MDML Homework 3

Due: 2019-10-17. Total Points: 100.

Instructions.

Please make one submission per group. Your submission should be a zip file, named in the following format: `hw3_lastname1_lastname2_lastname3.zip` (or `hw3_lastname1_lastname2.zip` etc. if you are a group of two). Please submit your zipped file on NYU classes.

The unzipped contents should **only** be the the following (do **not** submit data or figures).

1. `hw3_lastname1_lastname2_lastname3/scripts/poll_models.R`
2. `hw3_lastname1_lastname2_lastname3/scripts/pr_weapon_given_cpw_model.R`
3. `hw3_lastname1_lastname2_lastname3/scripts/question_b4.R`
4. `hw3_lastname1_lastname2_lastname3/written_responses.pdf`

Written Responses.

List each group member's full name at the top of the first page, and indicate the question number for each written answer.

Code.

All of your scripts will be executed from a directory **containing** your submission folder. That is, we will run your scripts from the command line with, e.g.:

`Rscript hw3_lastname1_lastname2_lastname3/scripts/poll_models.R`.

This means that any directories you set should be relative to the submission directory's *parent* directory (i.e., the directory one level above the folder `hw3_lastname1_lastname2_lastname3/`).

Writing data

We will ask you to output dataframes to `.csv` files for various problems. When you write these `.csv` files, remember to include column headers; however, do not include the row indices.

Grading.

Assignments submitted after the beginning of class (11am) on the due date are considered late. Please see the syllabus for the late policy. You will be graded on the following: how accurately you followed instructions; correctness, completeness, and clarity of your code and written answers; creativity (when applicable); and quality of visualizations. Note that **all group members** should discuss and work together on **each part** of the assignment.

Part A. [50 points]

In this question, we'll apply a logistic regression model to make predictions using polling data collected before the 2012 U.S. presidential election.

Question A1: Setup [5 points].

1. Create the following directories.
    - Your submission directory, named in the following format: `hw3_lastname1_lastname2_lastname3/`.
    - A `scripts/` directory in your *submission* directory.
    - A `data/` directory in your *submission* directory.
    - A `figures/` directory in your *submission* directory.
    - A `data_hw3/` directory in your submission's *parent* directory.
2. Download the `poll_data.tsv` and `poll_data_full.tsv` files from NYU classes and move them into your `data_hw3` directory.
3. Create a script called `poll_models.R` in the `scripts/` directory of your submission. *You will do all the work for questions A1, A2, A3, and A4 in this script.* First, read the data from `data_hw3/poll_data.tsv`, and call the resulting tibble `poll_data`. The path should be relative to your submission's *parent* directory.
4. Convert `vote_2008` into a factor, making `"john mcCain"` the reference category.

Question A2: Fitting a model [10 points].

1. Fit a binary logistic regression model that estimates individuals' probabilities of voting for Obama in the 2008 presidential election, using all the other features in the dataset.
2. Store the coefficient names and the estimates (including the intercept) in a tibble with two columns: `coefficient_name` and `coefficient_estimate`. Order the rows of this tibble alphabetically according to `coefficient_name` and save it to `data/question_a2_coefficients.csv` within your submission directory.
3. Create a tibble with 3 columns: `variable`, `number_of_levels`, `number_of_fitted_coefficients`. Store the 8 predictor variable names into the `variable` column, the number of unique values each variable takes on in the corresponding row in the `number_of_levels` column, and the number of fitted coefficients for that variable in the `number_of_fitted_coefficients` column. Save this tibble to `data/question_a2_levels_vs_coefficients.csv` within your submission directory.
4. In your written responses, write one paragraph interpreting some of the coefficients. If the model did not fit some coefficients, explain why in your write up.

Question A3: Evaluating the Model [15 points].

1. Use the logistic regression model from Question A2 to calculate predicted probabilities of voting for Obama in 2008 for all of the individuals in the dataset. Save these probabilities as a column called `predicted_probability`.
2. Convert the probabilistic predictions for each individual into binary predictions based on the candidate they are most likely to vote for. Save these binary predictions as a column called `predictions_point_5`.
3. Repeat the previous step, but now convert each individual's prediction to a binary prediction for Obama only if the individual's probability of voting for Obama is at least 0.7. Save these binary predictions as a column called `predictions_point_7`.
4. Write just the three columns `predicted_probability`, `predictions_point_5`, and `predictions_point_7` to `data/question_a3.csv` within your submission directory.
5. Compute the accuracy, precision, and recall metrics for the binary predictions produced in steps 2 and 3 (without using a package that does it automatically). Report these values and any notable differences between the metrics corresponding to the predictions in steps 2 and 3 in your writeup.

Question A4: Another model [20 points].

Not everyone votes for major party candidates in elections, so a binary prediction isn't always the best approach for predicting votes.

1. Read in `poll_data_full.tsv`, and call the resulting tibble `poll_data_full` (these data include individuals who voted 'other'). Using all available features, build a binary logistic regression model to predict whether an individual voted for a major-party candidate in the 2008 elections (both Obama and McCain are major party candidates). Save these predictions as a column called `pr_major`.
2. Filter `poll_data_full` to only individuals who actually voted for major party candidates. *On this subset*, use all features to build a binary logistic regression model to predict whether an individual voted for Obama. This model allows us to estimate Pr(voted Obama | voted major party candidate). Using this model, generate estimates of Pr(voted Obama | voted major party candidate) for *every* individual in `poll_data_full`. Save these predictions as a column called `pr_obama_given_major`.
3. Use `pr_major` and `pr_obama_given_major` to compute three numbers for each individual in `poll_data_full`: the probability that the individual votes for Obama, the probability that the individual votes for McCain, and the probability that the individual votes for 'Other'. Save these predictions as columns `pr_obama`, `pr_mccain`, and `pr_other`. Next, generate categorical predictions for each individual based on these probabilities, and save these categorical predictions in a column called `predictions`. Report the accuracy of your classifier in your writeup.
4. Write just the six columns `pr_major`, `pr_obama_given_major`, `pr_obama`, `pr_mccain`, `pr_other`, and `predictions` (for every individual in `poll_data_full`) out to `data/question_a4.csv` within your submission directory.
5. Create one figure with two subplots: a histogram of `pr_major`, and a histogram of `pr_obama_given_major`. Save this figure to `figures/question_a4.png` within your submission, and briefly describe the histograms in your writeup.
6. It might have occurred to you that we could generate `pr_obama` and `pr_mccain` for every individual in `poll_data_full` in an easier way: we could have just fit separate logistic regression models on the entire dataset, the first predicting Pr(voted Obama), and the second Pr(voted McCain). Along with the predictions from the model in step 1, we'd then have `pr_obama`, `pr_mccain`, and `pr_other` for every individual. What is the advantage of generating `pr_obama` and `pr_mccain` using the procedure in steps 2 and 3? (Hint: Add up `pr_obama`, `pr_mccain`, and `pr_other` from step 4.).

Part B. [50 points]

In this question, we will build models using the stop-and-frisk dataset from Assignment 2.

Question B1: Set up [5 points].

1. Download and unzip `sqf_08_16.csv` from NYU classes and move it into the `data_hw3` directory.
2. Create a `pr_weapon_given_cpw_model.R` script in the `scripts/` directory of your submission. *You will do all the work for questions B2 and B3 in this script.*
3. Create a `question_b4.R` script in the `scripts/` directory of your submission. *You will do all the work for question B4 in this script.*

Question B2. [5 points].

1. Read the data from `data_hw3/sqf_08_16.csv`, and restrict to stops where the suspected crime is 'cpw'.
2. Train a logistic regression model on all of 2008, where the outcome variable is whether or not a weapon is found and the predictor variables are the following (with real-valued attributes standardized by subtracting the mean and dividing by the standard deviation):

- precinct;
- whether the stop occurred in transit, housing, or on the street;
- the ten additional stop circumstances (`additional.*`);
- the ten primary stop circumstances (`stopped.bc.*`);
- suspect age, build, sex, height, weight;
- whether the stop occurred inside;
- whether the stop was the result of a radio call;
- length of observation period;
- day, month, and time of day.

3. Store the ten largest and ten smallest coefficient names and estimates in a dataframe with two columns: `coefficient_name` and `coefficient_estimate`. Order the rows of this dataframe alphabetically according to `coefficient_name`. Save this dataframe to `data/question_b2_coefficients.csv` within your submission directory.
4. Give a precise statement interpreting one of these coefficients in your writeup.

Question B3. [20 points].

Use the model you built in Question B2 in each part of this question.

1. Suppose a 30 year old, six-foot tall, 165 lb man of medium build was stopped in the West 4th subway station on 10/4/2008 at 8pm (no weapon was found). Upon reviewing the UF-250 form filled out for his stop, you notice that he was suspected of criminal possession of a weapon, and was stopped because he had a suspicious bulge in his coat, and he was near a part of the station known for having a high incidence of weapon offenses. He was observed for 10 minutes before the stop was made, and the stop was not the result of a radio call. If your model were used to predict the ex-ante probability that this person were carrying a weapon, what would this probability be? What if this person were a woman, everything else being equal?[1] Report both of these numbers in your writeup.
2. Compute the AUC of this model on all data from 2009, using the ROCR package (as in lecture). Report this number in your writeup.
3. The AUC can be interpreted as the probability that a randomly chosen true instance will be ranked higher than a randomly chosen false instance. Check that this interpretation holds by sampling (with replacement) 10,000 random pairs of true (weapon is found) and false (weapon is not found) examples from 2009, and computing the proportion of pairs where your model predicts that the true example is more likely to find a weapon than the false example. Confirm that your answer is approximately equal to the answer computed in Question B3.2, and report this number in your write up.

Question B4. [20 points].

For this question, you will generate a recall-at-k% plot and a calibration plot (like the ones created during lecture) for a classifier of your choice by following the steps below.

1. Choose a target variable that is not `found.weapon` or `found.gun`. For example, you might predict whether a suspect is arrested, frisked, searched, whether a summons is issued, whether contraband is found, or whether force (or a specific type of force) is used.
2. Select a set of predictor variables. Feel free to generate your own features, e.g., interaction terms, but be sure to only use predictors that are determined before the outcome would have been known.
3. Create a train-test split of the data either randomly (e.g., train on a random 50% of rows and test on the other half) or temporally (e.g. 2008-2010 data for training and 2011 for testing).
4. Select a classification method of your choosing (e.g., logistic regression). If it makes sense, restrict to a subset of the data. For example, if your outcome measure is whether contraband is found, you may

[1] This suggests a statistical strategy for assessing discrimination. For example, if model-estimated ex-ante probabilities of weapon recovery were generally higher for women than for men, it might suggest that officers had a higher ‘bar’ for stopping women compared to men. However, this interpretation is complicated by several statistical and substantive issues.

want to restrict to just stops where the suspected crime involves criminal sale/possession of marijuana (or “marihuana”) and criminal sale/possession of a controlled substance.

5. Generate a *recall-at-k%* plot by sweeping over all possible thresholds, where for a given threshold, the x value represents the proportion of stops with estimated probability above the threshold, and the y-value represents the corresponding recall. For example, if half the stops have an estimated probability above the threshold of 0.3, and that subset of stops contains 3/4 of all positive cases, then (0.5, 0.75) would be a point on the plot. Note: sweeping over all thresholds is equivalent in practice to ranking stops in descending order by estimated probability, then for each stop, computing the proportion of stops that have greater or equal estimated probability, as well as the proportion of all positive cases contained in that subset. Save this to `figures/recall_at_k.png` in your submission.
6. Generate a *calibration* plot. To generate the plot, first round the model predictions to the nearest percentage point. For each resulting bin of rounded predictions, plot the average model prediction on the x-axis, and the empirical frequency of positive outcomes on the y-axis (points closer to the diagonal correspond to better calibration; make sure to plot the 45 degree line as well). Also, to see the distribution of model predictions, size each by the total number of events in that bin. Save this to `figures/calibration.png` in your submission.
7. Write at least one paragraph explaining what you did and what you found.

Checklist.

When executed from your submission’s parent directory, your scripts should generate the following. Do **not** include files in the submission—we will generate them by running your scripts.

1. `hw3_lastname1_lastname2_lastname3/data/question_a2_coefficients.csv`
2. `hw3_lastname1_lastname2_lastname3/data/question_a2_levels_vs_coefficients.csv`
3. `hw3_lastname1_lastname2_lastname3/data/question_a3.csv`
4. `hw3_lastname1_lastname2_lastname3/data/question_a4.csv`
5. `hw3_lastname1_lastname2_lastname3/data/question_b2_coefficients.csv`
6. `hw3_lastname1_lastname2_lastname3/figures/question_a4.png`
7. `hw3_lastname1_lastname2_lastname3/figures/recall_at_k.png`
8. `hw3_lastname1_lastname2_lastname3/figures/calibration.png`