INFO370 Problem Set: Categorical variables and logistic regression

February 13, 2022

Instructions

This problem set revolves around logistic regression and categorical variables. It has two main goals:

1. Learn to handle categorical variables
2. Learn to use and interpret logistic regression results

You are welcome to discuss and work together, but you still have to submit your own work! Please list your collaborators on the submission!

1 Who will win the elections? (60pt)

This question asks you to do a simple election model. We are looking for the U.S. 2020 presidential elections by counties. Your task is to model the winner (1/0 for democratic/republican candidate winning the presidential elections in this county), and explain the winner using population density, education level, income, and geographic differences (the census region).

The data file is called *us-elections_2000-2020.csv.bz2*. The variables are

FIPS county FIPS code

year election year

state state name

state2 2-letter state code

region census region (west, midwest, south, northeast)

county county name

office President (we look only at presidential elections)

candidate name of the candidate

party party of the candidate

candidatevotes votes received by this candidate for this particular party

totalvotes total number of votes cast in this county-year

income personal income, USD/per capita (BEA—Bureau of Economic Analysis—data)

population population, census estimate (BEA data)

LND010200D land area (sq.mi) at 2000 (Census data)

EDU600209D Persons 25 years and over, total 2005-2009

EDU695209D Educational attainment - persons 25 years and over - bachelor's degree 2005-2009

POP010210D Resident population (April 1 - complete count) 2010

POP220210D Population of one race - White alone 2010 (complete count)

POP250210D Population of one race - Black or African American alone 2010 (complete count)

POP320210D Population of one race - Asian alone 2010 (complete count)

POP400210D Hispanic or Latino population 2010 (complete count)

PST110209D Resident total population estimate, net change - April 1, 2000 to July 1, 2009

The obscure variable names are pulled from the US Census.

1. (2pt) Load data, and do basic sanity checks.

2. (7pt) You are going to work with 2020 data. However, some important information for 2020 is missing. Fill the missings with the most recent values that exist in the data.

   Hint: check out `DataFrame.fillna` method.

3. (3pt) Ensure you order your observations right and do not fill missings with values from other counties. Print out a few lines before and after filling missings, where you show that you have done this correctly: a) missings are filled with the previous value; and b) it is the previous value for this county, not another county.

   Note: printing out challenging cases like here is a good approach to data processing in many contexts, not just in this PS.

4. (3pt) If you did this correctly, then even after filling in NA-s there are a few cases missing. What is going on? Explain!

5. (10pt) Make a new data frame that only contains 2020 data, and that contains a binary variable: the democrats won in that county in 2020.

   Hint: You have to build that variable using two lines of data in the original data frame *by FIPS* after the data is ordered by year. The orignal data contains two lines for each county, one for democrats and one for republicans. They contain the party-specific number of votes but are otherwise similar. You may extract the rows for democrats, the rows for republicans, and then just compare these two rows county-wise to see who won there. Note that it is *not enough* to just check if democrats/republicans got more than 50% of votes.

   However, when you extract the vote numbers, it will be a series with an index. You may want to either reset the index (see examples in Combining data into data frames) or convert the series into a numpy array with the `.values` attribute.

6. (10pt) Create auxiliary variables: population density (population divided by land area); and percentage of college graduates. These can be made of different variables, and as none of these are changing fast, it should not have much of an impact.

7. (2pt) Ensure that the variables you are going to use are in a reasonable range!

   Hint: there are values that do not make sense. Use `min` and `max` to check to find such values and remove those.

8. (10pt) Estimate logistic regression model where you explain democrats' winning with population density, education level, income, and census region.

9. (13pt) Interpret the results. Which results are statistically significant?

   Note: you may want to change some of the units, e.g. you may want to measure population density in 1000/per sq mi, instead of persons per sq mi.

2 Model AirBnB Price (40pt)

Your next task is to analyze the Beijing AirBnB listing price (variable *price*) in Beijing. It is downloaded from Inside Airbnb but we suggest to use the version on canvas (*airbnb-beijing-listings.csv*). You have to work with several sorts of categorical variables, including those that contain way too many too small categories. You are also asked to do log-transforms and interpret the results.

1. (2pt) Load data. Select only relevant variables you need below. Even better, check out the `usecols` argument for `read_csv`. Do basic checks.

2. (5pt) Do the basic data cleaning:

   (a) convert *price* to numeric.

   (b) remove entries with missing or invalid price, bedrooms, and other variables you need below

3. (4pt) Analyze the distribution of *price*. Does it look like normal? Does it look like something else? Does it suggest you should do a log-transformation?

   Hint: consult lecture notes Section 4.1.8 *Interactions and Feature Transformations*.

4. (6pt) Convert the number of bedrooms into another variable with a limited number of categories only, such as 0, 1, 2, 3, 4+, and use these categories in the models below.

   Hint: consult the python notes `https://faculty.washington.edu/otoomet/machinelearning-py/cleaning-data.html`

5. (6pt) Run a linear regression where you explain the listing price with number of bedrooms where *bedrooms* uses these categories. Interpret the results, including R^2.

   Hint: if 0-BR is the reference category, the effect for 1BR should be -11.62 (but it depends on how exactly did you clean data).

6. (8pt) Now repeat the process with the model where you analyze log price instead of price. Interpret the results. Which model behaves better in the sense of R^2?

   Hint: if you cleaned the data the same way as me, you should see $R^2 = 0.32$.

   For the following task use either *log(price)* or *price*, depending on your answer here.

7. (9pt) Finally we just add three more variables to the model: *room type*, *accommodates*, and *bathrooms*. While room type only contains three values, the other two contain many different categories. Recode these as

   - accommodates: “1”, “2”, “3”, “4 and more”
   - bathrooms: “0”, “1”, “2”, “3 and more”, where the 0.5 is rounded up to the next integer, e.g. 0.5 becomes 1, and 1.5 becomes 2.

   Run this model. Interpret and comment the more interesting/important results. Do not forget to mention what are the relevant reference categories and R^2.

Finally tell us how many hours did you spend on this PS.

References