Regression

Put your name here

Put the date here

Introduction

In this assignment we will learn how to run a regression analysis. Regression provides a model for the linear relationship between two numerical variables.

Instructions

Presumably, you have already created a new project and downloaded this file into it. Please knit the document and work back and forth between this R Markdown file and the PDF output as you work through this module.

When you are finished with the assignment, knit to PDF one last time, proofread the PDF file **carefully**, export the PDF file to your computer, and then submit your assignment.

Sometimes you will be asked to add your own R code. That will appear in this document as a code chunk with a request for you to add your own code, like so:

```
## Add code here to [do some task]...
```

Be sure to remove the line `## Add code here to [do some task]...` when you have added your own code.

Sometimes you will be asked to type up your thoughts. That will appear in the document as follows:

Please write up your answer here.

Again, please be sure to remove the line "Please write up your answer here" when you have written up your answer. In these areas of the assignment, please use contextually meaningful full sentences/paragraphs (unless otherwise indicated) and proper spelling, grammar, punctuation, etc. This is not R code, but rather a free response section where you talk about your analysis and conclusions. If you need to use some R code as well, you can use inline R code inside the block between `\begin{answer}` and `\end{answer}`, or if you need an R code chunk, please go outside the `answer` block and start a new code chunk.

Load Packages

We load the standard `mosaic` package as well as the `openintro` package for the `bdims` data, the `reshape2` package for the `tips` data, and the `MASS` package for the `Rubber` data. The `broom` package gives us tidy output.

```
library(openintro)
library(reshape2)
library(MASS)
library(broom)
library(mosaic)
```

Research question

Can you predict someone's weight from their wrist diameter?

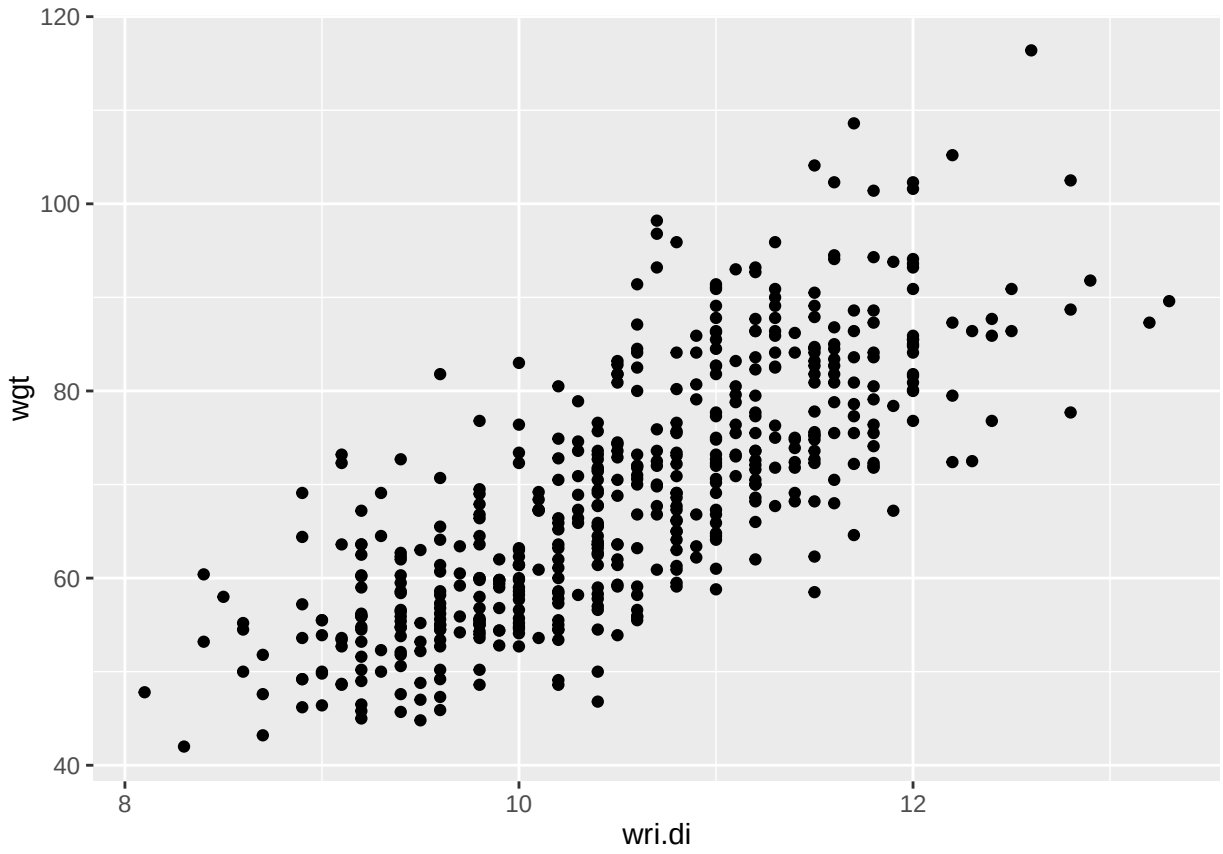
Regression

When we have a linear relationship between two numerical variables, it's helpful to model this relationship with an actual straight line. Such a line is called a regression line, or a best-fit line, or sometimes a least-squares line.

The mathematics involved in figuring out what this line should be is more complicated than we cover in this course. But R will do all the complicated calculations for us.

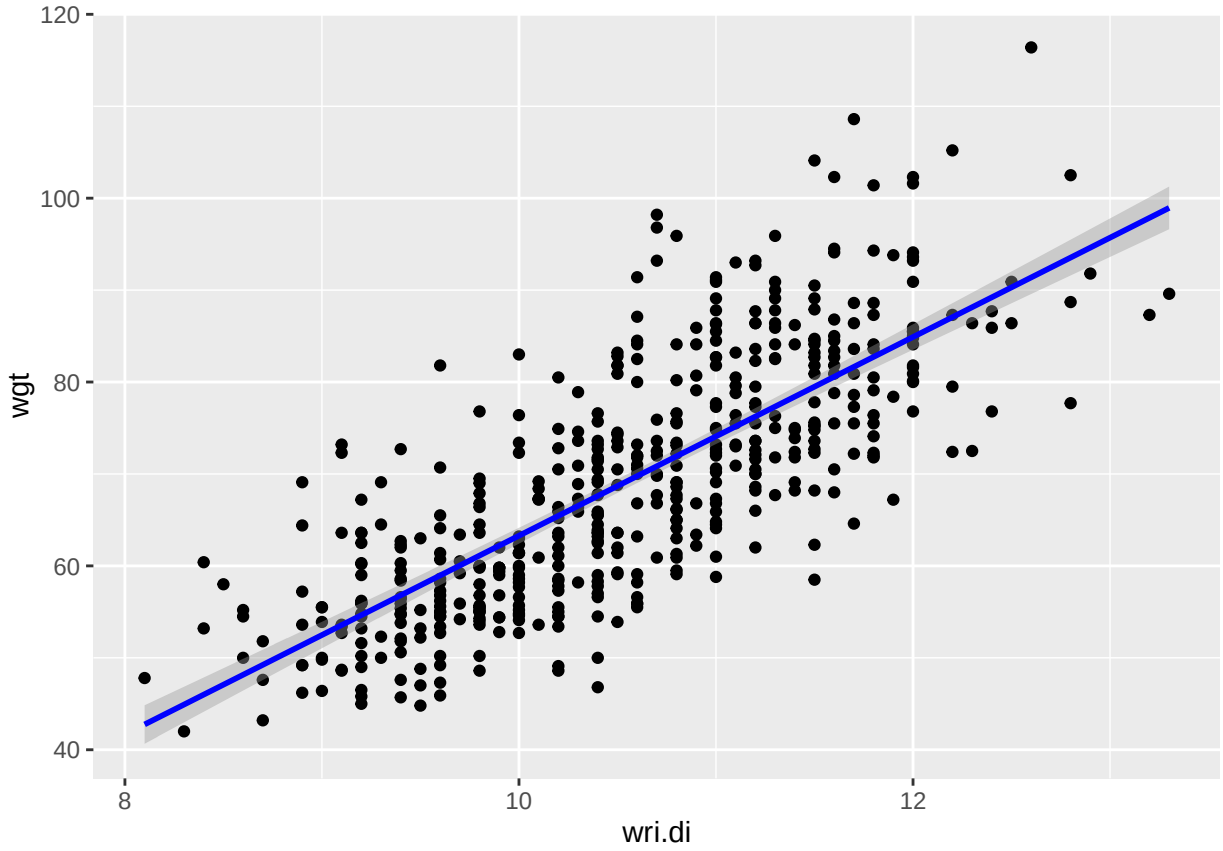
Let's look at a scatterplot of data from the `bdims` data set. These measurements come from 507 physically active individuals (a mix of males and females). The wrist diameter `wri.di` is measured in centimeters as the sum of both wrists, and the weight `wgt` is measured in kilograms.

```
ggplot(bdims, aes(x = wri.di, y = wgt)) +
    geom_point()
```



If certain conditions are met, we can graph a regression line; just add a `geom_smooth` layer to the scatterplot:

```
ggplot(bdims, aes(x = wri.di, y = wgt)) +
    geom_point() +
    geom_smooth(method = "lm", color = "blue")
```



The `method = "lm"` argument is telling `ggplot` to use a "linear model".

Of all possible lines, the blue line comes the closest to each point in the scatterplot. If we wiggled the line a little bit, it might get closer to a few points, but the net effect would be to make it further from other points. This is the mathematically optimal line of best fit. The gray region around the line is called a confidence band; the "true" population regression line is likely to lie inside this gray area.

What is the equation of this line? In your algebra class you learn that a line takes the form $y = mx + b$ where m is the slope and b is the y-intercept. Statisticians write the equation in a slightly different form:

$$\hat{y} = b_0 + b_1 x.$$

The intercept is b_0 and the slope is b_1. We use $\hat{y}$ instead of y because plugging in values of x does not give us back the exact values of y from the data. The line, after all, does not actually pass through most (if any) actual data points. Instead, this equation gives us "predicted" values of y that lie on the regression line. These predicted y values are called $\hat{y}$.

To run a regression analysis, we use the `lm` command in R. (`lm` stands for "linear model".) We wrap the results in the `tidy` command to make the output a little cleaner.

```
wrist_wt_tidy <- tidy(lm(wgt ~ wri.di,  data = bdims))
```

Interpreting the coefficients

Here is the output of the `tidy` command:

```
wrist_wt_tidy
```

```
##          term  estimate std.error statistic      p.value
## 1 (Intercept) -44.77000 4.2900420 -10.43580 3.214368e-23
## 2      wri.di  10.80545 0.4053047  26.66005 2.104556e-98
```

The `estimate` column of the output gives us the intercept and slope for the model regression line.

```
wrist_wt_tidy$estimate[1]
```

```
## [1] -44.77
```

```
wrist_wt_tidy$estimate[2]
```

```
## [1] 10.80545
```

The intercept is -44.7700047 and the slope is 10.8054466.

Therefore, the equation of the regression line could be written

$$\hat{y} = -44.77 + 10.81x.$$

When we report the equation of the regression line, we typically use words instead of x and y to make the equation more interpretable in the context of the problem. For example, for this data, we would write the equation as

$$\widehat{weight} = -44.77 + 10.81wrist.$$

The slope b_1 is always interpretable. This model predicts that one unit of increase in the x-axis corresponds to a change of 10.8054466 units in the y-direction. Let's phrase it this way:

> The model predicts that an increase of one cm in wrist diamater corresponds to a weight increase of 10.8054466 kg.

The intercept b_0 is a different story. There is always a literal interpretation:

> The model predicts that someone with a wrist diameter of zero will weigh -44.7700047 kg.

It is true that the model makes that prediction, so the preceding sentence is technically correct. However, that prediction is nonsensical. Aside from the fact that it is impossible for a wrist to have zero diameter and impossible for a weight to be negative, this is extrapolation—in other words, a prediction outside the range of the data.

The rest of the `tidy` output involves inference on the two parameters (intercept and slope). We'll get back to that later.

Checking conditions

We need to be careful here. We have not checked any conditions; therefore, it is inappropriate to fit a regression line yet. Once the line is seen, it cannot easily be "unseen", and it's crucial that you don't trick your reader into believing there is a linear relationship before checking the conditions that justify that belief.

The regression line we saw above makes no sense unless we know that regression is appropriate. The conditions for running a regression analysis include all the conditions you checked for a correlation analysis: "Random" and "10%" to ensure a good sample, "Linear association" to make sure our association follows a linear pattern, and an "Outlier" check because outliers can be bad. Let's check what he have so far:

- Random
  - We are not told how the sample was collected, so we can't say if it's a representative sample of all physically active individuals. We'll proceed with caution.
- 10%
  - Surely 507 people are less than 10% of all physically active people.
- Linear association
  - The scatterplot showed a clear linear relationship between wrist diameter and weight.
- Outliers
  - There are no apparent outliers in the scatterplot.

However, there is an additional condition to check to ensure that our regression model is appropriate. We need to check for...

- Patterns in the residuals

We discuss this below.

Residuals

Residuals are the vertical distances from each data point to the regression line. More formally, we're saying that the residual e is given by the following formula:

$$e = y - \hat{y}.$$

We know that some of the points are going to lie above the line (positive residuals) and some of the points will lie below the line (negative residuals). What we need is for there not to be any pattern among these residuals.

To calculate the residuals, we introduce a new function from the `broom` package. Whereas `tidy` serves up information about the parameters of interest (in this case, the intercept and the slope of the regression line), `augment` gives us extra information for each data point.

```
wrist_wt_aug <- augment(lm(wgt ~ wri.di,  data = bdims))
```

Examine the first few rows of output:

```
head(wrist_wt_aug)
```

```
##    wgt wri.di  .fitted   .se.fit     .resid        .hat   .sigma
## 1 65.6   10.4 67.60664 0.3867199  -2.006639 0.002017451 8.617911
## 2 71.8   11.8 82.73426 0.6371288 -10.934265 0.005476013 8.604526
## 3 80.7   10.9 73.00936 0.4088942   7.690637 0.002255444 8.611549
## 4 72.6   11.2 76.25100 0.4660527  -3.650997 0.002930084 8.616836
## 5 78.8   11.6 80.57318 0.5743536  -1.773175 0.004450089 8.618012
## 6 74.8   11.5 79.49263 0.5447794  -4.692631 0.004003607 8.615830
##        .cooksd .std.resid
## 1 5.501433e-05 -0.2332990
## 2 4.464709e-03 -1.2734648
## 3 9.038532e-04  0.8942473
## 4 2.649913e-04 -0.4246720
## 5 9.521952e-05 -0.2064074
## 6 5.994441e-04 -0.5461256
```

The first two columns are the actual data values we started with. But now we've "augmented" the original data with some new stuff too. The third column—here called `.fitted`—is $\hat{y}$, or the point on the line that corresponds to the given x value. Let's check and make sure this is working as advertised.

The regression equation from above is

$$\widehat{weight} = -44.77 + 10.81 wrist.$$

The first subject listed in row 1 has wrist diameter 10.4 cm. Plug that value into the equation above:

$$\widehat{weight} = -44.77 + 10.81(10.4) = 67.6.$$

The model predicts that a person with a wrist diameter of 10.4 will weigh 67.6 kg. The first number in the `.fitted` column is 67.60664, so that's correct.

Now skip over to the fifth column of the `augment` output, the one that says `.resid`. If this is the residual e, then it should be $y - \hat{y}$. Since y is the actual value of `wgt` and $\hat{y}$ is the value predicted by the model, we should get for the first row of output

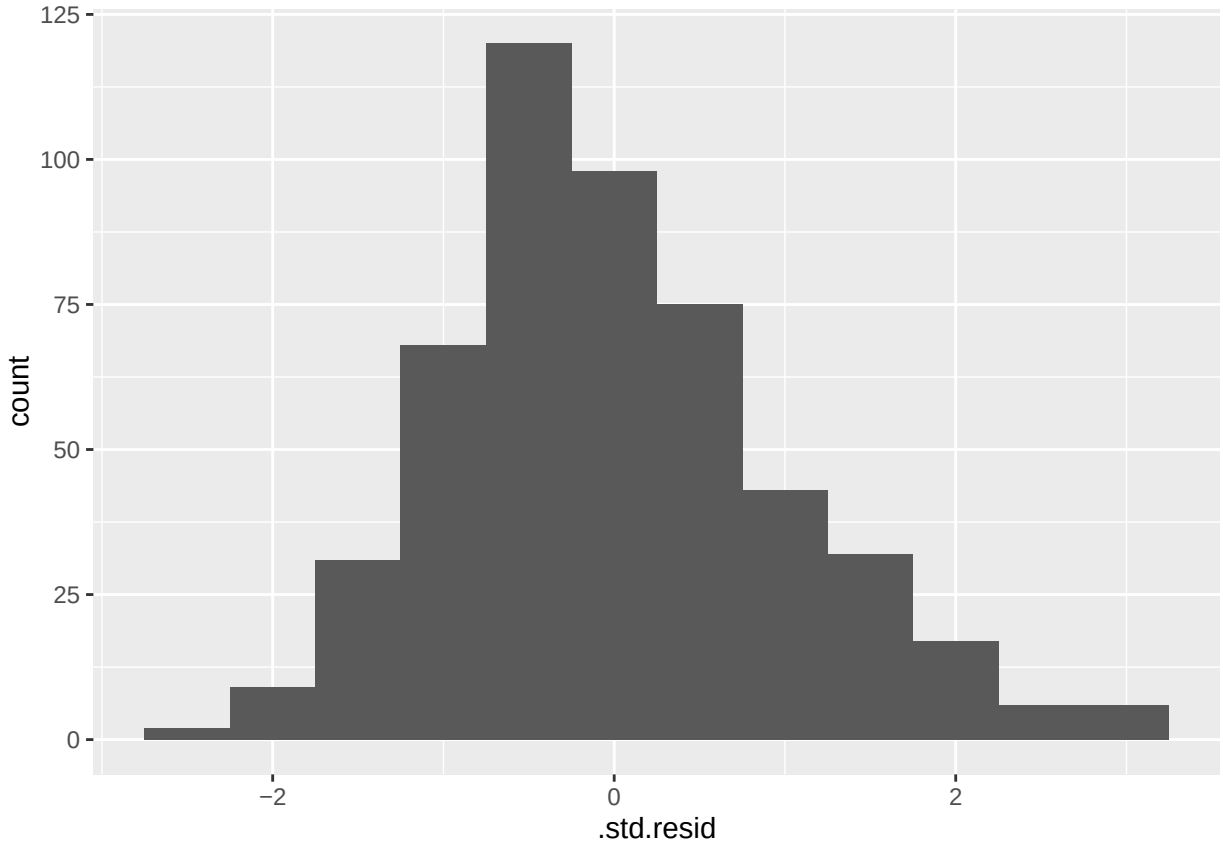
$$e = y - \hat{y} = 65.6 - 67.60664 = -2.00664.$$

Yup, it works!

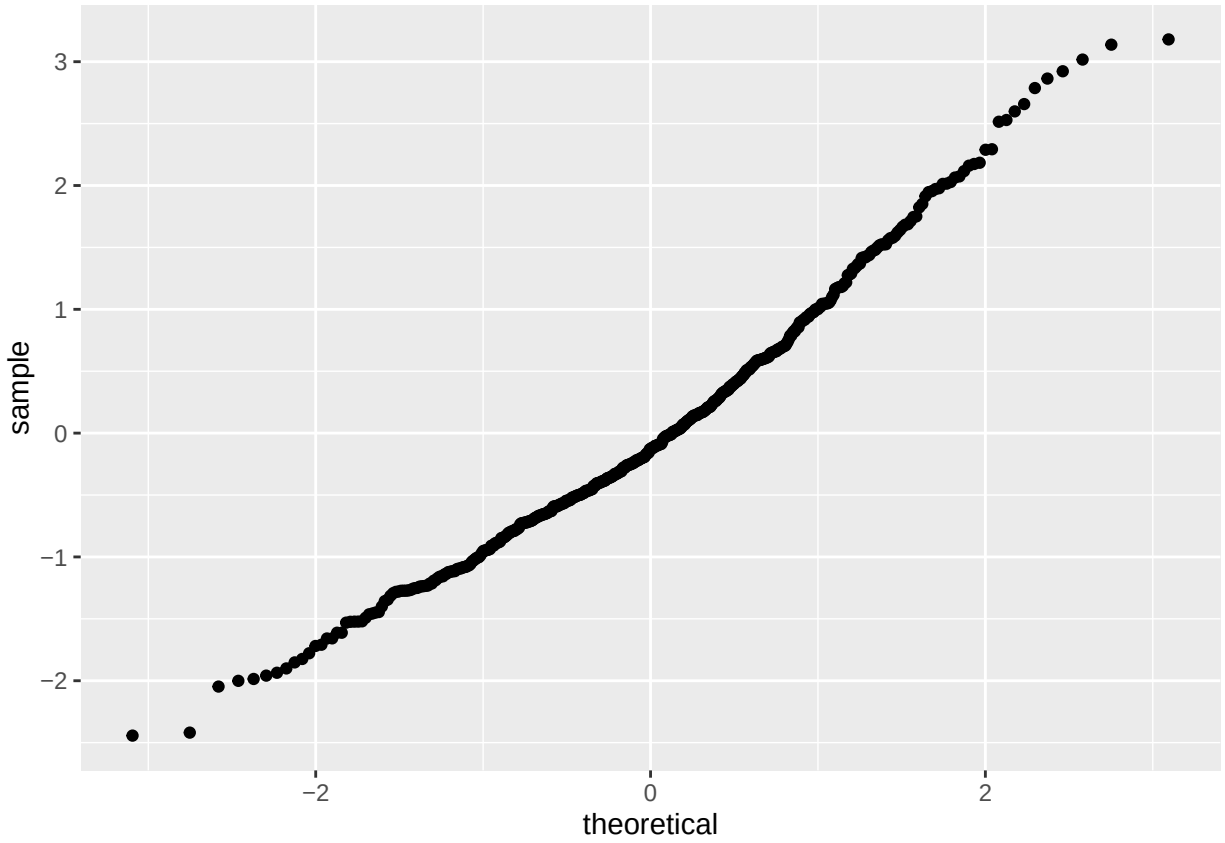
Skip to the last column of the output `.std.resid`. These are just the z-scores of the residuals, called "standardized residuals". It is common to use these to create our graphs. As z-scores, they are easy to interpret.

To check for patterns in the residuals, we'll use several different plots. First, we want our residuals to be normally distributed. We check this with a histogram and a QQ plot, as usual.

```
ggplot(wrist_wt_aug, aes(x = .std.resid)) +
    geom_histogram(binwidth = 0.5)
```



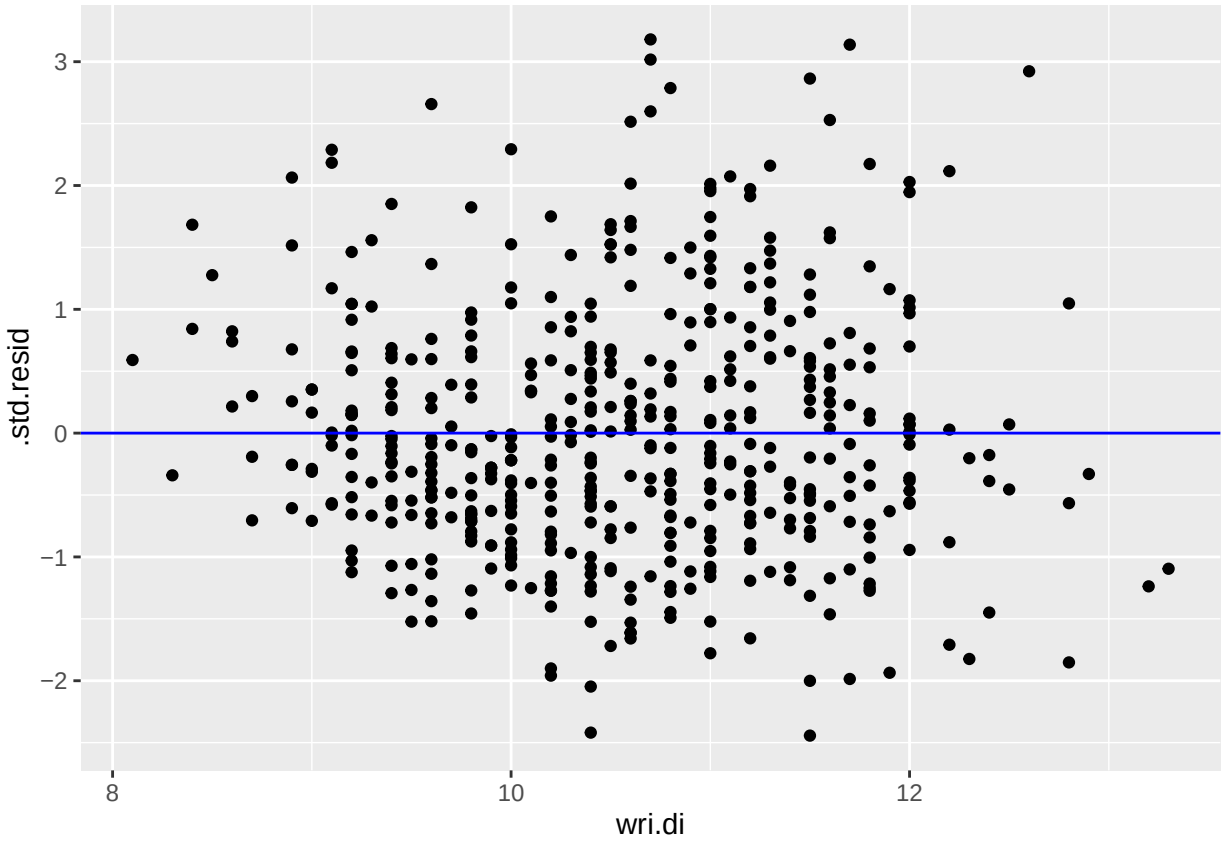

```
ggplot(wrist_wt_aug, aes(sample = .std.resid)) +
    geom_qq()
```



We see that the shape is mostly normal. There's a little bit of skew in the histogram, but nothing alarming.

We should also create a *residual plot*, which looks at the residuals above each value along the x-axis. (In the command below, we also add a horizontal reference line so that it is clear which points have positive or negative residuals.)

```
ggplot(wrist_wt_aug, aes(x = wri.di, y = .std.resid)) +
    geom_point() +
    geom_hline(yintercept = 0, color = "blue")
```

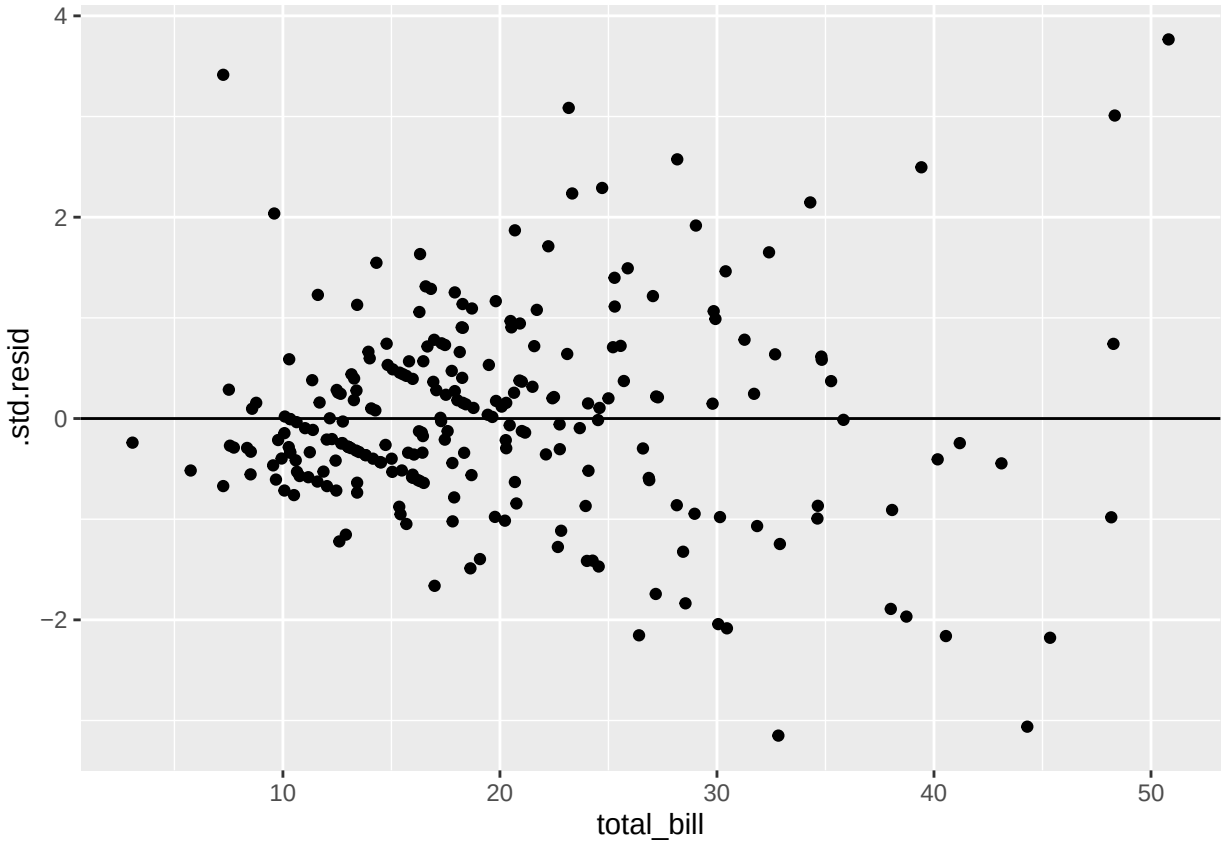


This looks good. There are no systematic patterns in the residuals. A residual plot should look like the most boring plot you've ever seen.

Residual patterns that are problematic often involve curved data (where the dots follow a curve around the horizontal reference line instead of spreading evenly around it) and *heteroscedasticity*, which is a fanning out pattern.

As an example of the latter, let's look at the residual plot for the tipping example from the correlation module.

```
tips <- reshape2::tips
bill_tip_aug <- augment(lm(tip ~ total_bill, data = tips))
ggplot(bill_tip_aug, aes(x = total_bill, y = .std.resid)) +
    geom_point() +
    geom_hline(yintercept = 0)
```



The residuals are quite small toward the left end of the residual plot, and then spread out and get larger toward the right end. This is a violation of the "patterns in the residuals" condition. It would be problematic to pursue a regression analysis of the tip data even though correlation was okay to report.

R^2

The correlation coefficient R is of limited utility. The number doesn't have any kind of intrinsic meaning; it can only be judged by how close it is to 0 or 1 in conjunction with a scatterplot to give you a sense of the strength of the correlation. In particular, some people try to interpret R as some kind of percentage, but it's not.

On the other hand, R^2 can be interpreted as a percentage. It represents the percent of variation in the y variable that can be explained by variation in the x variable.

Here we introduce the last of the `broom` functions: `glance`. Whereas `tidy` reports summary statistics related to parameters of the model, and `augment` reports values associated to each data point separately, the `glance` function gathers up summaries for the entire model.

```
wrist_wt_glance <- glance(lm(wgt ~ wri.di,  data = bdims))
wrist_wt_glance
```

```
##   r.squared adj.r.squared    sigma statistic      p.value df    logLik
## 1 0.5846215     0.5837989 8.609838  710.7585 2.104556e-98  2 -1809.923
##        AIC      BIC deviance df.residual
## 1 3625.846 3638.531  37435.3         505
```

A more advanced statistics course might discuss the other model summaries present in the `glance` output. The R^2 value is stored in `wrist_wt_glance$r.squared`. Its value is 0.5846215. We will word it this way:

> 58.4621457% of the variability in weight can be explained by variability in wrist diameter.

Thus, R^2 is a measure of the fit of the model. High values of R^2 mean that the line predicts the data values closely, whereas lower values of R^2 mean that there is still a lot of variability left in the residuals (presumably due to other factors that are not measured here). What we're saying here is that there are lots of factors that account for variability in weight, and not all that can be explained by the variability in wrist diameter.

Inference for the regression slope

The sample gives us the regression equation

$$\hat{y} = b_0 + b_1 x.$$

The idea of inference is that this line is meant to be an estimate of a true regression line

$$\hat{y} = \beta_0 + \beta_1 x.$$

In other words, if we plotted weight against wrist diameters for everybody in the world, there is, in theory, some perfect regression line that goes through the middle of all those points. And that ideal intercept and slope are the true population parameters β_0 and β_1. (β is the Greek letter "beta".)

We have already seen that the intercept is not particularly interesting, so we restrict attention to inference for the slope. As with correlation, the full inferential rubric for the regression slope is a little overkill. Nevertheless, the rubric forces us to be careful to identify our sample and population and state proper conclusions.

The sampling distribution for the slope parameter is somewhat complicated. There is a formula for the standard error of the sample slope estimates, and with that, one can compute t scores that are distributed as a t model with $n - 2$ degrees of freedom. We won't get into any of the mathematical details here.

Also note that a typical regression analysis will start by checking conditions so that the regression line can be calculated and graphed. Therefore, we will work through the rubric under the assumption that the conditions have already been checked.

Exploratory data analysis

Use data documentaton (help files, code books, Google, etc.), the str command, and other summary functions to understand the data.

[Type `library(openintro)`, then `?bdims` at the Console to read the help file.]

```
str(bdims)
```

```
## 'data.frame':    507 obs. of  25 variables:
##  $ bia.di: num  42.9 43.7 40.1 44.3 42.5 43.3 43.5 44.4 43.5 42 ...
##  $ bii.di: num  26 28.5 28.2 29.9 29.9 27 30 29.8 26.5 28 ...
##  $ bit.di: num  31.5 33.5 33.3 34 34 31.5 34 33.2 32.1 34 ...
##  $ che.de: num  17.7 16.9 20.9 18.4 21.5 19.6 21.9 21.8 15.5 22.5 ...
##  $ che.di: num  28 30.8 31.7 28.2 29.4 31.3 31.7 28.8 27.5 28 ...
##  $ elb.di: num  13.1 14 13.9 13.9 15.2 14 16.1 15.1 14.1 15.6 ...
```

```
##  $ wri.di: num  10.4 11.8 10.9 11.2 11.6 11.5 12.5 11.9 11.2 12 ...
##  $ kne.di: num  18.8 20.6 19.7 20.9 20.7 18.8 20.8 21 18.9 21.1 ...
##  $ ank.di: num  14.1 15.1 14.1 15 14.9 13.9 15.6 14.6 13.2 15 ...
##  $ sho.gi: num  106 110 115 104 108 ...
##  $ che.gi: num  89.5 97 97.5 97 97.5 ...
##  $ wai.gi: num  71.5 79 83.2 77.8 80 82.5 82 76.8 68.5 77.5 ...
##  $ nav.gi: num  74.5 86.5 82.9 78.8 82.5 80.1 84 80.5 69 81.5 ...
##  $ hip.gi: num  93.5 94.8 95 94 98.5 95.3 101 98 89.5 99.8 ...
##  $ thi.gi: num  51.5 51.5 57.3 53 55.4 57.5 60.9 56 50 59.8 ...
##  $ bic.gi: num  32.5 34.4 33.4 31 32 33 42.4 34.1 33 36.5 ...
##  $ for.gi: num  26 28 28.8 26.2 28.4 28 32.3 28 26 29.2 ...
##  $ kne.gi: num  34.5 36.5 37 37 37.7 36.6 40.1 39.2 35.5 38.3 ...
##  $ cal.gi: num  36.5 37.5 37.3 34.8 38.6 36.1 40.3 36.7 35 38.6 ...
##  $ ank.gi: num  23.5 24.5 21.9 23 24.4 23.5 23.6 22.5 22 22.2 ...
##  $ wri.gi: num  16.5 17 16.9 16.6 18 16.9 18.8 18 16.5 16.9 ...
##  $ age   : int  21 23 28 23 22 21 26 27 23 21 ...
##  $ wgt   : num  65.6 71.8 80.7 72.6 78.8 74.8 86.4 78.4 62 81.6 ...
##  $ hgt   : num  174 175 194 186 187 ...
##  $ sex   : int  1 1 1 1 1 1 1 1 1 1 ...
```

```
favstats(bdims$wri.di)
```

```
##  min  Q1 median   Q3  max    mean       sd   n missing
##  8.1 9.8   10.5 11.2 13.3 10.5426 0.944361 507       0
```

```
favstats(bdims$wgt)
```

```
##  min   Q1 median    Q3   max     mean       sd   n missing
##   42 58.4   68.2 78.85 116.4 69.14753 13.34576 507       0
```

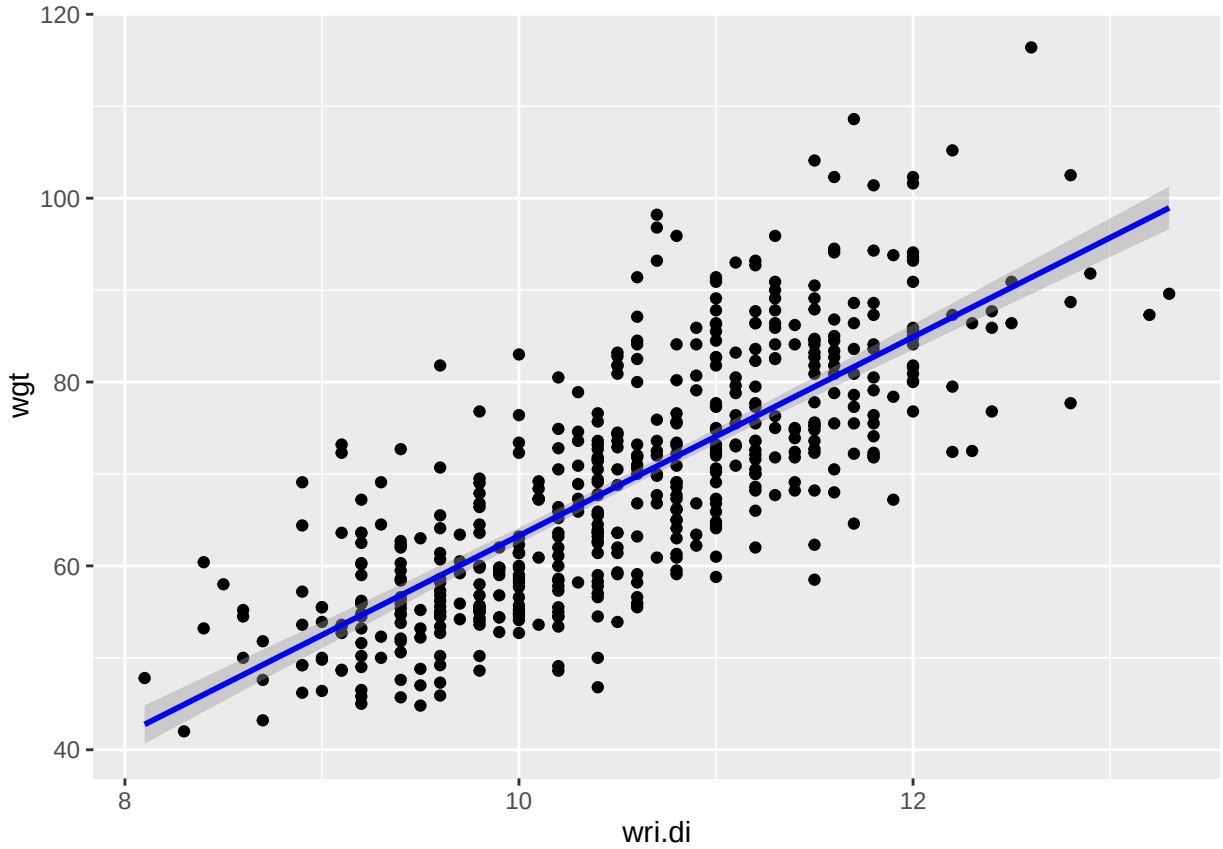
Prepare the data for analysis.

The two variables of interest, `wri.di` and `wgt` are already coded as numerical variables.

Make tables or plots to explore the data visually.

Here is the scatterplot with the regression line added:

```
ggplot(bdims, aes(x = wri.di, y = wgt)) +
    geom_point() +
    geom_smooth(method = "lm",  color = "blue")
```



Commentary: It would be inappropriate to show this regression line on the graph before the conditions have been checked. However, most regression analyses start by checking the conditions and writing down the equation of the regression line before doing inference on the slope parameter.

Hypotheses

Identify the sample (or samples) and a reasonable population (or populations) of interest.

The sample consists of 507 physically active individuals (a mix of females and males). The population is all physically active individuals.

Express the null and alternative hypotheses as contextually meaningful full sentences.

H_0: There is no relationship between wrist diameter and weight.

H_A: There is a relationship between wrist diameter and weight.

Express the null and alternative hypotheses in symbols.

$H_0 : \beta_1 = 0$

$H_A : \beta_1 \neq 0$

Commentary: We are performing a two-sided test here. One could perform a one-sided test if the question of interest was about a positive or a negative slope specifically. Unless otherwise specified, though, the default is to run a two-sided test.

Model

Identify the sampling distribution model.

We use a t model with 505 degrees of freedom.

Check the relevant conditions to ensure that model assumptions are met.

The conditions have already been checked.

Mechanics

Compute the test statistic.

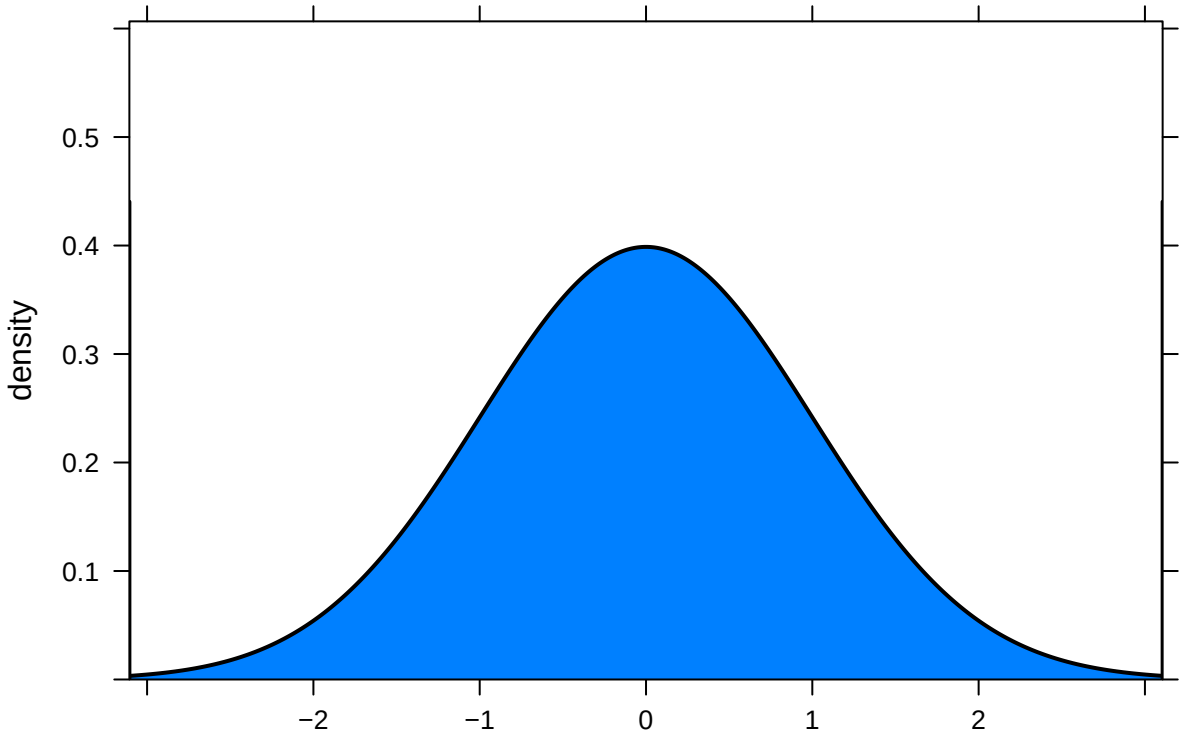
```
t <- wrist_wt_tidy$statistic[2]
```

The sample slope estimate has a t score of 26.6600546.

Commentary: `wrist_wt_tidy` has t scores for both the intercept and the slope. Be sure to grab the 2nd entry to get the slope test statistic.

Plot the null distribution.

```
pdist("t", df = wrist_wt_glance$df.residual, q =  c(-t, t))
```




```
## [1] 1.052278e-98 1.000000e+00
```

Commentary: The correct degrees of freedom are stored in `wrist_wt_glance$df.residual`, not `wrist_wt_glance$df` as you might expect.

Calculate the P-value.

```
wrist_wt_tidy$p.value[2]
```

```
## [1] 2.104556e-98
```

$P < 0.001$

Commentary: Not only are there two P-values in `wrist_wt_tidy` (one for the intercept and one for the slope), but there's one in `wrist_wt_glance` too. The one in the `glance` output, though, is the same as the P-value for the slope, so either `wrist_wt_tidy$p.value[2]` or `wrist_wt_glance$p.value` will give the right answer.

Conclusion

State the statistical conclusion.

We reject the null hypothesis.

State (but do not overstate) a contextually meaningful conclusion.

There is sufficient evidence that there is a relationship between wrist diameter and weight.

Identify the possibility of either a Type I or Type II error and state what making such an error means in the context of the hypotheses.

If we've made a Type I error, then there might not be any relationship between wrist diameter and weight, but our sample showed a relationship.

Confidence interval

Conditions

All the conditions have already been checked.

Calculation

Unlike our previous hypothesis tests, the confidence interval for the slope is not found in the output of the test. Instead, we use a special function `confint_tidy` that takes regression output and calculates a confidence interval for us.

```
CI <- confint_tidy(lm(wgt ~ wri.di,  data = bdims))
CI$conf.low[2]
```

```
## [1] 10.00916
```

```
CI$conf.high[2]
```

```
## [1] 11.60174
```

Conclusion

We are 95% confident that the true slope of the relationship between wrist diameter is captured in the interval (10.0091554, 11.6017377).

Your turn

The `Rubber` data set contains data on the testing of tires. (Since it was a British study, they tested "tyres".)

Explore the relationship between the hardness of the tire (measured in something called Shore units—Google it if you want to know more) and the loss of tire material in an abrasion test (measured in grams per hour).

Please perform the steps below, following the code examples from earlier in the module.

- Run the `lm` command three times wrapped in `tidy`, `augment`, and `glance` respectively. (It's technically incorrect to run regression before checking conditions, but we need the output of `lm` in order to check those conditions.)
- Check conditions for regression.

- If the conditions are met, plot the regression line on top of a scatterplot of the data.
- Write the regression equation mathematically (enclosing your answer in double dollar signs as above), using contextually meaningful variable names.
- Interpret the coefficients: interpret the slope, give a literal interpretation of the intercept, and then comment on the appropriateness of that interpretation.
- Report and interpret R^2.
- Run the full rubric for inference on the slope parameter.