CSE 258, Winter 2017: Homework 1

Instructions

Please submit your solution **by the beginning of the week 3 lecture (Jan 23).** Submissions should be made on **gradescope**. Please complete homework **individually**.

You will need the following files:

50,000 beer reviews : `http://jmcauley.ucsd.edu/cse258/data/beer/beer_50000.json`

UCI Wine Quality Dataset :
`http://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv`

Code examples : `http://jmcauley.ucsd.edu/cse258/code/week1.py` (regression) and `http://jmcauley.ucsd.edu/cse258/code/week2.py` (classification)

Executing the code requires a working install of Python 2.7 or Python 3 with the scipy packages installed. **Please include the code of (the important parts of) your solutions.**

Tasks — Regression (week 1):

In the first two questions, we'll see how ratings vary across different years in our dataset of 50,000 beer reviews. These questions should be completed on the *entire dataset.*

1. First, let's train a predictor that uses the year ('review/timeStruct'/'year') to predict the overall rating, i.e.,

   $$\text{review/overall} \simeq \theta_0 + \theta_1 \times \text{year}.$$

   You may use Python libraries to do so, so long as you include the code of your solutions. What are the fitted values of θ_0 and θ_1? (1 mark)

2. A simple regressor like the one above may not be very realistic—it assumes that ratings get linearly better or linearly worse over time. Can you come up with a better representation of the year variable?

   Describe your representation and write down an equation for it in terms of θ (like the equation from Q1 above). Compare the the new representation to the representation from Question 1 in terms of the Mean Squared Error (i.e., report the MSE for both representations) (1 mark).

Next, we'll use the *UCI Wine Quality Dataset* to train a regressor with a few more features. This data is in *CSV* format, and can be processed using the Python CSV library (`https://docs.python.org/3.6/library/csv.html`). See `https://archive.ics.uci.edu/ml/datasets/Wine+Quality` for a few more details about this dataset.

Start by splitting the data into 'train' and 'test' portions by taking the first half of the rows for training data and the remaining rows as test data.

3. Next, train a regressor that uses the first 11 features to predict the last feature ('quality'), i.e.,

   $$\text{quality} = \theta_0 + \theta_1 \times \text{'fixed acidity'} + \theta_2 \times \text{'volatile acidity'} + \theta_2 \times \text{'citric acid'} + \ldots + \theta_{11} \times \text{'alcohol'}.$$

   Write down the fitted coefficients on the training data, and the MSE on the train and test data (1 mark).

4. An *ablation* experiment consists of removing one feature from an experiment, in order to assess the amount of *additional* information that feature provides above and beyond the others. Repeat the experiment from Question 3 for all possible ablations (i.e., removing the 'fixed acidity' feature only, removing 'volatile acidity' only, etc.).

   (a) Report the MSEs (on the test set) of all 11 ablation experiments (1 mark).

   (b) Based on the test MSEs, Which features do you conclude provide the most and least additional information beyond what is present in the 11 other features? (1 mark)

Classification (week 2):

Finally, we'll treat the *Wine Quality* task from above as a *classification* task. Again, split the data so that the first half is used for training and the second half is used for testing as above.

To turn this into a classification problem, split the data so that 'negative' examples are those where quality ≤ 5 and 'positive' examples are those where quality > 5.

5. Again using the first 11 features, run an SVM classifier on the data (see the code provided in class) – remember to train on the first half and test on the second half. What is the accuracy (percentage of correct classifications) of the predictor on the train and test data (1 mark)?

6. **(Hard)** Finally, let's fit a model using logistic regression. A code stub has been provided to perform logistic regression using the above model on `http://jmcauley.ucsd.edu/cse258/code/homework1.py` Code for the log-likelihood has been provided in the code stub (`f`) but code for the derivative is incomplete (`fprime`)
   - Complete the code stub for the derivative (`fprime`) (1 mark).
   - What is the log-likelihood of after convergence, and what is the accuracy (on the test set) of the resulting model? (1 mark)