Aviation Data Analysis using Apache Pig

Problem Statement 1

Find out the top 5 most visited destinations.

Steps to find the top 5 most visited destinations.

- **In Line 1**: We are registering the *piggybank* jar in order to use the CSVExcelStorage class.
- **In Line 2:** we are defining org.apache.pig.piggybank.storage.CSVExcelStorage as CSVExcelStorage
- In relation **A**, we are loading the dataset using CSVExcelStorage because of its effective technique to handle double quotes and headers.
- In relation **B**, we are generating the columns that are required for processing and explicitly typecasting each of them.
- In relation **C**, we are filtering the null values from the "dest" column.
- In relation **D**, we are grouping relation C by "dest."
- In relation **E**, we are generating the grouped column and the count of each.
- Relation **F** and **Result** is used to order and limit the result to top 5.
- These are the steps to find the top 5 most visited destinations. However, adding few more steps in this process, we will be using another table to find the city name and country as well.
- In relation **A1**, we are loading another table to which we will look-up and find the city as well as the country.
- In relation **A2**, we are generating dest, city, and country from the previous relation.
- In relation **joined_table**, we are joining Result and A2 based on a common column, i.e., "dest"
- Finally, using dump, we are printing the result.

Pig Scripts

```
REGISTER  /usr/local/pig/contrib/piggybank/java/piggybank.jar;

DEFINE CSVExcelStorage  org.apache.pig.piggybank.storage.CSVExcelStorage;

A = load '/home/acadgild/pig/assignment_5.2_airline/DelayedFlights.csv' USING
CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

B = foreach A generate (int)$1 as year, (int)$10 as flight_num, (chararray)$17 as
origin,(chararray) $18 as dest;

C = filter B by dest is not null;

D = group C by dest;
```

E = foreach D generate group, COUNT(C.dest);

F = order E by $1 DESC;

Result = LIMIT F 5;

A1 = load '/home/acadgild/pig/assignment_5.2_airline/airports.csv' USING CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

A2 = foreach A1 generate (chararray)$0 as dest, (chararray)$2 as city, (chararray)$4 as country;

joined_table = join Result by $0, A2 by dest;

dump joined_table;

Output:

```
2017-10-28 18:18:23,612 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
(ATL,106898,ATL,Atlanta,USA)
(DEN,63003,DEN,Denver,USA)
(DFW,70657,DFW,Dallas-Fort Worth,USA)
(LAX,59969,LAX,Los Angeles,USA)
(ORD,108984,ORD,Chicago,USA)
grunt>
```

Result:
(ATL,106898,ATL,Atlanta,USA)
(DEN,63003,DEN,Denver,USA)
(DFW,70657,DFW,Dallas-Fort Worth,USA)
(LAX,59969,LAX,Los Angeles,USA)
(ORD,108984,ORD,Chicago,USA)

Problem Statement 2

Which month has seen the most number of cancellations due to bad weather?

Steps to find the month which seen most number of cancellation due to bad weather.

- **In Line 1**: We are registering *piggybank* jar in order to use the CSVExcelStorage class.
- **In Line 2:** we are defining org.apache.pig.piggybank.storage.CSVExcelStorage as CSVExcelStorage
- In relation **A**, we are loading the dataset using CSVExcelStorage because of its effective technique to handle double quotes and header.
- In relation **B**, we are generating the columns which are required for processing and explicitly typecasting each of them.
- In relation **C**, we are filtering the data based on cancellation and cancellation code, i.e., canceled = 1 means flight have been canceled and cancel_code = 'B' means the reason for cancellation is "weather." So relation C will point to the data which consists of canceled flights due to bad weather.

- In relation **D**, we are grouping the relation C based on every month.
- In relation **E**, we are finding the count of canceled flights every month.
- Relation **F** and **Result** is for ordering and finding the top month based on cancellation.

Pig Scripts

```
REGISTER /usr/local/pig/lib/piggybank.jar;

DEFINE CSVExcelStorage  org.apache.pig.piggybank.storage.CSVExcelStorage;

A = load '/home/acadgild/pig/assignment_5.2_airline/DelayedFlights.csv' USING
CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

B = foreach A generate (int)$2 as month,(int)$10 as flight_num,(int)$22 as
cancelled,(chararray)$23 as cancel_code;

C = filter B by cancelled == 1 AND cancel_code =='B';

D = group C by month;

E = foreach D generate group, COUNT(C.cancelled);

F= order E by $1 DESC;

Result = limit F 1;

dump Result;
```

Output:

```
2017-10-28 18:22:03,569 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
(12,250)
```

Result:

(12,250)

Problem Statement 3

Top ten origins with the highest AVG departure delay

Steps to find the top ten origins with the highest AVG departure delay.

- Explanation of first 3 lines are the same as explained in the previous 2 problem statements.

- In relation **C1**, we are removing the null values fields present if any.
- In relation **D1**, we are grouping the data based on column "origin."
- In relation **E1**, we are finding average delay from each unique origin.
- **In Line 2:** we are defining org.apache.pig.piggybank.storage.CSVExcelStorage as CSVExcelStorage
- Relations named **Result** and **Top_ten** are ordering the results in descending order and printing the top ten values.
- These steps are good enough to find the top ten origins with the highest average departure delay.
- However, rather than generating just the code of origin, we will be following a few more steps to find some more details like country and city.
- In the relation **Lookup**, we are loading another table to which we will look up and find the city as well as the country.
- In the relation L**ookup1,** we are generating the destination, city, and country from the previous relation.
- In the relation **Joined**, we are joining relation Top_ten and Lookup1 based on common a column, i.e., "origin."
- In the relation **Final,** we are generating required columns from the Joined table.
- Finally, we are ordering and printing the results.

Pig Scripts

```
REGISTER /usr/local/pig/contrib/piggybank/java/piggybank.jar;

DEFINE CSVExcelStorage  org.apache.pig.piggybank.storage.CSVExcelStorage;

A = load '/home/acadgild/pig/assignment_5.2_airline/DelayedFlights.csv' USING
CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

B1 = foreach A generate (int)$16 as dep_delay, (chararray)$17 as origin;

C1 = filter B1 by (dep_delay is not null) AND (origin is not null);

D1 = group C1 by origin;

E1 = foreach D1 generate group, AVG(C1.dep_delay);

Result = order E1 by $1 DESC;

Top_ten = limit Result 10;

Lookup = load '/home/acadgild/pig/assignment_5.2_airline/airports.csv' USING
CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

Lookup1 = foreach Lookup generate (chararray)$0 as origin, (chararray)$2 as city, (chararray)$4
as country;
```

```
Joined = join Lookup1 by origin, Top_ten by $0;

Final = foreach Joined generate $0,$1,$2,$4;

Final_Result = ORDER Final by $3 DESC;

dump Final_Result;
```

Output:

```
2017-10-28 18:27:32,838 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
(CMX,Hancock,USA,116.1470588235294)
(PLN,Pellston,USA,93.76190476190476)
(SPI,Springfield,USA,83.84873949579831)
(ALO,Waterloo,USA,82.2258064516129)
(MQT,NA,USA,79.55665024630542)
(ACY,Atlantic City,USA,79.3103448275862)
(MOT,Minot,USA,78.66165413533835)
(HHH,NA,USA,76.53005464480874)
(EGE,Eagle,USA,74.12891986062718)
(BGM,Binghamton,USA,73.15533980582525)
```

Result:

(CMX,Hancock,USA,116.1470588235294)
(PLN,Pellston,USA,93.76190476190476)
(SPI,Springfield,USA,83.84873949579831)
(ALO,Waterloo,USA,82.2258064516129)
(MQT,NA,USA,79.55665024630542)
(ACY,Atlantic City,USA,79.3103448275862)
(MOT,Minot,USA,78.66165413533835)
(HHH,NA,USA,76.53005464480874)
(EGE,Eagle,USA,74.12891986062718)
(BGM,Binghamton,USA,73.15533980582525)

Problem Statement 4

Which route (origin & destination) has seen the maximum diversion?

Steps to find the route (origin & destination) that has seen the maximum diversion.

- **In Line 1**: We are registering *piggybank* jar in order to use CSVExcelStorage class.
- **In Line 2:** we are defining org.apache.pig.piggybank.storage.CSVExcelStorage as CSVExcelStorage
- In relation **A**, we are loading the dataset using CSVExcelStorage because of its effective technique to handle double quotes and headers.
- In relation **B**, we are generating the columns which are required for processing and explicitly type-casting each of them.
- In relation **C**, we are filtering the data based on "not null" and diversion =1. This will remove the null records, if any, and give the data corresponding to the diversion taken.

- In relation **D**, we are grouping the data based on origin and destination.
- Relation **D** finds the count of diversion taken per unique origin and destination.
- Relations **F** and **Result** orders the result and produces top 10 results.

Pig Scripts

```
REGISTER /usr/local/pig/contrib/piggybank/java/piggybank.jar;

DEFINE CSVExcelStorage  org.apache.pig.piggybank.storage.CSVExcelStorage;

A = load '/home/acadgild/pig/assignment_5.2_airline/DelayedFlights.csv' USING
CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

B = FOREACH A GENERATE (chararray)$17 as origin, (chararray)$18 as dest, (int)$24 as
diversion;

C = FILTER B BY (origin is not null) AND (dest is not null) AND (diversion == 1);

D = GROUP C by (origin,dest);

E = FOREACH D generate group, COUNT(C.diversion);

F = ORDER E BY $1 DESC;

Result = limit F 10;

dump Result;
```

Output:

```
2017-10-28 18:30:29,636 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
((ORD,LGA),39)
((DAL,HOU),35)
((DFW,LGA),33)
((ATL,LGA),32)
((ORD,SNA),31)
((SLC,SUN),31)
((MIA,LGA),31)
((BUR,JFK),29)
((HRL,HOU),28)
((BUR,DFW),25)
```

Result:

((ORD,LGA),39)
((DAL,HOU),35)
((DFW,LGA),33)
((ATL,LGA),32)
((ORD,SNA),31)
((SLC,SUN),31)

((MIA,LGA),31)
((BUR,JFK),29)
((HRL,HOU),28)
((BUR,DFW),25)