: for example, a 90% confidence interval for $\mu_X - \mu_Z$ is

$$\overline{X} - \overline{Z} \pm S_P \sqrt{\frac{1}{10} + \frac{1}{11}} \cdot t_{19,0.95}.$$

- T statistic has the same form as before!

```r
sdP = sqrt((9*sd(treated)^2 +
    10*sd(placebo)^2)/19)
T = (mean(treated)-mean(placebo)-0) /
  (sdP * sqrt(1/10+1/11))
c(T, cutoff)
```

```
## [1] 1.634108 2.093024
```

- R has a built-in function to perform such *t*-tests.

```r
t.test(treated, placebo, var.equal=TRUE)
```

```
##
##  Two Sample t-test
##
## data:  treated and placebo
## t = 1.6341, df = 19, p-value = 0.1187
## alternative hypothesis: true difference in means is not
## 95 percent confidence interval:
##  -1.48077 12.02622
## sample estimates:
##  mean of x  mean of y
##  5.0000000 -0.2727273
```

- If we don't make the assumption of equal variance, `R` will give a slightly different result.

```r
t.test(treated, placebo)
```

```
##
##  Welch Two Sample t-test
##
## data:  treated and placebo
## t = 1.6037, df = 15.591, p-value = 0.1288
## alternative hypothesis: true difference in means is not
## 95 percent confidence interval:
##  -1.712039 12.257493
## sample estimates:
##  mean of x  mean of y
##  5.0000000 -0.2727273
```

Pooled estimate of variance

- The rule for the *SD* of differences is

$$SD(\overline{X} - \overline{Z}) = \sqrt{SD(\overline{X})^2 + SD(\overline{Z})^2}$$

- By this rule, we might take our estimate to be

$$\widehat{SD(\overline{X} - \overline{Z})} = \sqrt{\frac{S_X^2}{10} + \frac{S_Z^2}{11}}.$$

- The pooled estimate assumes $\mathbb{E}(S_X^2) = \mathbb{E}(S_Z^2) = \sigma^2$ and replaces the S^2's above with S_P^2, a better estimate of σ^2 than either S_X^2 or S_Z^2.

Where do we get $df = 19$?

- Well, the X sample has $10 - 1 = 9$ degrees of freedom to estimate σ^2 while the Z sample has $11 - 1 = 10$ degrees of freedom.
- Therefore, the total degrees of freedom is $9 + 10 = 19$.

Our first regression model

- We can put the two samples together:

$$Y = (X_1, \ldots, X_{10}, Z_1, \ldots, Z_{11}).$$

- Under the same assumptions as the pooled *t*-test:

$$\begin{aligned} Y_i &\sim N(\mu_i, \sigma^2) \\ \mu_i &= \begin{cases} \mu_X & 1 \leq i \leq 10 \\ \mu_Z & 11 \leq i \leq 21. \end{cases} \end{aligned}$$

- This is a (regression) model for the sample Y. The (qualitative) variable `Treatment` is called a *covariate* or *predictor*.
- The decrease in BP is the *outcome*.
- We assume that the relationship between treatment and average decrease in BP is simple: it depends only on which group a subject is in.
- This relationship is *modeled* through the mean vector $\mu = (\mu_1, \dots, \mu_{21})$.

```
print(summary(lm(Decrease ~ Treatment, data = calcium.table

##
## Call:
## lm(formula = Decrease ~ Treatment, data = calcium.table)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.7273  -4.7273  -0.7273   5.0000  13.0000
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)         5.000      2.335   2.141   0.0454 *
## TreatmentPlacebo   -5.273      3.227  -1.634   0.1187
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.
##
## Residual standard error: 7.385 on 19 degrees of freedom
## Multiple R-squared:  0.1232, Adjusted R-squared:  0.077
## F-statistic:  2.67 on 1 and 19 DF,  p-value: 0.1187
```

```r
print(sdP*sqrt(1/10+1/11))
```

```
## [1] 3.22667
```

```r
print(sdP)
```

```
## [1] 7.384842
```

References for this lecture

- Based on the lecture notes of Jonathan Taylor .