CS 696 Intro to Big Data
Fall Semester, 2016
Assignment 5

San Diego State University -- This page last updated 12/14/16

Spark Assignment

Due Dec 20 23:59

1. Russian voting data. We will use the Russian voting data from assignment 3. Using Spark compute the results described in a and b. In each case the output needs to be placed in an output file written by either the map or reduce function. The input and output directories need to be command line arguments to the command to run your program as in the WordCount sample programs. That is the input directory needs to be the first argument and the output directory needs to be the second argument. The result of part a should go into a subdirectory of the output directory called "a" and the output of part b goes in a subdirectory of the input directory called "b". The program needs to be able to run on AWS EMR. This means that each time the program is run multiple copies of both map and reduce may be used.

   a. Compute the number of voters in each voting district in Russia. There are 99 districts numbered from 1 to 99.

   b. Compute the mean number of voters in voting districts in Russia. You will have one number here not 99.

2. Spark Word Count. On the dataset page of course website you will find a link to Mark Twain's collected works which will be the input for our word count program. As with problem one your programs should have two command line arguments, the first is the input directory and the second is the output directory. In processing the words you should normalize the words by:

   - Removing the endings "'" (single quote), "--", "-", "'s", "ly", "ed", "ing", "ness", ")", "_", ";", "?", "!", ",", ".".
   - Convert all words to lower case
   - Remove leading "'" (single quote), "" (double quote), "(" and "_".

   a. Produce the standard word count but the output needs to be sorted by the number of times the word occurs in decreasing order. The final output should be in a single file even if we use multiple reducers.

   b. The second word program counts the occurrence of unordered word pairs. Each time two words occur next to each other we count that as a pair. For example in the sentence a cat a rat a bat" we have "a cat" occurs twice, "a rat" occurs twice, "a bat" occurs once as we do not consider the order of the words. The last word in a line is considered as occurring before the first word in the next line. A few word pairs will not be counted if the

file processed by multiple map instances. As in part a) we want the output sorted by the count in decreasing order.

3. Census data. On the dataset page of the course website you will find a link for 2012 US Tax Data. This is a csv file. This dataset contains information about income taxes filed in 2012. The data is categorized by state, zipcode and income level. There are six different income levels numbered 1 through 6. So each row in the file contains information about income taxes filed in a given state, in a given zipcode area and in one of 6 income levels. Here is a sample.

STATEFIPS	STATE	zipcode	AGI_STUB	N1	MARS1	MARS2	MARS4
1	AL	35004	1	1600.0000	990.0000	270.0000	300.0000
1	AL	35004	2	1310.0000	570.0000	400.0000	290.0000
1	AL	35004	3	900.0000	280.0000	500.0000	100.0000
1	AL	35004	4	590.0000	70.0000	490.0000	30.0000
1	AL	35004	5	480.0000	30.0000	440.0000	0.0000
1	AL	35004	6	50.0000	0.0000	50.0000	0.0000

The second column STATE give the abbreviation for the state. The third column indicates the zipcode. The fourth column, AGI_STUB, gives the adjusted gross income given in numbers 1-6 with the meaning listed below.

1 - $1 to $25,000
2 - $25,000 to $50,000
3 - $50,000 to $75,000
4 - $75,000 tor $100,000
5 - $100,000 to $200,000
6 - $200,000 or more

The fourth column, N1, givens the number of returns in the income level from the indicated zipcode. So there were 1600 tax returns from zip code 35004 with income between $1 and $25,000.

a. Write a Spark program to find the total number of tax returns filed in each state in each category. Again the program needs two command line argument, the first the input directory and the second the output directory. The input will be a file as described above. The output will be a file(s) sorted by state. Each state will have the six levels of income with the total of tax returns in each category.

AL	1	889920
AL	2	491150
AL	3	491150

AL	4	160160
AL	5	160160
AL	6	44840

What to turn in

Turn in the source code for each of the problems above. Include a jar file that contains running code for each program. The jar file should not include the Spark jar fiiles. Include a readme file which gives the command line argument to run each program. Submit one zip file containing the source code, jar files and readme file.

Late Penalty

An assignment turned in 1-7 days late, will lose 5% of the total value of the assignment per day late. The eight day late the penalty will be 40% of the assignment, the ninth day late the penalty will be 60%, after the ninth day late the penalty will be 90%. Once a solution to an assignment has been posted or discussed in class, the assignment will no longer be accepted. Late penalties are always rounded up to the next integer value.