Project 1.3 Titanic Data analysis

Introduction The data set contains information about passengers who boarded Titanic ship.

It contains data points like: • Passenger’s age • Their native place • Details of who survived • Fare details of various travel classes • Number of casualties from various classes etc.

Associated Data Files <https://drive.google.com/file/d/0ByJLBTmJojjzNmV0dk1EMmwwQ1U/view?usp=sharing>

DATA SET DESCRIPTION Column 1 : PassengerId Column 2 : Survived (survived=0 & died=1) Column 3 : Pclass Column 4 : Name Column 5 : Sex Column 6 : Age Column 7 : SibSp Column 8 : Parch Column 9 : Ticket Column 10 : Fare Column 11 : Cabin Column 12 : Embarked

Problem Statement You can use any of the technologies like Map Reduce, Pig or Hive of your choice.

Note:You need to copy the data set into HDFS using flume and send the screen shot of that with the project solution

**Copied data from local directory to HDFS**

***Flume.conf***

*agent.sources = localdir*

*agent.channels = channel\_f*

*agent.sinks = HDFS*

*# For each one of the sources, the type is defined*

*agent.sources.localdir.type = spooldir*

*# The channel can be defined as follows.*

*agent.sources.localdir.channels = channel\_f*

*agent.sources.localdir.spoolDir = /home/flumedata/flumespool*

*agent.sources.localdir.fileHeader = false*

*agent.sources.localdir.fileSuffix = .COMPLETED*

*# Each sink's type must be defined*

*agent.sinks.HDFS.type = hdfs*

*#Specify the channel the sink should use*

*agent.sinks.HDFS.channel = channel\_f*

*agent.sinks.HDFS.hdfs.path = hdfs://localhost/user/ec2-user/flume/titanic/*

*agent.sinks.HDFS.hdfs.fileType = DataStream*

*agent.sinks.HDFS.hdfs.writeFormat = Text*

*agent.sinks.HDFS.hdfs.batchSize = 1000*

*agent.sinks.HDFS.hdfs.rollSize = 268435456*

*agent.sinks.HDFS.hdfs.rollInterval = 0*

*agent.sinks.HDFS.hdfs.rollCount = 50000000*

*# Each channel's type is defined.*

*agent.channels.channel\_f.type = file*

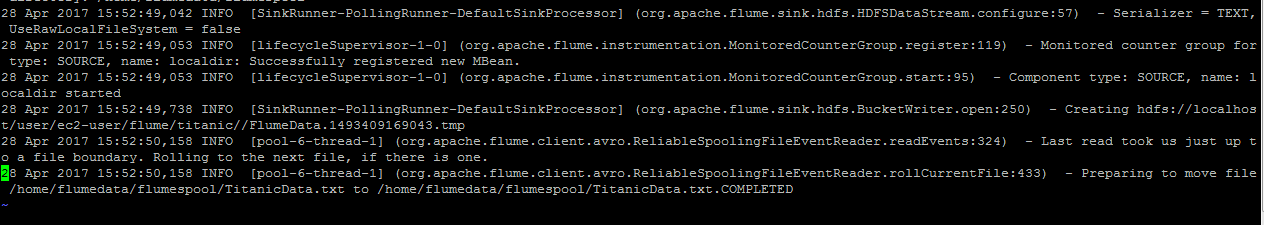
*# Other config values specific to each type of channel(sink or source)*

*# can be defined as well*

*# In this case, it specifies the capacity of the memory channel*

*agent.channels.memoryChannel.capacity = 10000*

*agent.channels.memoryChannel.transactionCapacity = 100*



USING PIG

titanic\_data = load '/user/ec2-user/flume/titanic/FlumeData.1493409169043' using PigStorage(',') as (passenger\_id:INT,survived:INT,pclass:chararray,name:chararray,sex:chararray,age:INT,sibsp:chararray,parch:chararray,ticket:chararray,fare:float,cabin:chararray,embarked:chararray);

**1.In this problem statement we will find the average fare of each class.**

group\_by\_class = GROUP titanic\_data by pclass;

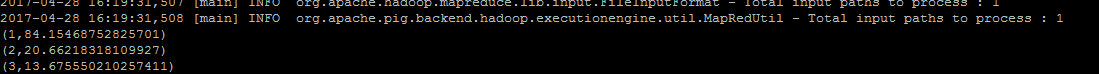
avg\_fare\_by\_class = foreach group\_by\_class generate group as class, AVG(titanic\_data.fare) as avgfare;

dump avg\_fare\_by\_class

(1,84.15468752825701)

(2,20.66218318109927)

(3,13.675550210257411)



**2.In this problem statement we will find the number of people alive in each class and are embarked in Southampton.**

As per problem statement , Survived (survived=0 & died=1)

people\_survived\_from\_S = FILTER titanic\_data by survived==0 AND embarked=='S';

grp = GROUP people\_survived\_from\_S all;

number\_of\_ppl\_survived = foreach grp generate COUNT(people\_survived\_from\_S);

(854)



**3.In this problem statement we will find out number of male and female people died in each class.**

female\_died = FILTER titanic\_data by survived==1 AND sex matches '(?i).\*female.\*';

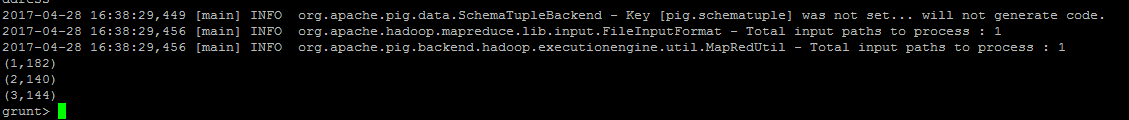
grp\_female\_died\_by\_class = GROUP female\_died by pclass;

num\_female\_died = foreach grp\_female\_died\_by\_class generate group as class, COUNT(female\_died);

(1,182)

(2,140)

(3,144)



male\_died = FILTER titanic\_data by survived==1 AND sex=='male';

grp\_male\_died\_by\_class = GROUP male\_died by pclass;

num\_male\_died = foreach grp\_male\_died\_by\_class generate group as class, COUNT(male\_died);

(1,90)

(2,34)

(3,94)

