CSC 582/782: Big Data

Assignment 2

Simple Moving Average

(50 points)

Due: Sunday, November 13

Introduction

The goal of this assignment is to implement a simple moving average that is an arithmetic mean of the values in that time period on Single host and on MapReduce. The simple moving average (SMA) is the well-known technical indicator used by stock traders. It is a measurement to decide potential buy and sell signals. The following table is a time series data for stock closing price.

Time series	Closing price
2015-08-03	11.30
2015-08-04	12.40
2015-08-05	19.00
2015-08-06	22.20
2015-08-07	23.50
2015-08-10	28.00
2015-08-11	27.00

For example, to compute simple moving average over three days, we sum the value of the time series in that time period, and divide by three. When calculating successive values, add a new value into the sum and an old value drops out. e.g.,

SMA#1 = (11.30+12.40+19.00)/3 = 14.23
SMA#2 = (12.40+19.00+22.20)/3 = 17.86
SMA#3 = (19.00+22.20+23.50)/3 = 21.56
… etc. …

The formal definition for an n-day moving average (https://en.wikipedia.org/wiki/Moving_average) is: given that stock closing prices are $p_M, p_{M-1}, \dots, p_{M-(n-1)}$,

$$\text{SMA} = \frac{p_M + p_{M-1} + \cdots + p_{M-(n-1)}}{n} = \frac{1}{n}\sum_{i=0}^{n-1} P_{M-i}$$

Write a Python MapReduce program that calculate a simple moving average with a moving average window size 3. The input to the MapReduce program is formatted such that [company-symbol][,][date][,][closing-price] e.g.,

FB,2015-03-02,89.40
FB,2015-03-03,90.30
FB,2015-03-04,91.10
YHOO,2015-03-02,24.40
YHOO,2015-03-03,25.11
YHOO,2015-03-04,26.33

The output from MapReduce program will have the following format: [company-symbol][,][date][,][simple-moving-average]. There is a sample input file you can use: sample.txt.

Hint: A possible solution is to design that i) mapper function will emit <key, value> as <company-symbol, {date, closing-price}> for the reducer and ii) the reducer function will aggregate values based on a window size and emit simple moving averages per key. How do you think you could have designed your MapReduce algorithm differently better than this solution?

Note: This is the example of the code snippet for Python. The code snippets are **NOT required** to write your program. If you can write better codes or optimize, it'll be OK. But, you can consider it as starting point so that it'll help write your SMA program.

Python

In SMA_Mapper.py

```
for line in sys.stdin:
   //parse each value from the line
   value = line.strip()
   //yield the pair (key, values) to stream
```

In SMA_Reducer.py

```
stock_record = list()
window_size = 3

for line in sys.stdin:
   (key, val, val2) = line.strip().split('\t')
   //parse an element and store to the list stock_record
   if len(stock_records) == window_size:
   //
   //calculate SMA
   //
   print '%s\t%s\t%.2f' % (key, va1, sma); //print out the result to stream
```

Programming Requirements

You are required to use Python for this assignment. Also, your program should compile and run on your virtual hadoop cluster from the first assignment.

Submission Requirements

The assignment must be done **individually**. Please submit a single zip (or tar) file that include all files. The submission should include:

- **Source codes**: your Python codes for MapReduce and shell script files (if applicable) that you use for testing. All programs (source code) must contain comments. Failure in using proper comments will result in reduced points assigned to the programming assignments.
- **Report:** A short summary on instruction to run your program and the provide a screenshot as testing results by your program. Note that screenshots must be readable for full credit.
- **Dataset**: a dataset for your results

Late Policy

Please check the late policy on the course syllabus.