Fall 2019, DS 5220

Homework 3
Instructor: Alina Oprea
TAs: Christopher Gomes, Yuxuan Wang

Due date: October 25, 2019

Instructions

- Submit a PDF writeup named "%LASTNAME%_HW3.pdf" in Gradescope. Please answer all the questions in the PDF. If a problem asks for graphs or other results, please extract the results from the code and include them in the report. The PDf report should have complete answers to all the questions.
- Please use Jupyter notebooks (either Python or R) as we will be selectively running your code. Include comments in the code, and a Readme file if needed.
- Submit your assignment code as a zip file here. Use the name [LASTNAME]_[FIRST]_code3.zip for your code file.

Course policy on collaboration and cheating:

- You may discuss the concepts with your classmates, but write up the answers entirely on your own.
- You cannot share your code with your classmates.
- You cannot use code from the Internet for your assignment.
- You can post questions on Piazza and are encouraged to come to the TA and Instructor office hours.

Dataset: The dataset for this assignment is the SPAMBASE dataset from the UCI repository available at: `https://archive.ics.uci.edu/ml/datasets/spambase`

The first 57 columns are features counting word frequencies (see documentation at `https://archive.ics.uci.edu/ml/machine-learning-databases/spambase/spambase.names`). The last column indicates 1 for the SPAM class and 0 for the HAM class.

Problem 1 [Logistic regression] - 20 points

Use an existing package of your choice to train and test a logistic regression model.

(a) Split the original data into 75% for training and 25% for testing. Choose the training set at random and ensure that the fraction of SPAM examples in the training set is close to the fraction of 39.4% SPAM examples in the entire dataset.

Train a logistic regression model on the training set and output the following **on the testing set**:

1. **Confusion matrix**
2. **True Positives**, **False Positives**, **True Negatives**, **False Negatives**
3. **Accuracy**, **Error**
4. **Precision**, **Recall**, **F1 score**

(b) Print the coefficients of the features in the model. Which features contribute mostly to the prediction? Which ones are positively correlated and which ones are negatively correlated with the SPAM class?

(c) Vary the decision threshold $T \in \{0.25, 0.5, 0.75, 0.9\}$ and report for each value the model accuracy, precision, and recall. Comment on how these metrics vary with the choice of threshold.

(d) Use your implementation of gradient descent from Homework 2 and adapt it for logistic regression. Take 3 values of the learning rate and report the cross-entropy loss objective after 10, 50, and 100 iterations. At 100 iterations, report the accuracy and F1 score for the 3 learning rates, and compare with the metrics given by the package.

Problem 2 [Comparing classifiers] - 20 points

In this problem, you will use existing packages of your choice for training and testing various classifiers, and then compare them. You will use the same SPAMBASE dataset.

You can use the same training and testing data as in Problem 1. Train the following classifiers using the training data:

1. Logistic regression
2. LDA
3. kNN
4. Naive Bayes
5. Decision tree

(a) Experiment with different values of k for kNN and report 2 metrics on the training and testing sets: **accuracy** and **error**. Choose the value of k that gives the highest accuracy in testing.

(b) Print the **accuracy** and **error** metrics for all 5 classifiers on both training and testing data. Which model is performing best? Which one is performing worst? Write down some observations.

(c) Generate a graph that includes 5 ROC curves (one for each of the 5 classifiers) on the testing set. Compute the Area Under the Curve (AUC) metric for all 5 classifiers.

Problem 3 [kNN] 20 points

In this problem, you will implement your own kNN (k Nearest Neighbors) model, using the Euclidian distance between points as a distance metric. You will also compare your model with the one trained with the kNN package of your choice (the same one you used in Problem 2).

Select 100 records from the dataset for training and 100 records for testing. Make sure that both the training and testing dataset have the same fraction of SPAM emails as the original data.

(a) Implement a function that computes the Euclidian Distance between 2 points with d features. If $x = (x_1, \ldots, x_d)$ and $y = (y_1, \ldots, y_d)$, the Euclidian Distance between points x and y is $\sqrt{\sum_{i=1}^{d}(x_i - y_i)^2}$.

(b) Write the implementation for the kNN classifier. Given the value of k and the training set, you should implement a **test** function that produces a predicted label for a new point x in the testing set.

(c) Pick several values of k (the same ones you picked in Problem 2) and print the **accuracy** and **error** metrics **on the test set** using your implementation of the kNN classifier.

(d) Compare the results obtained by your implementation with those obtained with the package (on the same dataset). Are the results similar or different? If there are differences, explain why.

(e) Report the running time of kNN testing averaged over all the points in the testing set.

Problem 4 [Cross validation] 20 points

In this problem, you will implement your own k-fold cross-validation algorithm and apply it to two linear classifiers (Logistic Regression and LDA).

(a) Implement k-fold cross-validation (CV) for training a model. The CV algorithm consists of the following steps:

  (a) Divide the entire data into k partitions of equal size.

  (b) Run k experiments. In each experiment $i \in \{1, \ldots, k\}$, train on $k - 1$ partitions and test on the validation set (partition i).

  (c) Record the validation error for each experiment.

  (d) Compute and print the **average validation error** across all k experiments.

(b) Run the CV experiment for logistic regression and LDA for $k \in \{5, 10\}$. You can use a package for training the logistic regression and LDA models. Print for each model the average validation error for each value of k.

(c) Which model performs better? Compare the results.

Problem 5 [Naive Bayes] 10 points

Assume you have the following training set with three binary features X_1, X_2 and X_3, and a binary response variable Y.

X_1	X_2	X_3	Y
0	1	1	1
1	0	0	1
1	1	0	1
0	1	1	1
1	1	0	0
1	0	1	0
0	0	1	0

(a) Estimate $P(Y = 1|X_1 = 1, X_2 = 0, X_3 = 1)$ and $P(Y = 1|X_1 = 1, X_2 = 1, X_3 = 1)$ using Bayes rule, with the Naive Bayes assumption (stating that the conditional probabilities of features given the label are independent). How many parameters are estimated and stored by the Naive Bayes classifier?

(b) Estimate $P(Y = 1|X_1 = 1, X_2 = 0, X_3 = 1)$ and $P(Y = 1|X_1 = 1, X_2 = 1, X_3 = 1)$ using Bayes rule, without the Naive Bayes assumption. How many parameters are estimated and stored in this case?

Problem 6 [Logistic regression] 10 points

Suppose we collect data for a group of students in a machine learning class with variables X_1 = hours studied, X_2 = GPA, and response variable Y = receive an A in the class. We fit a logistic regression and produce estimated coefficients: $\theta_0 = -6; \theta_1 = 0.05; \theta_2 = 1$.

(a) Estimate the probability that a student who studies for 40 hours and has a GPA of 3.5 gets an A in the class.

(b) How many hours would the student in part (a) need to study to have a 50 % chance of getting an A in the class?