Classwork 9: Love Island

The data for this week: a Love Island superfan recorded very detailed data on every contestant over the course of three seasons of the show. I had never seen Love Island before, but I was still pretty tickled about the idea of a data project on the topic. So I started watching the first episode of the 2016 series and I would like to make it clear, while I think this is sort of a fun data project, I do not necessarily recommend this show for everyone. If you're looking to get into reality TV, start with the Golden Bachelor instead.

Run this to get started:

```
library(tidyverse)
love <- read_csv("https://raw.githubusercontent.com/cobriant/320data/master/love.csv")
```

1. Dplyr: answer these questions using dplyr verbs.

1.1 Out of the three seasons, how many people won?

1.2 What is the minimum, maximum, and median age of contestants?

1.3 Are male contestants, on average, older than female contestants?

1.4 What are the three most common professions among contestants?

1.5 Continuing from 1.4, what are the three most common professions for male contestants and what are the three most common professions for female contestants?

1.6 What region of the UK are most of the contestants from?

1.7 Love Island seems to introduce people in waves, so people enter at different times. Show that in 2016, 42.3% of the cast arrived on day 1. Then in 2017, 34.4% arrived on day 1, and then in 2018, 28.9% arrived on day 1.

2 Explore: which characteristics seem to help people win the show?

2.1 There aren't really enough winners in only 3 seasons to do much inference,

so we'll expand the definition of "win" to include the runner ups and third place winners. Create a new column `win` that takes 1 if the person had an outcome of "winner", "runner up", or "third place", and 0 otherwise. These functions may be useful:

```
?qelp::mutate
?qelp::if_else
```

2.2 Draw a scatterplot plot with `age` on the x-axis and `win` on the

y-axis. Add a line of best fit with `geom_smooth(method = lm)`. Try replacing `geom_point` with `geom_jitter` since `win` takes on only 0 and 1, so `geom_point` will have an issue with overplotting. Experiment with `geom_jitter` arguments `height` and `width`.

2.3 Estimate the model $win_i = \beta_0 + \beta_1 age_i + u_i$ using `lm()`.

This model is called a "linear probability model" because the dependent variable `win` only takes on the values 0 or 1, so the fitted values can be interpreted as estimated probabilities of winning (although it would be absolutely possible to estimate "probabilities" under 0 or over 1). Give an interpretation of $\hat{\beta}_1$ and talk about

its statistical significance: "we (reject/fail to reject) the null hypothesis that age has no effect on success in the show at the .05 significance level". Recall that you can find p-values for an lm object by piping the lm object into the function `broom::tidy()`. You may need to install broom: `install.packages("broom")`.

```
library(broom)
```

2.4 Repeat the procedure above to answer the question, does joining the show earlier seem to help your chances of winning?

Draw a scatterplot with a line of best fit, estimate the model, and interpret the results.

2.5 One more question: Does being a model help your chances of winning?

To identify all the models, you might want to use `str_detect`.

3 Try a different measure of success and repeat the analysis from question 2. Do you get the same results?

You might look at the number of days the person was on the show, or the number of dates they went on.