Computational Social Science: Randomized Experiments

Your Name Here

10/20/2020

```
set.seed(13)
library(dplyr)
library(ggplot2)
library(purrr)
```

In this lab, we are going to discuss Randomized Experiments. Causal inference methods can be used for observational data, but it is easier to first consider them in the context of randomized experiments.

Simulation

Say that we have a new pain medication, AspiTyleCedrin, and we would like to know how effective it is at treating migraines. Ideally, we would like to observe the entire population's experience of migraines without using the medication, then turn back time and observe the entire population's experience with migraines with the medication, and compare the results.

Let's simulate this scenario. The following code chunk creates a dataframe containing:

- `A`: Treatment variable indicating whether the individual i took AspiTyleCedrin ($A_i = 1$) or not ($A_i = 0$).
- `Y_0`: Potential outcome variable Y_0 = Migraine|No AspiTyleCedrin indicating whether the individual i would experience a migraine ($Y_{i0} = 1$) or not ($Y_{i0} = 0$) if they DO NOT take AspiTyleCedrin.
- `Y_1`: Potential outcome variable Y_1 = Migraine|AspiTyleCedrin indicating whether the individual i would experience a migraine ($Y_{i1} = 1$) or not ($Y_{i1} = 0$) if they DO take AspiTyleCedrin.
- `W1`: Variable representing sex assigned at birth, with $W1 = 0$ indicating AMAB (assigned male at birth), $W1 = 1$ indicating AFAB (assigned female at birth), and $W1 = 2$ indicating an X on the birth certificate, possibly representing an intersex individual or left blank.
- `W2`: Variable representing simplified racial category, with $W2 = 0$ indicating White, $W2 = 1$ indicating Black or African American, $W2 = 2$ indicating Non-White Hispanic or Latinx, $W2 = 3$ indicating American Indian or Alaska Native, $W2 = 4$ indicating Asian, and $W2 = 5$ indicating Native Hawaiian or Other Pacific Islander.

```
n = 1e6 # Number of individuals


# NOTE: Don't worry too much about how we're creating this dataset, this is just for an example.
df <- data.frame(W1 = sample(0:2, size = n, replace = TRUE,
                             prob = c(0.49,0.50,0.01)),
                 W2 = sample(0:5, size = n, replace = TRUE,
                             prob = c(0.60,0.13,0.19,0.06, 0.015, 0.005)))
df <- df %>%
  mutate(A = as.numeric(rbernoulli(n,
```

```
                                    p = (0.50 + 0.07*(W1 > 0) + 0.21*(W2 == 0))),
         Y_0 = as.numeric(rbernoulli(n,
                                     p = (0.87 + 0.035*(W1 > 0) + 0.05*(W2 > 0)))),
         Y_1 = as.numeric(rbernoulli(n,
                                     p = (0.34 + 0.035*(W1 > 0) + 0.3*(W2 > 0)))))

head(df)
```

```
##   W1 W2 A Y_0 Y_1
## 1  0  2 1   0   1
## 2  1  0 1   1   0
## 3  1  1 1   1   0
## 4  1  0 1   1   0
## 5  0  2 1   1   1
## 6  1  0 1   1   0
```

```
summary(df)
```

```
##        W1               W2               A               Y_0
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:1.0000
##  Median :1.0000   Median :0.0000   Median :1.0000   Median :1.0000
##  Mean   :0.5192   Mean   :0.7736   Mean   :0.6616   Mean   :0.9077
##  3rd Qu.:1.0000   3rd Qu.:2.0000   3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :2.0000   Max.   :5.0000   Max.   :1.0000   Max.   :1.0000
##       Y_1
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.0000
##  Mean   :0.4772
##  3rd Qu.:1.0000
##  Max.   :1.0000
```

Causal Types

Let's take a look at a frequency table of the two output columns `Y_0` and `Y_1`:

```
# Add a column indicating the causal type of each individual
df <- df %>%
  mutate(type = as.factor(4 - (Y_1*2 + Y_0)))

# Create a frequency table showing how many individuals there are of each causal type
df.freq <- df %>%
  count(Y_1, Y_0, type) %>%
  group_by(Y_1) %>%
  mutate(prop = n/nrow(df)) %>%
  arrange(type)
df.freq
```

```
## # A tibble: 4 x 5
## # Groups:   Y_1 [2]
```

```
##     Y_1   Y_0 type       n   prop
##   <dbl> <dbl> <fct>  <int>  <dbl>
## 1     1     1 1     437134 0.437
## 2     1     0 2      40097 0.0401
## 3     0     1 3     470568 0.471
## 4     0     0 4      52201 0.0522
```

```
# Save the proportions of each causal type
p_1 <- df.freq$prop[1]
p_2 <- df.freq$prop[2]
p_3 <- df.freq$prop[3]
p_4 <- df.freq$prop[4]
```

This shows us how many individuals in our population of interest had each of four possible sets of outcomes with and without the use of AspiTyleCedrin, which we may refer to as four different causal "types":

- **Type 1 or "doomed":** These individuals experience a migraine regardless of whether they take AspiTyleCedrin. In our population of interest there are c(437134, 40097, 470568, 52201) such individuals. The proportion of these individuals out of the entire population of interest is $p_1 \approx 0.437$.
- **Type 2 or "causal":** These individuals experience a migraine if and only if they take AspiTyleCedrin. In our population of interest there are 1:4 such individuals. The proportion of these individuals out of the entire population of interest is $p_2 \approx 0.04$.
- **Type 3 or "preventive":** These individuals experience a migraine if and only if they do not take AspiTyleCedrin. In our population of interest there are c(1, 0, 1, 0) such individuals. The proportion of these individuals out of the entire population of interest is $p_3 \approx 0.471$.
- **Type 4 or "immune":** These individuals do not experience a migraine regardless of whether they take AspiTyleCedrin. In our population of interest there are c(1, 1, 0, 0) such individuals. The proportion of these individuals out of the entire population of interest is $p_4 \approx 0.052$.

Causal Parameters

Individual-level Treatment Effect (ITE)

The Individual-level Treatment Effect is simply the difference between the potential outcomes for a particular individual i, that is:

$$\text{ITE}_i = Y_{i1} - Y_{i0}$$

Question 1: Use the `mutate()` function to add a column to `df` named `ITE` which contains the individual-level treatment effect for each row.

```
df <- df %>%
  mutate(ITE = Y_1 - Y_0)
```

Average Treatment Effect (ATE)

A common causal parameter of interest is the Average Treatment Effect, which is the average difference in the pair of potential outcomes averaged over the entire population of interest (at a particular moment in time), or rather, it is just the average (or expected value) of the individual-level treatment effect.

$$\text{ATE} = E[Y_{i1} - Y_{i0}]$$

Question 2: Use the `ITE` column you just added to `df` to find the average treatment effect of AspiTyleCedrin on migraines in this population and assign it to the variable name `ATE`.

```
ATE <- mean(df$ITE)
ATE
```

```
## [1] -0.430471
```

You should have gotten a result around ATE ≈ -0.43. This means that for our entire population of interest, AspiTyleCedrin decreases migraines by about 43% on average.

Notice that, since the expected value is a linear operator:

$$\text{ATE} = E[Y_{i1} - Y_{i0}] = E[Y_{i1}] - E[Y_{i0}]$$

Question 3: Confirm this using `R`.

```
ATE == mean(df$Y_1 - df$Y_0)
```

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

Let us consider this in terms of the proportions of causal types above, that is, p_1, p_2, p_3, p_4. The expected value of Y_{i1} is simply the proportion of the entire population of interest that experiences migraines if everyone takes AspiTyleCedrin, so it is really the sum of both types that experience migraines when everyone takes AspiTyleCedrin (i.e. the sum of the "doomed" and the "causal" groups):

$$E[Y_{i1}] = p_1 + p_2$$

Similarly, the expected value of Y_{i0} is simply the proportion of the entire population of interest that experiences migraines if nobody takes AspiTyleCedrin, so it is really the sum of both types that experience migraines when nobody takes AspiTyleCedrin (i.e. the sum of the "doomed" and the "preventive" groups):

$$E[Y_{i0}] = p_1 + p_3$$

Therefore, the average treatment effect can be re-written as follows:

$$\begin{aligned}
\text{ATE} &= E[Y_{i1} - Y_{i0}] \\
&= E[Y_{i1}] - E[Y_{i0}] \\
&= (p_1 + p_2) - (p_1 + p_3) \\
&= p_2 - p_3
\end{aligned}$$

Or rather, the average treatment effect is equivalent to the difference between the proportions of the "causal" and "preventive" groups.

Question 4: Again, confirm this using `R`.

```
ATE == p_2 - p_3
```

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

Average Treatment Effect on the Treated (ATT)

Another common causal parameter that we may be interested in is the Average Treatment Effect on the Treated which, instead of considering the entire population of interest, only considers those who would actually be treated.

Question 5: Use the `filter()` function to create a new data frame `df.treat` which is a subset of `df` containing only those who receive the treatment. Then find the average treatment effect on the treated of AspiTyleCedrin on migraines in this population and assign it to the variable name `ATT`.

```
# Create a subset of the dataset containing only those individuals who received the treatment
df.treat <- df %>%
  filter(A == 1)

ATT <- mean(df.treat$ITE)
ATT
```

```
## [1] -0.4490738
```

You should have gotten a result around ATT ≈ -0.45. This means that among those actually treated, AspiTyleCedrin decreases migraines by about 45% on average.

Heterogeneous Treatment Effects

In many cases, treatment effects may look very different among different groups, so it may be worth considering whether there are heterogenous treatment effects within your population of interest. For example, say we have reason to suspect AspiTyleCedrin is more or less effective among different racial groups.

Question 6: Use the `group_by()` and `summarize()` functions to calculate separate average treatment effects for each race category. Discuss your results.

```
df.group <- df %>%
  group_by(W2) %>%
  summarize(ATE = mean(ITE))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
df.group
```

```
## # A tibble: 6 x 2
##      W2    ATE
##   <int>  <dbl>
## 1     0 -0.531
## 2     1 -0.278
## 3     2 -0.280
## 4     3 -0.283
## 5     4 -0.277
## 6     5 -0.274
```

We see a clear difference in the average treatment effect for White individuals than other racial groups. The average treatment effect of AspiTyleCedrin among White people is an approximately 53% decrease in migraine incidence whereas for people of color it is only an approximately 28% decrease in migraine incidence. AspiTyleCedrin appears to be less effective for people of color than it is for White people.

Experimental Designs

All of the above calculations were made using complete information about the experience of migraines for the entire population of interest both with and without the use of AspiTyleCedrin. In reality we of course would not have this information. Instead of knowing all the values in `Y_0` and `Y_1`, we would instead only know a single outcome `Y_i` for each individual, as well as the treatment and covariate columns.

For example, using our `A` column, representing the observed treatment variable indicating whether each individual took AspiTyleCedrin, we can create an observed outcome column `Y_obs`:

```
df <- df %>%
  mutate(Y_obs = as.numeric((A & Y_1) | (!A & Y_0)))
```

Now we consider a frequency table of `A` versus `Y_obs`:

```
df %>%
  select(A, Y_obs) %>%
  ftable()
```

```
##   Y_obs      0      1
## A
## 0        29087 309269
## 1       360338 301306
```

Or, we can lay it out slightly differently, as below. Confirm for yourself that these two tables are equivalent.

```
df %>%
  count(A, Y_obs) %>%
  group_by(A)
```

```
## # A tibble: 4 x 3
## # Groups:   A [2]
##       A Y_obs      n
##   <dbl> <dbl>  <int>
## 1     0     0  29087
## 2     0     1 309269
## 3     1     0 360338
## 4     1     1 301306
```

The above table is the only information (aside from covariates, etc.) that we would be able to obtain in reality, even if we were able to sample the entire population of interest, as we have here.

How does this table relate to the different causal types? Let's consider this same table, but separate out each causal type.

```
df.freq.obs.type <- df %>%
  count(A, Y_obs, type) %>%
  group_by(type)
df.freq.obs.type
```

```
## # A tibble: 8 x 4
## # Groups:   type [4]
```

```
##       A Y_obs type       n
##   <dbl> <dbl> <fct>  <int>
## 1     0     0 2      13638
## 2     0     0 4      15449
## 3     0     1 1     162287
## 4     0     1 3     146982
## 5     1     0 3     323586
## 6     1     0 4      36752
## 7     1     1 1     274847
## 8     1     1 2      26459
```

Thus we can see that each of the cells in the original table is actually composed of a mixture of two different causal types, each, and conversely, that each causal type makes up part of the cell count for two different arrangements of `A` and `Y_obs`. In other words:

- Those that DID NOT take AspiTyleCedrin and DID NOT experience a migraine could be either *"causal"* or *"immune"*.
- Those that DID NOT take AspiTyleCedrin and DID experience a migraine could be either *"doomed"* or *"preventive"*.
- Those that DID take AspiTyleCedrin and DID NOT experience a migraine could be either *"preventive"* or *"immune"*.
- Those that DID take AspiTyleCedrin and DID experience a migraine could be either *"doomed"* or *"causal"*.

And, importantly, in reality, for each of these categories **we have no way of knowing which is which** of the corresponding two causal types. This makes it difficult to estimate the average treatment effect in practice. For that reason, there are a number of different estimators used to estimate this parameter, but for now we will consider a very naive one, that is:

$$\hat{\text{ATE}} = E[Y_i|A_i = 1] - E[Y_i|A_i = 0]$$

Question 7: Use the `group_by()` and `summarize()` functions to find the estimated average treatment effect using the `A` and `Y_obs` columns. Compare this result to the actual ATE calculated earlier.

```
# Find the mean of Y_obs among different values of A
est.obs <- df %>%
  group_by(A) %>%
  summarize(mean = mean(Y_obs))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ATE.obs <- est.obs$mean[2] - est.obs$mean[1]
ATE.obs
```

```
## [1] -0.4586444
```

Our estimate $\hat{\text{ATE}} \approx$ -0.4586444 is close but definitely not the same as the true parameter $ATE \approx$ -0.430471.

Independence Assumption and Random Assignment

If we consider this estimator in terms of the causal groups, we note that $E[Y|A=1]$ is composed of parts of causal types 1 and 2, and $E[Y|A=0]$ is composed of parts of causal types 1 and 3. Or rather:

$$\begin{aligned}\hat{\text{ATE}} &= E[Y_i|A_i=1] - E[Y_i|A_i=0] \\ &= \frac{(?n_1 + ?n_2)}{(?n_1 + ?n_2 + ?n_3 + ?n_4)} - \frac{(?n_1 + ?n_3)}{(?n_1 + ?n_2 + ?n_3 + ?n_4)}\end{aligned}$$

Where each question mark is some unknown fraction, and n_{type} is the number of individuals of that causal type. If we could somehow be assured that all of the unknown fractions were equivalent, then the following would also be true:

$$\begin{aligned}\hat{\text{ATE}} &= \frac{(?n_1 + ?n_2)}{(?n_1 + ?n_2 + ?n_3 + ?n_4)} - \frac{(?n_1 + ?n_3)}{(?n_1 + ?n_2 + ?n_3 + ?n_4)} \\ &= \frac{?(n_1 + n_2)}{?(n_1 + n_2 + n_3 + n_4)} - \frac{?(n_1 + n_3)}{?(n_1 + n_2 + n_3 + n_4)} \\ &= \frac{(n_1 + n_2)}{(n_1 + n_2 + n_3 + n_4)} - \frac{(n_1 + n_3)}{(n_1 + n_2 + n_3 + n_4)} \\ &= \left(\frac{n_1}{(n_1 + n_2 + n_3 + n_4)} + \frac{n_2}{(n_1 + n_2 + n_3 + n_4)}\right) - \left(\frac{n_1}{(n_1 + n_2 + n_3 + n_4)} + \frac{n_3}{(n_1 + n_2 + n_3 + n_4)}\right) \\ &= (p_1 + p_2) - (p_1 + p_3) \\ &= p_2 - p_3\end{aligned}$$

And thus we could rest assured that our estimator $\hat{\text{ATE}}$ is actually equivalent to the parameter ATE.

Unfortunately, we've seen that these assumptions do not hold for our observed data. However, in an experimental design, we can control the assignment of the intervention variable `A` (AspiTyleCedrin, in our case). If we are somehow able to ensure that the assignment of the intervention `A` is **independent** of the causal type, then these assumptions will hold.

Question 8: Explain why this is true. That is, show mathematically that if `A` and `type` are independent, then all of the question marks above are equivalent.

If `A` and `type` are independent, then by definition, $P(\text{type}|A=0) = P(\text{type}|A=1)$. As we saw in `df.freq.obs.type`, each causal type is split into two groups, one for $A=0$ and one for $A=1$. Therefore, if $P(\text{type}|A=0) = P(\text{type}|A=1)$, then both of these probabilities must be 0.5. This applies to all four causal types, so:

$$\begin{aligned}\frac{(?n_1 + ?n_2)}{(?n_1 + ?n_2 + ?n_3 + ?n_4)} - \frac{(?n_1 + ?n_3)}{(?n_1 + ?n_2 + ?n_3 + ?n_4)} &= \frac{(0.5n_1 + 0.5n_2)}{(0.5n_1 + 0.5n_2 + 0.5n_3 + 0.5n_4)} - \frac{(0.5n_1 + 0.5n_3)}{(0.5n_1 + 0.5n_2 + 0.5n_3 + 0.5n_4)} \\ &= p_2 - p_3\end{aligned}$$

Question 9: Briefly explain why we can ensure this independence assumption by randomly assigning individuals to take AspiTyleCedrin.

If our intervention variable `A` (taking AspiTyleCedrin) is randomly assigned, then members of each causal group have an equal chance of being assigned to take AspiTyleCedrin. Therefore, $P(A_i|\text{Type } 1_i) = P(A_i|\text{Type } 2_i) = P(A_i|\text{Type } 3_i) = P(A_i|\text{Type } 4_i)$, so by definition of independence, $A \perp \text{Type}$.

Completely Randomized Designs

There are different methods of randomization. The most conceptually simple is complete randomization. That is, randomly assigning the intervention for the entire sample.

We will simulate this by creating a new assignment variable `A_comp`, and a new outcome variable `Y_comp`:

```
df <- df %>%
  mutate(A_comp = as.numeric(rbernoulli(n, p =0.5)),
         Y_comp = as.numeric((A_comp & Y_1) | (!A_comp & Y_0)))
```

Question 10: Create a new frequency table for `A_comp` and `Y_comp` (of whichever layout you prefer) and then assign the new estimated average treatment effect to the variable `ATE.comp`. Then briefly compare your result to the true parameter `ATE`.

```
df %>%
  select(A_comp, Y_comp) %>%
  ftable()
```

```
##         Y_comp      0      1
## A_comp
## 0               46035 453421
## 1              261355 239189
```

```
df %>%
  count(A_comp, Y_comp) %>%
  group_by(A_comp)
```

```
## # A tibble: 4 x 3
## # Groups:   A_comp [2]
##   A_comp Y_comp      n
##    <dbl>  <dbl>  <int>
## 1      0      0  46035
## 2      0      1 453421
## 3      1      0 261355
## 4      1      1 239189
```

```
est.comp <- df %>%
  group_by(A_comp) %>%
  summarize(mean = mean(Y_comp))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ATE.comp <- est.comp$mean[2] - est.comp$mean[1]
ATE.comp
```

```
## [1] -0.4299716
```

This is much closer to the true `ATE`.

Cluster Randomized Designs

Another method of randomization is cluster randomization. In this method, individuals are broken up into cluster, in which all members of the same cluster are assigned the same treatment (i.e. all $A = 0$ or all $A = 1$) but *clusters* are randomized, such that each cluster has an equal chance of being assigned to the intervention.

To simulate this, we will create a new `cluster` column which contains a number 1 through 100 to indicate which of 100 clusters each individual belongs to.

```
df <- df %>%
  mutate(cluster = rep(1:100, each = n/100))
```

Question 11: Create a new assignment variable `A_clus`, and a new outcome variable `Y_clus`, making sure that all members of a given cluster have the same value for `A_clus`. Then, create a new frequency table for `A_clus` and `Y_clus`, assign the new estimated average treatment effect to the variable `ATE.clus`, and briefly compare your result to the true parameter `ATE`.

```
df <- df %>%
  mutate(A_clus = rep(as.numeric(rbernoulli(100, p =0.5)), each = n/100),
         Y_clus = as.numeric((A_clus & Y_1) | (!A_clus & Y_0)))
```

```
df %>%
  select(A_clus, Y_clus) %>%
  ftable()
```

```
##         Y_clus      0      1
## A_clus
## 0               44429 435571
## 1              271909 248091
```

```
df %>%
  count(A_clus, Y_clus) %>%
  group_by(A_clus)
```

```
## # A tibble: 4 x 3
## # Groups:   A_clus [2]
##   A_clus Y_clus      n
##    <dbl>  <dbl>  <int>
## 1      0      0  44429
## 2      0      1 435571
## 3      1      0 271909
## 4      1      1 248091
```

```
est.clus <- df %>%
  group_by(A_clus) %>%
  summarize(mean = mean(Y_clus))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ATE.clus <- est.clus$mean[2] - est.clus$mean[1]
ATE.clus
```

```
## [1] -0.4303415
```

This too is much closer to the true `ATE` than the observed estimate.

Note that this example is simply clustering individuals by location in our dataframe, and thus individuals in a given cluster are not statistically more similar to each other than they are to individuals in other clusters. In reality, clusters usually are statistically more similar to each other than to individuals in other clusters (for example, perhaps each cluster represents patients of a particular hospital). This is in fact a common reason *why* a cluster randomized design is used.

Block Randomized Designs

Block randomized designs can be thought of as the reverse of cluster designs. That is, individuals are broken up into blocks, and then randomization occurs *within each block*.

To simulate this, we will create a new `block` column which contains a number 1 through 20 to indicate which of 20 blocks each individual belongs to.

```
df <- df %>%
  mutate(block = rep(1:20, each = n/20))
```

Question 12: Create a new assignment variable `A_bloc`, and a new outcome variable `Y_bloc`. Then, create a new frequency table for `A_bloc` and `Y_bloc`, assign the new estimated average treatment effect to the variable `ATE.bloc`, and briefly compare your result to the true parameter `ATE`. Hint: Before creating `A_bloc`, take a look at the help page for the `replicate()` function.

```
df <- df %>%
  mutate(A_bloc = as.numeric(replicate(20, rbernoulli(n/20, p =0.5))),
         Y_bloc = as.numeric((A_bloc & Y_1) | (!A_bloc & Y_0)))
```

```
df %>%
  select(A_bloc, Y_bloc) %>%
  ftable()
```

```
##         Y_bloc      0      1
## A_bloc
## 0                45910 453662
## 1               261651 238777
```

```
df %>%
  count(A_bloc, Y_bloc) %>%
  group_by(A_bloc)
```

```
## # A tibble: 4 x 3
## # Groups:   A_bloc [2]
##   A_bloc Y_bloc      n
##    <dbl>  <dbl>  <int>
## 1      0      0  45910
## 2      0      1 453662
## 3      1      0 261651
## 4      1      1 238777
```

```r
est.bloc <- df %>%
  group_by(A_bloc) %>%
  summarize(mean = mean(Y_bloc))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
ATE.bloc <- est.bloc$mean[2] - est.bloc$mean[1]
ATE.bloc
```

```
## [1] -0.4309558
```

Yet again this is much closer to the true `ATE` than the observed estimate.

Note again that this example is simply blocking individuals by location in our dataframe, and thus individuals in a given block are not statistically more similar to each other than they are to individuals in other blocks. In reality, blocks usually are statistically more similar to each other than to individuals in other blocks (for example, perhaps each block represents a geographical region).

Statistical Tests of Difference

A common statistical question (in fact, often the actual research question of interest) is whether some variable in our dataset (usually the dependent or outcome variable) varies by one or more other (usually independent) variables.

There are many different statistical tests to try to answer such questions, and choosing among these tests depends on the context. In particular, choosing a test depends upon the types of variables you are comparing, and the assumptions you are able to make about the underlying distributions of the continuous variables. The following table is a quick guide for a few different scenarios. The first column lists tests that can be used when assuming Normality of any continuous variables, the second column lists non-paramtric tests that do not make distributional assumptions.

Table 1: Tests for Different Types of Variables

Variable Types	**Normal**	**Non-Parametric**
2 Categorical	χ^2 Test	χ^2 Test
1 Categorical (2 levels), 1 Continuous	t-Test (means)	Rank-Sum Test (medians)
1 Categorical (>2 levels), 1 Continuous OR >1 Categorical, 1 Continuous	ANOVA a.k.a. F-Test (means)	Kruskal-Wallis Test (medians)
2 Continuous	Pearson Correlation	Spearman Correlation

For our dataset, if we want to explore whether the experience of migraines varies among those who took AspiTyleCedrin or not, or among simplified racial groups or sex assigned at birth, we could use either χ^2 or ANOVA, since these are all categorical variables, some with more than two levels. Let's take a look at both methods using the completely randomized dataset.

χ^2 Test

Let's first do a simple χ^2 test just to confirm that there is evidence of a difference in migraine experience among those who took AspiTyleCedrin or not.

```
chisq.test(table(df$Y_comp, df$A_comp))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(df$Y_comp, df$A_comp)
## X-squared = 217088, df = 1, p-value < 2.2e-16
```

Since the p-value from this test is so small, we see that there is evidence that migraine incidence is indeed different between these two groups.

Now let's see if migraine experience is different among different sexes assigned at birth.

```
chisq.test(table(df$Y_comp, df$W1))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(df$Y_comp, df$W1)
## X-squared = 1556.1, df = 2, p-value < 2.2e-16
```

Yet again we see a very small p-value, meaning that we do indeed have evidence of a difference in migraine experience among these groups.

Question 13: Run a χ^2 test to see if there is a difference in migraine experience among simplified racial groups. Briefly discuss your results.

```
chisq.test(table(df$Y_comp, df$W2))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(df$Y_comp, df$W2)
## X-squared = 34720, df = 5, p-value < 2.2e-16
```

Yet again we see a very small p-value, meaning that we do indeed have evidence of a difference in migraine experience among these groups.

Analysis of Variance (ANOVA) a.k.a. F-Test

Our outcome of migraine experience is categorical, but we can still use methods like ANOVA which can handle continuous outcomes. This is especially useful if we wish to use two-way ANOVA, which we will in a moment. First, let's replicate the χ^2 tests we just performed using one-way ANOVA.

```
# Migraine difference between AspiTyleCedrin use and not
summary(aov(Y_comp ~ A_comp, data = df))
```

```
##                Df Sum Sq Mean Sq F value Pr(>F)
## A_comp      1e+00  46219   46219  277286 <2e-16 ***
## Residuals   1e+06 166683       0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Migraine difference among sexes assigned at birth
summary(aov(Y_comp ~ W1, data = df))
```

```
##                Df Sum Sq Mean Sq F value Pr(>F)
## W1          1e+00    323   322.5    1517 <2e-16 ***
## Residuals   1e+06 212579     0.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Migraine difference among simplified racial groups
summary(aov(Y_comp ~ W2, data = df))
```

```
##                Df Sum Sq Mean Sq F value Pr(>F)
## W2          1e+00   5550    5550   26768 <2e-16 ***
## Residuals   1e+06 207351       0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that each of these p-values are again quite small, indicating evidence for a difference in migraine experience among the different groups.

Two-way ANOVA allows us to test for a difference in the dependent (outcome) variable among multiple independent variables, including interaction terms. For example, if we want to test for a difference in migraine experience among both sex assigned at birth and simplified racial group, we can do the following.

```
summary(aov(Y_comp ~ W1*W2, data = df))
```

```
##                Df Sum Sq Mean Sq  F value Pr(>F)
## W1          1e+00    323     323  1557.90 <2e-16 ***
## W2          1e+00   5551    5551 26813.35 <2e-16 ***
## W1:W2       1e+00      0       0     0.76  0.383
## Residuals   1e+06 207028       0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that we again found significant p-values for each of the two variables sex assigned at birth and simplified racial group, but we did not find a significant p-value for the interaction term. Note that this does not give us evidence that there is no interaction, it just does not give us evidence that there is interaction.

Question 14: Repeat the one-way and two-way ANOVA tests using the observed data (`A` and `Y_obs`) instead of the completely randomized data (`A_comp` and `Y_comp`). Briefly discuss your results.

```
summary(aov(Y_obs ~ A, data = df))
```

```
##                Df Sum Sq Mean Sq F value Pr(>F)
## A           1e+00  47092   47092  246969 <2e-16 ***
## Residuals   1e+06 190681       0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(aov(Y_obs ~ W1, data = df))
```

```
##                Df Sum Sq Mean Sq F value   Pr(>F)
## W1          1e+00      5   5.198   21.86 2.93e-06 ***
## Residuals   1e+06 237768   0.238
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(aov(Y_obs ~ W2, data = df))
```

```
##                Df Sum Sq Mean Sq F value Pr(>F)
## W2          1e+00  15952   15952   71913 <2e-16 ***
## Residuals   1e+06 221821       0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(aov(Y_obs ~ W1*W2, data = df))
```

```
##                Df Sum Sq Mean Sq  F value   Pr(>F)
## W1          1e+00      5       5    23.43 1.29e-06 ***
## W2          1e+00  15952   15952 71919.91  < 2e-16 ***
## W1:W2       1e+00     15      15    67.58  < 2e-16 ***
## Residuals   1e+06 221801       0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We see in the observed data the same results, that there is evidence of a difference in migraine experience among the different groups, but no evidence of an interaction between sex assigned at birth and simplified racial group. The p-values for sex assigned at birth are slightly different here than they were for the completely randomized data though.