Big Data Analytics

Exercise Sheet 10

Prof. Dr.Dr. Lars Schmidt-Thieme, Mohsan Jameel

Information Systems and Machine Learning Lab University of Hildesheim

June 30th, 2016

Submission until July 6th, 2016, to mohsan.jameel@ismll.de

Setup and getting ready:

1) Reading material

```
http://spark.apache.org/docs/latest/programming-guide.html
http://training.databricks.com/workshop/itas_workshop.pdf
extra reading: https://stanford.edu/~rezab/sparkclass/
```

Exercise: Spark Essentials (5 marks)

a) Spark has Resilient Distributed Datasets (RDD) which works on lazy transformation mechanism. Explain what this lazy transformation is and what its benefits are.

b) Spark performs in-memory computation and manipulation thanks to RDD. But when the problem requires to access computation again and again between different jobs, RDD get recomputed each time. How can this problem be avoided?

c) Suppose we have distributed data across multiple nodes and each file contains historical information about monthly salaries payed by an employer. Your task is to find the total sum of salaries payed by the employer in the record.

Here is the code snippet, explain why it will work or not work. Present your solution to remedy (if any) problem in the given code. (Scala code)

```
var counter = 0
var rdd = sc.textFile("data.txt")
rdd.foreach(x => counter += x)
println("Counter value: " + counter)
```

Exercise: count a specific word with Spark (5 marks)

Write a code (either in Java or Scale) to calculate occurrence of Spark in "README.md" and "CHANGES.md" files in spark home folder. It will be a modified version of the word count example. Some requirements for solving this task are as follows:

1) You will not use generic word count. In this task you only have to develop logic so that your reduce phase only count occurrences of keyword “Spark”.
2) You have to combine the results from “README.md” and “CHANGES.md” RDDs

Exercise: KMeans clustering with Spark (5 marks)

A KMeans algorithm along with its parallel version was explained in tutorial 2 (tutorial lecture 2). In this task you are required to implement KMeans clustering with Spark. You can choose any language of choice i.e. Java, Scala or Python. Please provide the code and main output of your program i.e. listing final centroids.