Lab Course Machine Learning
Exercise Sheet 8

Prof. Dr. Dr. Lars Schmidt-Thieme, Mohsan Jameel
Information Systems and Machine Learning Lab
University of Hildesheim

December 14th, 2016
Submission on January 11th, 2016 at 11:55pm, (on moodle, course code 3112)

Instructions

Datasets

1. **Classification Datasets:** You can use one of the two datasets (or optionally, both datasets).
   (a) Iris dataset D_1: Target attribute **class**:$\{$Iris Setosa, Iris Versicolour, Iris Virginica$\}$. `https://archive.ics.uci.edu/ml/datasets/Iris`
   (b) Wine Quality dataset D_2: Target attribute **quality**:$\{0$ to $10\}$. `https://archive.ics.uci.edu/ml/datasets/Wine+Quality`

Note: Dataset D_2 can also be used for a regression problem.

Exercise 1: Implement K-Nearest Neighbor (KNN) (10 Points)

Your task is to implement KNN algorithm. To implement KNN you have to

- Split data into a train and a test split (70% and 30% respectively).
- Implement a similarity (or a distance) measure. To begin with you can implement the Euclidean Distance.
- Implement a function that returns top K Nearest Neighbors for a given query (data point).
- You should provide the prediction for a given query (for a classification task you can use majority voting and for a regression you can use mean).
- Measure the quality of your prediction. [Hint: You have to choose a quality criterion according to the task you are solving i.e. a regression or a classification task].

Exercise 2: Optimize and Compare KNN algorithm. (10 Points)

Part A: (5 Points): Determine Optimal Value of K in KNN algorithm. In this exercise you have to provide the optimal value of K for given datasets.

- How you can choose value of K for KNN. Give a criterion to choose an optimal value of K.
- Implement the criterion for choosing the optimal value of K.
- Experimentally, give evidence that your chosen value is better than other values of K. [Hint: run your experiment with different values of K and plot the error measure for each value].

Part A: (5 Points): Compare KNN algorithm with Tree based method. In this task you are allowed to use scikit learn. In particular you have to use Nearest Neighbor and Decision Tree implementation provided by scikit learn.

- You should be able to use Nearest Neighbor and Decision Tree provided by scikit learn to solve classification task for two datasets.
- You have to provide the optimal hyperparameters for both the methods. [Hint: use Grid Search and cross validation and present results for them to support your solution].
- Present the comparison of the two methods using evaluation results on test datasets. [Hint: Better to use cross validation to ascertain your results]

Bonus: Recommender system using similarity measures (10 Points)

Recommender Datasets: You can use one of the two datasets (or optionally, both datasets).

1. movielens 100k dataset D_1: Rating prediction dataset (rating scale 1-5). `http://grouplens.org/datasets/movielens/100k/`
2. movielens 1m D_2: Rating prediction dataset (rating scale 1-5).`http://grouplens.org/datasets/movielens/1m/`
3. The RMSE score for rating prediction is available at **Mymedialite website** `http://www.mymedialite.net/examples/datasets.html`

In this task you are required to build a recommender system based on KNN. You will be required to

- As usual, split your data into train and test sets.
- *User KNN cosine:* Using k nearest neighbor users for a given query (user and item pair) and predict the rating. Note: that you have to modify your KNN algorithm implementation in Exercise 1 for User based KNN using cosine similarity. Calculate Test RMSE and compare it with results presented at Mymedialite.
- *Item KNN cosine:* Using k nearest neighbor items for a given query (user and item pair) and predict the rating. Note: that you have to modify your KNN algorithm implementation in Exercise 1 for Item based KNN using cosine similarity. Calculate Test RMSE and compare it with results presented at Mymedialite.
- In the above two tasks you might also want to perform a hyperparameter search to get an optimal value of K.
- Finally, present your results in a tabular form i.e. listing methods, hyper-parameters and Test RMSE scores.
- *Hints:* If you have a less powerful machine you can use movielens 100k dataset (or sub sample of it). Read papers in Annex section to learn more about User-KNN and Item-KNN for recommender system. You have to implement this yourself and you cannot use scikit-learn or anyother off-the-self softwares/implementations.

Annex

1. Following lecture is relevant this exercise `https://www.ismll.uni-hildesheim.de/lehre/ml-16w/script/ml-06-A5-nearest-neighbor.pdf`
2. Recommender reference 1: `http://files.grouplens.org/papers/www10_sarwar.pdf`

3. Recommender reference 2: `http://siplab.tudelft.nl/sites/default/files/sigir06_similarityfusion.pdf`