Statistical Analysis in R

1 Purpose

The purpose of this activity is to provide you with an understanding of statistical inference and to both develop and apply that knowledge to the use of the R statistical programming environment.

2 Overview

This lab can be completed during class time and at-home. You should allocate time to complete the relevant portions of the lab in line with the scheduled topics for each week. The two quantitative problem sets in LT relate to, roughly, the first half and second half of the material covered in this lab.

3 Your Task

Using R as instructed, complete the following activities.

3.1 Sampling

1. As we talked about in lecture, simple random sampling is the easiest design-based strategy for ensuring that your sample data are *representative* of the population from which those observations are drawn. The first of today's activities reiterates this idea using R. Start by defining an R vector that is going to contain our "population" values. This vector will store numerical values between 0 and 20; you can imagine that they represent the number of years of formal education obtained my members of our population:

```
set.seed(1)      # this makes sure we all get the same answer
x <- sample(0:20, 1e7, TRUE, c(1,2,3,4,5,6,7,8,9,10,11,12,13,12,11,10,9,8,7,6,5))
```

2. Use the `length()` to verify how many people there are in our population.

3. Use `mean()` to calculate the "true" population mean of this population. How many years of education, on average, does this population have?

4. Now, we will draw a small sample from this population using the `sample()` function. (Note: we used this above to generate some fake population data; now we will use in a different way.). Start by drawing a small sample of just ten observations:

```
s1 <- sample(x, 5, FALSE)
```

5. Use **ggplot2** to draw a histogram of these data:

```
ggplot(, aes(x = s1)) + geom_histogram(bins = 21)
```

6. What is the sample mean of this sample?

7. Calculate the element variance of the data: $Var(Y) = s_Y^2 = \sum_{i=1}^{n} \frac{(Y_i - \bar{Y})^2}{n-1}$. In R this is just `sum((mean(s1) - s1)2)/(length(s1)-1)`. Or, more simply, `var(s1)`. Calculate the element standard deviation (you can use `sd()`).

8. Now, recall the formula for the standard error of the mean: $SE_{\bar{y}} = \sqrt{\frac{s^2}{n}}$, where s^2 = sample (element) variance, and n = sample size.[1] Calculate the standard error of the population mean for your sample data.

9. Recall the definition of the margin of error. For a typical margin of error, we simply double the standard error to create an interval within which we estimate the population mean to be. What is the margin of error for your sample mean? Is the population mean in your sample mean?

10. Now, repeat the above exercises, but draw a larger sample size of size 100 (call this vector `s2`). How large is the margin of error of this larger sample compared to that of the smaller sample from before? Why?

11. Recall that the standard error is meant to capture the idea that if we repeated our sampling process and calculated our statistic of interest (in this case, the mean) on each sample, the standard deviation of those estimates around the true mean would be equal to our standard error. To get a better grasp on this idea, we are going to simulate the process of drawing random samples from our population and then compare the standard deviation of our estimates from each sample to the standard errors we calculate above.

12. To do this, we are going to use the `replicate()` function. This function allows us to repeat a calculate multiple times and return the results in a convenient form. To understand how it works, try generating a single sum of two random numbers: `rnorm(1) + rnorm(1)`. Then, use `replicate()` to do this five times:

    ```
    replicate(5, rnorm (1) + rnorm (1))
    ```

    Note how the result is simply a vector.

13. Now, we want to apply this function to the calculation of the sample mean, as above. To do so, we simply write:

    ```
    # five samples of size n = 5
    replicate(5, mean(sample(x, 5, FALSE)))

    # 1,000 samples of size n = 5
    dist5 <- replicate(1000, mean(sample(x, 5, FALSE)))

    # 100 samples of size n = 100
    dist100 <- replicate(100, mean(sample(x, 100, FALSE)))
    ```

    This vector of sample means is often called the "sampling distribution" of the mean. This term refers to the distribution of a given statistic across repeated samples of the same size from a population.[2] Note that these operations may take some time. When they are done, examine the results:

    - What does the histogram look like: `ggplot(, aes(x = dist5)) + geom_histogram(bins = 21)` ?

[1] Note we are sampling a very small proportion of our population, so we ignore the finite population correction.

[2] So hear we focus on the sampling distribution of the mean, but we could also create a sampling distribution for the maximum of each sample, for the count of observations with 10 years of education, the proportion with more than 12 years, etc. The process works the same for any sample statistic, we just focus here on the mean.

- Are the sample means "unbiased" (meaning the mean of the sample means is close to the population mean): `mean(dist5)` `mean(dist100)`?
- How does the standard deviation of the sample means correspond to the standard errors you calculated above: `sd(dist5)` and `sd(dist100)` ?

14. The margin of error is a form of "interval estimation" in which we express our uncertainty about the value of a parameter by stating the range of values that the parameter is expected to be in based upon our sample estimate. The interval that is equivalent to estimate +/- 2 times the standard error is also called a 95% confidence interval (for reasons we will return to below). You can compare the confidence interval from your data (mean +/- 2 SE) to the distribution of estimated sample means (`quantile(dist100, c(0.025, 0.975))`) to see how closely that interval compares to the interval of estimated values from repeated sampling.

15. To ensure you are comfortable with these ideas, try repeating all of the above but for a different kind of variable. Rather than using an ordinal or interval measure (as above), try using a binary variable. You can create one using `rbinom()`:

```
# the 'prob' argument controls the ratio of 1s and 0s
y <- rbinom(1000000, 1, prob = .5)
```

3.2 Descriptive Statistics

1. For the second major part of this activity, we will look at some real data from the Quality of Government project. This dataset contains country-level data on a very large number of economic, social, health, and political indicators. We will import the data using the `import()` function from the "rio" package. You may need to install this package using: `install.packages()`. Once the data are loaded, we can examine the data themselves by just confirming that they are loaded correctly:

```
library("rio")
d <- import("http://www.qogdata.pol.gu.se/data/qog_std_cs_jan16.dta")
dim(d)
nrow(d)
ncol(d)
names(d)
str(d)
```

2. To obtain some simple descriptive statistics about a few variables, we can use the `summary()` function:

```
summary(d$fh_polity2)  # Polity scores (a democracy measure)
summary(d$gle_cgdpc)   # GDP
summary(d$dpi_finter)  # executive term limits
summary(d$bti_cr)      # civil rights index
summary(d$bl_asy15f)   # female educational attainment
```

3. Use ggplot2 to create a histogram of the distributions of these variables (see code above). You may want to play with the `bins` argument to control the look of the histograms.

4. Use the R functions `mean()`, `median()`, and `table()` to inspect the central tendency and distribution of these variables.

5. To assess the dispersion of each variable, use the functions we used above: `var()` and `sd()`.

6. If you're feeling ambitious, you can create some of your own ("user-defined") functions to calculate the skew and kurtosis statistics described by Kellstedt and Whitten:

```
skew <-  function(x) {
  m3 <- mean((x-mean(x))^3)
  skew <- m3/(sd(x)^3)
  skew
}
skew(d$gle_cgdpc)

kurtosis <- function(x) {
  m4 <- mean((x-mean(x))^4)
  kurt <- m4/(sd(x)^4)-3
  kurt
}
kurtosis(d$gle_cgdpc)
```

7. Skew and kurtosis are, in essence, meant to compare against a distribution known as the "normal distribution." It is "normal" for statistical reasons that have little to do with "normality" in the sense of common English. It is also called the "Gaussian" distribution, and that can sometimes be a less confusing (those less commonly used label). A normal distribution looks like the following graph:

```
curve(dnorm, from = -4, to = 4, col = "red", lwd = 2)
```

   If a variable follows the normal distribution, its histogram will follow a very specific bell shape. We can "eyeball" this, but a more formal way to compare is by drawing what is called a "Q-Q plot". This is a special scatterplot drawn against a theoretical normal distribution based on the "quantiles" of the data (see `quantile()`). If the scatterplot has a straight line, then the data are normally distributed. If it deviates from that, then the data are skewed or "peaked" in a way that deviates from "normality". You can try it on two of the QoG variables:

```
# on two of our observed variables from QoG:
ggplot(d, aes(sample = gle_cgdpc)) + geom_qq()
ggplot(d, aes(sample = bl_asy15f)) + geom_qq()

# on a vector of random numbers drawn to follow the normal curve:
ggplot(, aes(sample = rnorm(1e5))) + geom_qq()
```

8. Now repeat all of the above for the variables mentioned, and possibly explore other variables in the dataset. A codebook is available here: `http://qog.pol.gu.se/data/datadownloads/qogstandarddata`

9. Now, estimate the correlation between two variables. To do this, use `cor()`:

```
cor(d$gle_cgdpc, d$bl_asy15f)
```

   We can also generate a "correlation matrix" showing the correlation between many variables, but this requires specifying the data in a slightly different way:

```
cor(d[, c("gle_cgdpc", "bl_asy15f", "fh_polity2")])
```

10. Based on the correlations, imagine what the scatterplots might look like (keeping in mind what the correlation coefficient measures). If the data are categorical (rather than interval), you may want to use a cross-tabulation rather than correlation coefficient to summarize the results:

```
table(d$dpi_finter, d$bti_cr)
```

Note: You can also use `ftable()` to produce a slightly different looking table. You might also want to consider summarizing this relationship visually using a boxplot:

```
ggplot(d) + aes(x = factor(dpi_finter), y = bti_cr) + geom_boxplot()
```

11. Use ggplot to create a scatterplot of the relationship between two variables. Here's an example showing the relationship between GDP (x-axis) and average female educational attainment:

```
ggplot(d) + aes(x = gle_cgdpc, y = bl_asy15f) + geom_point()
```

12. You will note that R prints a warning message when producing this plot. This relates to missing values in the dataset, where one or both of these variables are unobserved for a particular country. To see which countries we are missing data for, try the following:

```
d$cname[is.na(d$gle_cgdpc)]  # for GDP
d$cname[is.na(d$bl_asy15f)]  # for educational attainment
```

You can also look at the data directly to see where these missing values are. You can use `table(is.na(`*`var`*`))` to count how many missing values there are in the variable. What does the presence of these missing values do for our ability to analyze the data? to estimate the values of population parameters? to represent the population? to draw a causal inference?

13. You may want to adjust the axis scales using, for example:

```
ggplot(d) + aes(x = gle_cgdpc, y = bl_asy15f) + geom_point() + scale_x_log10()
```

14. You can modify the appearance of the plot in many, many ways. A common way to do this is by adding `aes()` features (see `? aes`) or by changing the plot theme (see `? theme`). Experiment with different plots until you feel comfortable with the various options.

15. One useful feature of ggplot2 is the ability to create multiple "panels" or "facets" (visual designer Edward Tufte calls these "small multiples"). To do this, you use the `facet_wrap()` function and specify a "formula" including the variable you would like to split the data by. This example create subpanels for different regions of the world:

```
ggplot(d) + aes(x = gle_cgdpc, y = bl_asy15f) + geom_point() + facet_wrap(~ht_region)
```

16. Pause for a moment to consider how each facet represents a subset of the dataset. In this way, each facet is a summary of the dataset for only a subset of the dataset. If we want to summarize data in this way without plotting, we might consider using the `aggregate()` function. For example to calculate the mean level of GDP by region, you can do:

```
aggregate(gle_cgdpc ~ ht_region, data = d, FUN = mean)
```

Try this aggregate command using different variables and using a different value of the `FUN` argument (which takes the name of a function, such as `mean`, `sd`, etc. without the parentheses) until you feel comfortable with the process of generating data summaries.

3.3 Statistical Significance

1. For the final activity, we will examine the idea of "statistical significance." Statistical significance is a concept related to *statistical* hypothesis testing. If you recall from much earlier in the course (MT Week 6), we discussed two different "flavours" of hypothesis testing — one associated with Fisher and one associated with Neyman and Pearson. We will see how both kinds of hypothesis testing manifest and how current statistical practice is a blend of these two perspectives. That practice most closely approximates Fisher's ideas (the calculation of a p-value) but differs in other ways (the estimation of a "confidence interval").

2. To develop an initial understanding of the idea of statistical hypothesis testing, we will think about the idea of "outliers." Outliers are unusual values in a variable (or set of variables). For example, consider the following variable:

   ```
   v <- c(1,1,1,1,2,2,2,2,2,3,3,3,3,10)
   ```

   It should be clear that the value 10 here is an outlier. If we draw a boxplot of this variable, it becomes even more clear:

   ```
   ggplot(, aes(x = 1, y = v)) + geom_boxplot() + scale_y_continuous(limits = c(0,12))
   ```

3. If we expected all the values of `v` to be relatively similar, we could point out that the value 10 is more than 3 standard deviations away from the mean of the variable: `(10-mean(v))/sd(v)`. This idea that a value is an outlier relative to some assumed central tendency is the basic logic of statistical hypothesis testing.

4. In statistical hypothesis testing, however, we are instead interested to know how unusual a *statistic* is relative to our baseline expectation of what the value of the statistic should be. Thus instead of looking at the values of each observation in a variable and looking for how unusual they are, we want to know how unlikely we are to see a given statistic in our sample of values. But how do we determine whether a statistic is unusual? To figure that out, we have to describe — in Fisher's language — a a "null hypothesis." This is a value that we think the parameter might have in the population. Often, the null hypothesis is set to something that would be theoretically uninteresting: a proportion is 0.5, a difference between two means is 0, a correlation between two variables is 0, the mean of a variable is 0, etc. We can pick any value, but these theoretically uninteresting values are conventional. We are therefore trying to see whether the statistic in our sample data differs from this (hypothetical) null population parameter.

5. But how do we know if our statistic differs from the null hypothesis value? We are worried that a difference between our sample statistic and the null hypothesis value might be erroneous. For example, we have a small sample so our data appear to differ from the null hypothesis but if we collected more observations the sample statistic on this larger sample might be closer to our null hypothesis for the population parameter. Statistical hypothesis testing therefore asserts that we only consider a sample statistic to differ from a null hypothesis value when it is quite far from the null hypothesis value. But how far exactly is "quite far"?

   To decide that, we return to the idea of repeated sampling that we examined earlier. We consider a difference "statistically significant" when it differs more from our null expectation than the variation in sample statistics we would expected to observe across repeated samples were our null hypothesis true (in this case, that there were actually no difference between the groups). For a mean (like the one we looked at earlier), this would be stated as a null expectation that the mean level of education in a country is 12 years. If we collect a sample of data and find that the mean is different from 12, we would consider that mean to be statistically significantly different from our null expectation of 12 years if the sample mean was further from 12 (i.e., much larger or much

smaller) that the variation in sample means we could to estimate from same-sized samples drawn from a population where the true mean was 12.[3]

6. Return to our vector of education values, `x`, from earlier and to our set of sample means of size 100 from that population, `dist100`. What is the standard deviation of the sampling distribution `dist100` (i.e., what is the standard error of the sample mean)? In statistical hypothesis testing, we will say that an estimate is statistically significantly different from the null hypothesis value when the sample statistic is more than a certain number of of standard errors away from the null hypothesis value. But how far is far enough to be considered *significant*? Take a look at how many observations in our `dist100` vector are more than 3 standard errors, more than 2 standard errors, and more than 1 standard error above or below the true mean of `x`:

```
dist100[dist100 > (mean(x) + 3*sd(dist100))]
dist100[dist100 > (mean(x) + 2*sd(dist100))]
dist100[dist100 > (mean(x) + 1*sd(dist100))]
dist100[dist100 < (mean(x) - 3*sd(dist100))]
dist100[dist100 < (mean(x) - 2*sd(dist100))]
dist100[dist100 < (mean(x) - 1*sd(dist100))]
```

   Many sample means are within 1 standard error of the mean, fewer of them are further than 1 standard error from the population mean, even fewer are further than 2 standard errors, and — in this example — none or almost none are further than 3 standard errors from the mean. Thus when we repeatedly sample from this population, it is unusual to see a sample statistic more than 2 standard errors from the population mean and even more unusual to see a statistic more than 3 standard errors from the population mean.

7. Now here is where a short moment of consideration is required before we proceed. In statistical hypothesis testing, we are declaring a null hypothesis value as a *known* population and seeing how likely different sample estimates are when draw repeated random samples from that population. We declare a sample statistic "statistically significant" when the sample statistic is an outlier in that distribution because it would seem to suggest that the sample is drawn from a population with a different population parameter value one with a population parameter equal to the null hypothesis. In other words, the sample statistic is so unusual for a the population from which we — under the null hypothesis — believe that the data are drawn from, that we decide that the data are actually drawn from a population with a different population mean. That's the essence of statistical hypothesis testing.

   But, there's a leap being made here: in practice we don't generally have population data in front of us. We don't actually know the population mean (or any population parameter), so we can't draw repeated random samples from the population to create a sampling distribution. We just have the data in front of us. So, if we set a null expectation of the population parameter value and find that our sample statistic differs from that considerably, we are inclined to "reject the null hypothesis" and believe the population parameter value is different from our null expectation. But, this judgment is based on seeing a sample statistic that is *unusual*, not *impossible*, under the null hypothesis. We might therefore reject the null hypothesis sometimes when the population parameter actually is equal to the null hypothesis value. To avoid doing this too often, we have to define an "error rate" or "significance level" that only allows us to make these kind of "false positive" judgments quite rarely.

8. We'll call this significance level α in order to explore different possible values and what consequences that has for the probability of obtaining a "false positive." A commonly used value of α

[3] If we were talking about a different statistic, the logic would be the same. For a proportion, we might have a null expectation that the proportion is 0.5. An observed proportion in our data would be considered statistically significantly different from 0.5 if the proportion were larger than the variation in estimated proportions we would see across multiple same-sized samples from a population where the parameter was 0.5.

is 0.05. This means that we will only declare a sample statistic to be "statistically significant" when it is as far or farther from the null hypothesis value as 5% of the possible sample statistics generated from random samples from a population with a parameter equal to the null hypothesis value. In essence, we're looking for an outlier statistic that is farther than the 2.5% percentile or 97.5% percentile of the sampling distribution. If we look at our sampling distribution, we can see how far a sample statistic would have to be in order for us declare it statistically significantly different from the population mean:

```
quantile(dist100, c(0.025, 0.975))
```

Check your understanding by assessing why these particular quantiles are used when $\alpha = 0.05$. Those quantiles reflect a "two-tailed" hypothesis test in which we look to see whether a sample statistic is an outlier in either direction. What quantiles would we consider if we were doing a "one-tailed" hypothesis to see if the sample statistic was an outlier only on the upper end of the distribution? only on the lower end of the distribution?

9. Different values of α are possible. Small values mean we want a lower chance of a "false positive" judgment, at the expense of making many for "false negatives" (judgments where we say a sample statistic is consistent with the null hypothesis when in fact it is drawn from a different population). Try some different values. Why are these two rates (false positives and false negatives) trade-offs?

10. But recall, we rarely have access to the population mean and we rarely have the ability to repeatedly sample from a population in order to create a sampling distribution. We therefore rely on "distributional" assumption. Rather than generating a sampling distribution from repeatedly sampling, rely on a mathematical theorem — the central limit theorem — that shows that the sampling distribution of the mean is normally distributed. You can get a sense of this by repeating our repeated sampling exercise above by collecting more samples. The more samples we draw from the population, the more and more the shape of the sampling distribution will resemble the normal distribution's bell curve:

```
ggplot(, aes(x = replicate(100, mean(sample(x, 10, FALSE))))) +
    geom_histogram(bins = 21)
ggplot(, aes(x = replicate(500, mean(sample(x, 10, FALSE))))) +
    geom_histogram(bins = 21)
ggplot(, aes(x = replicate(5000, mean(sample(x, 10, FALSE))))) +
    geom_histogram(bins = 21)
```

This property enables us to calculate how unusual a sample estimate is against a null hypothesis by simply calculating the probability of seeing different statistic values given the normal distribution (which has a well-defined formula). In R we can calculate these probabilities using the `pnorm()` function. If we are thinking about sample means, we can calculate the probability of different sample means against any particular null hypothesis parameter value. To do so, however, requires that we express the mean as a difference from the null hypothesis value and rescale to the scale of standard errors. In our example, we have to convert the sample mean to number of years different from the population mean and then rescale to units in number of standard errors: i.e., from `mean(s2)` to

```
tstat <- (mean(s2) - mean(x)) / ( sd(s2)/sqrt(length(s2)) )}
```

This value is called a *t*-statistic. If we had a different null hypothesis value (e.g., that the population mean was 3), we would use that instead of `mean(x)` in the above calculation.

11. To see how unusual this t-statistic is, we plug it into the `pnorm()` function: `pnrom(tstat)`. The output is called a p-value and can be understood as the probability of seeing a test statistic this far or farther from the null hypothesis value. Formally, it is "the probability of a t-statistic as extreme as the one we observed, if the null hypothesis was true." When this p-value is smaller than α level, we judge the sample mean to be "statistically significantly different from zero." When the p-value is larger than α, we declare that the test statistic is not statistically significantly different from the null hypothesis value. However, it is important to remember when reading this that:

    - The probability that a hypothesis is true or false
    - A reflection of our confidence or certainty about the result
    - The probability that the true mean is in any particular range of values
    - A statement about the importance or substantive size of the effect

    Those are all common misconceptions of how to interpret a p-value.

12. To see how large a t-statistic has to be to cross different α levels, you can explore the `qnorm()` function, which takes a p-value as input and returns the corresponding t-statistic.[4]

```
qnorm(0.025) # 5% significance level
qnorm(0.05)  # 10% significance level
qnorm(0.33)  # 33% significance level
qnorm(0.25)  # 50% significance level
```

13. We now come to the idea of a "confidence interval," which is an interval estimate like the interval created by calculating a margin of error as in the first part of this lab activity. A confidence interval (or "CI") is simply a range, centered on our sample estimate that tells us about the likely location of the population parameter within a stated range of uncertainty.[5] To formalize this, a confidence interval tells us:

    > Were we to repeat our procedure of sampling, analyzing the sample to produce a sample estimate and standard error, and transforming those estimates into a confidence interval *repeatedly* from the population, a fixed percentage of the resulting intervals would include the true population-level parameter.

    This does not say for sure whether the estimated confidence interval *this time* actually includes that true population parameter.

[4] Note: We are being a bit lose here. Technically, `qnorm()` looks at the normal distribution, but in very large samples, the normal distribution and the t distribution are identical.

[5] Because it is just a transformation of the margin of error, it is based on the variability of the data, sampling procedures, and — most importantly — sample size.

14. To get at the notion of the confidence interval, we are going back to our population data `x` and repeatedly drawing new samples. This time, however, we are going to store our results in a data.frame and we are going to save not only how far the sample mean deviates from the population mean, but we are also going to save the standard error calculated from each sample:

```
alpha <- qnorm(0.025) # 5% significance level; 95% confidence interval

n <- 100  # number of samples to draw and estimate statistic on
ci <- data.frame(i = 1:n,
                 means = numeric(n),
                 se = numeric(n),
                 off = logical(n))
for (i in 1:n){                              # Take 100 samples from our distribution
  tmp <- sample(x, 100, replace=FALSE)       # Store samples in 'temp'
  ci$means[i] <- mean(tmp)-mean(x)           # calculate and store mean
  ci$se[i] <- (sd(tmp)/sqrt(length(tmp)))    # calculate and store upper CI limit
}
```

You can explore this new object, `ci`, perhaps using `summary(ci)`.

15. What we have generated here is a data.frame of "centered" sample means: we are expressing how far our sample mean is from the population mean. This allows us to calculate a so-called "confidence" interval. A confidence interval is an interval — recall how we have already encountered it in the form of a margin of error — that attempts to provide information about the location of the true population value. The confidence interval is sized based upon the variability of our data, the size of the sample we draw, and α. The choice of α, as in the rest of statistical hypothesis testing, is important because it dictates our error rate.

    When we set the width of the confidence level as $1 - \alpha$, we are saying we will allow α proportion of our confidence (where we to sample an infinite number of times) to not include the true population parameter. If $\alpha = 0.05$, then we are drawing 95% confidence intervals. Thus only 5% of the intervals we draw from this sample are likely to "miss" the true population parameter. If we therefore find in our particular sample that the interval differs from our null expectation (e.g., the sample mean does not equal zero and the 95% confidence interval based upon our sample data does not cover 0), then we would say the sample mean difference is statistically significantly different from zero. (The p-value in this case would be less than 0.05.) So this either indicates that the population parameter is truly not equal to zero or that our particular sample happens to have produced one of the 5% of confidence intervals that are expected (given our sampling procedure, sample size, and α level) to not cover the true population parameter, simply due to chance. We cannot know with certainty which interval we have.

    How many of our confidence intervals do not cover the population mean? (Recall we have rescaled the mean to be a different from the true value.):

```
ci$off <- ((ci$means-(abs(alpha)*ci$se)) > 0 & (ci$means+(abs(alpha)*ci$se)) > 0) |
          ((ci$means-(abs(alpha)*ci$se)) < 0 & (ci$means+(abs(alpha)*ci$se)) < 0)
table(ci$off)
```

16. We can graph all of our confidence intervals to see which include the true mean and which do not (colored by the `off` variable we just generated):

```
ggplot(ci, aes(x = i, y = means, colour = off)) +
    geom_errorbar(aes(ymin = (means - alpha*se),
                      ymax = (means + alpha*se)), width=.1) +
    geom_point() + coord_flip()
```

This exercise shows that if a particular parameter value is true (in this case, the population mean is zero), we can draw confidence intervals for that mean to try to estimate where the mean is located. Most of these intervals will "cover" the true population parameter value, but not all of them. The number that cover the true population parameter value depends on the width of the confidence interval we draw. If we draw a wider confidence interval (say 95%), then 95% of the confidence intervals drawn from samples of this size from the population will cover the true population parameter value. If we draw a narrower confidence interval (say 50%), then only 50% of the confidence intervals drawn from samples of this size from the population will cover the true population value.

17. Try to repeat the above sampling and graphing procedure but tweak the values of α and sample size. As α increases, fewer of the confidence intervals drawn from the population with cover the true population parameter. That means that we are more likely to make "false positive" judgments and to have incorrect beliefs about the location of the population parameter.

18. We can also see, in the above data, the equivalence of a confidence interval, a t-statistic, and a p-value. When our confidence interval does not overlap the null hypothesis value, we describe it as statistically significant. The p-value is generated from a "test statistic" which translates our statistic (in the above example, the sample mean). For a sample mean, the t statistic is simply the sample mean divided by the standard error. In our data that would be:

```
ci$mean / ci$se
```

When this test statistic exceeds the critical value, α, then the statistic is deemed statistically significantly different from the null hypothesis value. The critical value is simply based on a hypothetical distribution, in this case the t-distribution (which, in large samples, is identical to the normal or Gaussian distribution noted earlier). When we earlier selected a value of α we were in essence setting the critical value of our t-statistic. Recall for our 95% confidence interval, we selected an α value of `qnorm(0.025)` (approximately 1.96). Thus, when our sample t-statistic exceeds this value, then we have a statistically significant result because our test statistic is very far from the null hypothesis value of 0. Our null hypothesis implies a distribution of test statistics that follows the t distribution and, given that expected distribution of possible test statistics, a sample test statistic larger than 1.96% would be quite rare (less than 5% of test statistics observed for samples from a population of mean 0 would have test statistics that large or larger).

19. This t-statistic can then be translated into the p-value: `1-pnorm(1.96)`[6] We can calculate the p-values for the t-statistic from each of our samples as:

```
pnorm(ci$mean / ci$se)
```

You should now clearly be able to see a correspondence between t-statistics, the confidence intervals, and the p-values for each of the samples. The t-statistics are large when the confidence interval does not overlap zero:

```
cbind.data.frame(ci$means/ci$se, ci$off)
```

and the p-value for each sample is small in those same cases.

20. Returning to the QoG data from earlier, apply what we've just learned to assess whether the mean years of female educational attainment differ from a null hypothesis value. You can do this quickly in R using the `t.test()` function, where `mu` is the value of your null hypothesis:

[6]Here we see a "two-tailed" hypothesis test. We are allowing the test statistic to differ from the null expectation in either direction (see `1-pnorm(1.96)` and `pnorm(-1.96)`).

```
t.test(d$bl_asy15f, mu = 12)
```

Try out different possible values of `mu`.

21. A further test we might be interested in is whether countries' average educational attainment for men differs from the educational for women. This is what is known as a two-sample t-test. Like exercise we just performed to generate confidence intervals, this is based on a null hypothesis about the mean-difference, typically that the difference between two group means is zero. Try it out:

```
t.test(x = d$bl_asy15f, y = d$bl_asy15m)
```

22. Another further test we might be interested in is whether two subsets of cases differ from one another. To capture this, we can also use `t.test()`, this time with a formula notation. For example we might test whether the countries with high and low levels of democracy differ in female educational attainment:

```
dem_high <- factor(d$fh_polity2 > mean(d$fh_polity2, na.rm = TRUE))
t.test(d$bl_asy15f ~ dem_high)
```

23. Try to interpret all of the above results.

24. A major caveat in any discussion of *statistical* significance is that we can never forget *substantive* significance. Statistical significance tells us whether an estimate, a relationship, or an effect is large relative to a hypothetical distribution of test statistics corresponding to null expectation. This says nothing about whether that effect is large or important in substantive terms.[7] If we find that democracy and non-democracies differ by $50 in per capita GDP that this difference is statistically different from zero, that is a statistically significant difference. Whether that difference is large or important depends upon the state of broader scientific understanding, the amount of dispersion in the data (is the difference large if measured in number of standard deviations), the size of other differences (do regions of the world vary more than one another on this variable), the research context, and our own judgment. $50 may be a substantively small effect when talking about GDP but it may be a large effect when talking about the cost of tonight's dinner. This is something for you to consider.

[7] If we have enough data (i.e., our sample is large enough), almost any test statistic will "statistically significant" but that does not mean that the estimated parameter is large or important.