CS 6350- ASSIGNMENT 5

Please read the instructions below before starting the assignment.

- There are 3 parts in this assignment. Please use a separate folder for each.
- You have to submit link to the dataset used for each question, your code and the analysis.
- You should use a cover sheet, which can be downloaded at: http://www.utdallas.edu/~axn112530/cs6350/CS6350_CoverPage.docx
- You are allowed to work in pairs i.e. a group of two students is allowed. Please write the names of the group members on the cover page. Only one submission per team is required.
- You have a total of 4 free late days for the entire semester. You can use at most 2 days for any one assignment. After that, there will be a penalty of 10% for each late day. The submission for this assignment will be closed 2 days after the due date.
- Please ask all questions on Piazza, and not through email to the instructor or TA.

PART I - UNDERSTANDING RECOMMENDER SYSTEMS

(5 points)

Read the paper "Matrix Factorization Techniques for Recommender Systems" by Koren et al available from any of the sources below
http://ieeexplore.ieee.org/document/5197422/

http://dl.acm.org/citation.cfm?id=1608614

https://datajobs.com/data-science-repo/Recommender-Systems-%5BNetflix%5D.pdf

Answer the following questions:

1. Recommender Systems (RS) are based on one of two strategies – content filtering and collaborative filtering. Write down key features and properties of each of them.

2. Two primary areas of collaborative filtering are neighborhood methods and latent factor models. Write down key features of each of them.

3. What do the "factors" in latent factor represent? How are these factors discovered?

4. What is the difference between explicit and implicit feedback for RS? What do you think are advantages and disadvantages of each?

5. In the latent factor model for dimension f, the item i and user u are represented as vectors q_i and $p_u \in R^f$. How is the interaction between item i and user u calculated?

6. What does the learning system try to minimize? Understand the meaning of each term in the equation.

7. There are 2 learning algorithms for latent factorization – stochastic gradient descent (SGD) and alternating least squares (ALS). What are the advantages of ALS over SGD?

8. Read about the Netflix competition and the authors' entry. What were the most descriptive dimensions (features) that their models discovered? Summarize briefly. Also mention what metric do they use to check the performance of their models.

PART II-
IMPLEMENTING RECOMMENDER SYSTEMS

(20 points)

In this part of the assignment, you will implement Recommender Systems (RS) using Apache Spark MLlib. For the first section of this assignment, you will finish a notebook of Databricks and for the second section you will implement RS on another dataset.

Section I:

Open the community edition of Databricks and complete all of the steps in this notebook: https://docs.databricks.com/_static/notebooks/cs100x-2015-introduction-to-big-data/module-5--machine-learning-lab.html

Remember to click on the top right corner of the page and import the notebook into your cluster environment. The dataset for this part is the famous movielens dataset. **The steps will work best if you use a Spark 2.0 (NOT Spark 2.1) cluster**. The dataset for this part is the famous movielens dataset. After you have completed all the steps, be sure to publish your notebook and then submit the public URL of your notebook.

Section II:

After you have finished section I and understood the basics of RS using Spark MLlib on movielens dataset, you will repeat the steps on another dataset. You are free to use the method performed in Section I or using the approach followed below:
http://spark.apache.org/docs/latest/mllib-collaborative-filtering.html

Here are the steps:

1. Choose another dataset from this URL: https://gist.github.com/entaroadun/1653794

Of course, you cannot choose the movielens dataset again.

2. Examine the dataset, especially the format and field separators.

3. Output the top 10 items e.g. movies, songs, books with the highest ratings.

4. Split the data in 80:20 ratio for training and testing.

5. Train an ALS model on the training part of the dataset and apply the model on the test part. Report RMSE values for different choices of parameters.

6. Submit the URL for your notebook in case of Databricks and your code in case of other clusters. Also submit the URL and name of the dataset you used.

PART III-
CLASSIFICATION AND CLUSTERING

(25 points)

Section I: Classification

For the classification part, you have to choose a dataset from the UCI Machine Learning repository, which is one of the most popular repositories and is available at:
https://archive.ics.uci.edu/ml/datasets.html

For selecting appropriate datasets, choose "classification" from the "Default task" options at the top left of the page. You are free to choose any dataset that you want.

Below are the requirements:
1. You need to use **pipelines** for pre-processing (e.g. converting categorical attributes to numerical, handling NA or missing values) and model creation.

2. You need to use a **ParamGridBuilder** object to create a parameter grid for the various parameters and their values. Be sure to use at least 2 parameters and at least 3 values for each parameter.

3. You need to use a **cross-validator** object for choosing the best parameters and setting the number of folds. You should use at least a 5-fold cross validation.
You can see an example of the above being used for a toy dataset here:
http://spark.apache.org/docs/latest/ml-tuning.html#cross-validation

4. You will need to run **any two** of the classification algorithms available on Spark on the dataset:
http://spark.apache.org/docs/latest/ml-classification-regression.html#classification

5. Output classification metrics of your model. An example of how to do this can be viewed here:
http://spark.apache.org/docs/latest/mllib-evaluation-metrics.html

6. You have to run each classification algorithm twice as follows:
a. In the first attempt, use all of the attributes (features) while training and testing the model.

b. In the second attempt, perform **dimensionality reduction** and reduce the dimensions to some chosen value K. Then, perform the training and testing on these K dimensions and report your results. The details of dimensionality reduction are available here: https://spark.apache.org/docs/latest/mllib-dimensionality-reduction.html

The results should be reported in a tabular format like the one shown below:

Dataset name and URL: _____

Number of instances	Total number of attributes	Number of attributes used (in case of dimensionality reduction)	Classification Method e.g. Logistic Regression	Parameters	Cross-Validation Times	Evaluation Metric1	Evaluation Metric2 …

Section II: Clustering

In this part, you will perform clustering task. You will select another dataset by going to the UCI Machine Learning repository and filtering the default task to clustering. Select a dataset and perform k-means clustering on it.

Below are the requirements:

1. You have to choose at least 5 different values of K and evaluate your results based on the WSSE metric.

Report the output of your code in tabular format. Be sure to mention the URL of your dataset.