R For SNA

Workshop 2

Eric Brewe

Associate Professor of Physics at Drexel University

11 August 2020, last update: 2020-08-10

Welcome Back!

Workshop #1 was a sprint!

This should be a bit more reasonably paced.

First of all, we will not need to build the network all over again.

But we should start with some project-oriented workflow.

The blog post by Jenny Bryan linked above describes the basic logic for a project orinted workflow.

Each project gets:

- It's own folder
- A project file (which you never open)
- As many Rmarkdown or R script files as needed
- Subfolders for:
    - data
    - plots
    - reports/papers etc.

Let's build on our project oriented workflow

Our project is about building and analyzing the Workshop network

So we will **not** need a new project folder, we will continue to use RForSNA, which you should have set up in WS1.

This time we will add a new .Rmd file, give it a title like "node_properties.Rmd"

So navigate to your RForSNA file and create a new .Rmd file.

Then, add a code chunk to load the libraries necessary and run this chunk.

```
library(tidyverse) #tools for cleaning data
library(igraph)  #package for doing network analysis
library(tidygraph) #tools for doing tidy networks
library(here) #tools for project-based workflow
library(ggraph) #plotting tools for networks
```

Let's load our data

This workshop assumes you did the work from week 1 and you know how to install libraries, read csv files into data, and to manipulate data in R.

The benefit of having done the work previously is that we don't have to re-do it (unless we want to change something)

So I took the network at the end of workshop #1 and saved it as a .rds file. I could have saved it as a pair of csv files and rebuilt the network, but if I save as rds it will be ready to go!

To load an rds file the syntax is a little different:

```
gr <- readRDS(here("data", "WokshopNetwork.rds"))
```

Let's inspect our data

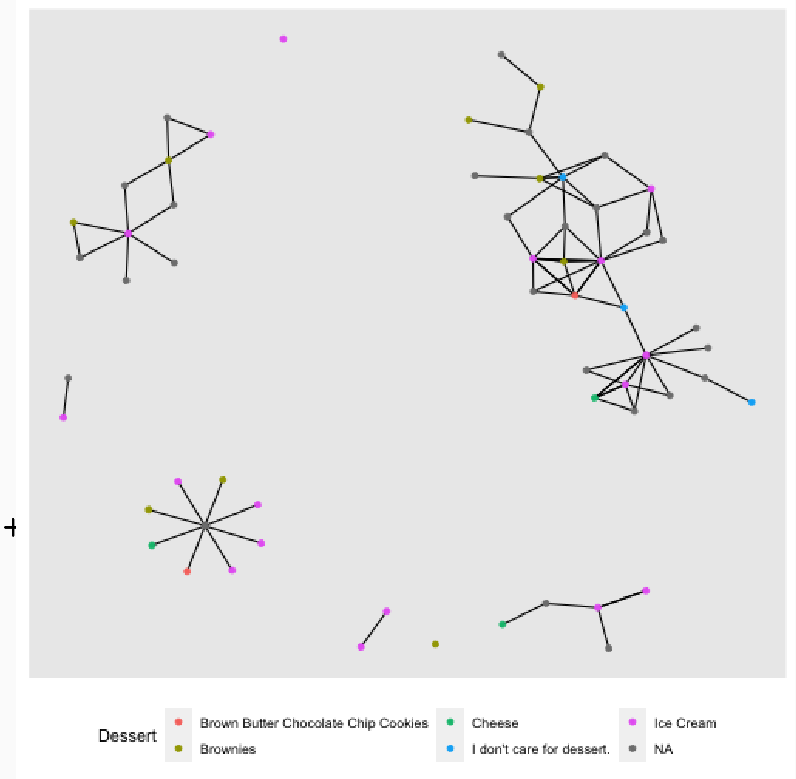
```
gr
```

```
## # A tbl_graph: 60 nodes and 101 edges
## #
## # A directed multigraph with 8 components
## #
## # Node Data: 60 x 4 (active)
##   name  AMPM             Dessert                    Pages
##   <chr> <chr>            <chr>                      <dbl>
## 1 5106  morning person. Brownies                     350
## 2 6633  morning person. I don't care for dessert.    12
## 3 7599  night owl.       Ice Cream                    300
## 4 4425  morning person. I don't care for dessert.     0
## 5 2495  morning person. Brownies                     264
## 6 6355  night owl.       Ice Cream                      4
## # … with 54 more rows
## #
## # Edge Data: 101 x 2
##    from    to
##   <int> <int>
## 1     1     1
```

Let's plot it quickly

Which node is the most important?

```r
gr %>%
  ggraph(layout = "kk") +
  geom_edge_link() +
  geom_node_point(aes(color = Dessert)) +
  theme(legend.position="bottom")
```



Let's explore centrality

One view of importance is how many people you are connected to.

- **Degree** = number of edges a node is involved in.
- **In Degree** = number of incoming edges
- **Out Degree** = number of outgoing edges

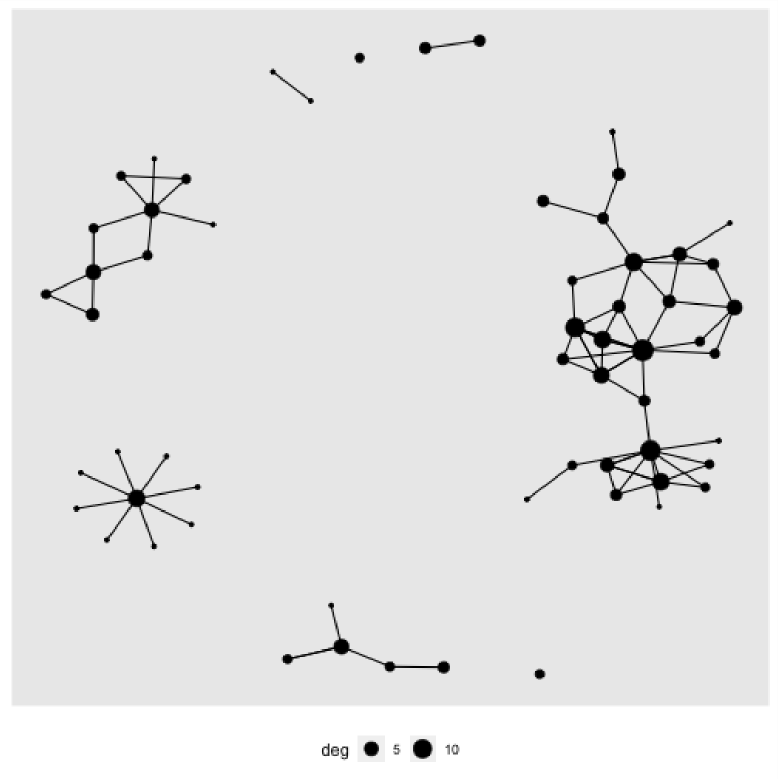
Lets calculate degree

```
gr %>%
  activate(nodes) %>%
  mutate(deg = centrality_degree(mode = "total")) %>%
  arrange(desc(deg)) -> gr

gr
```

```
## # A tbl_graph: 60 nodes and 101 edges
## #
## # A directed multigraph with 8 components
## #
## # Node Data: 60 x 5 (active)
##    name  AMPM            Dessert                           Pages   deg
```

Let's use the degree in a plot

```r
gr %>%
  ggraph(layout = "kk") +
  geom_edge_link() +
  geom_node_point(aes(size = deg )) +
  theme(legend.position="bottom")
```



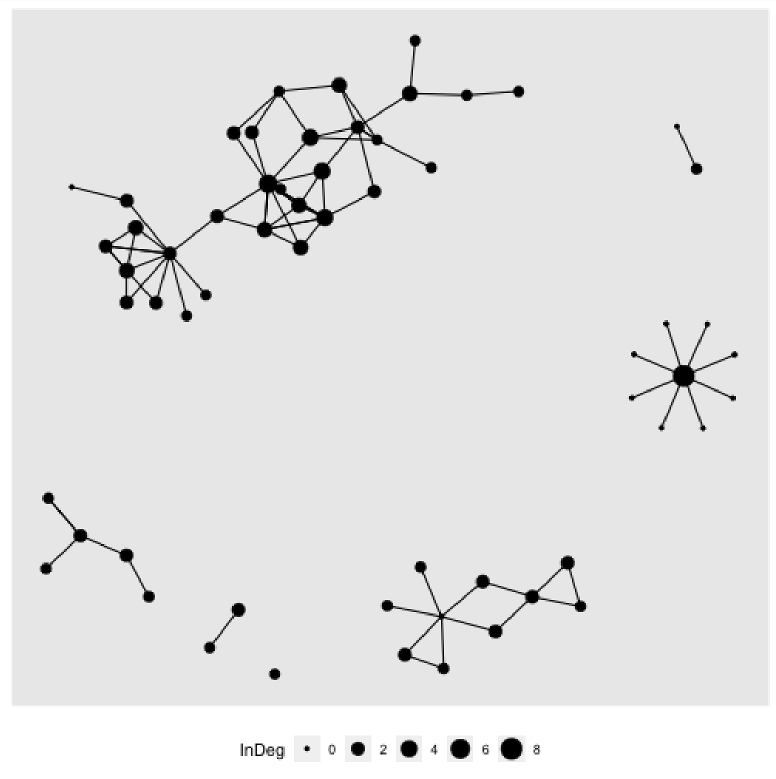
Let's see about just In-degree

```
gr %>%
  activate(nodes) %>%
  mutate(InDeg = centrality_degree(mode = "in")) %>%
  arrange(desc(InDeg)) -> gr

gr
```

```
## # A tbl_graph: 60 nodes and 101 edges
## #
## # A directed multigraph with 8 components
## #
## # Node Data: 60 x 6 (active)
##   name  AMPM              Dessert    Pages   deg InDeg
##   <chr> <chr>             <chr>      <dbl> <dbl> <dbl>
## 1 5844  <NA>              <NA>          NA     8     8
## 2 7743  morning person.   Ice Cream    300    14     5
## 3 1386  night owl.        Ice Cream    500    11     4
## 4 2169  <NA>              <NA>          NA     4     4
## 5 4755  <NA>              <NA>          NA     4     4
## 6 9929  morning person.   Brownies     527     8     3
## # … with 54 more rows
```

Let's use in degree in our plot

```r
gr %>%
  ggraph(layout = "kk") +
  geom_edge_link() +
  geom_node_point(aes(size = InDeg )) +
  theme(legend.position="bottom")
```



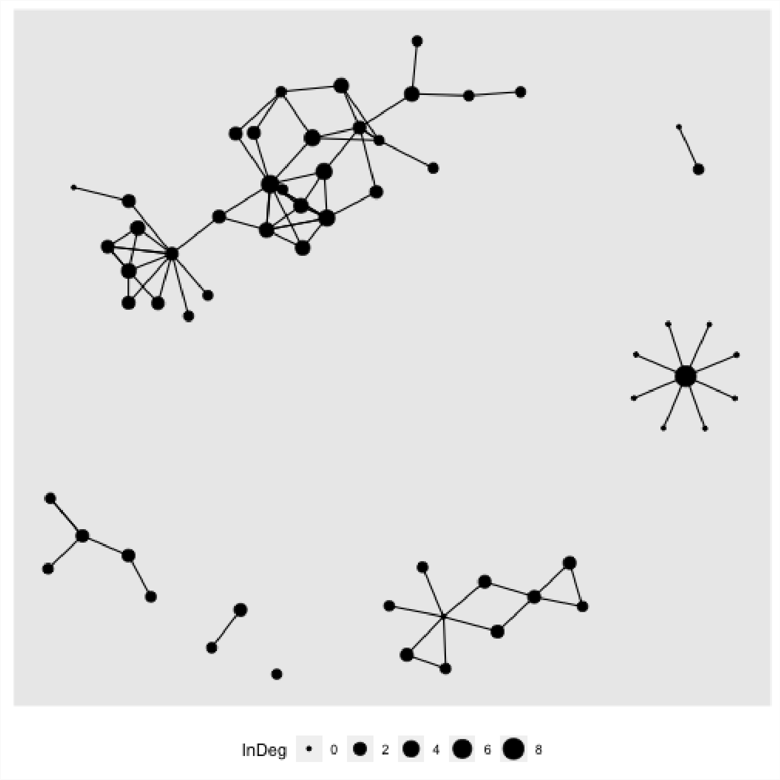
Lets compare centrality plots.

You might have noticed, the plots change layouts

To address this we set a layout...

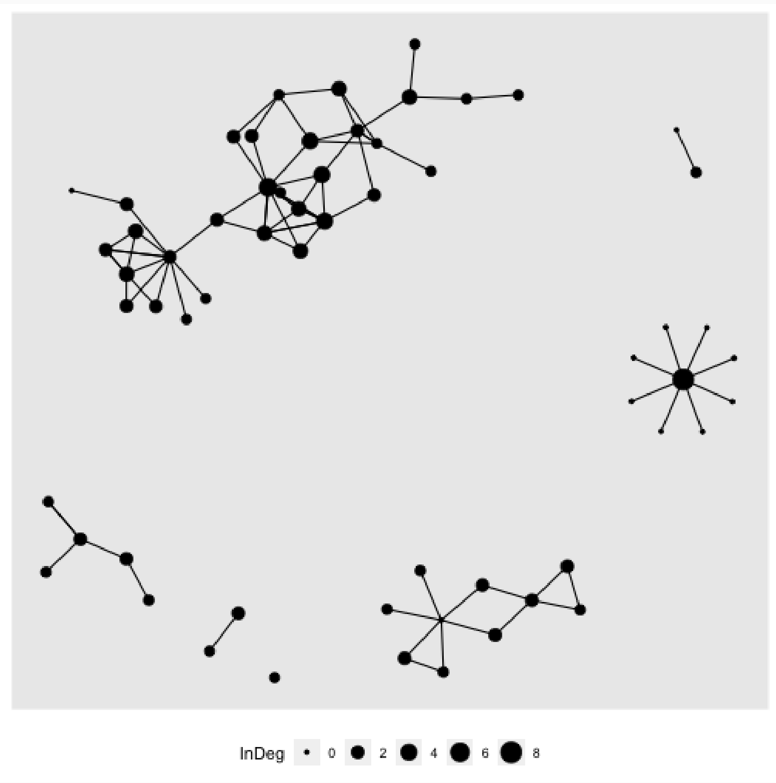
```r
GrLayout = create_layout(gr,
                layout = "kk")

ggraph(GrLayout) +
  geom_edge_link() +
  geom_node_point(aes(size = InDeg)) +
  theme(legend.position="bottom")
```

The only issue is, we then can't change nodes/attributes.

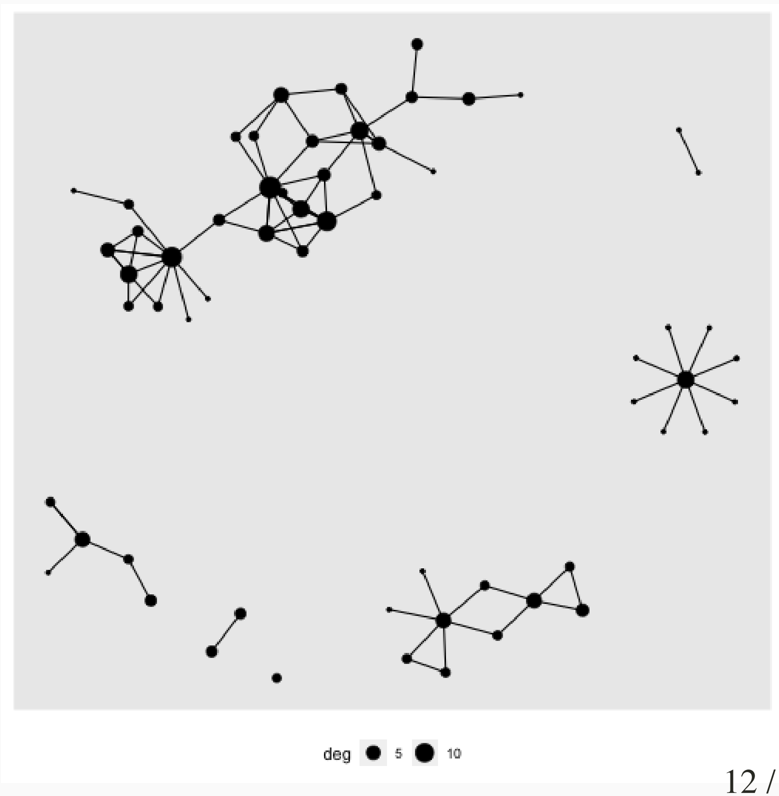


Let's compare in-degree and out-degree

```r
ggraph(GrLayout) +
  geom_edge_link() +
  geom_node_point(aes(size = InDeg)
  theme(legend.position="bottom")
```




```r
ggraph(GrLayout) +
  geom_edge_link() +
  geom_node_point(aes(size = deg))
  theme(legend.position="bottom")
```



Let's explore some other centrality measures

Here is a list of all of the centrality measures that you have access to in tidygraph:

http://finzi.psych.upenn.edu/R/library/tidygraph/html/centrality.html

This is a workshop and thus far I've done most of the work! It is your turn!

1. I will randomly assign people to breakout rooms.
2. As a group, choose one of the centrality measures from the list.
3. Calculate the centrality measure, use it in the plotting recipe we've been using, and be ready to describe what you think this centrality metric measures.
4. You will have 10-15 minutes.

Let's use centrality in hypothesis testing

We should establish a research question. How about:

Are morning people or night owls more central in our network?

To test this we might use:

1. An independent samples t-test to see if the means of a centrality metric is different.
2. A linear model to explore the relationships between the grouping variable and the outcome variable.

Either way, we need to choose a centrality metric.

Our winner...centrality_eigen!

Let's first calculate and plot eigenvector

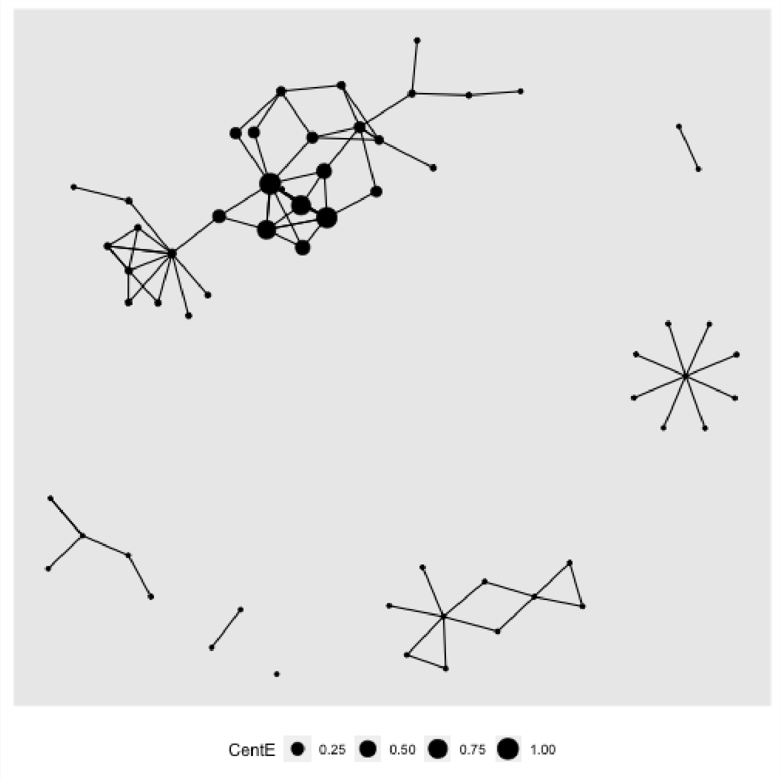
Eigenvector centrality is interesting because it allows a node to inherit centrality from neighbors. It works with weighted networks, and but it treats directed networks as undirected.

We want to calculate eigenvector centrality, then add it as an attribute to our graph, and add it to our layout dataframe.

```
gr %>%
  activate(nodes) %>%
  mutate(CentE = centrality_eigen()) %>%
  arrange(desc(CentE)) -> gr

#This makes a dataframe of names and CentE
gr %>%
  activate(nodes) %>%
  select(name, CentE) %>%
  as_tibble()-> CentEdf

#This adds it to the GrLayout dataframe.
GrLayout = left_join(GrLayout, CentEdf, by = "name")
```

Let's plot with eigenvector centrality

```r
ggraph(GrLayout) +
  geom_edge_link() +
  geom_node_point(aes(size = CentE)) +
  theme(legend.position="bottom")
```



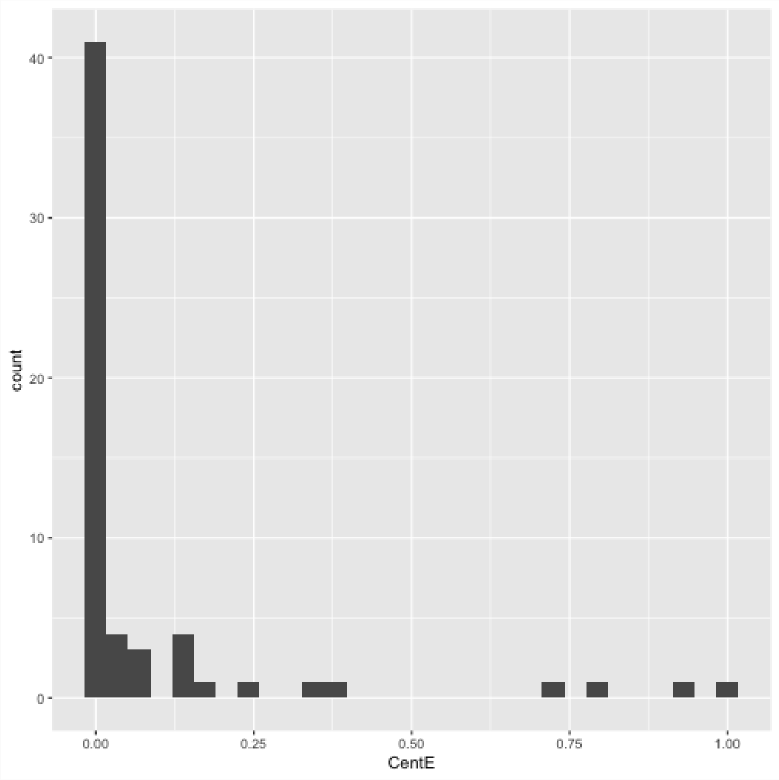
Cool, different nodes are important!

Let's get ready to use in linear model

First, let's see how it is distributed

````
```r
gr %>%
  select(AMPM,CentE) %>%
  as_tibble() %>%
  ggplot(., aes(x = CentE)) +
  geom_histogram()
```
````

Not Normal!



Let's reflect

Remember when I said...

"Nodes and interactions are interdependent*"

and

"* Violates basic assumption of inferential statistics"

Well, this is the problem, because network metrics are interdependent, they are invariably not normally distributed. Which means we have to take some additional precautions about how we use them in hypothesis testing. Which means...

Bootstrapping!

I said it here: https://ericbrewe.com/slides/rforsna/rforsna_ws1.html#8

Let's get ready to bootstrap

What is bootstrapping?

Bootstrapping is a resampling method, where you draw samples based on your existing dataset, calculate your measure of interest, store this, and then rinse and repeat.

R has a package for that.

Let's install the package "boot" and load it.

```
```r
install.packages("boot")
library(boot)
```
```

Let's set up a bootstrapped model

Our goal was to answer the question, "Are morning people or night owls more central in our network?"

So we will run a bootstrapped linear model to look at this relationship.

First, get our data and assign it to its own dataframe (you don't have to, but.)

```
GrLayout %>%
  select(AMPM,CentE) -> LmData
```

Next, we need to define the linear model as a function. Why? I'll explain in next slide.

```
bs <- function(formula, data, indices) {
  d <- data[indices,]
  fit <- lm(formula, data = d)
  return(coef(fit))
}
```

Let's run this linear model

Finally, we use the boot command to call the function

Since we want to run this once, and keep our results around, we should do two things.

1. Set our seed so that we can replicate if need be.
2. Assign our results to a data structure.

```
set.seed(327)

results <- boot(data = LmData,
     statistic = bs,
     R = 50,
     formula = CentE ~ AMPM)
```

Let's explore the results

First, what is the result if we don't use bootstrapping?

```
lm(CentE ~ AMPM, data = LmData)
```

```
##
## Call:
## lm(formula = CentE ~ AMPM, data = LmData)
##
## Coefficients:
##    (Intercept)  AMPMnight owl.
##       0.121912       -0.003964
```

Now, what about the bootstrap?

```
results
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = LmData, statistic = bs, R
##     AMPM)
##
##
## Bootstrap Statistics :
##         original        bias     std. e
## t1*  0.121911858  0.009475059  0.0557
## t2* -0.003963586 -0.019188955  0.0793
```

```
boot.ci(results, type = "basic")
```

Let's figure out what this means

- Original is the estimate without resampling.
- Bias is the difference between the bootstrap mean and the original.
- t1* is the intercept
- t2* is the coefficient for the AMPM variable

The confidence interval on the coefficients are (-0.0285, 0.2213)

So, are morning people more central than night owls?

Probabaly not

😞