**Take-home Final Exam**

Ishwar Bhat Kattur

**IFT598: Analyzing Big Data**

Prof. Robert Rucker

05/04/2017

**Question 1 Part a**

*Before starting this regression question, it would help to be clear about the components and their meaning and uses: Using the components of Q1 to illustrate your answers:*

*0. entities ( these may be implicit as in the array below), observations, features*

*1. Independent and dependent variables*

*2. Categorical versus numerical features*

*3. Labels*

*4. Labeled versus unlabeled observations*

*5. Describe the connection between a regression algorithm and a regression model, and illustrate this with the classes used in the analysis below..*

**Entities** in Machine Learning are the crux of the problem for which we are trying to find a solution or build a model for. For example, in the Spam-Ham classification problem, Emails are the entities. Also, each entity is associated with real-valued label or a binary label (depending on whether we are working on a linear regression or logistic regression problem).

The number of training and test data we have make up the **observations**. For the given problem in 1b, fileMat has 5 **observations**. Usually, the more observations we have (assuming they are accurate), the better we can train the data.

Wikipedia says that a **feature** is an individual measurable property of the phenomenon being observed. In simple terms, features are inputs and **label** is the output. We use features to train our model to predict the **labels**.

**Categorical features** have categories (say “Gender”, “Sexual Orientation” or “Marital Status”) as features. **Numerical features** have numbers (like “age” which has to be a positive number) as feature values.

**Labels**, as mentioned above, are the outputs. For email classification, “Spam” and “Ham” are the two possible labels. **Labeled observations** have their observations labeled. These are used in supervised learning, where the model is trained with both inputs and the actual output. **Unlabeled observations** on the other hand are used in unsupervised learning, like clustering. We (or the ML model) have to figure out how to group the data, so that when a new input arrives, we can classify it to the correct group.

A **regression algorithm** is an algorithm for predicting relationship among variables. A **regression model** is an actual implementation of the algorithm, which when trained with the training dataset, can predict the output given a test dataset.

In this question down below, LinearRegressionWithSGD() is the algorithm (the set of rules which lay down the steps for building the model). model is the regression model (which we train by regression.run(parsedData)). This **model** is later used to predict output for custom input (val prediction = model.predict(point.features))

**Question 1 Part b**

We have the data as below:



First step is cleaning the data.



Next step: Split and get the X and Y vectors



Slope and intercept are calculated using First Principles:



**Question 1 Part c**

Constructing the RDD:



Scaled parsedData:



weights and intercepts using LinearRegressionWithSGD():



Testing with custom as well as given data:



**Question 2: DataFrames and Datasets**

The bodies.csv input file:



The case class required:



Defining the StructType:



DataFrame:



Using UDF to create dataset:



**Part b:** Surface area column added:



**Question 3: Sentiment Analysis**

Guller pg. 199



Using the SparkContext and SparkConf to create RDDs:



Parsing data:



DataFrame:



Making sure data is populated:



Tokenizer:



HashingTF:



LogisticRegression Pipeline:



Training and Test:



aucTraining and aucTest:



ParamGridBuilder:



Cross Validator:





Best Model, which can be used to classify new reviews:



**Question 4: Summing Up**

I was a bit apprehensive in the beginning about this course (because in my experience, most senior Professors like to go the old school way, and I hate to say this, but turn what could be potentially interesting topics into boring courses), but not for long. This class turned out to be a pleasant surprise. Working on the latest technologies like Spark and Scala instead of going the tried and tested way of Hadoop/Java was great fun. I personally would have stuck with Python and never would have went for Scala if not for this course. I now understand how powerful a functional programming language Scala can be, of which I’m a big fan now (But I still hate the Scala IDE, which sometimes stops showing output for no apparent reason. Since REPL works just on Save and there’s no option to run the code like other languages, most of these times I’m left wondering why there’s no output when all the syntax of my Scala worksheet is fine). In a broader sense, Thank You for expanding our horizons by forcing us learn this awesome new (for me, at least) language and broadening our narrow mindset of sticking with what we know best! I also believe in what you said in the last class, that Spark is going to be the next big thing in practical Data Science. Your insights from your vast professional experience before academi

Continuing on from the “Managing the Cloud” course where, as a part of the project work we learnt how to set up a Spark cluster on cloud and perform some basic Spark operations on RDDs using Python. I for one, sure did not know the nuances of RDDs, DataFrames and Datasets and their evolution in the Spark ecosystem, of which I am confident now.

While I appreciate the intent of familiarizing the students with the whole Spark ecosystem, I felt a little bit rushed in the second half of the semester, what with going through modules like MLlib, GraphX and Streaming week after week, without doing much meaningful work. I felt that it would be better if one module like MLlib was taken and students were made to work on practically. Even better: something like a mini project maybe in groups of two or three (maybe even individually) with regular weekly or bi-weekly deadlines for modules to keep the students engaged would be pretty cool, and students could show off their work at the end of their semesters and even on their resumes.

A side note on student engagement, you seemed pretty frustrated when students didn’t interact much in class. I feel that it is just a cultural shock for most of us Indian students, who aren’t much used to openly disagreeing with the Professor’s ideas or even asking questions in class. But have no doubt that we are soaking up all the information you give out. I know that we also need to work on our communication skills and speak up more in class. I’m working on it.

Finally, I would like to say that this was one of the best classes I’ve taken here at ASU. Thank you for making this available! And it was an honor to have you as our professor, and would love to work for/with you in future, if I got a chance. So long!