Problem set 4

due Monday, November 20 at 11:59pm

> **Estimated time: 6 hours**
>
> Allocate about **1 hour per problem**, though some will take longer than others. You may need more time if programming is completely new to you, or less if you have some experience already.

Instructions Upload your `.ipynb` notebook to gradescope by 11:59pm on the due date.

- Note that each problem will be graded according to this rubric. Solutions that include packages or functions not covered in this course will recieve a score no higher than 2.
- You may collaborate with any of your classmates, but you must write your own code/solutions, understand all parts of the problem, and name your collaborators.
- You should also cite any outside sources you consulted, like Stack Overflow or ChatGPT, with a comment near the relevant lines of code (see example below). Recycled code that has not been cited will be considered plagerism and receive a zero.

```
# code here was inspired by user2554330 on stack overflow:
# https://stackoverflow.com/questions/69091812/is-everything-a-vector-in-r
```

Problem 0

not graded

Create a new colab R notebook. Please include the title "Problem set 4", your name, the date, and any collaborators somewhere at the top.

Problem 1

Import the data available at

```
"https://kathrynschuler.com/datasets/model-reliability-cubic.csv"
```

Problem 2

Explore the data with (at least) glimpse and a scatterplot. Include a visualization of a simple linear model (y ~ x) using geom_smooth. You may include any other explorations you wish to perform.

Problem 3

Fit a cubic polynomial model using `poly()` to the data and store your results as `observed_fit`. Use whichever of the three methods we learned in class that you prefer. Be sure to return the fitted model so we can see the parameter estimates.

Problem 4

Estimate the accuracy of the model on the population using bootstrapping or k-fold cross validation (choose one, not both). Use the `collect-metrics()` function to return the R^2 value.

Problem 5

Use `infer` to get a bootstrapped 68% confidence interval around the parameter estimates of your model. Visualize your bootstrapped distribution and shade the confidence interval.

Problem 6

Replot your scatterplot of the data and this time plot the cubic polynomial with `geom_smooth`. Use the `level` argument to include the 68% confidence interval.