Day 2: Introduction to Computational Statistics in Python

Objectives:

(1) Understand the basics of linear and logistic regression for computational modeling of behavioral data, and their appropriate application
(2) Explore the utility of the following python libraries: scikit-learn, statsmodel.api

Lesson Plan:

Module 1: Descriptive Stats & Their Short Falls

Descriptive Stats:
Begin by walking students through the use of excel, a-priori, to analyze both averages and X^2 statistics of the Fresh-Start dataset. Walk students through data-visualization via generating graphs using both excel and transitioning to python's matplotlib library.

The pitfalls of descriptive stats:
Ask students the following question—(1) do you see any trends in the data here with respect to how survey participants responded to questions b&c in the dataset, and how they responded to question a? (2) can you see any causal trends in this data?

Question (2) is in fact a trap, and explain why—"descriptive statistics can show us that there is a trend, but they fundamentally fall short of saying why this trend exists. To better understand why a trend is there, we need to really look at inferential statistical methods, which look at variables in the data to help use parse out why people acted the way they did in our dataset."

15-minute break

Module 2: Descriptive Statistical Methods

Logistic vs. Linear Regression:
Begin by asking students what the difference is between a category and a scale. Have them describe it verbally for a few minutes, but then invite a student to diagram it out on the board.

Ask students, "How do you know if something occurs on a scale versus in a category?" and write their responses on the board in two columns: "categories" and "scales". At the end of this brainstorming session, explain to the students that the point in this was to illustrate that there are different kinds of ways to analyze bits of evidence depending on whether a thing is a category or a scale. At this point describe linear regression including a visual graph on the board showing how different variables contribute to a smooth location in the graph. Now show logistic regression, and show how the slopes of different variables cuts the graph space into categorical quadrants.

Allow time for questions and discussion.

Logistic Regression in Python:
Have students open up their pycharm project libraries and create a new .py file in the folder day 2. Make sure all students have the requisite datasets (the fresh start data) Use the attached .py file to walk students through setting up a logistic regression model in scikit-learn.

Explain how they could do the same with a linear regression model, by changing the linear_model.<function> called. If time permits, have students walk through the census data set using only the numerical values to illustrate a linear regression model in action.

15-minute break

Module 3: Choosing a library

This is the closer, so it need not be particularly long. In this module, explain to students the difference between scikitlearn and statsmodels.api . . . "while scikitlearn allows us to build a robust engine for generating predictions from data, statsmodels is useful in analyzing those models as a whole. Statsmodels tells us if we're looking at a real trend, or if the model is lacking in validity, by looking at the statistical significance of the factors in our model.

In a new .py file in day 2, walk students in their pycharm project through importing statsmodels.api and running the fresh stat dataset through the model.

Conclusion & Individual questions period.