CPSC 531- Advance Database Management System

Final Project

Airline Analysis

Web URL: https://airline-delay-analysis.netlify.app/

GitHub URL:

https://github.com/LencyLakhani/advancedatabase

Team Members:

Hetal Patel

885868455

hetal-patel.1994@csu.fullerton.edu

Lency Lakhani

885196055

lencylakhani@csu.fullerton.edu

Mentor:

Prof. Tseng-Ching Shen

Table Of Content

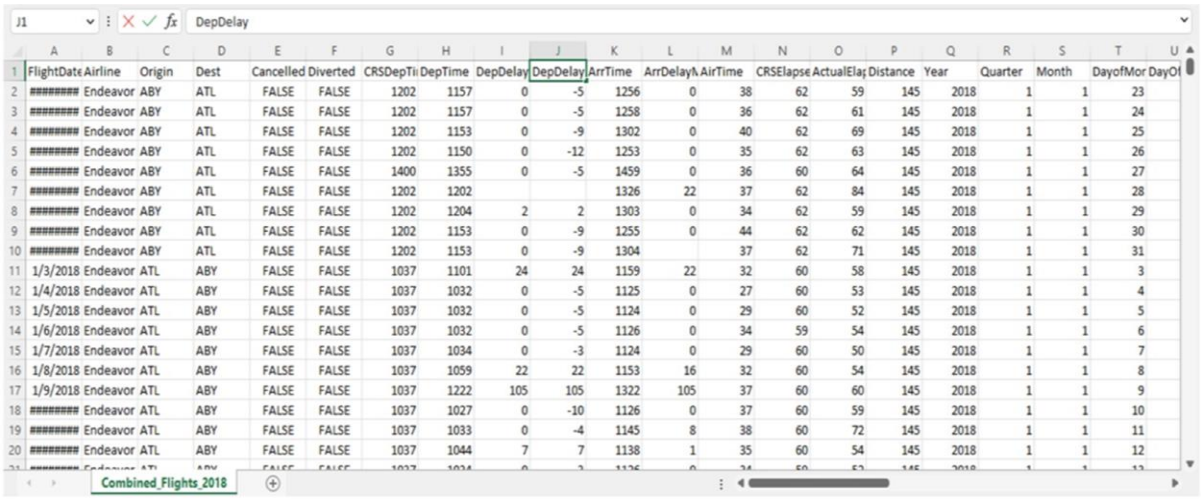
Problem Statement

Nowadays, there are many reasons for the delay and cancellation of flights. The Federal Aviation Administration estimates flight delays cost airlines $22 billion yearly in the United States. Flight delays are inconvenient for passengers as well. The airline analysis allows for comparisons between different airlines, their time, arrival, and departure.

We analysed an airline dataset from 2018 to identify flights that were delayed for over 60 minutes, had a minimum airtime of 100 minutes, and examined their destinations. Our findings provide detailed information on specific airline flights that meet these criteria.

Dataset

We utilised a 1.9 GB dataset from the Kaggel website consisting of 1048575 records and 61 title columns. After cleaning the source CSV file, we extracted only the necessary fields such as Airline, Origin, Destination, Airtime, and departure delay time for the final output CSV file following analysis.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	FlightDate	Airline	Origin	Dest	Cancelled	Diverted	CRSDepTi	DepTime	DepDelay	DepDelay	ArrTime	ArrDelayN	AirTime	CRSElapse	ActualEla	Distance	Year	Quarter	Month	DayofMor	DayOf
2	########	Endeavor	ABY	ATL	FALSE	FALSE	1202	1157	0	-5	1256	0	38	62	59	145	2018	1	1	23	
3	########	Endeavor	ABY	ATL	FALSE	FALSE	1202	1157	0	-5	1258	0	36	62	61	145	2018	1	1	24	
4	########	Endeavor	ABY	ATL	FALSE	FALSE	1202	1153	0	-9	1302	0	40	62	69	145	2018	1	1	25	
5	########	Endeavor	ABY	ATL	FALSE	FALSE	1202	1150	0	-12	1253	0	35	62	63	145	2018	1	1	26	
6	########	Endeavor	ABY	ATL	FALSE	FALSE	1400	1355	0	-5	1459	0	36	60	64	145	2018	1	1	27	
7	########	Endeavor	ABY	ATL	FALSE	FALSE	1202	1202			1326	22	37	62	84	145	2018	1	1	28	
8	########	Endeavor	ABY	ATL	FALSE	FALSE	1202	1204	2	2	1303	0	34	62	59	145	2018	1	1	29	
9	########	Endeavor	ABY	ATL	FALSE	FALSE	1202	1153	0	-9	1255	0	44	62	62	145	2018	1	1	30	
10	########	Endeavor	ABY	ATL	FALSE	FALSE	1202	1153	0	-9	1304		37	62	71	145	2018	1	1	31	
11	1/3/2018	Endeavor	ATL	ABY	FALSE	FALSE	1037	1101	24	24	1159	22	32	60	58	145	2018	1	1	3	
12	1/4/2018	Endeavor	ATL	ABY	FALSE	FALSE	1037	1032	0	-5	1125	0	27	60	53	145	2018	1	1	4	
13	1/5/2018	Endeavor	ATL	ABY	FALSE	FALSE	1037	1032	0	-5	1124	0	29	60	52	145	2018	1	1	5	
14	1/6/2018	Endeavor	ATL	ABY	FALSE	FALSE	1037	1032	0	-5	1126	0	34	59	54	145	2018	1	1	6	
15	1/7/2018	Endeavor	ATL	ABY	FALSE	FALSE	1037	1034	0	-3	1124	0	29	60	50	145	2018	1	1	7	
16	1/8/2018	Endeavor	ATL	ABY	FALSE	FALSE	1037	1059	22	22	1153	16	32	60	54	145	2018	1	1	8	
17	1/9/2018	Endeavor	ATL	ABY	FALSE	FALSE	1037	1222	105	105	1322	105	37	60	60	145	2018	1	1	9	
18	########	Endeavor	ATL	ABY	FALSE	FALSE	1037	1027	0	-10	1126	0	37	60	59	145	2018	1	1	10	
19	########	Endeavor	ATL	ABY	FALSE	FALSE	1037	1033	0	-4	1145	8	38	60	72	145	2018	1	1	11	
20	########	Endeavor	ATL	ABY	FALSE	FALSE	1037	1044	7	7	1138	1	35	60	54	145	2018	1	1	12	

Technology & Tools Used

Backend:

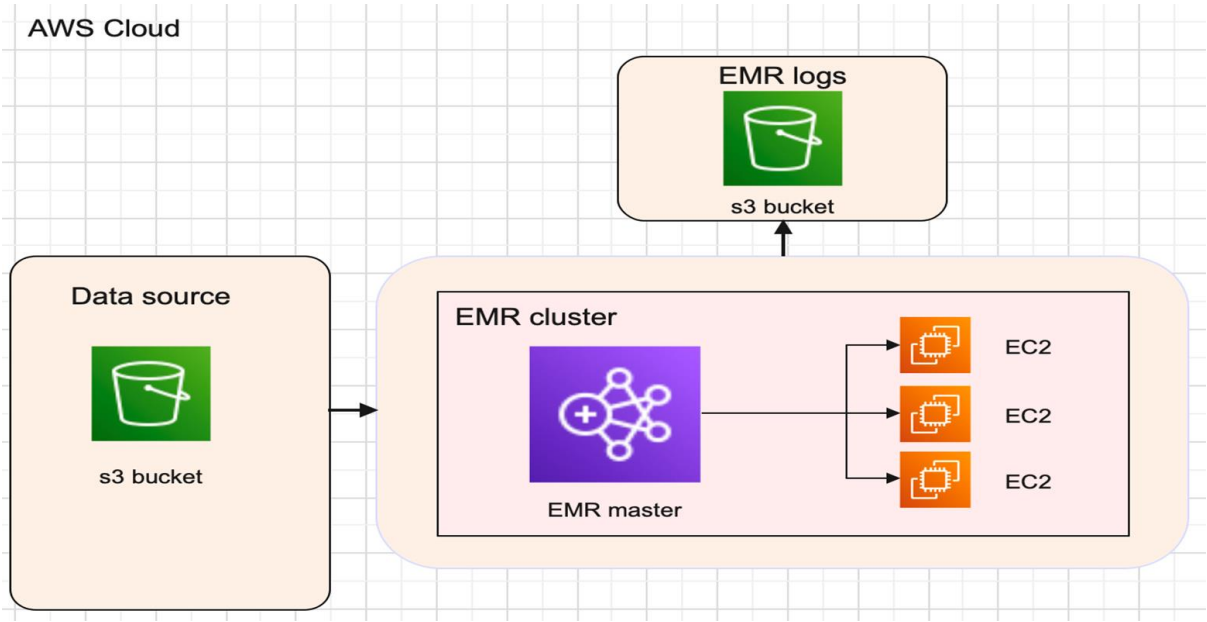
- AWS EMR (Amazon Elastic MapReduce)
- AWS S3 (Amazon Simple Storage Service)
- AWS EC2 (Amazon Elastic Compute Cloud)
- Apache Spark
- Python Pyspark SQL libraries

Frontend:

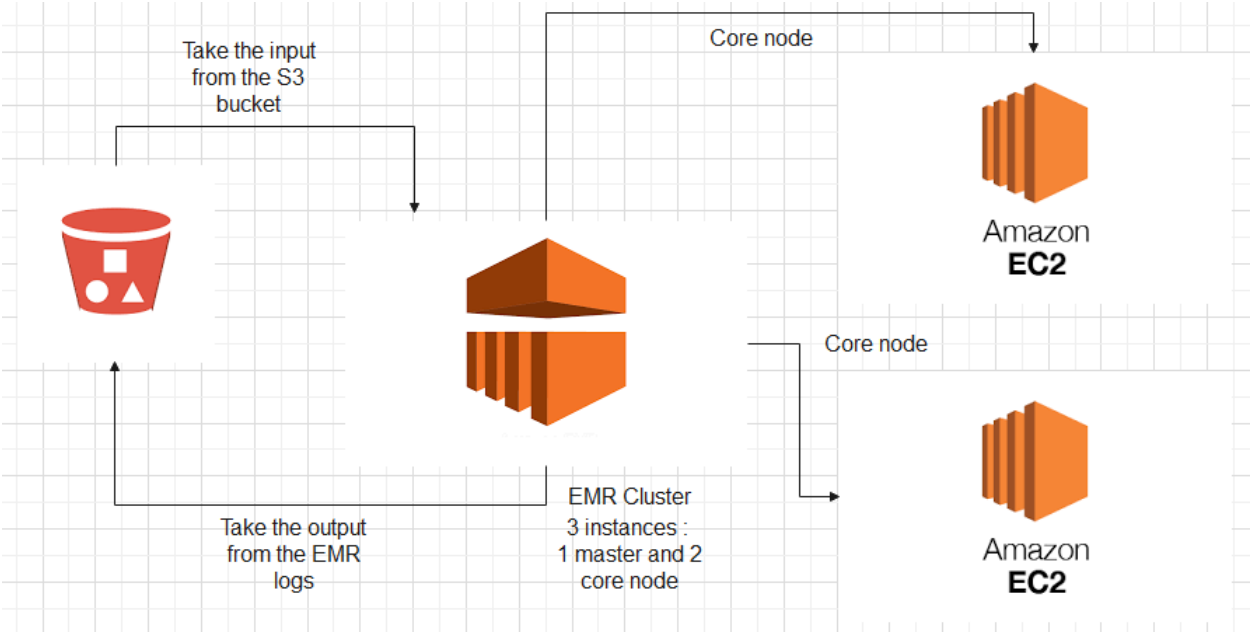
- HTML
- Javascript
- Bootstrap
- CSS

Overview Of Design

We used AWS EMR to handle large data servers on the cloud efficiently. This is possible because EMR comes equipped with pre-installed and pre-configured Spark and Hadoop. EMR necessitates worker nodes to execute the task, which we provided through AWS EC2 instances. To tackle a vast dataset, we carefully selected an EC2 instance with a configuration that offers 1 Master and 2 core nodes of m5.xlarge. Furthermore, the AWS S3 bucket served as our file system for storing project source, output, and Python pyspark files. Our project is represented in the diagram below, where EMR employs 3 instances of EC2 to execute the spark job. We accessed the source file through the S3 bucket and processed the data using the Python Pyspark file. EMR also generates logs in an S3 bucket.

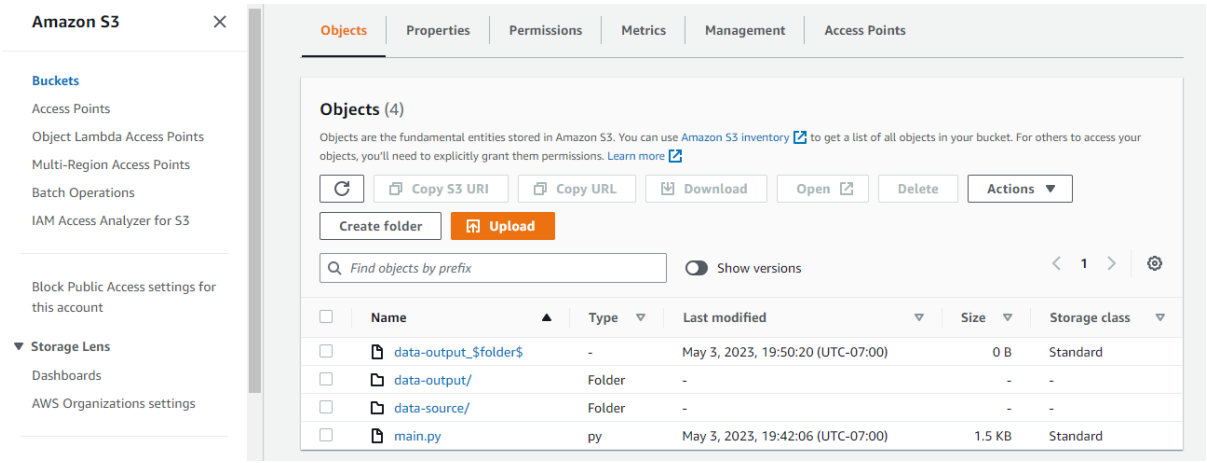


Architecture & Design

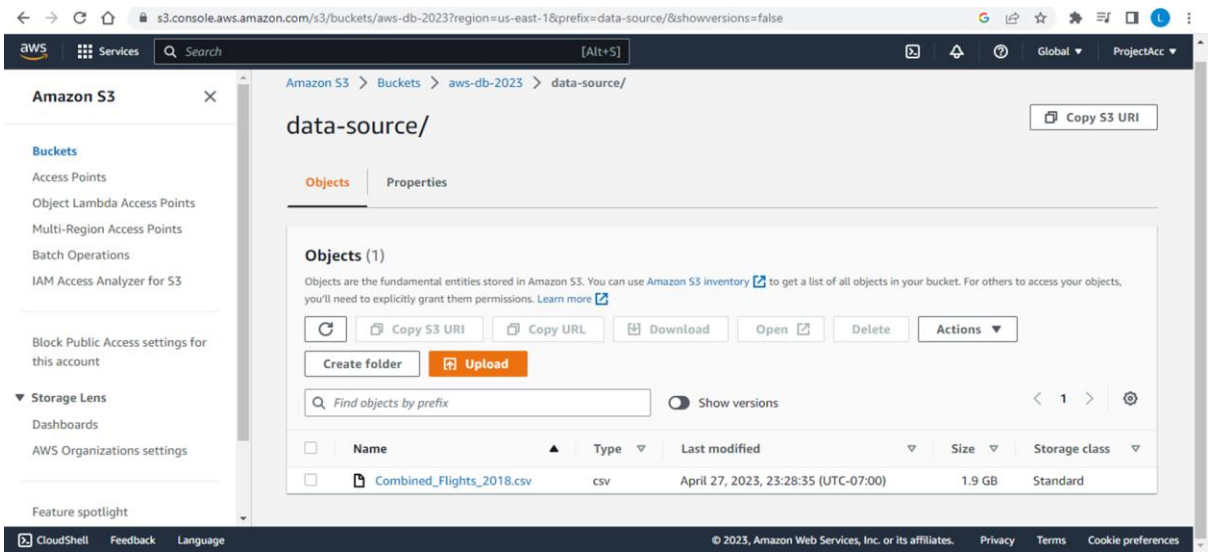


Implementation Overview

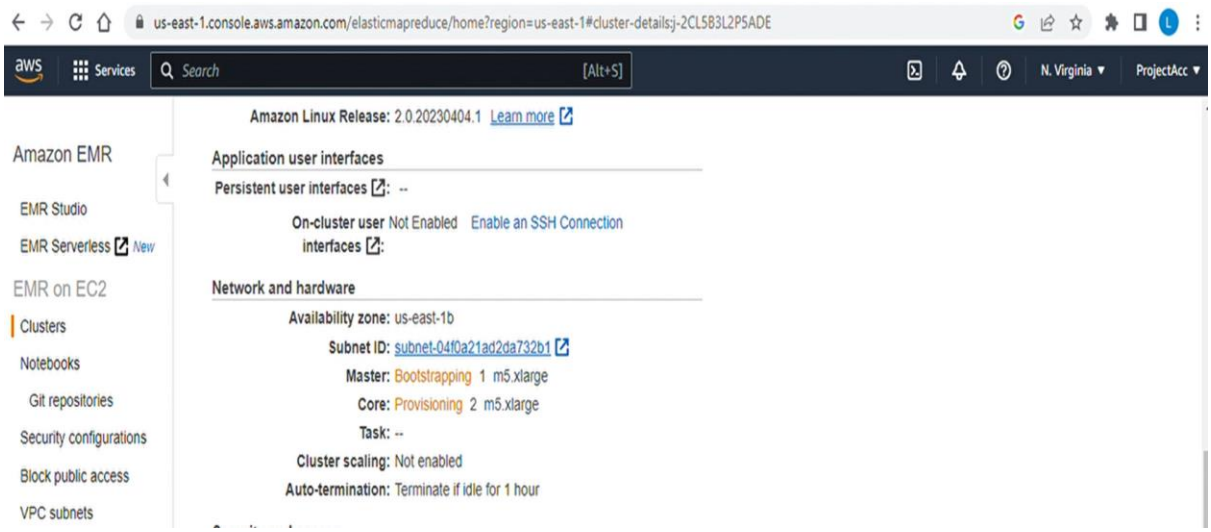
AWS S3 bucket console with data-source folder for storing airline source csv file, data-output folder to store the output csv file generated after the job completion of EMR cluster and the python file to perform spark job.



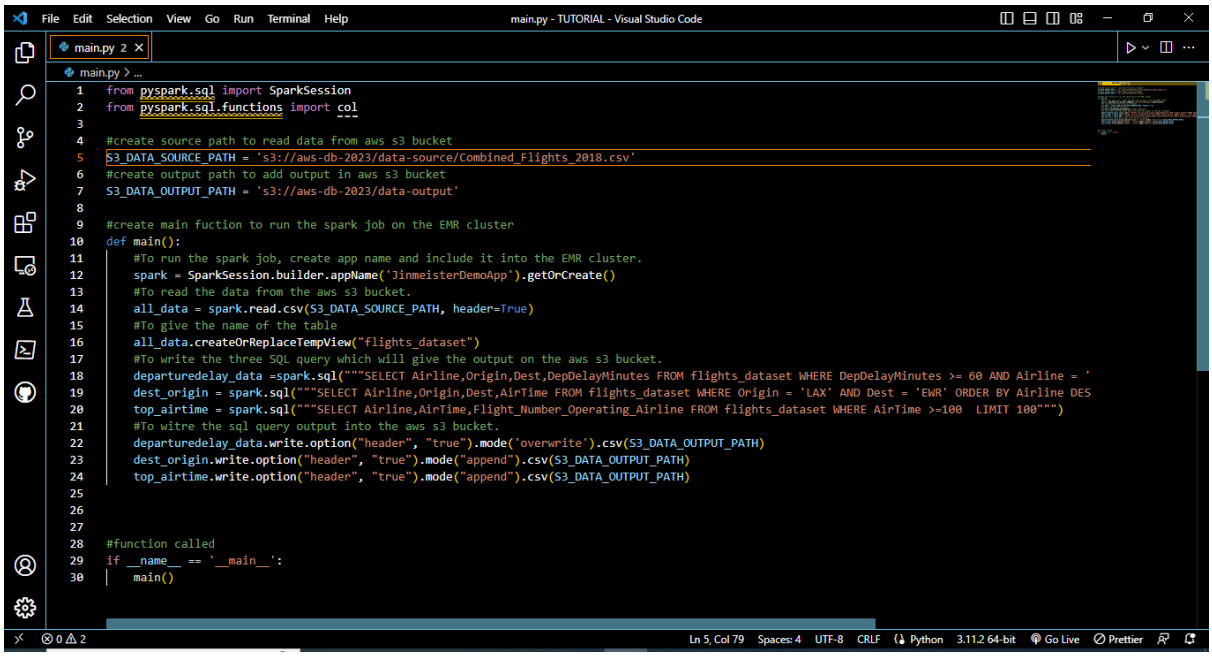
Uploaded the airline source CSV file, 1.9GB in size, to the data-source folder in S3 bucket.



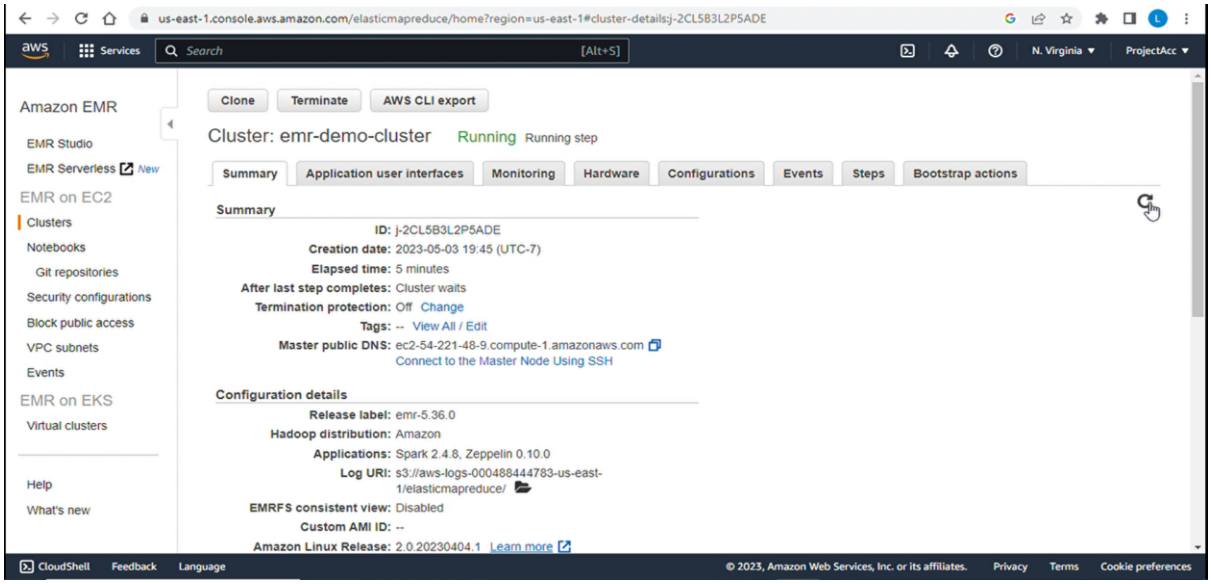
EMR cluster using three EC2 instances, consisting of one master node and two core nodes of m5.xlarge.



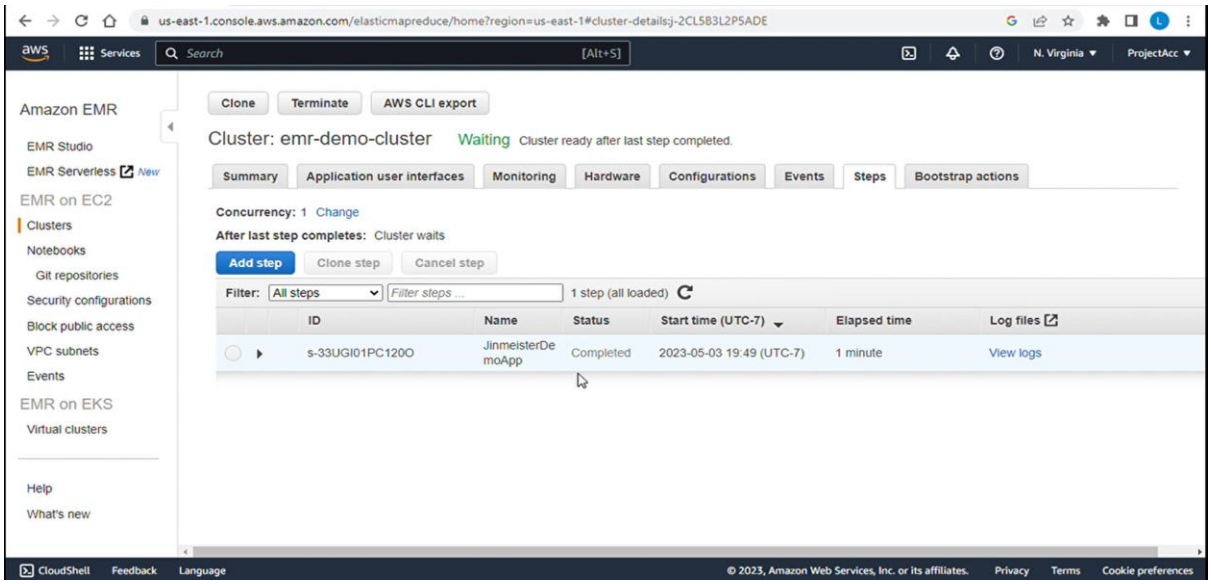
Python pyspark code to perform the spark job and process output data.



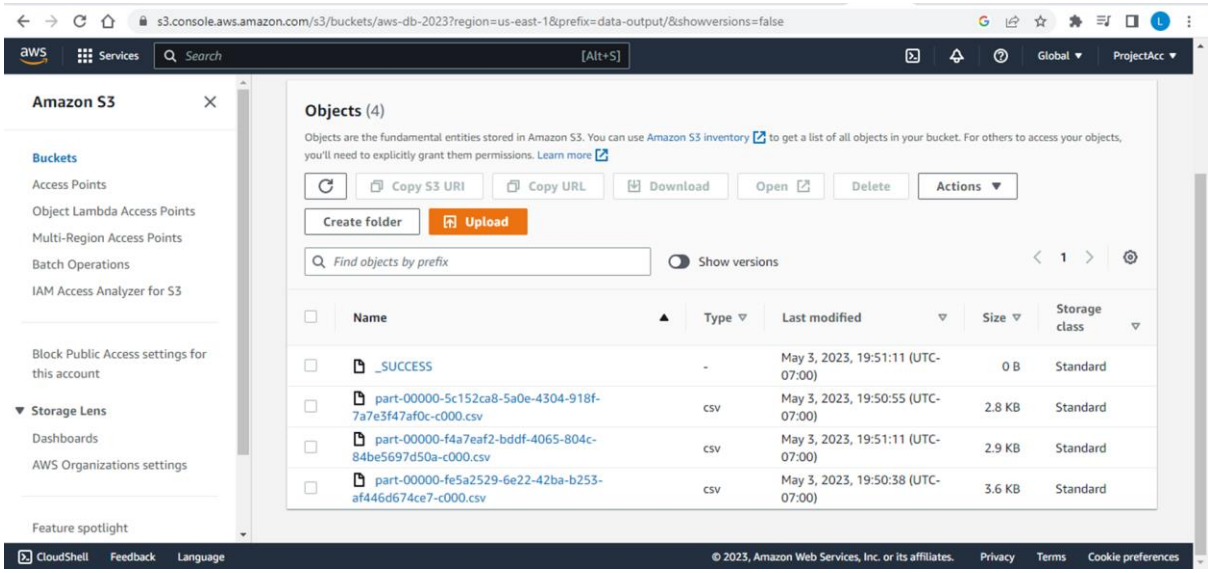
EMR running the python pyspark job file cluster instances to process the output.



EMR cluster ready with completed jb execution.

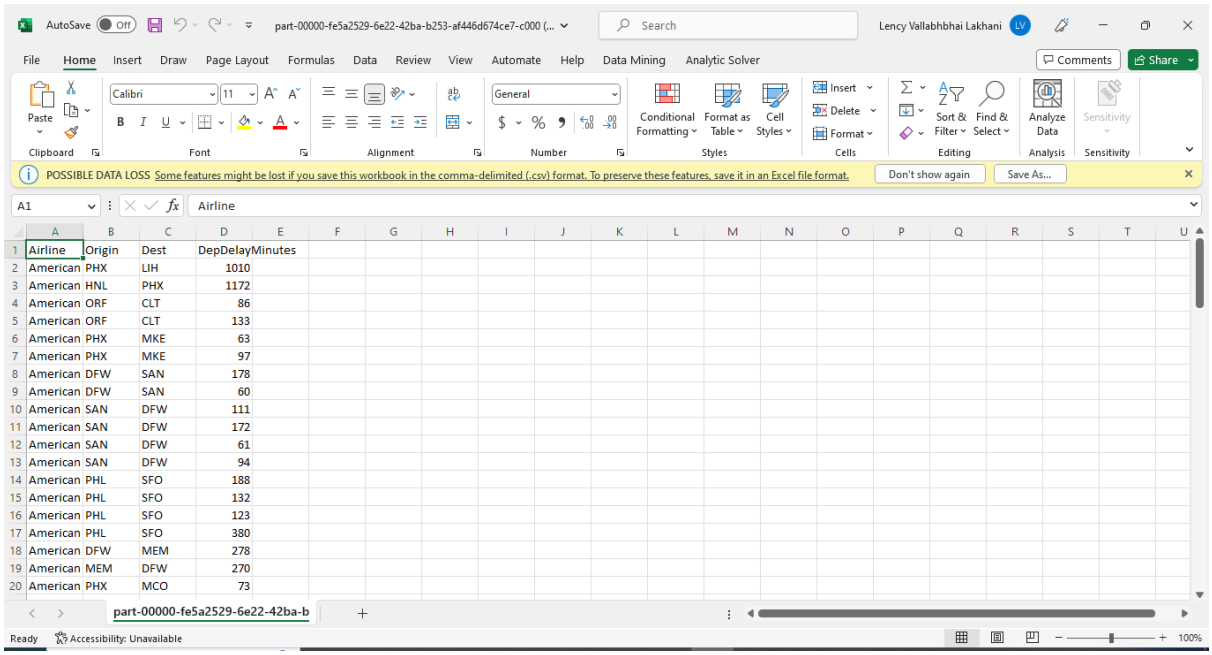


Output CSV files generated in the data-output folder of S3 bucket.



Output File 1: Delay Analysis Output for below query:

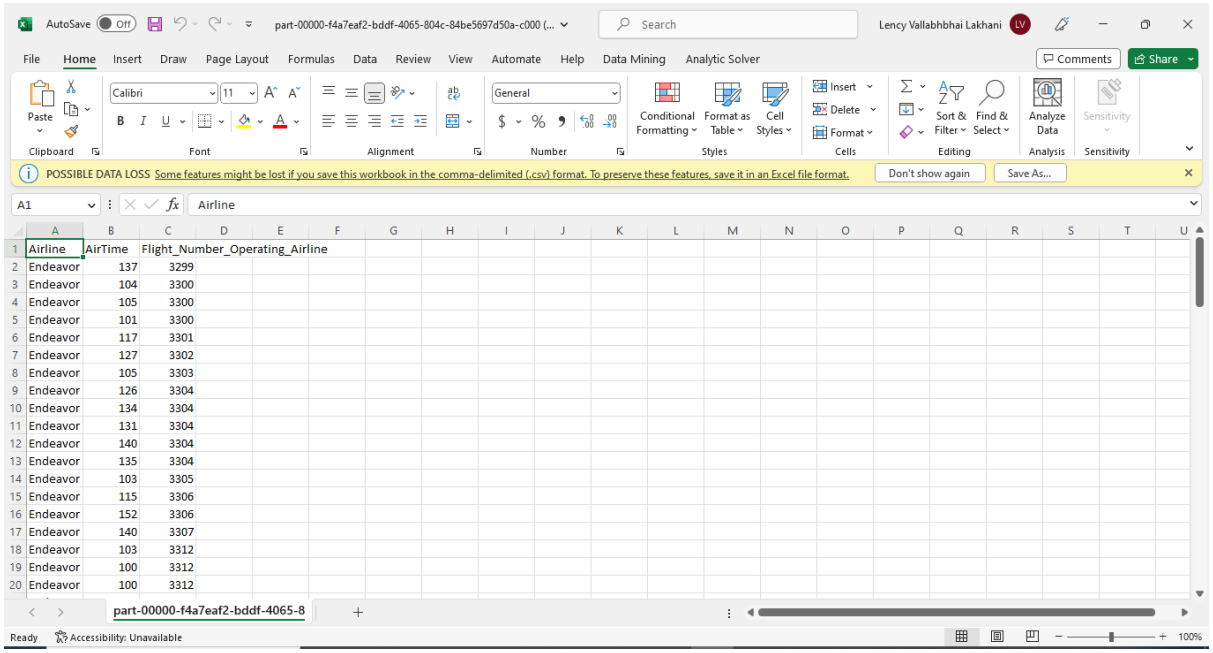
SELECT Airline,Origin,Dest,DepDelayMinutes FROM flights_dataset WHERE DepDelayMinutes >= 60 AND Airline = 'American Airlines Inc.' LIMIT 100



	A	B	C	D
1	Airline	Origin	Dest	DepDelayMinutes
2	American	PHX	LIH	1010
3	American	HNL	PHX	1172
4	American	ORF	CLT	86
5	American	ORF	CLT	133
6	American	PHX	MKE	63
7	American	PHX	MKE	97
8	American	DFW	SAN	178
9	American	DFW	SAN	60
10	American	SAN	DFW	111
11	American	SAN	DFW	172
12	American	SAN	DFW	61
13	American	SAN	DFW	94
14	American	PHL	SFO	188
15	American	PHL	SFO	132
16	American	PHL	SFO	123
17	American	PHL	SFO	380
18	American	DFW	MEM	278
19	American	MEM	DFW	270
20	American	PHX	MCO	73

Output File 2: Airtime Analysis Output for below query:

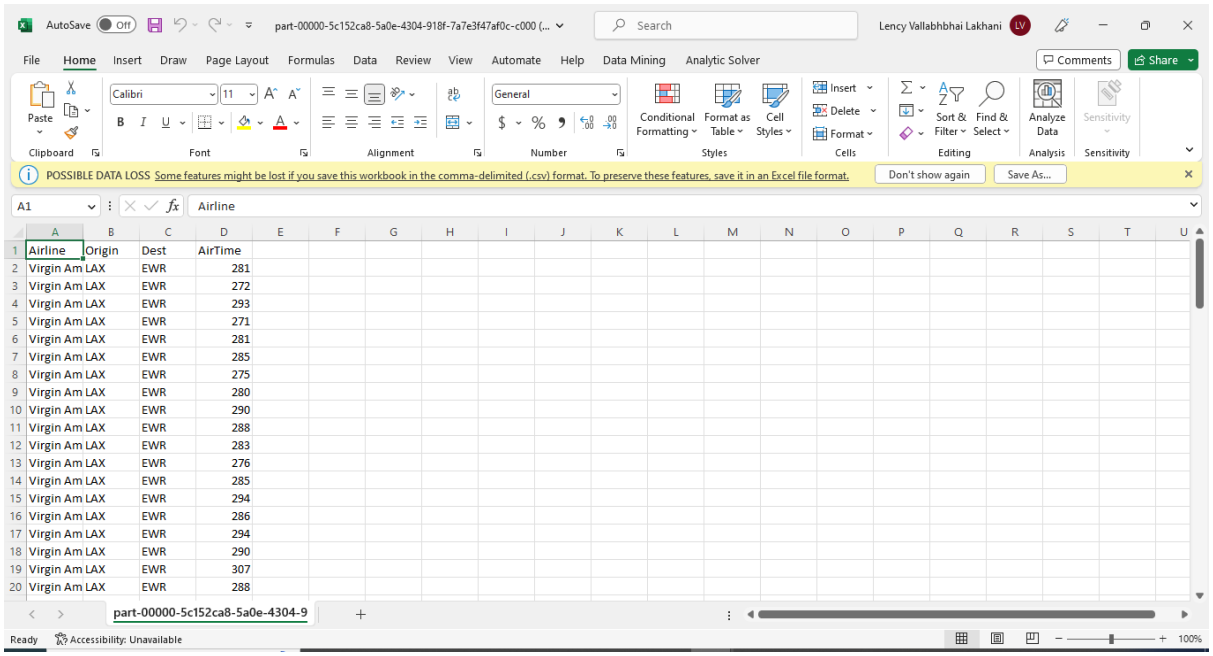
SELECT Airline,AirTime,Flight_Number_Operating_Airline FROM flights_dataset WHERE AirTime >=100 LIMIT 100



Airline	AirTime	Flight_Number_Operating_Airline
Endeavor	137	3299
Endeavor	104	3300
Endeavor	105	3300
Endeavor	101	3300
Endeavor	117	3301
Endeavor	127	3302
Endeavor	105	3303
Endeavor	126	3304
Endeavor	134	3304
Endeavor	131	3304
Endeavor	140	3304
Endeavor	135	3304
Endeavor	103	3305
Endeavor	115	3306
Endeavor	152	3306
Endeavor	140	3307
Endeavor	103	3312
Endeavor	100	3312
Endeavor	100	3312

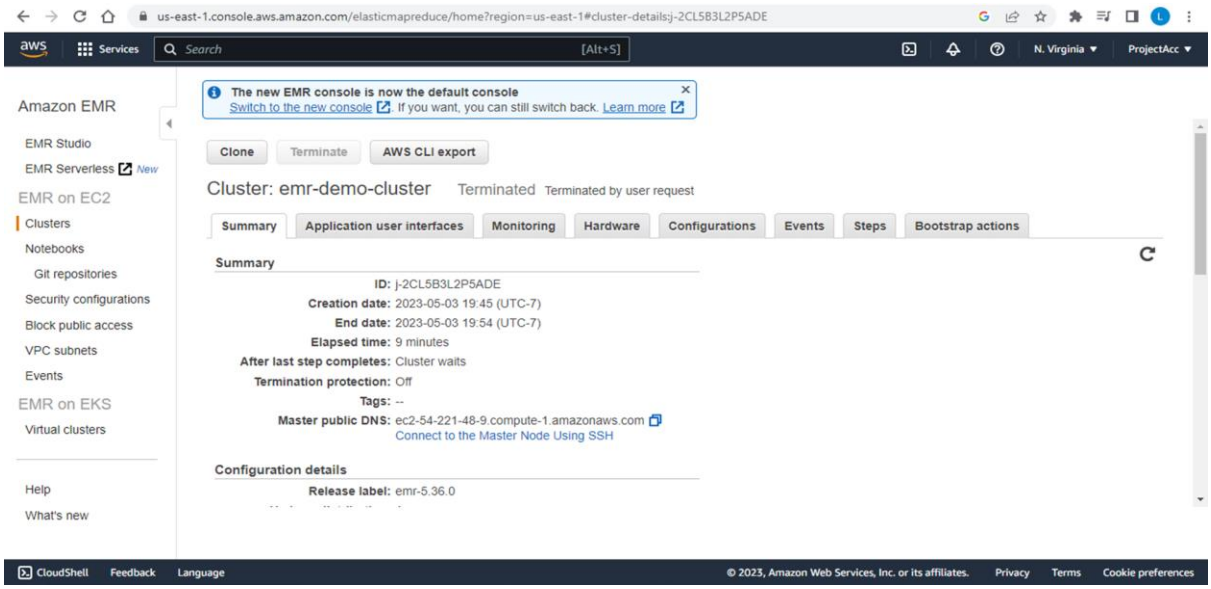
Output File 3: Destination Analysis Output for below query:

SELECT Airline,Origin,Dest,AirTime FROM flights_dataset WHERE Origin = 'LAX' AND Dest = 'EWR' ORDER BY Airline DESC LIMIT 100



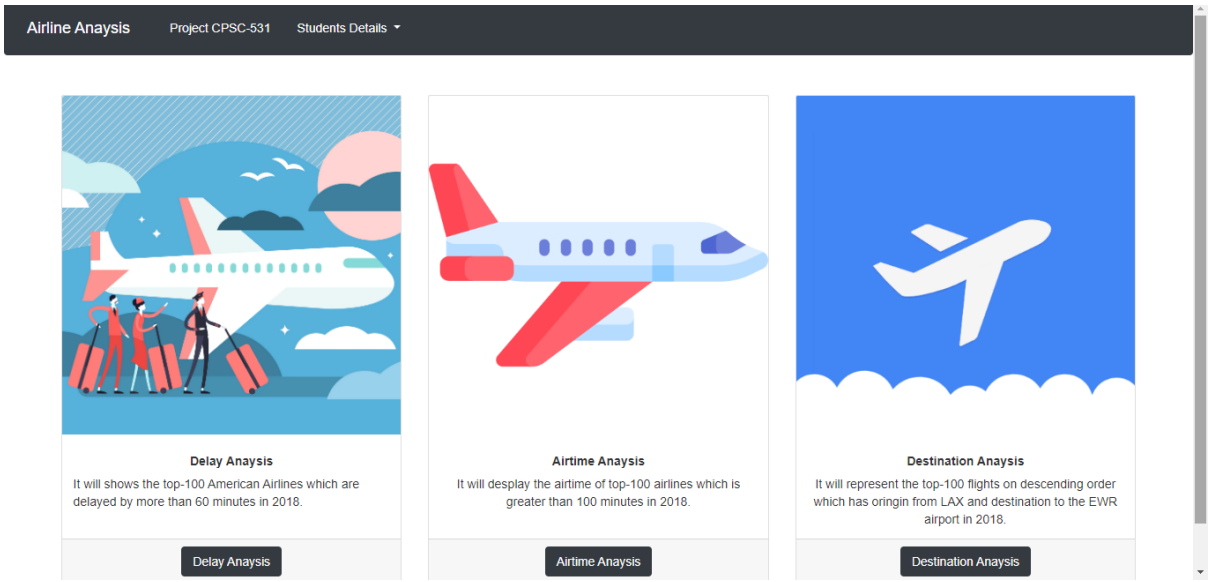
Airline	Origin	Dest	AirTime
Virgin Am	LAX	EWR	281
Virgin Am	LAX	EWR	272
Virgin Am	LAX	EWR	293
Virgin Am	LAX	EWR	271
Virgin Am	LAX	EWR	281
Virgin Am	LAX	EWR	285
Virgin Am	LAX	EWR	275
Virgin Am	LAX	EWR	280
Virgin Am	LAX	EWR	290
Virgin Am	LAX	EWR	288
Virgin Am	LAX	EWR	283
Virgin Am	LAX	EWR	276
Virgin Am	LAX	EWR	285
Virgin Am	LAX	EWR	294
Virgin Am	LAX	EWR	286
Virgin Am	LAX	EWR	294
Virgin Am	LAX	EWR	290
Virgin Am	LAX	EWR	307
Virgin Am	LAX	EWR	288

Terminate the cluster after job completion.



User Interface

We have created a user interface using javascript, HTML, CSS, and Bootstrap. This User interface consists of airline analysis which we have done using AWS services for 1.9 GB of airline dataset of 2008. It includes the three types of analysis “Delay analysis” , “Airtime analysis” and “Destination Analysis”. When the user clicks on each of the analysis, it will redirect to the page which will show the airline analysis and their related scatter graph.

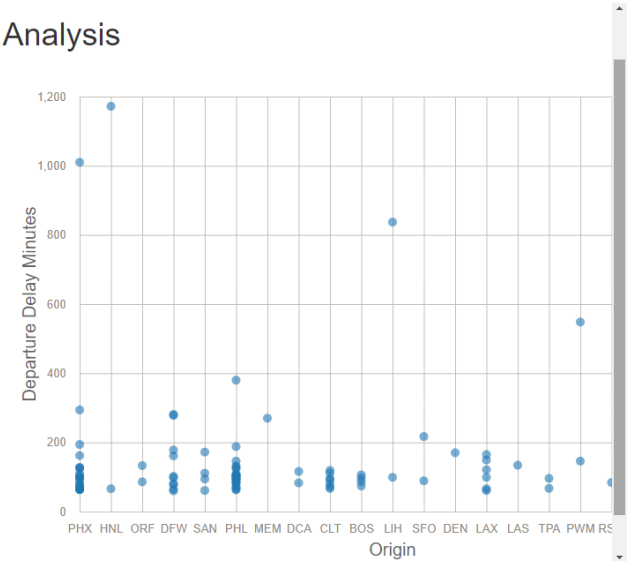


1.Delay analysis

It will show the top-100 American Airlines which were delayed by more than 60 minutes in 2008.

Delay Analysis

Airline	Origin	Dest	DepDelayMinutes
American Airlines Inc.	PHX	LIH	1010
American Airlines Inc.	HNL	PHX	1172
American Airlines Inc.	ORF	CLT	86
American Airlines Inc.	ORF	CLT	133
American Airlines Inc.	PHX	MKE	63
American Airlines Inc.	PHX	MKE	97
American Airlines Inc.	DFW	SAN	178
American Airlines Inc.	DFW	SAN	60

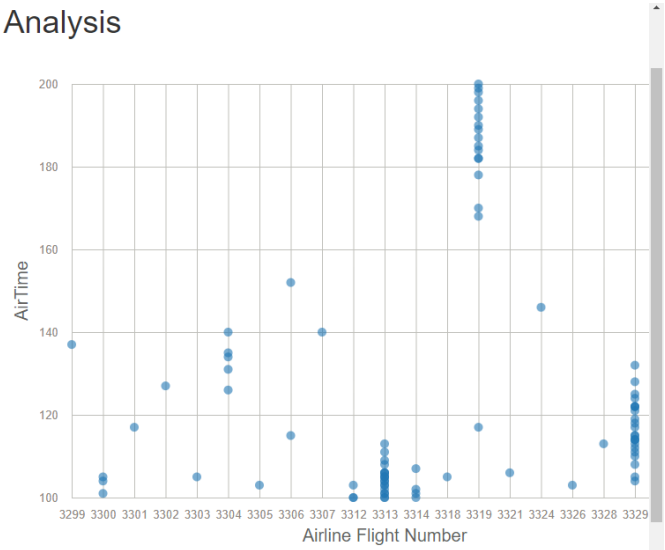


2.Airtime Analysis

It will display the airtime of top-100 airlines which is greater than 100 minutes in 2008.

Airtime Analysis

Airline	AirTime	Flight_Number_Operating_Airline
Endeavor Air Inc.	137.0	3299
Endeavor Air Inc.	104.0	3300
Endeavor Air Inc.	105.0	3300
Endeavor Air Inc.	101.0	3300
Endeavor Air Inc.	117.0	3301
Endeavor Air Inc.	127.0	3302
Endeavor Air Inc.	105.0	3303
Endeavor Air Inc.	126.0	3304
Endeavor Air Inc.	134.0	3304
Endeavor Air Inc.	131.0	3304
Endeavor Air Inc.	140.0	3304
Endeavor Air Inc.	135.0	3304

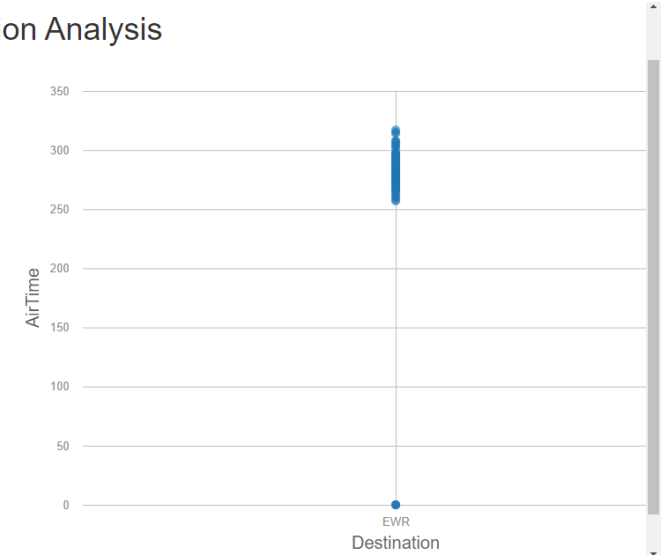


3.Destination Analysis

It will represent the top-100 flights in descending order which has origin from LAX and destination to the EWR airport in 2018.

Destination Analysis

Airline	Origin	Dest	AirTime
Virgin America	LAX	EWR	281
Virgin America	LAX	EWR	272
Virgin America	LAX	EWR	293
Virgin America	LAX	EWR	271
Virgin America	LAX	EWR	281
Virgin America	LAX	EWR	285
Virgin America	LAX	EWR	275
Virgin America	LAX	EWR	280
Virgin America	LAX	EWR	290
Virgin America	LAX	EWR	288
Virgin America	LAX	EWR	283
Virgin America	LAX	EWR	276



Step to run the code

1. Add the main.py file into the AWS s3 bucket.
2. In the EMR cluster, go to the steps.
3. In steps, follow the “add steps” button and then add the path of the main.py file which is stored in AWS s3 bucket.
4. Start the EMR cluster.
5. Waiting to the running step of the cluster.
6. Go to the s3 bucket output file.
7. In the last step, terminate the cluster.