**1. Analyzing structured data using Pig 50%**

Apache Pig makes it possible to write complex queries over big datasets using a language named Pig Latin. The queries are transparently compiled into MapReduce jobs and run with Hadoop.

While doing this assignment, you are invited to check the official documentation http://pig.apache.org/docs/r0.12.0/).

Connect to your Virtual Machine and copy the files to a local directory on your machine.

Note: Depending on your machine resources, you may choose to run pig in **local** mode or in **mapreduce** mode.

**Question 1:**

Write a Pig script to load the files into two relations named **data** (all rides) and **stations** (stations), and remove the header lines of the csv files from the Pig relations.

dataRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Rides' USING PigStorage(',')

AS (start\_date:chararray, start\_station\_code:chararray, end\_date:chararray, end\_station\_code:chararray, duration\_sec:int, is\_member:chararray);

data = FILTER dataRaw BY NOT (start\_date == 'start\_date');

stationsRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Stations\_2017.csv' USING PigStorage(',')

AS (code:chararray, name:chararray, latitude:double, longitude:double);

stations = FILTER stationsRaw BY NOT (code == 'code');

DUMP data;

DUMP stations;

**Question 2:**

Write a Pig script to compute and print on the screen the number of rows for each relations (data, stations).

dataRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Rides' USING PigStorage(',')

AS (start\_date:chararray, start\_station\_code:chararray, end\_date:chararray, end\_station\_code:chararray, duration\_sec:int, is\_member:chararray);

data = FILTER dataRaw BY NOT (start\_date == 'start\_date');

stationsRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Stations\_2017.csv' USING PigStorage(',')

AS (code:chararray, name:chararray, latitude:double, longitude:double);

stations = FILTER stationsRaw BY NOT (code == 'code');

data\_group = GROUP data ALL;

data\_count = FOREACH data\_group GENERATE COUNT(data.start\_date);

stations\_group = GROUP stations ALL;

stations\_count = FOREACH stations\_group GENERATE COUNT(stations.code);

DUMP data\_count;

DUMP stations\_count;

**Question 3:**

Write a Pig script to split the data relation into two relations **members** and **notmembers**. The members relation should contain only peoples that are members and the notmembers relation should contains all non-member people.

dataRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Rides' USING PigStorage(',')

AS (start\_date:chararray, start\_station\_code:chararray, end\_date:chararray, end\_station\_code:chararray, duration\_sec:int, is\_member:chararray);

data = FILTER dataRaw BY NOT (start\_date == 'start\_date');

members = FILTER data BY (is\_member == '1');

notmembers = FILTER data BY (is\_member == '0');

DUMP members;

DUMP notmembers;

**Question 4:**

Write a Pig script to compute and print on the screen the number of rows for each relation (members, notmembers).

dataRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Rides' USING PigStorage(',')

AS (start\_date:chararray, start\_station\_code:chararray, end\_date:chararray, end\_station\_code:chararray, duration\_sec:int, is\_member:chararray);

data = FILTER dataRaw BY NOT (start\_date == 'start\_date');

members = FILTER data BY (is\_member == '1');

notmembers = FILTER data BY (is\_member == '0');

members\_group = GROUP members ALL;

members\_count = FOREACH members\_group GENERATE COUNT(members.start\_date);

notmembers\_group = GROUP notmembers ALL;

notmembers\_count = FOREACH notmembers\_group GENERATE COUNT(notmembers.start\_date);

DUMP members\_count;

DUMP notmembers\_count;

members\_group = GROUP members ALL;

members\_count = FOREACH members\_group GENERATE COUNT(members.start\_date);

notmembers\_group = GROUP notmembers ALL;

notmembers\_count = FOREACH notmembers\_group GENERATE COUNT(notmembers.start\_date);

DUMP members\_count;

DUMP notmembers\_count;

**Question 5:**

Write a Pig script that given some station’s code (hard-coded constant) will return the stations record if found.

stationsRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Stations\_2017.csv' USING PigStorage(',')

AS (code:chararray, name:chararray, latitude:double, longitude:double);

stations = FILTER stationsRaw BY NOT (code == 'code');

station\_detail = FILTER stations BY (code == '6706');

DUMP station\_detail;

**Question 6:**

Write a Pig script that will calculate the number of rides departures per station. The output does not have to be sorted.

dataRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Rides' USING PigStorage(',')

AS (start\_date:chararray, start\_station\_code:chararray, end\_date:chararray, end\_station\_code:chararray, duration\_sec:int, is\_member:chararray);

data = FILTER dataRaw BY NOT (start\_date == 'start\_date');

station\_cluster = GROUP data by start\_station\_code;

station\_count = FOREACH station\_cluster GENERATE FLATTEN(group) as start\_station\_code, COUNT($1);

DUMP station\_count;

**Question 7:**

Write a Pig script that will calculate the count of rides per station (start station) and the MIN, MAX, AVG of the ride’s duration for members and non-members.

dataRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Rides' USING PigStorage(',')

AS (start\_date:chararray, start\_station\_code:chararray, end\_date:chararray, end\_station\_code:chararray, duration\_sec:int, is\_member:chararray);

data = FILTER dataRaw BY NOT (start\_date == 'start\_date');

members = FILTER data BY (is\_member == '1');

notmembers = FILTER data BY (is\_member == '0');

member\_station\_cluster = GROUP members by start\_station\_code;

member\_station\_desc = FOREACH member\_station\_cluster GENERATE group, COUNT(members.start\_station\_code), MIN(members.duration\_sec), MAX(members.duration\_sec), AVG(members.duration\_sec);

notmember\_station\_cluster = GROUP notmembers by start\_station\_code;

notmember\_station\_desc = FOREACH notmember\_station\_cluster GENERATE group, COUNT(notmembers.start\_station\_code), MIN(notmembers.duration\_sec), MAX(notmembers.duration\_sec), AVG(notmembers.duration\_sec);

DUMP member\_station\_desc;

DUMP notmember\_station\_desc;

**Question 8:**

Write a Pig script that will list the Top 5 names of the start station for members and for non-member

stationsRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Stations\_2017.csv' USING PigStorage(',')

AS (code:chararray, name:chararray, latitude:double, longitude:double);

stations = FILTER stationsRaw BY NOT (code == 'code');

dataRaw = LOAD '/home/cloudera/Desktop/a1/Bixi Data/Rides' USING PigStorage(',')

AS (start\_date:chararray, start\_station\_code:chararray, end\_date:chararray, end\_station\_code:chararray, duration\_sec:int, is\_member:chararray);

data = FILTER dataRaw BY NOT (start\_date == 'start\_date');

data\_join = JOIN data BY start\_station\_code, stations BY code;

members = FILTER data\_join BY (is\_member == '1');

notmembers = FILTER data\_join BY (is\_member == '0');

members\_cluster = GROUP members by name;

members\_count = FOREACH members\_cluster GENERATE group, COUNT(members.start\_station\_code) as cnt;

notmembers\_cluster = GROUP notmembers by name;

notmembers\_count = FOREACH notmembers\_cluster GENERATE group, COUNT(notmembers.start\_station\_code) as cnt;

members\_count\_ordered = ORDER members\_count BY cnt DESC;

notmembers\_count\_ordered = ORDER notmembers\_count BY cnt DESC;

lim\_members\_count = LIMIT members\_count\_ordered 5;

lim\_notmembers\_count = LIMIT notmembers\_count\_ordered 5;

DUMP lim\_members\_count;

DUMP lim\_notmembers\_count;

**2. Analyzing unstructured data using Pig 50%**

You are given an unstructured text file to import it into a structured data analyzing tool such as Hive or Impala.

Your task is:

• To remove all the comments lines (lines starting with #)

• To remove the header line

• To output all the columns of the file to disk.

NOTE: I am answering this question using Apache Pig since we have not covered Hive or Impala yet

A = LOAD '/home/cloudera/Desktop/a1/Protocol Data/protocols.txt' USING PigStorage('\n') AS line:chararray;

B = FOREACH A GENERATE REGEX\_EXTRACT($0,'^(?m)([^#]+?)(?:\t+?#)',1) as (urlString:chararray);

C = FILTER B BY NOT ($0 == '');

D = FILTER C BY NOT ($0 == 'ip\t0\tIP');

E = FOREACH D GENERATE STRSPLIT($0, '\t', 3);

STORE E INTO '/home/cloudera/Desktop/a1/P2Q1\_output' using PigStorage(',');