50.570 Machine Learning, Fall 2017

Assignment 1

Last update: Sunday 24th September, 2017 22:23

Grading Policy and Due Date

- You are required to submit: 1) a report that summarizes your experimental results and findings, based on each of the following question asked; 2) your implementation (source code) of the algorithms.
- You are free to choose any programming language you prefer.
- Submit your assignment report and code to eDimension.
- This assignment is an individual assignment. Discussions amongst yourselves are allowed and encouraged, but you should write your own code and report.
- Submit your assignment to eDimension by 7 October 2017 11.59pm. This is a hard deadline. Late submissions will be heavily penalized (20% deduction per day).

Task: Document Classification

In classes we have discussed the issue of classification as a class of *supervised learning* algorithms. One important application of classification is to identify the underlying topics associated with each document by doing *document classification*.

Download `data.zip` and unzip it. In the `data` folder, there are two main directories: `train` and `test`. They contain data used for training and testing respectively. Each directory consists of four sub-folders, where each sub-folder contains several documents related to a particular topic (the four topics are: `acq`, `coffee`, `fuel`, `housing`). The goal of document classification is to learn a model (classifier) based on the labeled documents in the `train` folder only, such that the model can be used to predict the correct label (i.e., topic) associated with each new document appeared in the `test` folder.

1. (5 pt) Discuss how to formulate this task as a classification problem – describe what are the inputs (x) and what are the outputs (y). Discuss how you would convert the input data into vector representations using the approach discussed in class, where only word unigrams are considered for constructing the input vectors.
2. (10 pt) Consider only the documents that appeared in these two sub-folders: `acq` and `housing` (i.e., ignore documents from `fuel` and `coffee` for now). Implement the perceptron algorithm to perform binary classification based on the data from these two sub-folders. Evaluate the model's performance on the training and test set respectively. Discuss clearly in your report when to stop the algorithm and whether this has any effect on the performance on the test set. Try to use the

averaged perceptron algorithm discussed in class (*Note: in averaged perceptron, instead of using the final θ and θ_0 parameters, we keep track of all intermediate parameters after each update, and then take the average. Such averaged parameters are then used during the testing phase.*), and report the performance when such an algorithm is used. Compare its performance with that of the standard version of the perceptron.

3. (10 pt) Implement the stochastic gradient descent algorithm that minimizes the loss (or *the empirical risk*) involving the hinge loss to tackle the same binary classification problem. Evaluate your model's performance using the test data. Discuss its learning behavior: the effect of using different learning rates, how fast the algorithm converges. Discuss when to stop the algorithm and whether this has any effect on the performance on the test set. Also compare its performance with that of the perceptron.

4. (10 pt) Now, consider the multi-class classification problem (i.e., consider all documents from all 4 topics). Think of a way to perform such a multi-class classification task using what you have learned so far. Describe and implement your algorithm, and report its performance on training and test set. (*Hint: How to cast a multi-class classification problem into binary classification problems?*)

5. (10 pt) The objective function involving the hinge loss is defined as follows:

$$\mathcal{R}_n(\bar{\theta}, \theta_0) = \frac{1}{n} \sum_{t=1}^{n} \max\{1 - y^{(t)}(\bar{\theta} \cdot \bar{x}^{(t)} + \theta_0), 0\}$$

   Now let us introduce a so-called *regularization term* to the above objective function, leading to the following new objective function:

$$\mathcal{R}'_n(\bar{\theta}, \theta_0) = \lambda ||\bar{\theta}||^2 + \frac{1}{n} \sum_{t=1}^{n} \max\{1 - y^{(t)}(\bar{\theta} \cdot \bar{x}^{(t)} + \theta_0), 0\}$$

   where λ is a constant.

   Present your new learning algorithm to optimize this new objective function. (*Note: if you would like to use stochastic sub-gradient descent, how do you calculate the per-instance gradient correctly with the regularization term? We briefly discussed this in class.*) Provide necessary derivations for any update equations used in your algorithm. Train and evaluate your new model based on the documents from the two sub-folders `acq` and `housing`. Try to set λ to different values (such as 0.001, 0.01, 0.1, 1, 10 etc) and report the learned model's performance on both the training set and the test set. Try to explain why the performances on the training and test set have such behaviors as we change the value of λ.

6. (10 pt) The approach described in class for representing documents as vectors is the so-called "bag-of-words (BOW)" approach. It is only one of the many ways to represent input documents as vectors. Think of some other ways to represent the input documents as vectors, and discuss the effect of using such alternative representations when the perceptron algorithm is used. Try to explain why the performance on the test set becomes better or worse with your alternative representations.

7. (bonus 5 pt)* This is a very open question. Try to think of something else interesting to explore based on what you have learned and the data provided. For example, in class we discussed that we can use gradient descent with hinge loss to optimize the empirical risk. You can think of a different loss function to replace the hinge loss and see what performance you can obtain.