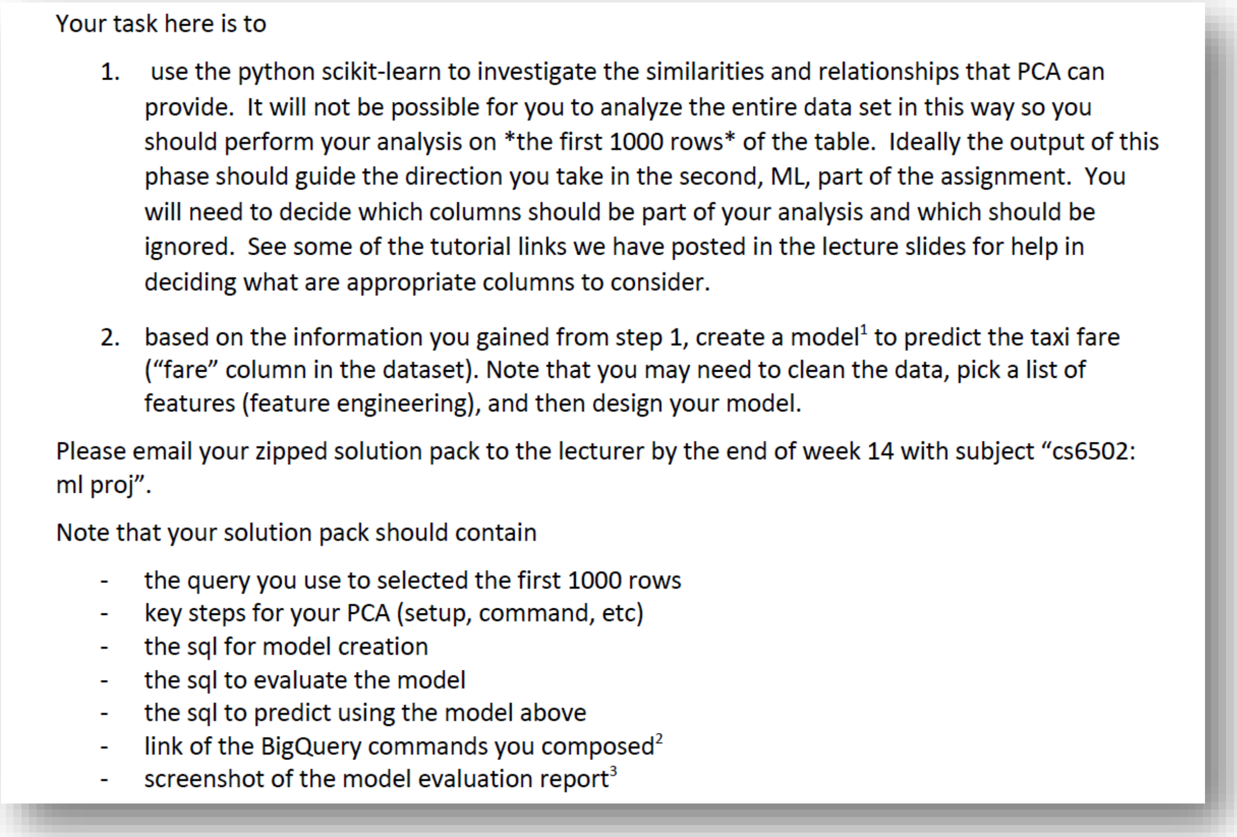
CS6502

Applied Big Data & Visualization

ML Project

KAILASH MURALIDHARAN 19116608



Your task here is to

1. use the python scikit-learn to investigate the similarities and relationships that PCA can provide. It will not be possible for you to analyze the entire data set in this way so you should perform your analysis on *the first 1000 rows* of the table. Ideally the output of this phase should guide the direction you take in the second, ML, part of the assignment. You will need to decide which columns should be part of your analysis and which should be ignored. See some of the tutorial links we have posted in the lecture slides for help in deciding what are appropriate columns to consider.

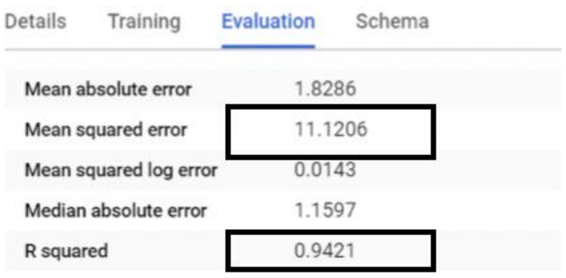
2. based on the information you gained from step 1, create a model[1] to predict the taxi fare ("fare" column in the dataset). Note that you may need to clean the data, pick a list of features (feature engineering), and then design your model.

Please email your zipped solution pack to the lecturer by the end of week 14 with subject "cs6502: ml proj".

Note that your solution pack should contain

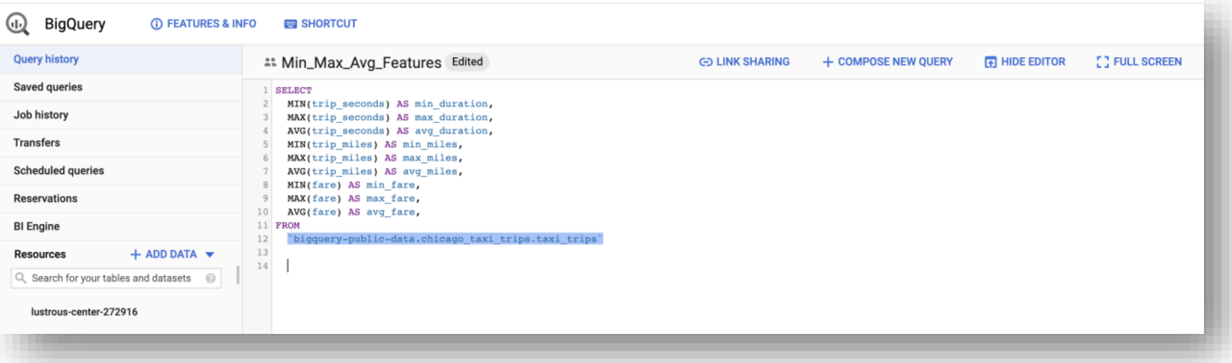
- the query you use to selected the first 1000 rows
- key steps for your PCA (setup, command, etc)
- the sql for model creation
- the sql to evaluate the model
- the sql to predict using the model above
- link of the BigQuery commands you composed[2]
- screenshot of the model evaluation report[3]

Summary

	Item	Links
1.	Query used to select 1000 rows	https://console.cloud.google.com/bigquery?sq=3103539222:ed4b7d29551040a1a3441a2b2cf59031
2.	Results of the query containing 1000 rows	https://docs.google.com/spreadsheets/d/1xc51HUUrw51UBCwIfbpPT343E8GyODNaQc2khHywu30/edit?ts=5eb593ba#gid=2010390566
3.	Query used to create the Final Model	https://console.cloud.google.com/bigquery?sq=3103539222:88ef2a11074c451db05d9e1598f90d55
4.	Query used for Model Evaluation	https://console.cloud.google.com/bigquery?sq=3103539222:4c44667fd77d489e8a8164b3a02b843c
5.	Query used for Model Prediction	https://console.cloud.google.com/bigquery?sq=3103539222:4be462936e3c40eba4f80b02cc69c4b4
6.	Final Model Evaluation Report	 Mean absolute error 1.8286 Mean squared error 11.1206 Mean squared log error 0.0143 Median absolute error 1.1597 R squared 0.9421

1. Query to select 1000 rows

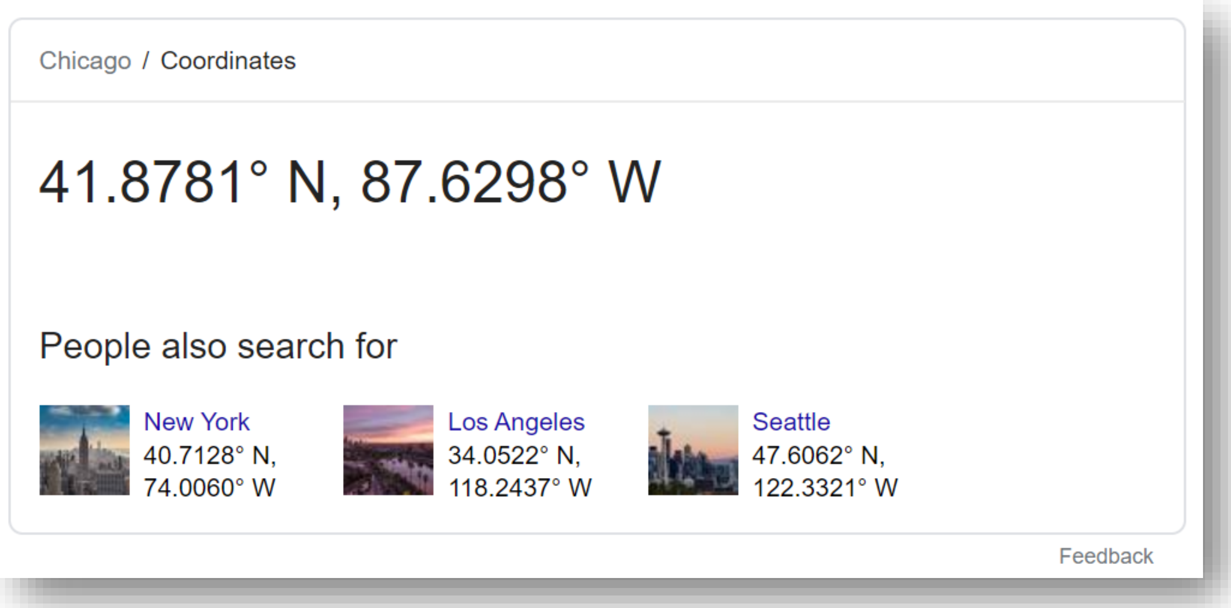
I have used a query to check for the minimum, maximum and average of three numeric columns.



```sql
SELECT
  MIN(trip_seconds) AS min_duration,
  MAX(trip_seconds) AS max_duration,
  AVG(trip_seconds) AS avg_duration,
  MIN(trip_miles) AS min_miles,
  MAX(trip_miles) AS max_miles,
  AVG(trip_miles) AS avg_miles,
  MIN(fare) AS min_fare,
  MAX(fare) AS max_fare,
  AVG(fare) AS avg_fare,
FROM
  `bigquery-public-data.chicago_taxi_trips.taxi_trips`

  |
```

Using the latitude and longitude values of Chicago, to consider only those trips that started and ended inside the city range.


Chicago / Coordinates

41.8781° N, 87.6298° W

People also search for

New York
40.7128° N,
74.0060° W

Los Angeles
34.0522° N,
118.2437° W

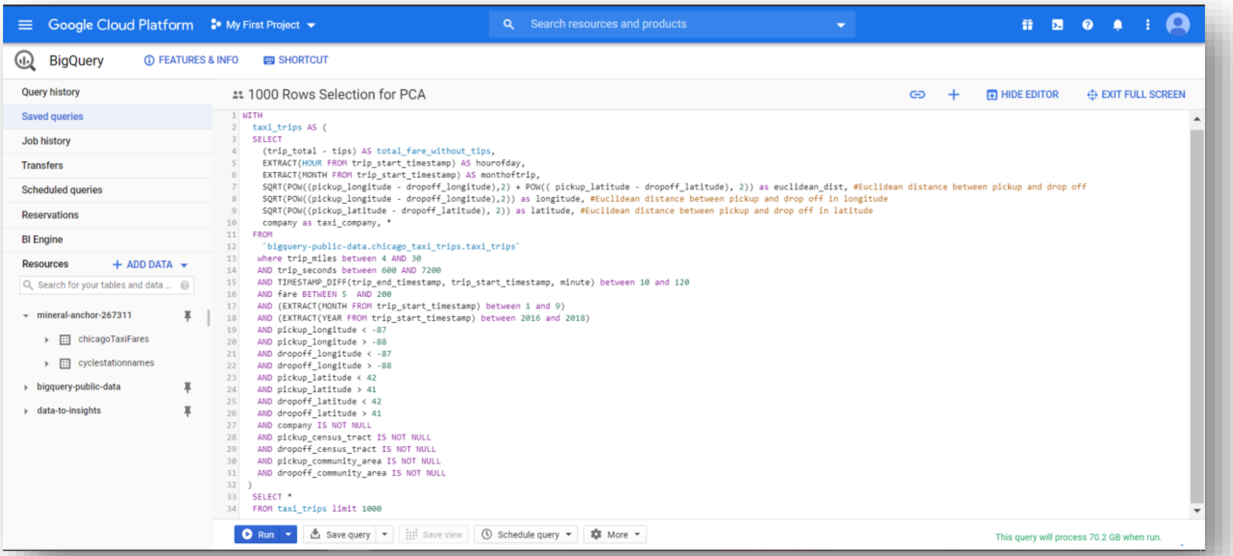
Seattle
47.6062° N,
122.3321° W

Feedback

<u>Conditions applied</u>:

- Picked values to apply for where conditions based on the min, max, avg summary and Chicago's location coordinates.
- Years filtered between 2016 and 2018, and months between January and September.
- Rest of the columns – NOT NULL condition
- In addition to the existing columns, following additional columns have been created
    - **total_fare_without_tips** (the value to be predicted), since the tips amount varies from customer to customer (created a column by subtracting tips from the trip_total)

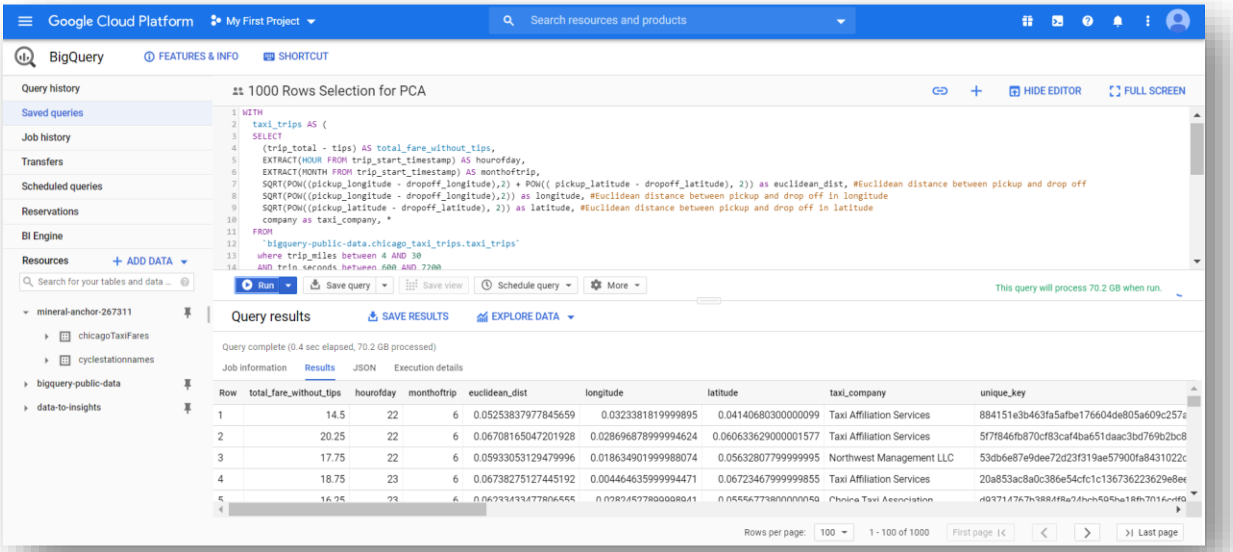
- **hour_of_day**, from the timestamp
- **month_of_trip**, also from the timestamp
- **euclidean_distance**, the distance between the pickup and dropoff points
- **longitude,** the distance between the pickup and dropoff points longitude
- **latitude,** the distance between the pickup and dropoff points latitude
- **taxi_company**, the name of the taxi company



1000 Rows Selection for PCA

```sql
WITH
  taxi_trips AS (
  SELECT
    (trip_total - tips) AS total_fare_without_tips,
    EXTRACT(HOUR FROM trip_start_timestamp) AS hourofday,
    EXTRACT(MONTH FROM trip_start_timestamp) AS monthoftrip,
    SQRT(POW((pickup_longitude - dropoff_longitude),2) + POW(( pickup_latitude - dropoff_latitude), 2)) as euclidean_dist, #Euclidean distance between pickup and drop off
    SQRT(POW((pickup_longitude - dropoff_longitude),2)) as longitude, #Euclidean distance between pickup and drop off in longitude
    SQRT(POW((pickup_latitude - dropoff_latitude), 2)) as latitude, #Euclidean distance between pickup and drop off in latitude
    company as taxi_company, *
  FROM
    `bigquery-public-data.chicago_taxi_trips.taxi_trips`
   where trip_miles between 4 AND 30
   AND trip_seconds between 600 AND 7200
   AND TIMESTAMP_DIFF(trip_end_timestamp, trip_start_timestamp, minute) between 10 and 120
   AND fare BETWEEN 5  AND 200
   AND (EXTRACT(MONTH FROM trip_start_timestamp) between 1 and 9)
   AND (EXTRACT(YEAR FROM trip_start_timestamp) between 2016 and 2018)
   AND pickup_longitude < -87
   AND pickup_longitude > -88
   AND dropoff_longitude < -87
   AND dropoff_longitude > -88
   AND pickup_latitude < 42
   AND pickup_latitude > 41
   AND dropoff_latitude < 42
   AND dropoff_latitude > 41
   AND company IS NOT NULL
   AND pickup_census_tract IS NOT NULL
   AND dropoff_census_tract IS NOT NULL
   AND pickup_community_area IS NOT NULL
   AND dropoff_community_area IS NOT NULL
 )
  SELECT *
  FROM taxi_trips limit 1000
```

This query will process 70.2 GB when run.



1000 Rows Selection for PCA

```sql
WITH
  taxi_trips AS (
  SELECT
    (trip_total - tips) AS total_fare_without_tips,
    EXTRACT(HOUR FROM trip_start_timestamp) AS hourofday,
    EXTRACT(MONTH FROM trip_start_timestamp) AS monthoftrip,
    SQRT(POW((pickup_longitude - dropoff_longitude),2) + POW(( pickup_latitude - dropoff_latitude), 2)) as euclidean_dist, #Euclidean distance between pickup and drop off
    SQRT(POW((pickup_longitude - dropoff_longitude),2)) as longitude, #Euclidean distance between pickup and drop off in longitude
    SQRT(POW((pickup_latitude - dropoff_latitude), 2)) as latitude, #Euclidean distance between pickup and drop off in latitude
    company as taxi_company, *
  FROM
    `bigquery-public-data.chicago_taxi_trips.taxi_trips`
   where trip_miles between 4 AND 30
   AND trip_seconds between 600 AND 7200
```

This query will process 70.2 GB when run.

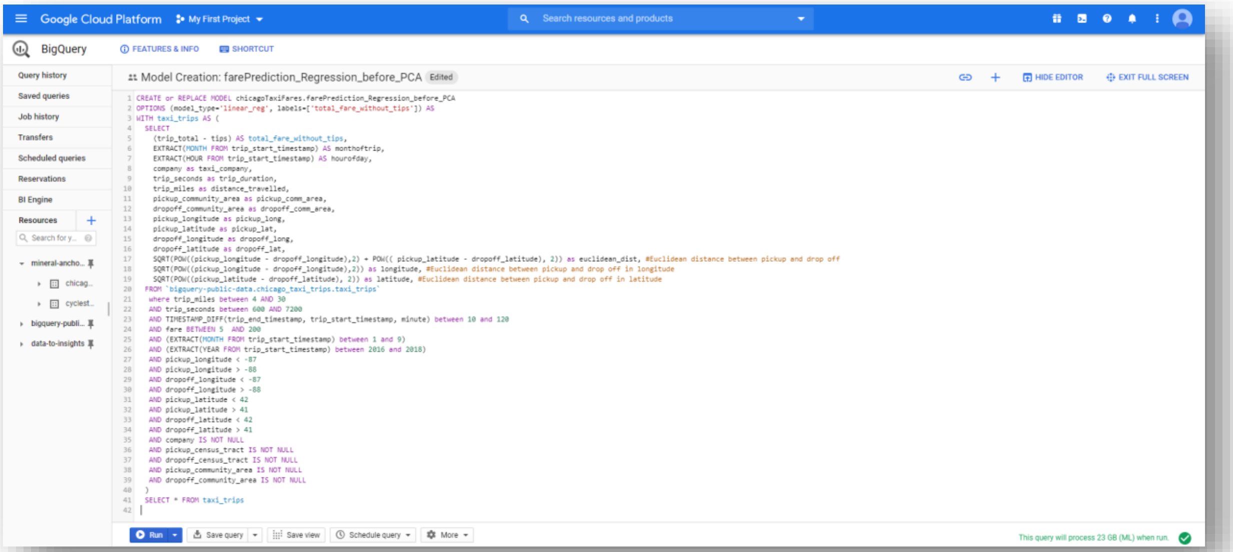
Query results

Query complete (0.4 sec elapsed, 70.2 GB processed)

Row	total_fare_without_tips	hourofday	monthoftrip	euclidean_dist	longitude	latitude	taxi_company	unique_key
1	14.5	22	6	0.05253837977845659	0.0323381819999895	0.04140680300000099	Taxi Affiliation Services	884151e3b463fa5afbe176604de805a609c257a
2	20.25	22	6	0.06708165047201928	0.028696878999994624	0.060633629000001577	Taxi Affiliation Services	5f7f846fb870cf83caf4ba651daac3bd769b2bc8
3	17.75	22	6	0.05933053129479996	0.018634901999988074	0.05632807799999995	Northwest Management LLC	53db6e87e9dee72d23f319ae57900fa8431022c
4	18.75	23	6	0.06738275127445192	0.004464635999994471	0.06723467999999855	Taxi Affiliation Services	20a853ac8a0c386e54cfc1c136736223629e8ee
5	16.25	23	6	0.06233433477806555	0.02824527899998941	0.05556773800000059	Choice Taxi Association	d93714767b3884f8e24bcb595be18fb7016cdf9

Rows per page: 100 1 - 100 of 1000 First page Last page

2. Base Model before PCA

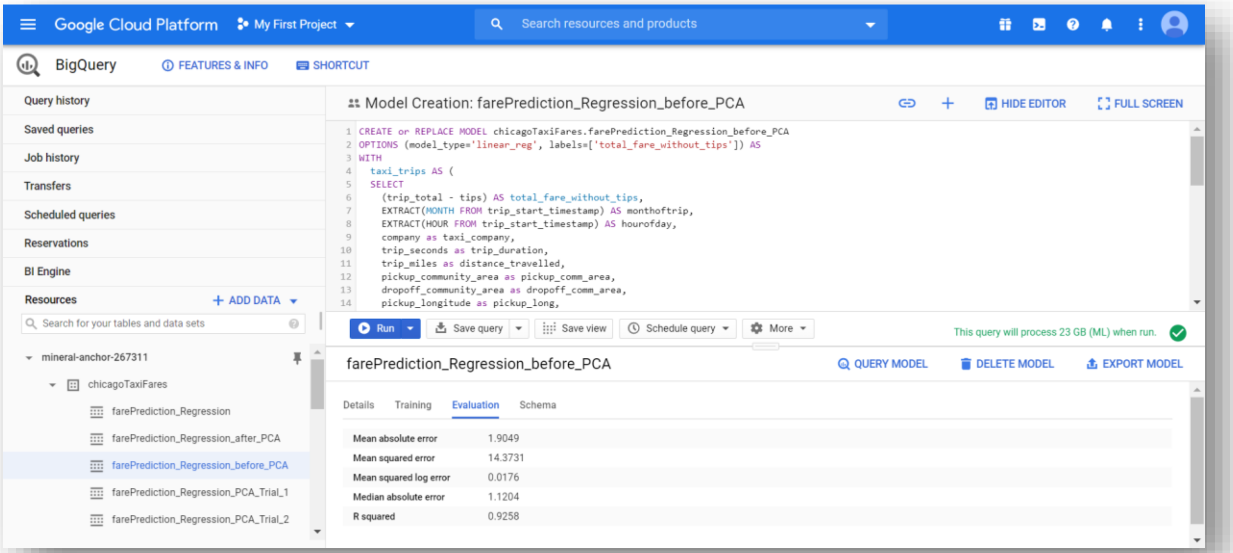


Model Creation: farePrediction_Regression_before_PCA

```sql
CREATE or REPLACE MODEL chicagoTaxiFares.farePrediction_Regression_before_PCA
OPTIONS (model_type='linear_reg', labels=['total_fare_without_tips']) AS
WITH taxi_trips AS (
  SELECT
    (trip_total - tips) AS total_fare_without_tips,
    EXTRACT(MONTH FROM trip_start_timestamp) AS monthoftrip,
    EXTRACT(HOUR FROM trip_start_timestamp) AS hourofday,
    company as taxi_company,
    trip_seconds as trip_duration,
    trip_miles as distance_travelled,
    pickup_community_area as pickup_comm_area,
    dropoff_community_area as dropoff_comm_area,
    pickup_longitude as pickup_long,
    pickup_latitude as pickup_lat,
    dropoff_longitude as dropoff_long,
    dropoff_latitude as dropoff_lat,
    SQRT(POW((pickup_longitude - dropoff_longitude),2) + POW(( pickup_latitude - dropoff_latitude), 2)) as euclidean_dist, #Euclidean distance between pickup and drop off
    SQRT(POW((pickup_longitude - dropoff_longitude),2)) as longitude, #Euclidean distance between pickup and drop off in longitude
    SQRT(POW((pickup_latitude - dropoff_latitude), 2)) as latitude, #Euclidean distance between pickup and drop off in latitude
  FROM `bigquery-public-data.chicago_taxi_trips.taxi_trips`
   where trip_miles between 4 AND 30
   AND trip_seconds between 600 AND 7200
   AND TIMESTAMP_DIFF(trip_end_timestamp, trip_start_timestamp, minute) between 10 and 120
   AND fare BETWEEN 5  AND 200
   AND (EXTRACT(MONTH FROM trip_start_timestamp) between 1 and 9)
   AND (EXTRACT(YEAR FROM trip_start_timestamp) between 2016 and 2018)
   AND pickup_longitude < -87
   AND pickup_longitude > -88
   AND dropoff_longitude < -87
   AND dropoff_longitude > -88
   AND pickup_latitude < 42
   AND pickup_latitude > 41
   AND dropoff_latitude < 42
   AND dropoff_latitude > 41
   AND company IS NOT NULL
   AND pickup_census_tract IS NOT NULL
   AND dropoff_census_tract IS NOT NULL
   AND pickup_community_area IS NOT NULL
   AND dropoff_community_area IS NOT NULL
  )
  SELECT * FROM taxi_trips
```

This query will process 23 GB (ML) when run.

Evaluation Report:



Model Creation: farePrediction_Regression_before_PCA

```sql
CREATE or REPLACE MODEL chicagoTaxiFares.farePrediction_Regression_before_PCA
OPTIONS (model_type='linear_reg', labels=['total_fare_without_tips']) AS
WITH
  taxi_trips AS (
  SELECT
    (trip_total - tips) AS total_fare_without_tips,
    EXTRACT(MONTH FROM trip_start_timestamp) AS monthoftrip,
    EXTRACT(HOUR FROM trip_start_timestamp) AS hourofday,
    company as taxi_company,
    trip_seconds as trip_duration,
    trip_miles as distance_travelled,
    pickup_community_area as pickup_comm_area,
    dropoff_community_area as dropoff_comm_area,
    pickup_longitude as pickup_long,
```

This query will process 23 GB (ML) when run.

farePrediction_Regression_before_PCA

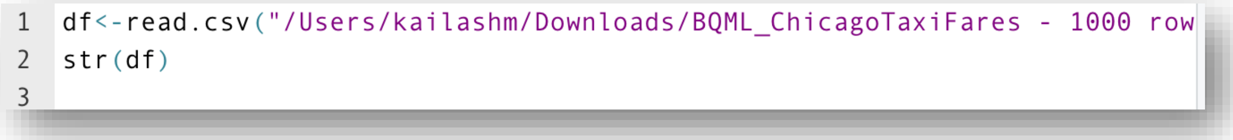
Details | Training | Evaluation | Schema

Mean absolute error	1.9049
Mean squared error	14.3731
Mean squared log error	0.0176
Median absolute error	1.1204
R squared	0.9258

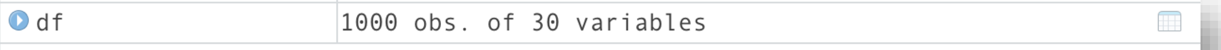
- MSE= **14.3731**
- **R squared value** = **0.9258**, about **92.58**% of the variability in the dependent variable is explained by our model.

3. PCA using R statistical package

- Load the dataset into the dataframe df

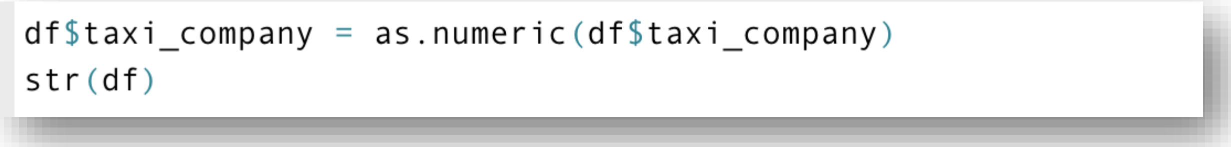


```
df<-read.csv("/Users/kailashm/Downloads/BQML_ChicagoTaxiFares - 1000 row
str(df)
```

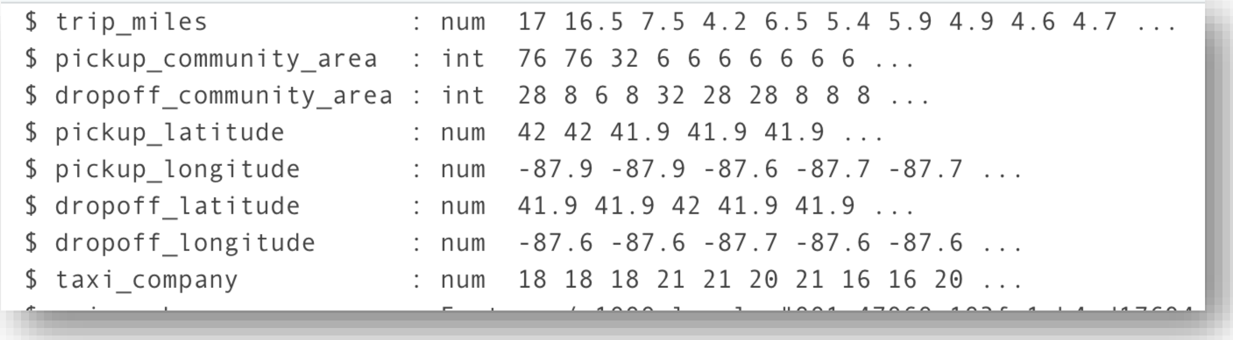


```
df                         1000 obs. of 30 variables
```

- The taxi_company column is changed to be of numeric data type (to be included in creating the model)

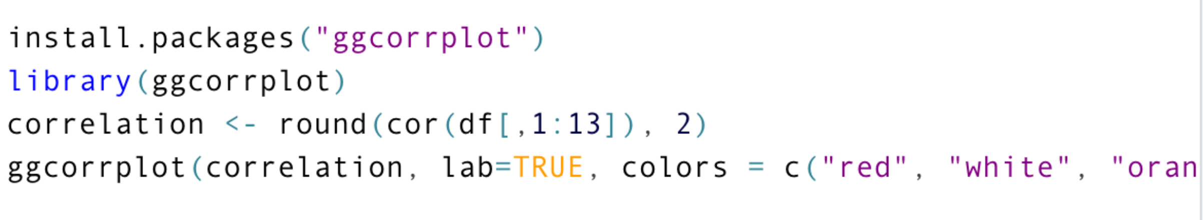


```
df$taxi_company = as.numeric(df$taxi_company)
str(df)
```



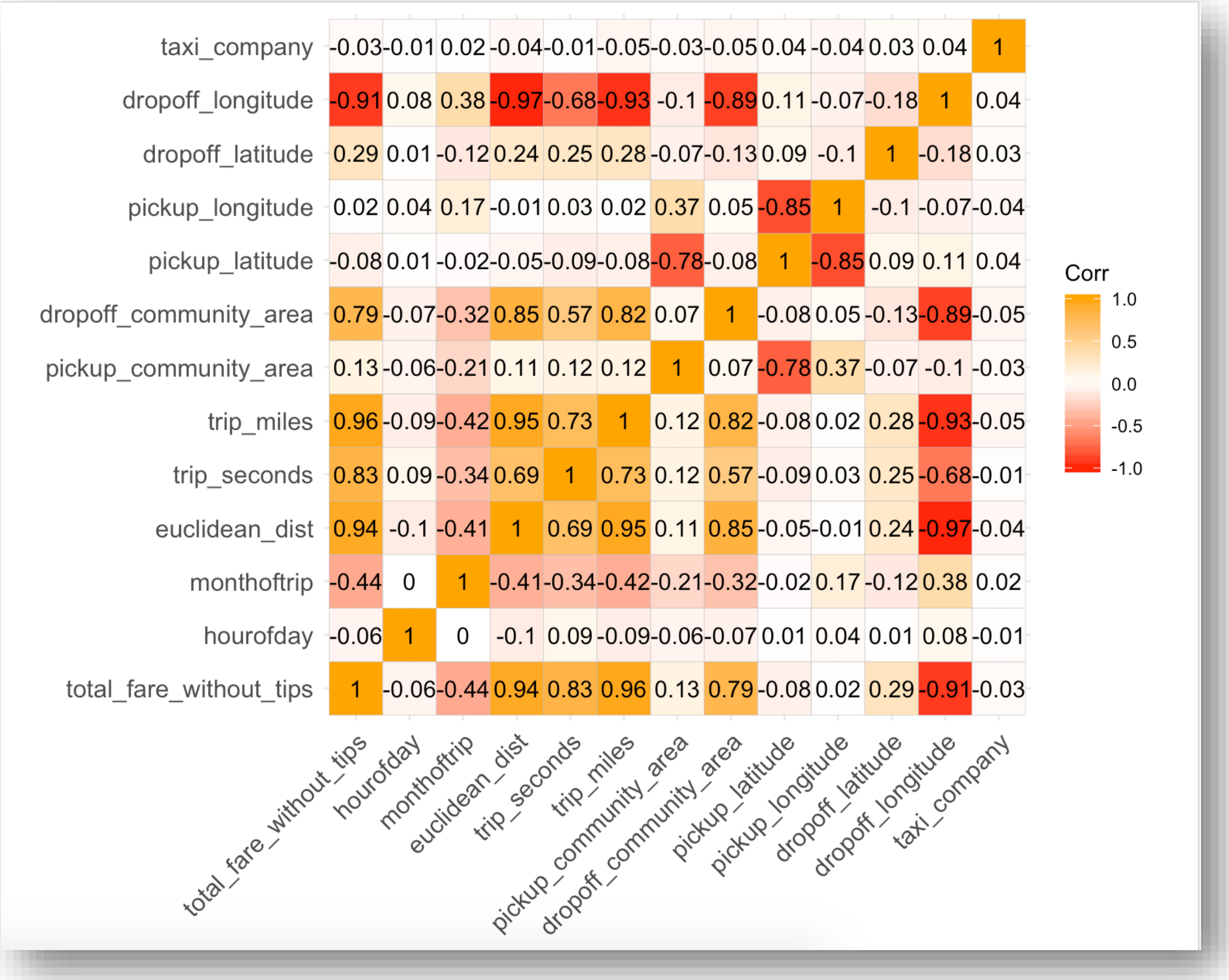
```
$ trip_miles             : num  17 16.5 7.5 4.2 6.5 5.4 5.9 4.9 4.6 4.7 ...
$ pickup_community_area  : int  76 76 32 6 6 6 6 6 6 6 ...
$ dropoff_community_area : int  28 8 6 8 32 28 28 8 8 8 ...
$ pickup_latitude        : num  42 42 41.9 41.9 41.9 ...
$ pickup_longitude       : num  -87.9 -87.9 -87.6 -87.7 -87.7 ...
$ dropoff_latitude       : num  41.9 41.9 42 41.9 41.9 ...
$ dropoff_longitude      : num  -87.6 -87.6 -87.7 -87.6 -87.6 ...
$ taxi_company           : num  18 18 18 21 21 20 21 16 16 20 ...
```

- Correlation Matrix



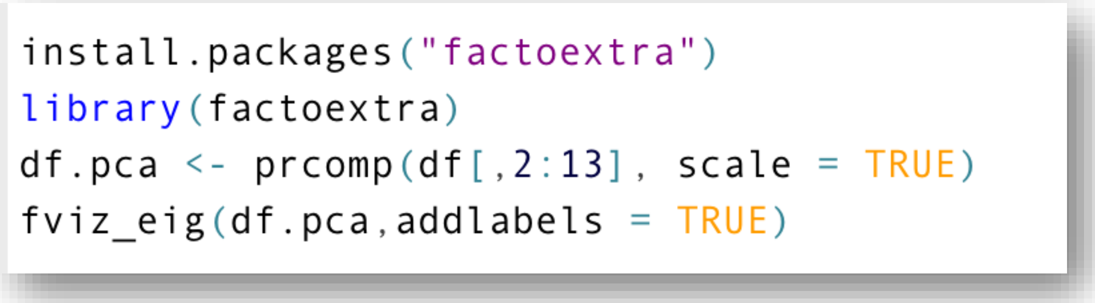
```
install.packages("ggcorrplot")
library(ggcorrplot)
correlation <- round(cor(df[,1:13]), 2)
ggcorrplot(correlation, lab=TRUE, colors = c("red", "white", "oran
```

The below matrix shows the correlation between the numeric variables in the dataset.



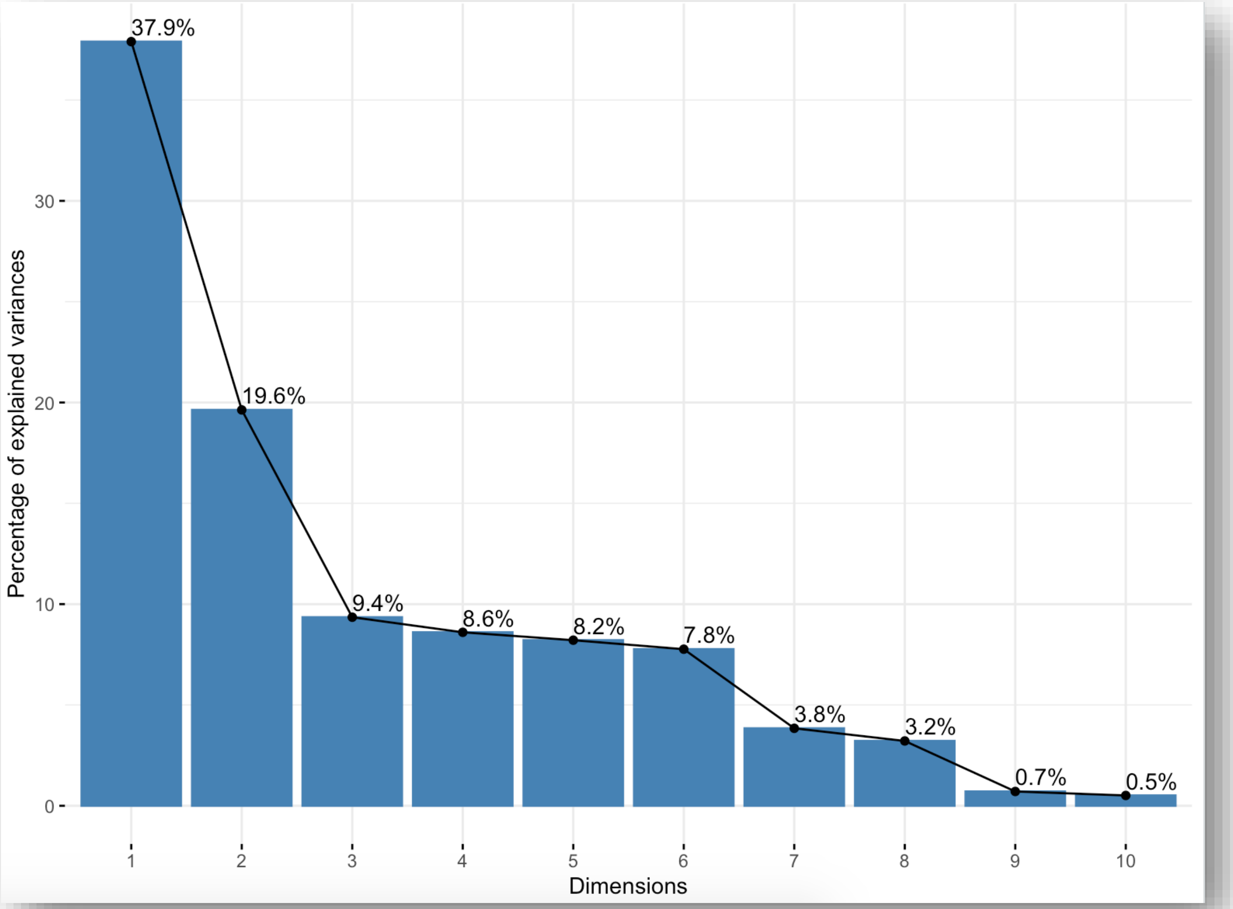
	total_fare_without_tips	hourofday	monthoftrip	euclidean_dist	trip_seconds	trip_miles	pickup_community_area	dropoff_community_area	pickup_latitude	pickup_longitude	dropoff_latitude	dropoff_longitude	taxi_company
taxi_company	-0.03	-0.01	0.02	-0.04	-0.01	-0.05	-0.03	-0.05	0.04	-0.04	0.03	0.04	1
dropoff_longitude	-0.91	0.08	0.38	-0.97	-0.68	-0.93	-0.1	-0.89	0.11	-0.07	-0.18	1	0.04
dropoff_latitude	0.29	0.01	-0.12	0.24	0.25	0.28	-0.07	-0.13	0.09	-0.1	1	-0.18	0.03
pickup_longitude	0.02	0.04	0.17	-0.01	0.03	0.02	0.37	0.05	-0.85	1	-0.1	-0.07	-0.04
pickup_latitude	-0.08	0.01	-0.02	-0.05	-0.09	-0.08	-0.78	-0.08	1	-0.85	0.09	0.11	0.04
dropoff_community_area	0.79	-0.07	-0.32	0.85	0.57	0.82	0.07	1	-0.08	0.05	-0.13	-0.89	-0.05
pickup_community_area	0.13	-0.06	-0.21	0.11	0.12	0.12	1	0.07	-0.78	0.37	-0.07	-0.1	-0.03
trip_miles	0.96	-0.09	-0.42	0.95	0.73	1	0.12	0.82	-0.08	0.02	0.28	-0.93	-0.05
trip_seconds	0.83	0.09	-0.34	0.69	1	0.73	0.12	0.57	-0.09	0.03	0.25	-0.68	-0.01
euclidean_dist	0.94	-0.1	-0.41	1	0.69	0.95	0.11	0.85	-0.05	-0.01	0.24	-0.97	-0.04
monthoftrip	-0.44	0	1	-0.41	-0.34	-0.42	-0.21	-0.32	-0.02	0.17	-0.12	0.38	0.02
hourofday	-0.06	1	0	-0.1	0.09	-0.09	-0.06	-0.07	0.01	0.04	0.01	0.08	-0.01
total_fare_without_tips	1	-0.06	-0.44	0.94	0.83	0.96	0.13	0.79	-0.08	0.02	0.29	-0.91	-0.03

- Computing PCA using prcomp()

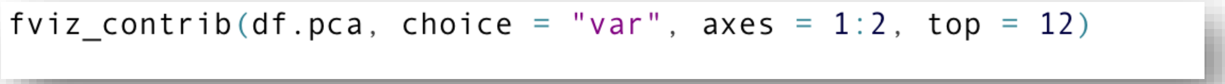


```
install.packages("factoextra")
library(factoextra)
df.pca <- prcomp(df[,2:13], scale = TRUE)
fviz_eig(df.pca,addlabels = TRUE)
```

This plot shows the percent contribution of the principle components. We can see that the first two principle components explain about **57.5%** variation.

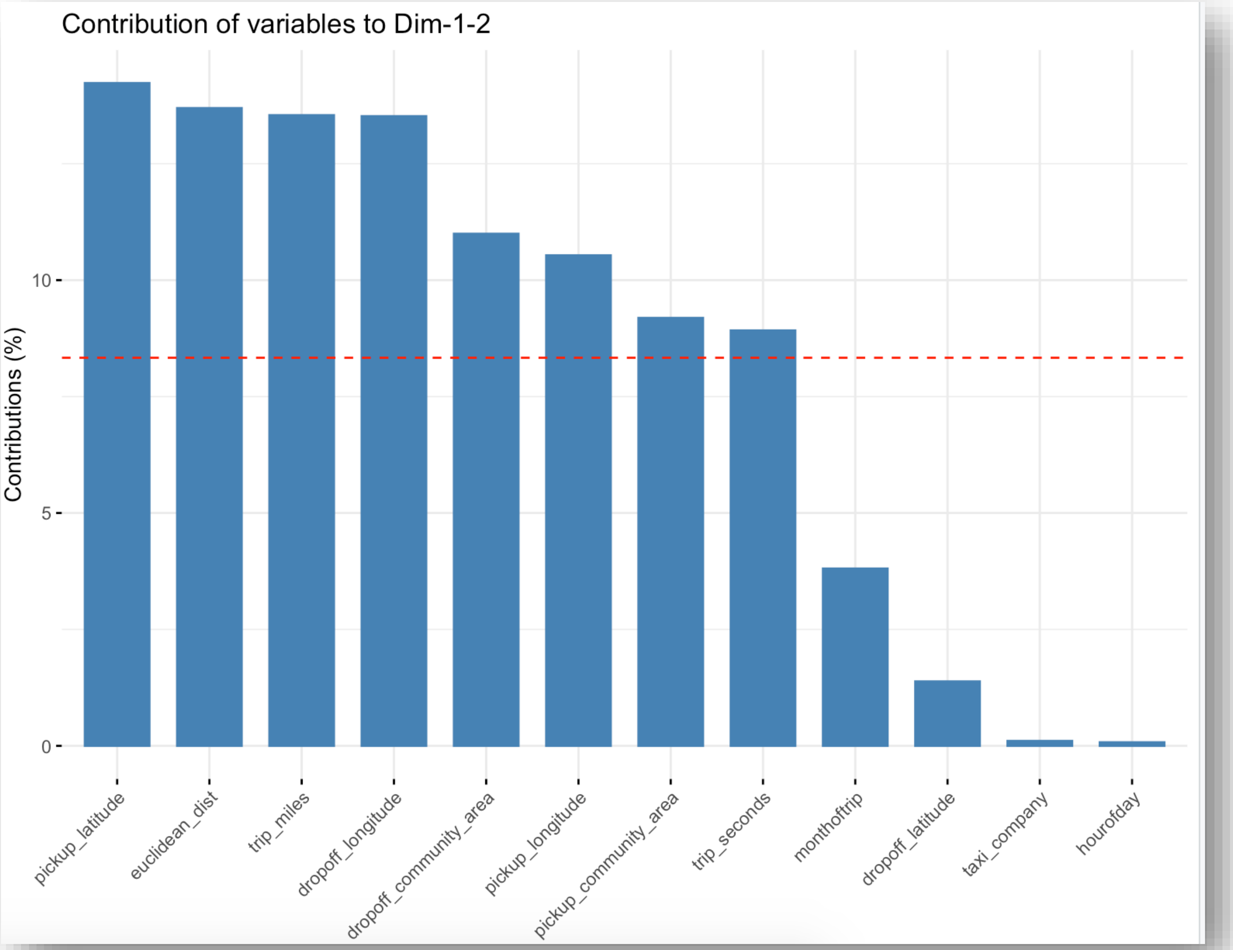


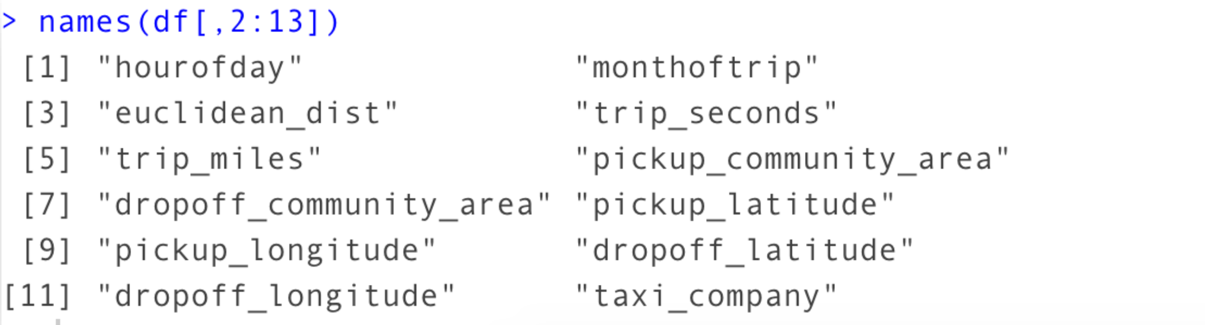
- We visualize the first two principle components to check the **% contribution of each variable** (features from the original dataset) **in both the principle components**.



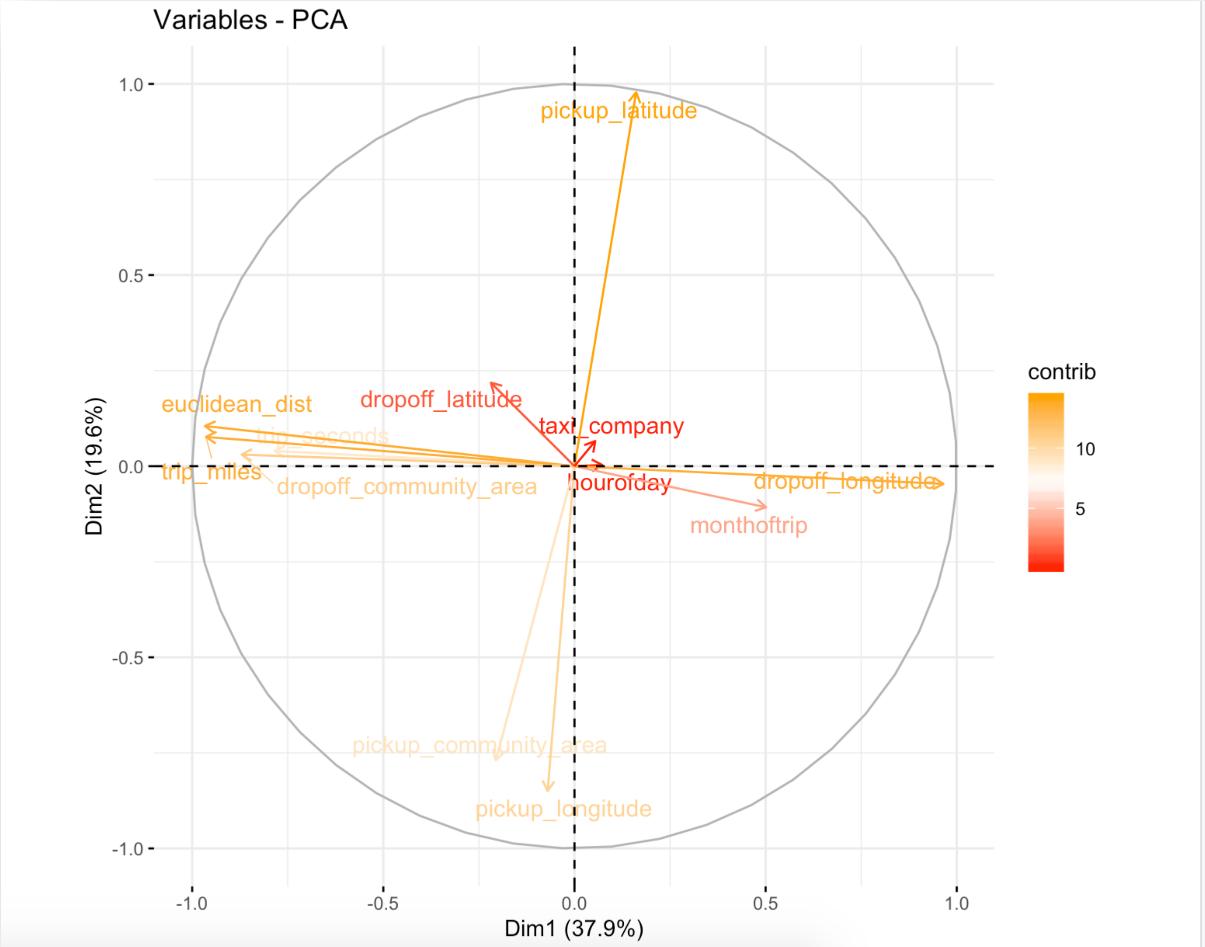
```
fviz_contrib(df.pca, choice = "var", axes = 1:2, top = 12)
```

Output Screenshot:

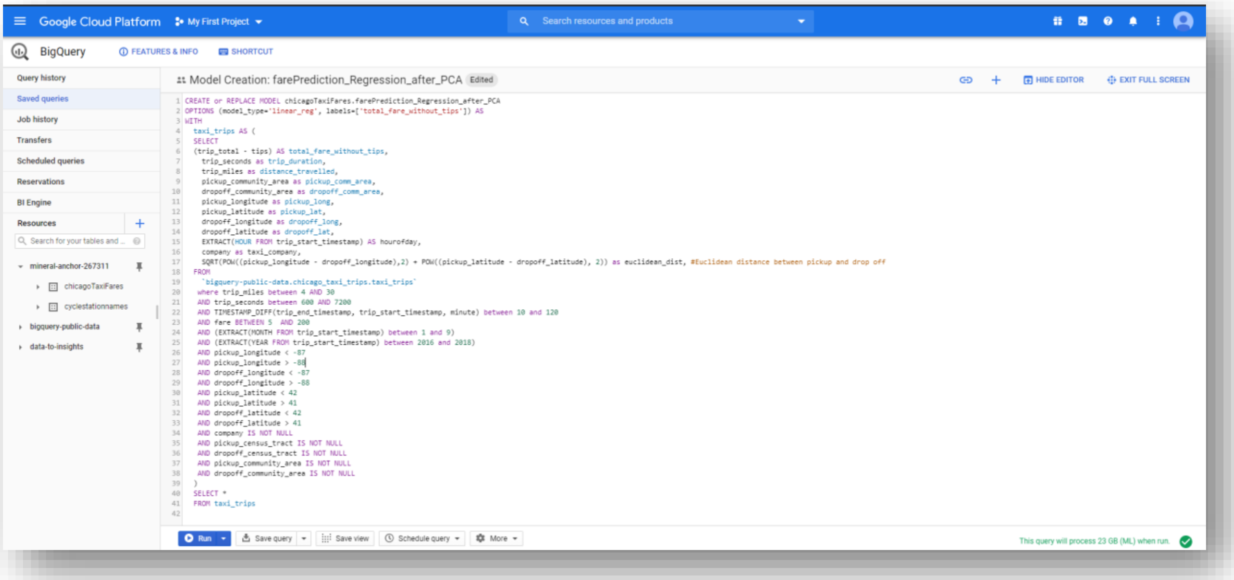




```
> names(df[,2:13])
 [1] "hourofday"              "monthoftrip"
 [3] "euclidean_dist"         "trip_seconds"
 [5] "trip_miles"             "pickup_community_area"
 [7] "dropoff_community_area" "pickup_latitude"
 [9] "pickup_longitude"       "dropoff_latitude"
[11] "dropoff_longitude"      "taxi_company"
```


Variables - PCA
pickup_latitude
euclidean_dist
dropoff_latitude
trip_seconds
taxi_company
trip_miles
dropoff_community_area
hourofday
dropoff_longitude
monthoftrip
pickup_community_area
pickup_longitude
Dim2 (19.6%)
Dim1 (37.9%)
1.0
0.5
0.0
-0.5
-1.0
-1.0
-0.5
0.0
0.5
1.0
contrib
10
5

4. Final Model after PCA and feature selection

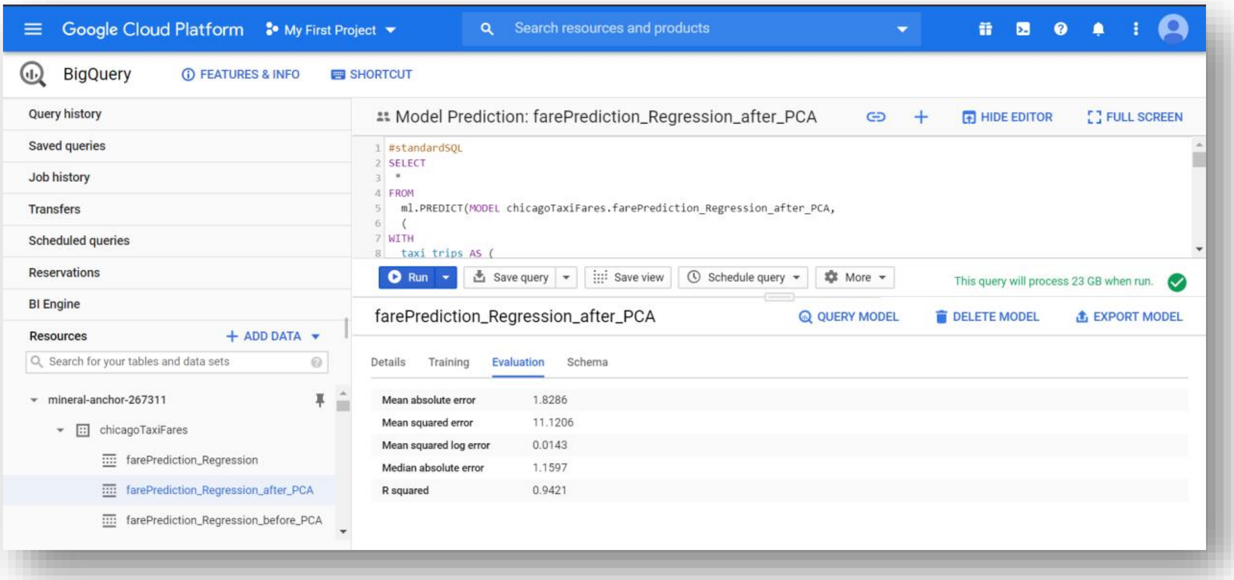


Model Creation: farePrediction_Regression_after_PCA Edited

```
CREATE or REPLACE MODEL chicagoTaxiFares.farePrediction_Regression_after_PCA
OPTIONS (model_type='linear_reg', labels=['total_fare_without_tips']) AS
WITH
  taxi_trips AS (
  SELECT
  (trip_total - tips) AS total_fare_without_tips,
    trip_seconds as trip_duration,
    trip_miles as distance_travelled,
    pickup_community_area as pickup_comm_area,
    dropoff_community_area as dropoff_comm_area,
    pickup_longitude as pickup_long,
    pickup_latitude as pickup_lat,
    dropoff_longitude as dropoff_long,
    dropoff_latitude as dropoff_lat,
    EXTRACT(HOUR FROM trip_start_timestamp) AS hourofday,
    company as taxi_company,
    SQRT(POW((pickup_longitude - dropoff_longitude),2) + POW((pickup_latitude - dropoff_latitude), 2)) as euclidean_dist, #Euclidean distance between pickup and drop off
  FROM
    `bigquery-public-data.chicago_taxi_trips.taxi_trips`
   where trip_miles between 4 AND 30
   AND trip_seconds between 600 AND 7200
   AND TIMESTAMP_DIFF(trip_end_timestamp, trip_start_timestamp, minute) between 10 and 120
   AND fare BETWEEN 5  AND 200
   AND (EXTRACT(MONTH FROM trip_start_timestamp) between 1 and 9)
   AND (EXTRACT(YEAR FROM trip_start_timestamp) between 2016 and 2018)
   AND pickup_longitude < -87
   AND pickup_longitude > -88
   AND dropoff_longitude < -87
   AND dropoff_longitude > -88
   AND pickup_latitude < 42
   AND pickup_latitude > 41
   AND dropoff_latitude < 42
   AND dropoff_latitude > 41
   AND company IS NOT NULL
   AND pickup_census_tract IS NOT NULL
   AND dropoff_census_tract IS NOT NULL
   AND pickup_community_area IS NOT NULL
   AND dropoff_community_area IS NOT NULL
  )
  SELECT *
  FROM taxi_trips
```

This query will process 23 GB (ML) when run.

<u>Final Prediction Model - Evaluation Report</u>:



Model Prediction: farePrediction_Regression_after_PCA

```
#standardSQL
SELECT
  *
FROM
  ml.PREDICT(MODEL chicagoTaxiFares.farePrediction_Regression_after_PCA,
  (
  WITH
    taxi_trips AS (
```

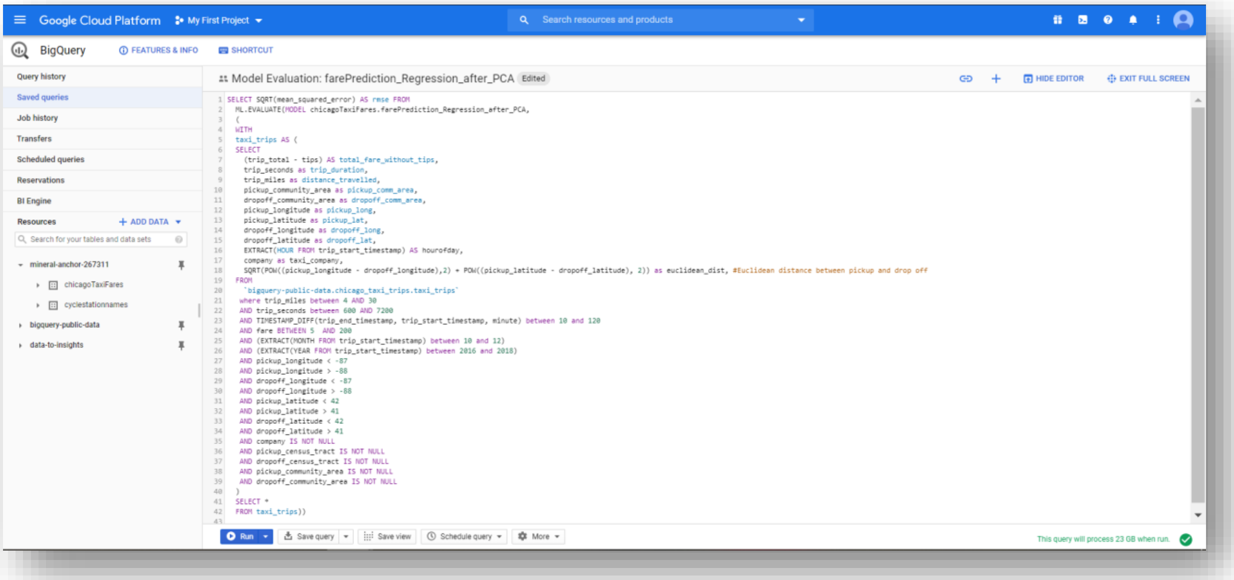
This query will process 23 GB when run.

farePrediction_Regression_after_PCA

Mean absolute error	1.8286
Mean squared error	11.1206
Mean squared log error	0.0143
Median absolute error	1.1597
R squared	0.9421

- The MSE of the prediction model **after PCA,** is **11.1206**, **which is less than** that of the **base model MSE = 14.3731**
- The **R squared value** of the prediction model has **increased** from **0.9258** to **0.9421**, which suggests that the final model captures more variation in the **total_fare_without_tips** than the base model

5. Final Model - Evaluation Query

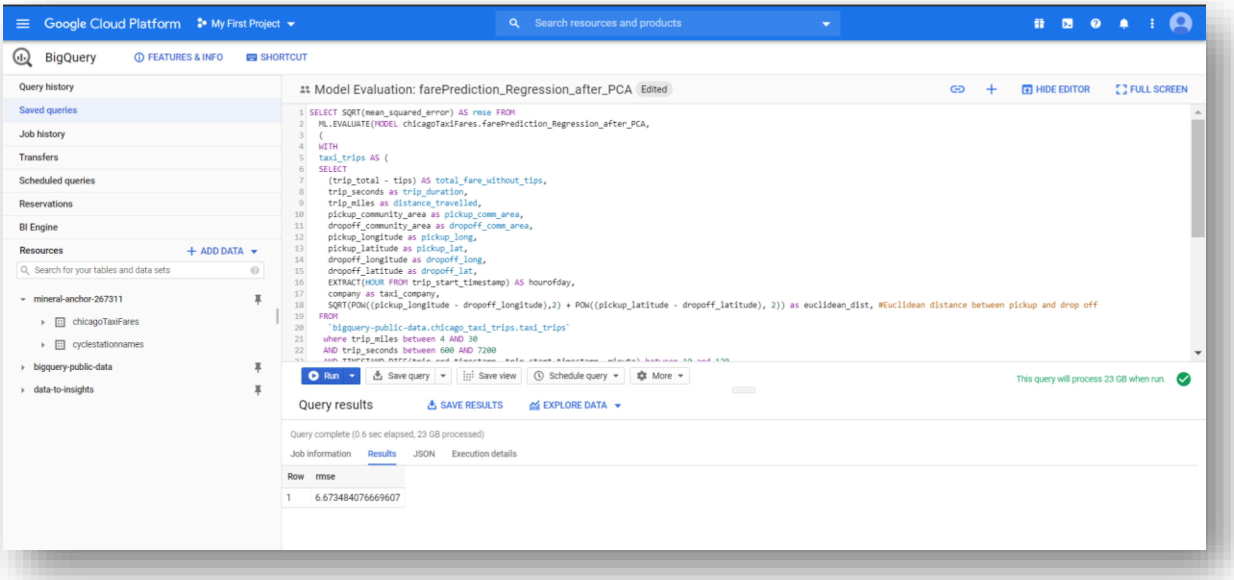


Model Evaluation: farePrediction_Regression_after_PCA Edited

```sql
SELECT SQRT(mean_squared_error) AS rmse FROM
 ML.EVALUATE(MODEL chicagoTaxiFares.farePrediction_Regression_after_PCA,
 (
 WITH
 taxi_trips AS (
 SELECT
   (trip_total - tips) AS total_fare_without_tips,
   trip_seconds as trip_duration,
   trip_miles as distance_travelled,
   pickup_community_area as pickup_comm_area,
   dropoff_community_area as dropoff_comm_area,
   pickup_longitude as pickup_long,
   pickup_latitude as pickup_lat,
   dropoff_longitude as dropoff_long,
   dropoff_latitude as dropoff_lat,
   EXTRACT(HOUR FROM trip_start_timestamp) AS hourofday,
   company as taxi_company,
   SQRT(POW((pickup_longitude - dropoff_longitude),2) + POW((pickup_latitude - dropoff_latitude), 2)) as euclidean_dist, #Euclidean distance between pickup and drop off
 FROM
   `bigquery-public-data.chicago_taxi_trips.taxi_trips`
  where trip_miles between 4 AND 30
  AND trip_seconds between 600 AND 7200
  AND TIMESTAMP_DIFF(trip_end_timestamp, trip_start_timestamp, minute) between 10 and 120
  AND fare BETWEEN 5  AND 200
  AND (EXTRACT(MONTH FROM trip_start_timestamp) between 10 and 12)
  AND (EXTRACT(YEAR FROM trip_start_timestamp) between 2016 and 2018)
  AND pickup_longitude < -87
  AND pickup_longitude > -88
  AND dropoff_longitude < -87
  AND dropoff_longitude > -88
  AND pickup_latitude < 42
  AND pickup_latitude > 41
  AND dropoff_latitude < 42
  AND dropoff_latitude > 41
  AND company IS NOT NULL
  AND pickup_census_tract IS NOT NULL
  AND dropoff_census_tract IS NOT NULL
  AND pickup_community_area IS NOT NULL
  AND dropoff_community_area IS NOT NULL
 )
 SELECT *
 FROM taxi_trips))
```

This query will process 23 GB when run.

Output Screenshot:



Model Evaluation: farePrediction_Regression_after_PCA Edited

```sql
SELECT SQRT(mean_squared_error) AS rmse FROM
 ML.EVALUATE(MODEL chicagoTaxiFares.farePrediction_Regression_after_PCA,
 (
 WITH
 taxi_trips AS (
 SELECT
   (trip_total - tips) AS total_fare_without_tips,
   trip_seconds as trip_duration,
   trip_miles as distance_travelled,
   pickup_community_area as pickup_comm_area,
   dropoff_community_area as dropoff_comm_area,
   pickup_longitude as pickup_long,
   pickup_latitude as pickup_lat,
   dropoff_longitude as dropoff_long,
   dropoff_latitude as dropoff_lat,
   EXTRACT(HOUR FROM trip_start_timestamp) AS hourofday,
   company as taxi_company,
   SQRT(POW((pickup_longitude - dropoff_longitude),2) + POW((pickup_latitude - dropoff_latitude), 2)) as euclidean_dist, #Euclidean distance between pickup and drop off
 FROM
   `bigquery-public-data.chicago_taxi_trips.taxi_trips`
  where trip_miles between 4 AND 30
  AND trip_seconds between 600 AND 7200
```

This query will process 23 GB when run.

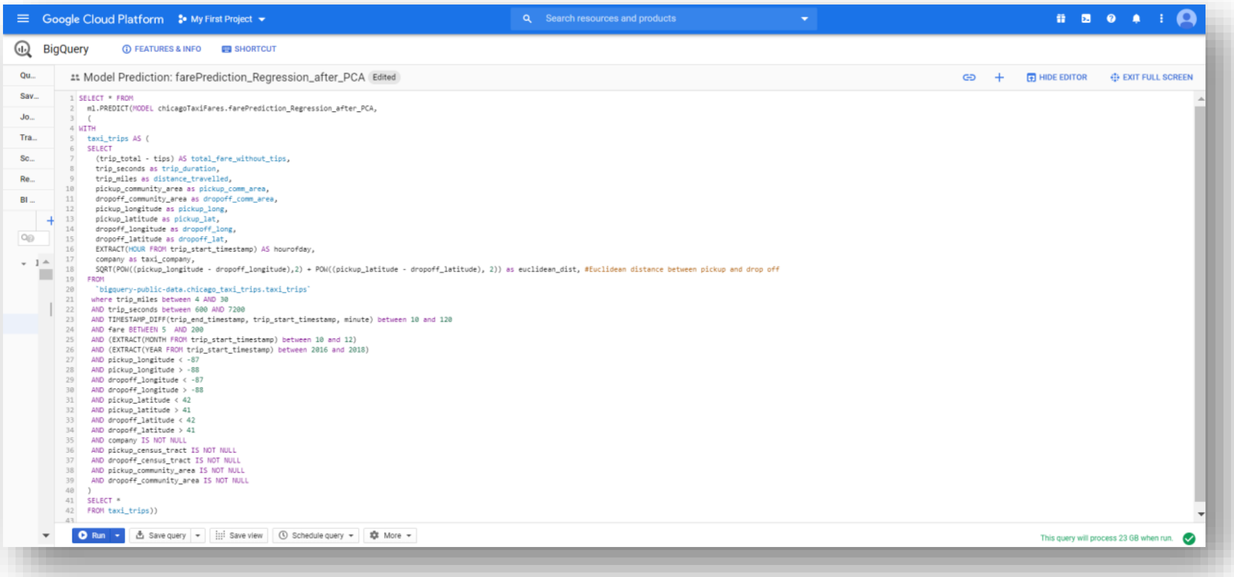
Query results

Query complete (0.6 sec elapsed, 23 GB processed)

Row	rmse
1	6.673484076669607

6. Final Model - Prediction Query

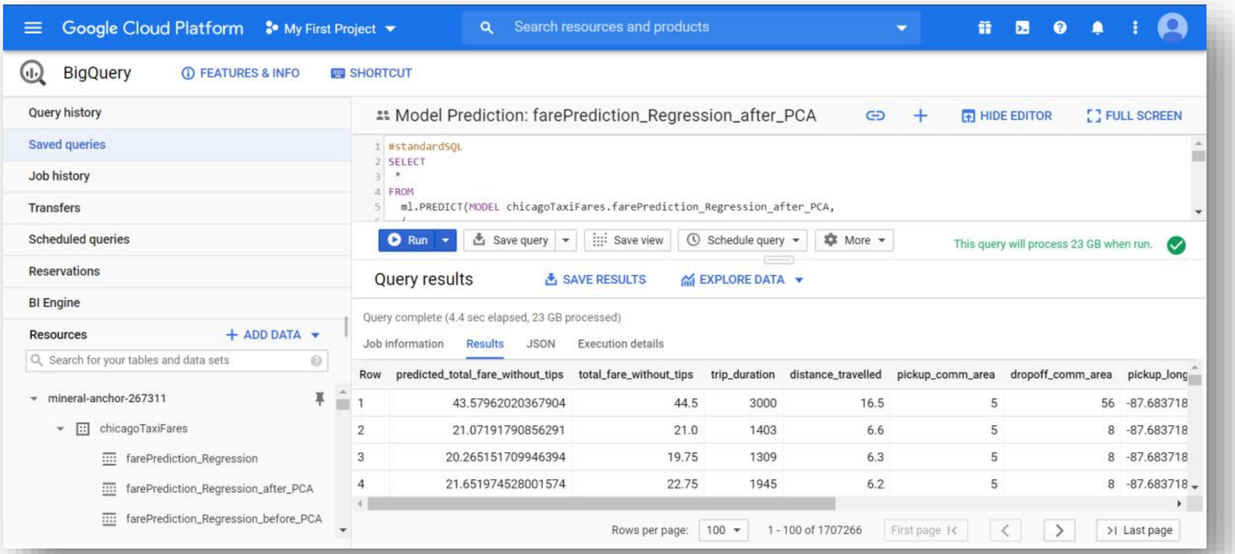
The prediction model uses training data from months 1 to 9 and the test data from months 10, 11, 12.



```
SELECT * FROM
  ml.PREDICT(MODEL chicagoTaxiFares.farePrediction_Regression_after_PCA,
  (
WITH
  taxi_trips AS (
  SELECT
    (trip_total - tips) AS total_fare_without_tips,
    trip_seconds as trip_duration,
    trip_miles as distance_travelled,
    pickup_community_area as pickup_comm_area,
    dropoff_community_area as dropoff_comm_area,
    pickup_longitude as pickup_long,
    pickup_latitude as pickup_lat,
    dropoff_longitude as dropoff_long,
    dropoff_latitude as dropoff_lat,
    EXTRACT(HOUR FROM trip_start_timestamp) AS hourofday,
    company as taxi_company,
    SQRT(POW((pickup_longitude - dropoff_longitude),2) + POW((pickup_latitude - dropoff_latitude), 2)) as euclidean_dist, #Euclidean distance between pickup and drop off
  FROM
    `bigquery-public-data.chicago_taxi_trips.taxi_trips`
   where trip_miles between 4 AND 30
   AND trip_seconds between 600 AND 7200
   AND TIMESTAMP_DIFF(trip_end_timestamp, trip_start_timestamp, minute) between 10 and 120
   AND fare BETWEEN 5  AND 200
   AND (EXTRACT(MONTH FROM trip_start_timestamp) between 10 and 12)
   AND (EXTRACT(YEAR FROM trip_start_timestamp) between 2016 and 2018)
   AND pickup_longitude < -87
   AND pickup_longitude > -88
   AND dropoff_longitude < -87
   AND dropoff_longitude > -88
   AND pickup_latitude < 42
   AND pickup_latitude > 41
   AND dropoff_latitude < 42
   AND dropoff_latitude > 41
   AND company IS NOT NULL
   AND pickup_census_tract IS NOT NULL
   AND dropoff_census_tract IS NOT NULL
   AND pickup_community_area IS NOT NULL
   AND dropoff_community_area IS NOT NULL
  )
  SELECT *
  FROM taxi_trips))
```

This query will process 23 GB when run.

Below screenshot shows total_fare_without_tips and the predicted_total_fare_without_tips using the final prediction model.



```
#standardSQL
SELECT
  *
FROM
  ml.PREDICT(MODEL chicagoTaxiFares.farePrediction_Regression_after_PCA,
```

This query will process 23 GB when run.

Query complete (4.4 sec elapsed, 23 GB processed)

Row	predicted_total_fare_without_tips	total_fare_without_tips	trip_duration	distance_travelled	pickup_comm_area	dropoff_comm_area	pickup_long
1	43.57962020367904	44.5	3000	16.5	5	56	-87.683718
2	21.07191790856291	21.0	1403	6.6	5	8	-87.683718
3	20.265151709946394	19.75	1309	6.3	5	8	-87.683718
4	21.651974528001574	22.75	1945	6.2	5	8	-87.683718