MDML Homework 3

Due: 03-10-2022. Total Points: 100.

Instructions.

Your submission should be a zip file called `hw3_`[firstname]`_`[lastname]`.zip` (e.g., `hw3_ravi_shroff.zip`). You may discuss this assignment with your professor, course assistant, and classmates, but you must turn in your own work. In particular, *you may not directly copy anyone else's code*; that would constitute plagiarism. The unzipped folder should contain only the following files (in particular, do **not** submit any data files):

1. `poll_models.R`
2. `sqf_models.R`
3. `written_responses.pdf`

Written Responses.

Submit a pdf file containing your written responses as `written_responses.pdf`. Include your full name, and indicate the question number for each written answer.

Code.

Please comment your code clearly and extensively and make sure it runs without error. If we can't understand what you're doing, we won't be able to grade that part of your assignment.

Grading.

Submit your zipped assignment folder on Brightspace. Assignments submitted after the beginning of class (10:15am) 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 (when applicable). Note that there is 10 points worth of extra credit available on this assignment.

Part A. Predicting election outcomes [50 points]

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

Question A1: Estimate the probability of voting for Obama [35 points].

1. Create a folder on your computer for this assignment. Within this folder, create a `scripts/` directory, a `data/` directory, and a `figures/` directory. Download the `poll_data.tsv` and `poll_data_full.tsv` files from Brightspace and move them into the `data/` directory.
2. Create a script called `poll_models.R` in the `scripts/` directory (you will do all the work for Part A in this script). Read the data from `data/poll_data.tsv`, and call the resulting tibble `poll_data`. Then, convert `vote_2008` into a factor, making `"john mcCain"` the reference category.
3. Fit a logistic regression model that estimates individuals' probabilities of voting for Obama in the 2008 presidential election, using all the other features in the dataset. 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.

4. Use this logistic regression model to calculate predicted probabilities of voting for Obama in 2008 for *all* the individuals in the dataset. Add these probabilities to `poll_data` as a column called `predicted_probability`. Then, use these probabilistic predictions to create *binary* predictions for each individual for the candidate they are most likely to vote for. Add these binary predictions to `poll_data` as a column called `predictions_point_5`.
5. Now, create another vector of binary predictions, where an individual is predicted to vote for Obama if their `predicted_probability` from the previous step is at least 0.7. Add these binary predictions to `poll_data` as a column called `predictions_point_7`.
6. Compute the accuracy, precision, and recall metrics for the binary predictions produced in steps 4 and 5 (don't use a package that does it automatically). Report these values and comment briefly in your writeup on any notable differences between the metrics corresponding to the predictions in steps 4 and 5.

Question A2: Building a classifier for more than two classes [15 points].

Not everyone votes for major party candidates in elections, so a binary prediction isn't always the best approach for predicting votes. In this question, you'll estimate probabilities of an individual voting for Obama, McCain, or 'other'.

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). Add these predictions to `poll_data_full` 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 (except `pr_major`) 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`. Add these predictions to `poll_data_full` 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'. Add these predictions to `poll_data_full` as columns `pr_obama`, `pr_mccain`, and `pr_other`, respectively. 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 this classifier in your writeup.

Part B. Prediction and "Stop-and-Frisk" [50 points]

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

Question B1: Predicting weapon recovery [30 points].

1. Download and unzip `sqf_08_16.csv` from Brightspace and move it into the `data/` directory. Create a `sqf_models.R` script in your `scripts/` directory; you will do all the work for Part B in this script. Read the data from `data/sqf_08_16.csv`, and restrict to stops where the suspected crime is 'cpw'. Remove stops in Precinct 121, then `select` just the following variables:
    - whether a weapon was found;
    - 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.

Then, restrict to complete cases, so no row has any missing information. Call this tibble `sqf`.

2. Train a logistic regression model on all of the stops in `sqf` from **2008**, where the outcome variable is whether or not a weapon is found and the predictor variables are everything else (standardize real-valued attributes by subtracting the mean and dividing by the standard deviation). Give a precise statement interpreting one of these coefficients in your writeup.

3. 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 the logistic regression model you just built was 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. (Hint: remember to standardize the real-valued attributes).

4. Compute the AUC of this model on all data from **2009**, using the ROCR package (as in lecture). Report this number in your writeup.

5. [**5 points extra credit**] 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 Step 4, and report this number in your write up.

Question B2. Assessing model performance [20 points].

For this question, you will generate *either* a recall-at-k% plot *or* a calibration plot (like the ones created during lecture) for a classifier of your choice by following the steps below. In other words, do *either* part 5 *or* part 6 below. You may choose to do *both* parts (i.e., to generate *both* plots) for 5 points extra credit.

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. If it makes sense, restrict to a subset of the data. For example, if your outcome measure is whether contraband is found, you may 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.

2. Select a set of predictor variables. Feel free to generate your own features, e.g., interaction terms, but *be sure to only use variables 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). You may choose to restrict to a subset of years if you want.

4. Select a classification method of your choice (e.g., logistic regression), fit it on the training data, and make predictions for the test data.

[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 'threshold' for stopping women compared to men. However, this interpretation is complicated by several statistical and substantive issues.

5. Generate a *recall-at-k%* plot (like we covered in class) 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 in `figures/recall_at_k.png`.
6. Generate a *calibration* plot (like we covered in class). 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, each point's size should represent the total number of events in that bin. Save this in `figures/calibration.png`.
7. Write at least one paragraph explaining what you did and what you found.