Spark Plane Distances Estimate

Team Nexus:
Paul J. Anderson
Rachel Hughes

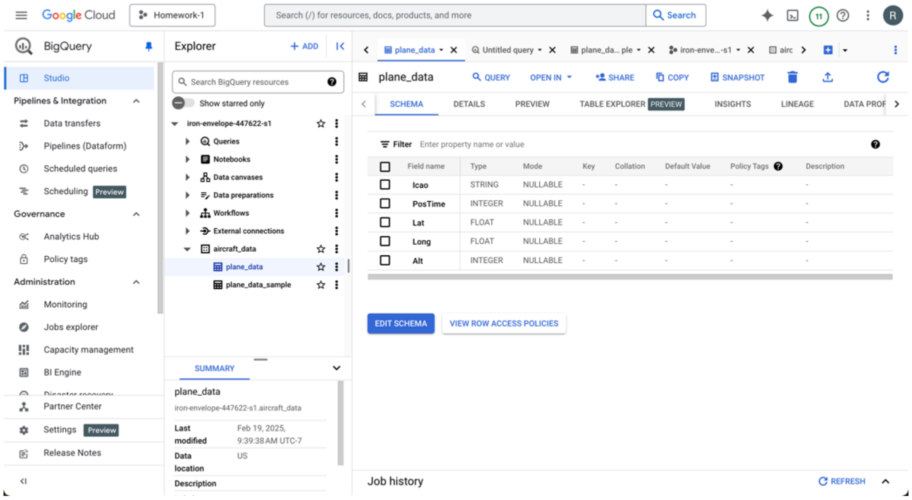
CS 512, Winter 2025
Oregon State University

Abstract:

This project is designed to explore the power of Google Cloud Platform services, especially using Dataproc to run Spark jobs on managed clusters. Using the provided dataset of airplane traffic, the ultimate goal of the project is to determine the total amount of distance traveled by all airplanes that flew during the 24 hours of 15 July 2018. Results indicate that aircraft traveled an estimated 148,665,089 km in those 24 hours.

Obtain:

The project imports a complete 24-hour dataset of airplane traffic for January 15, 2018 into BigQuery. Previous classwork involved scrubbing the data in DataPrep to get a standardized JSON format and to eliminate excess data that was not needed for the project. As documented previously, attempts to import into BigQuery failed. Instead, a .csv file was provide and uploaded into BigQuery. This data was then used for the rest of the project. The schema for the imported data is below – the data is stored in BigQuery table plane_data.



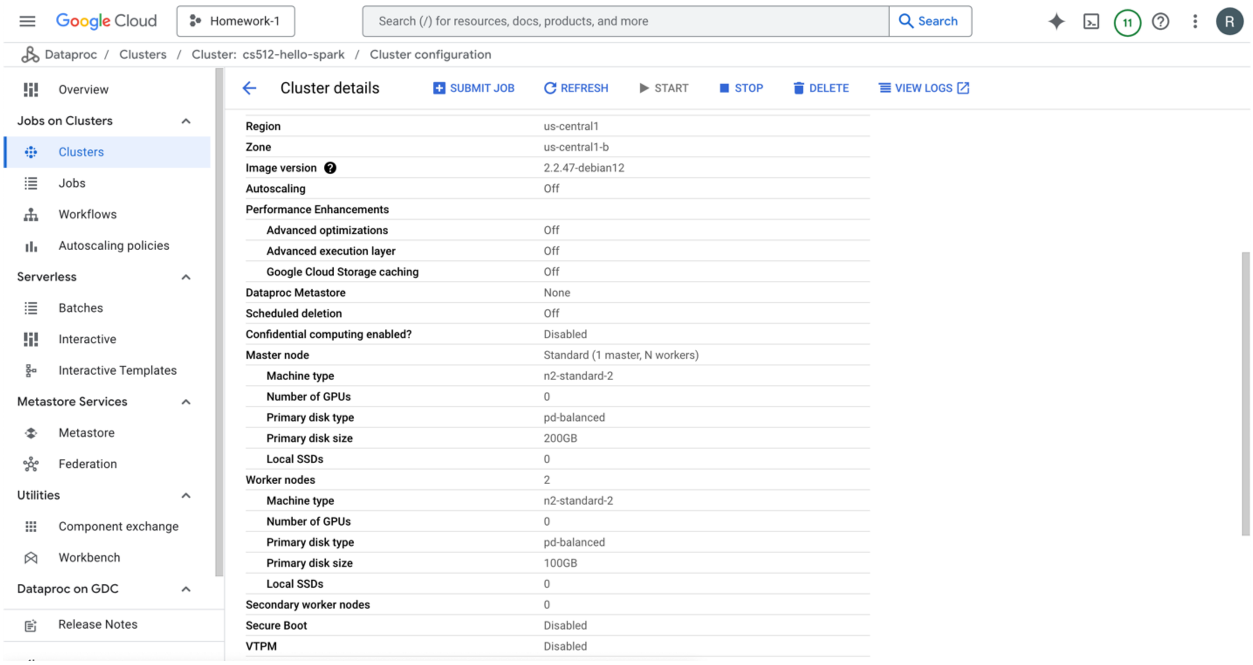
plane_data

□	Field name	Type	Mode	Key	Collation	Default Value	Policy Tags	Description
□	Icao	STRING	NULLABLE	-	-	-	-	-
□	PosTime	INTEGER	NULLABLE	-	-	-	-	-
□	Lat	FLOAT	NULLABLE	-	-	-	-	-
□	Long	FLOAT	NULLABLE	-	-	-	-	-
□	Alt	INTEGER	NULLABLE	-	-	-	-	-

plane_data
iron-envelope-447622-s1.aircraft_data

Last modified	Feb 19, 2025, 9:39:38 AM UTC-7
Data location	US
Description	

The work this week continues use of the managed clusters set up in prior weeks, as shown below, to determine an estimate of the total distance traveled by plane on 15 July 2018.



Region	us-central1
Zone	us-central1-b
Image version	2.2.47-debian12
Autoscaling	Off
Performance Enhancements	
Advanced optimizations	Off
Advanced execution layer	Off
Google Cloud Storage caching	Off
Dataproc Metastore	None
Scheduled deletion	Off
Confidential computing enabled?	Disabled
Master node	Standard (1 master, N workers)
Machine type	n2-standard-2
Number of GPUs	0
Primary disk type	pd-balanced
Primary disk size	200GB
Local SSDs	0
Worker nodes	2
Machine type	n2-standard-2
Number of GPUs	0
Primary disk type	pd-balanced
Primary disk size	100GB
Local SSDs	0
Secondary worker nodes	0
Secure Boot	Disabled
VTPM	Disabled

Scrub:

There were two steps to the data scrubbing process.

(1) The data were scrubbed in BigQuery using a query statement to select all rows of data that had no null values. There were a number of latitude and longitude values that were 0.0. This was unlikely as the odds of many planes in a single day reading exactly at 0N or 0W, for example, are slim. Therefore, these 0.0 values were considered to be an error and also removed using the following SQL query in BigQuery.

```
CREATE TABLE `projectID.bucketID.outputdf` -- insert project bucket output ID
AS SELECT *
FROM `projectID.bucketID.inputdf` -insert df ID
WHERE (PosTime IS NOT NULL
AND Icao IS NOT NULL
AND Lat IS NOT NULL
AND Long IS NOT NULL
AND Alt IS NOT NULL
AND Lat != 0.0
AND Long != 0.0)
```

(2) Assuming that no aircraft traveled farther than the longest commercial aircraft distance known, data were selectively chosen within the pyspark script that was used to determine the total sum of distance traveled by aircraft. The longest distance traveled by commercial aircraft is currently Singapore Changi Airport to JFK International Airport (NY, USA), covering an average of 15,349 km. Relevant snippets of the filtering within the pyspark script are reproduced here, the complete script is filed separately.

For a read-in BigQuery table of `planes`, with the variable `dist` calculated from a haversine function, the `SUM(dist)` by `Icao` isolates the distance traveled by each aircraft by flight, and only distances less than the longest distance flight of 15349 km are added.

```
# determine if distance traveled is greater than the longest distance flight
# from Singapore to JFK (15349 km)
top = spark.sql("""
   SELECT Icao, SUM(dist) as dist
   FROM planes
   GROUP BY Icao
   HAVING SUM(dist) < 15349
   ORDER BY dist DESC
   LIMIT 10
""")
top = top.rdd.map(tuple)
pprint.pprint(top.collect())

# sum the distances for all planes, selecting only distances less than the maximum distance flight
miles = spark.sql("""
   SELECT SUM(dist)
   FROM (
      SELECT dist
      FROM planes
      WHERE dist < 15349
   ) AS tmp
""")
pprint.pprint(miles.collect())
```

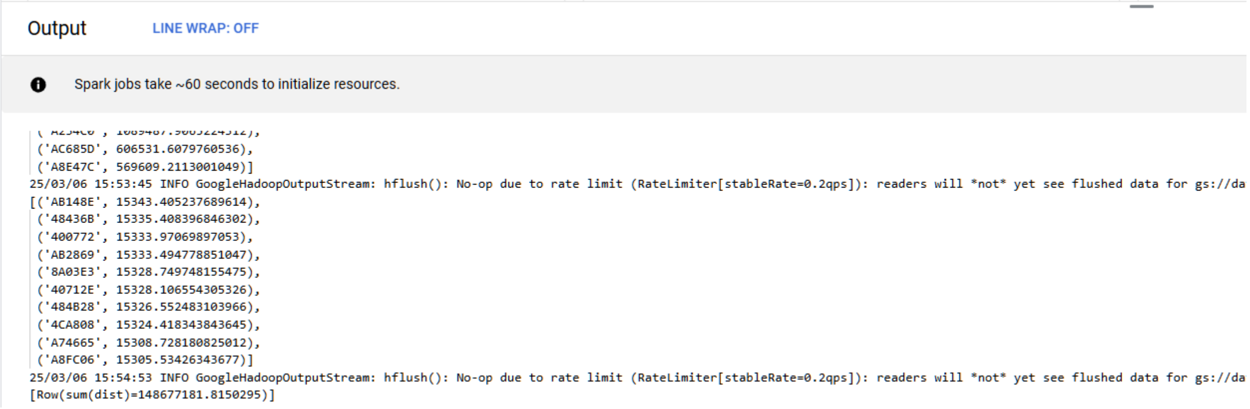
Explore:

This week's analysis did not require exploration, but the test output of the ten longest flights given here as proof of completion.

Output:

```
[('AB148E', 15343.405237689614),
 ('48436B', 15335.187286323555),
 ('400772', 15333.97069897053),
 ('AB2869', 15333.494778851047),
 ('8A03E3', 15328.749748155475),
 ('40712E', 15328.10720559783),
 ('484B28', 15326.552483103966),
 ('4CA808', 15324.418343843645),
 ('A74665', 15308.728180825012),
 ('A8FC06', 15305.53426343677)]

[Row(sum(dist)=148665089.38591665)] # team member A
AND
[Row(sum(dist)=148677181.8150295)] # team member B
```



```
 ('A234C0', 1089487.9003224312),
 ('AC685D', 606531.6079760536),
 ('A8E47C', 569609.2113001049)]
25/03/06 15:53:45 INFO GoogleHadoopOutputStream: hflush(): No-op due to rate limit (RateLimiter[stableRate=0.2qps]): readers will *not* yet see flushed data for gs://da
[('AB148E', 15343.405237689614),
 ('48436B', 15335.408396846302),
 ('400772', 15333.97069897053),
 ('AB2869', 15333.494778851047),
 ('8A03E3', 15328.749748155475),
 ('40712E', 15328.106554305326),
 ('484B28', 15326.552483103966),
 ('4CA808', 15324.418343843645),
 ('A74665', 15308.728180825012),
 ('A8FC06', 15305.53426343677)]
25/03/06 15:54:53 INFO GoogleHadoopOutputStream: hflush(): No-op due to rate limit (RateLimiter[stableRate=0.2qps]): readers will *not* yet see flushed data for gs://da
[Row(sum(dist)=148677181.8150295)]
```

Model:

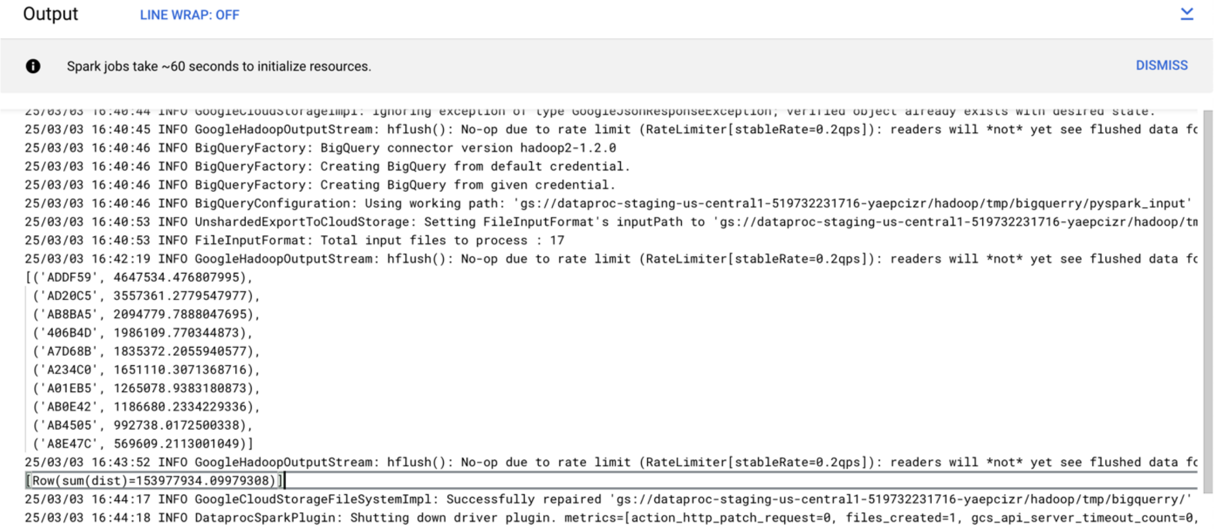
The model step in the OSEMN framework does not apply to this portion of the study.

Interpret:

The interpret step in the OSEMN framework does not apply to this portion of the study.

Obstacles Encountered in Work:

A first attempt at filtering the bad data was attempted within the haversine function, where an if statement was used to set any distances greater than the longest flight commercial distance (15,349 km) to zero. This did not impact the final flight total, and the longest 10 flights were far above the longest commercial flight (see below).

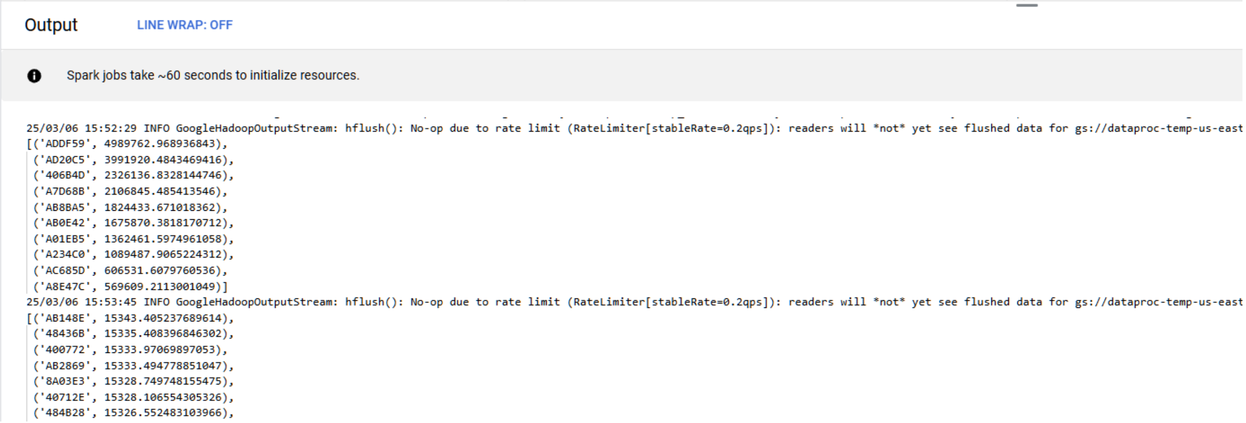


```
25/03/03 16:40:44 INFO GoogleCloudStorageImpl: Ignoring exception of type GoogleJsonResponseException; verified object already exists with desired state.
25/03/03 16:40:45 INFO GoogleHadoopOutputStream: hflush(): No-op due to rate limit (RateLimiter[stableRate=0.2qps]): readers will *not* yet see flushed data fc
25/03/03 16:40:46 INFO BigQueryFactory: BigQuery connector version hadoop2-1.2.0
25/03/03 16:40:46 INFO BigQueryFactory: Creating BigQuery from default credential.
25/03/03 16:40:46 INFO BigQueryFactory: Creating BigQuery from given credential.
25/03/03 16:40:46 INFO BigQueryConfiguration: Using working path: 'gs://dataproc-staging-us-central1-519732231716-yaepcizr/hadoop/tmp/bigquerry/pyspark_input'
25/03/03 16:40:53 INFO UnshardedExportToCloudStorage: Setting FileInputFormat's inputPath to 'gs://dataproc-staging-us-central1-519732231716-yaepcizr/hadoop/tm
25/03/03 16:40:53 INFO FileInputFormat: Total input files to process : 17
25/03/03 16:42:19 INFO GoogleHadoopOutputStream: hflush(): No-op due to rate limit (RateLimiter[stableRate=0.2qps]): readers will *not* yet see flushed data fc
[('ADDF59', 4647534.476807995),
 ('AD20C5', 3557361.2779547977),
 ('AB8BA5', 2094779.7888047695),
 ('406B4D', 1986109.770344873),
 ('A7D68B', 1835372.2055940577),
 ('A234C0', 1651110.3071368716),
 ('A01EB5', 1265078.9383180873),
 ('AB0E42', 1186680.2334229336),
 ('AB4505', 992738.0172500338),
 ('A8E47C', 569609.2113001049)]
25/03/03 16:43:52 INFO GoogleHadoopOutputStream: hflush(): No-op due to rate limit (RateLimiter[stableRate=0.2qps]): readers will *not* yet see flushed data fc
[Row(sum(dist)=153977934.09979308)]
25/03/03 16:44:17 INFO GoogleCloudStorageFileSystemImpl: Successfully repaired 'gs://dataproc-staging-us-central1-519732231716-yaepcizr/hadoop/tmp/bigquerry/'
25/03/03 16:44:18 INFO DataprocSparkPlugin: Shutting down driver plugin. metrics=[action_http_patch_request=0, files_created=1, gcs_api_server_timeout_count=0,
```

This was because the function was applied for each segment of a flight, not the whole flight. Once this was ascertained, the additional filtering was done within the summation part of the pyspark script. It took a number of iterations to obtain a functional SQL query for each response

– both the total summation and the longest 10 flights.

We left increased visibility for this in the final code, by adding a print line after the initial longest distance query (see below for output).



Spark jobs take ~60 seconds to initialize resources.

```
25/03/06 15:52:29 INFO GoogleHadoopOutputStream: hflush(): No-op due to rate limit (RateLimiter[stableRate=0.2qps]): readers will *not* yet see flushed data for gs://dataproc-temp-us-east
[('ADDF59', 4989762.968936843),
 ('AD20C5', 3991920.4843469416),
 ('406B4D', 2326136.8328144746),
 ('A7D68B', 2106845.485413546),
 ('AB8BA5', 1824433.671018362),
 ('AB0E42', 1675870.3818170712),
 ('A01EB5', 1362461.5974961058),
 ('A234C0', 1089487.9065224312),
 ('AC685D', 606531.6079760536),
 ('A8E47C', 569609.2113001049)]
25/03/06 15:53:45 INFO GoogleHadoopOutputStream: hflush(): No-op due to rate limit (RateLimiter[stableRate=0.2qps]): readers will *not* yet see flushed data for gs://dataproc-temp-us-east
[('AB148E', 15343.405237689614),
 ('48436B', 15335.408396846302),
 ('400772', 15333.97069897053),
 ('AB2869', 15333.494778851047),
 ('8A03E3', 15328.749748155475),
 ('40712E', 15328.106554305326),
 ('484B28', 15326.552483103966),
```

Distribution of Work:

Rachel and Paul consulted on difficulties along the way and built on prior weeks' joint efforts, completing the assignment as directed and co-writing this document. Completed work was submitted as a joint PDF and as a .py code file to the instructors.

References:

Wolford, J. (2025). Spark Plane Distances. CS512. Computer Science, Oregon State University.

Coding Sources:

Wolford, J. (2025). Starter code for Spark Plane Distances CS512. Department of Computer Science, Oregon State University.