Airplane Mode

Taylor Turner and Alec Meyer

What is the data set?

- The U.S. Department of Transportation's (DOT) Bureau of Transportation Statistics tracks the on-time performance of domestic flights operated by large air carriers
- This data set contains flight data from 2015
- 5,819,079 rows and 31 columns

	YEAR	MONTH	DAY	DAY_OF_WEEK	AIRLINE	FLIGHT_NUMBER	TAIL_NUMBER	ORIGIN_AIRPORT	DESTINATION_AIRPORT	SCHEDULED_DEPARTUR
0	2015	1	1	4	AS	98	N407AS	ANC	SEA	
1	2015	1	1	4	AA	2336	N3KUAA	LAX	PBI	1
2	2015	1	1	4	US	840	N171US	SFO	CLT	2
3	2015	1	1	4	AA	258	N3HYAA	LAX	MIA	2
4	2015	1	1	4	AS	135	N527AS	SEA	ANC	2
...	...	...	...	...	...	...	...	...	...	
5819074	2015	12	31	4	B6	688	N657JB	LAX	BOS	235
5819075	2015	12	31	4	B6	745	N828JB	JFK	PSE	235
5819076	2015	12	31	4	B6	1503	N913JB	JFK	SJU	235
5819077	2015	12	31	4	B6	333	N527JB	MCO	SJU	235
5819078	2015	12	31	4	B6	839	N534JB	JFK	BQN	235

5819079 rows × 31 columns

Project Questions

- How does the day of the week affect the chance of having a delayed flight or cancellation?
- How does the month affect the chance of having a delayed flight or cancellation?
- How does the airline affect the chance of having a delayed flight or cancellation?

Benefits of this Project

- Identify which day of the week or month is most likely to have a flight delay or cancellation so fliers know when is best to fly
- Identify which airline is most likely to have a flight delay or cancellation so fliers can choose the best airline

Data Preparation

- Data obtained from Kaggle
- Additional data is not required at this time, however this project could be repeated with updated flight information
- Created new dataframe with only necessary columns

Cleaning the Data

- Dropped any duplicates and reset index
- Separated on-time flights, delayed flights, and cancelled flights
- Null values are okay, as they represent a cancelled flight

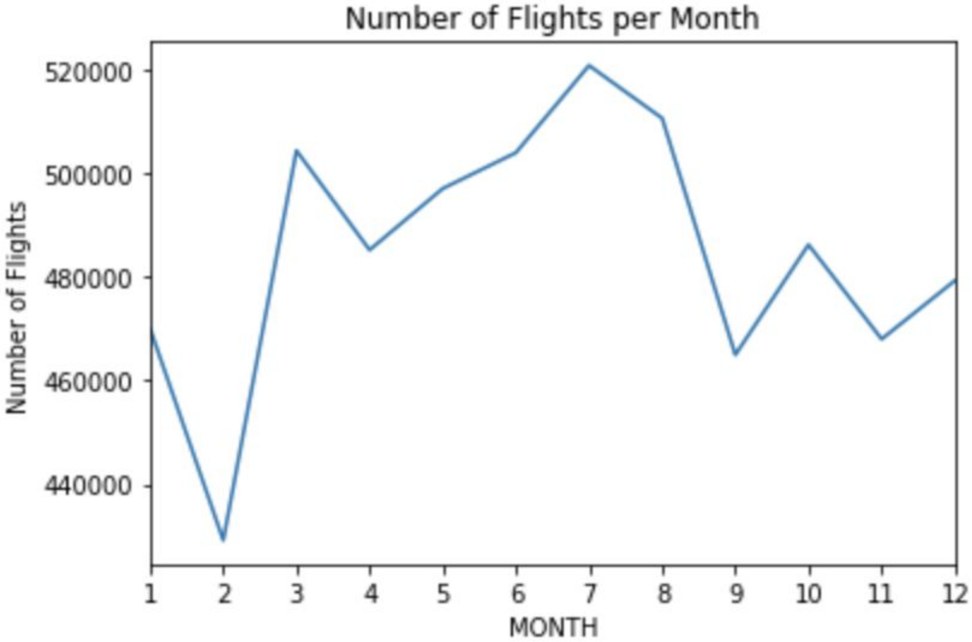
```
# data cleaning
rawData = pd.read_csv("flights.csv", low_memory=False)
# drop duplicate entries
df = rawData.drop_duplicates()
# reset index if any entries were dropped
df = rawData.reset_index(drop = True)
# dataframe with only necessary columns
df = rawData[['MONTH', 'DAY', 'DAY_OF_WEEK', 'AIRLINE', 'CANCELLED','DEPARTURE_DELAY']]
# creating data frame with flights that were cancelled
cancelled = df[df['DEPARTURE_DELAY'].isnull()]
# creating data frame with flights that were delayed
delayed = df[df['DEPARTURE_DELAY'] != 0 & df['DEPARTURE_DELAY'].notnull()]
```

Hypothesis

- Months that typically have bad flying weather will have the most delays and cancellations (snow, storms, etc)
- Months or days that have a greater number of flights will have more delays and cancellations
- Airlines that have a greater number of flights will have more delays and cancellations

Descriptive Analysis

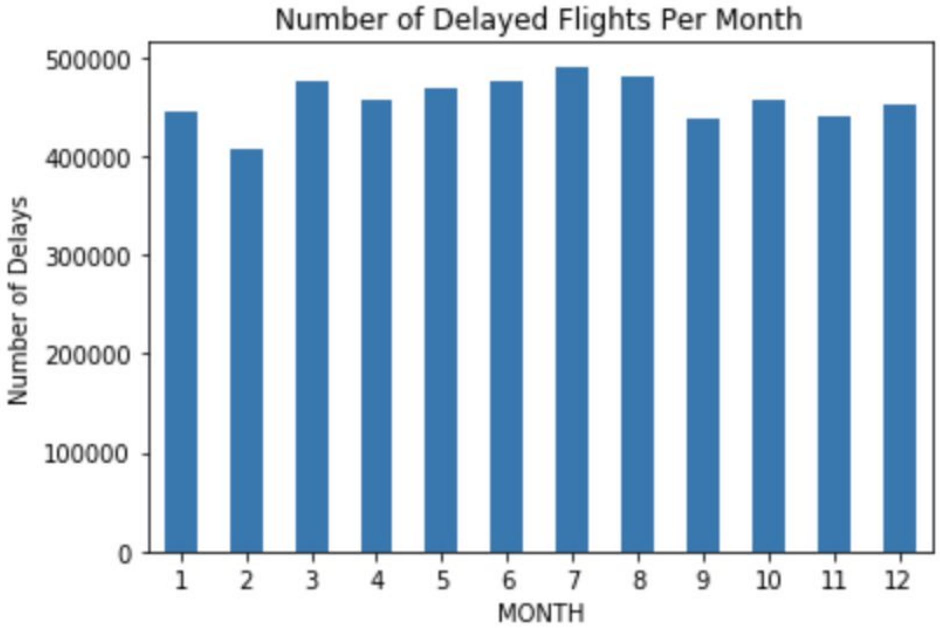
Flights per Month



1 - Jan
2 - Feb
3 - March
4 - April
5 - May
6 - June
7 - July
8 - Aug
9- Sep
10 - Oct
11 - Nov
12 - Dec

7	520718
8	510536
3	504312
6	503897
5	496993
10	486165
4	485151
12	479230
1	469968
11	467972
9	464946
2	429191

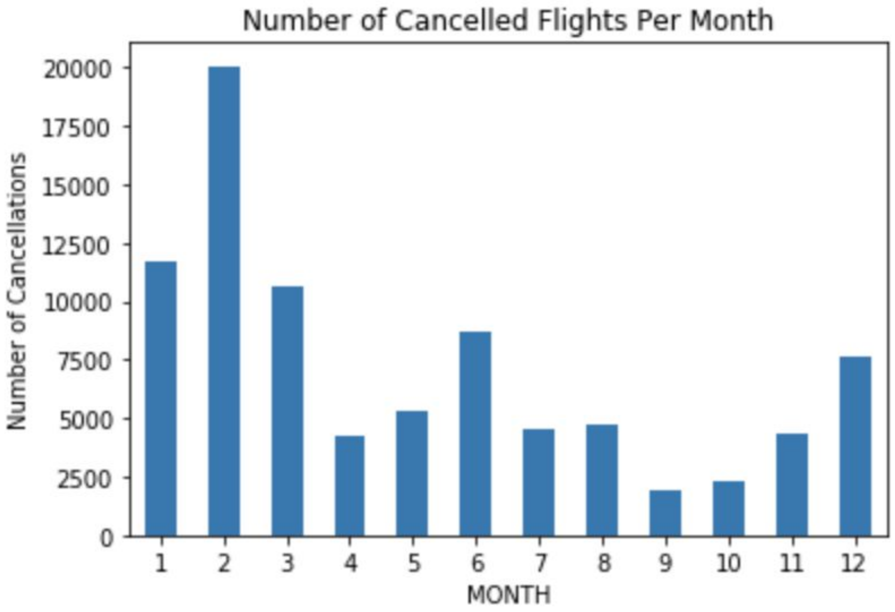
Delays per Month



1 - Jan
2 - Feb
3 - March
4 - April
5 - May
6 - June
7 - July
8 - Aug
9- Sep
10 - Oct
11 - Nov
12 - Dec

7	490877
8	480665
3	475937
6	475547
5	469374
4	458155
10	456430
12	451353
1	445683
11	440488
9	438408
2	406802

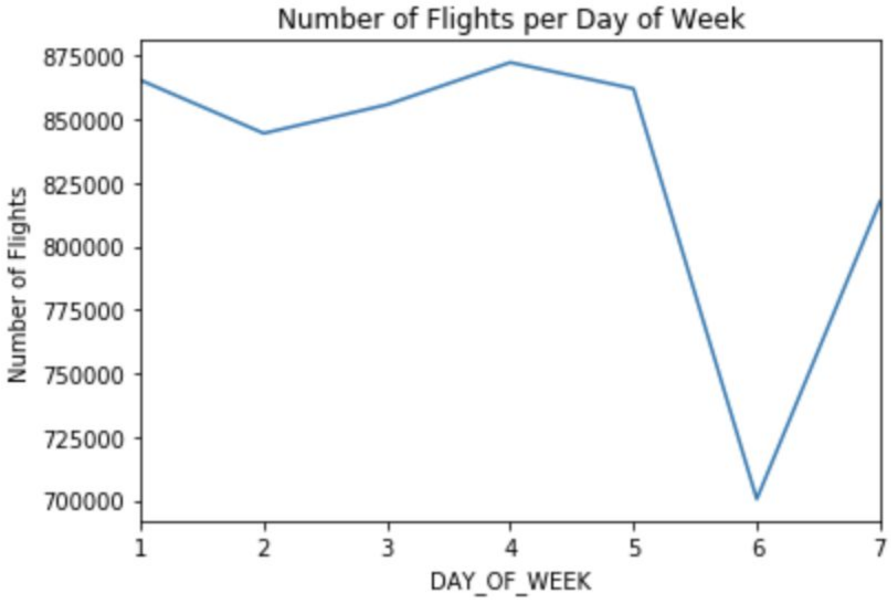
Cancellations per Month



1 - Jan
2 - Feb
3 - March
4 - April
5 - May
6 - June
7 - July
8 - Aug
9- Sep
10 - Oct
11 - Nov
12 - Dec

2	20059
1	11657
3	10639
6	8698
12	7679
5	5336
8	4719
7	4507
11	4339
4	4253
10	2339
9	1928

Flights per Day of Week



1 - Sunday

2 - Monday

3 - Tuesday

4 - Wednesday

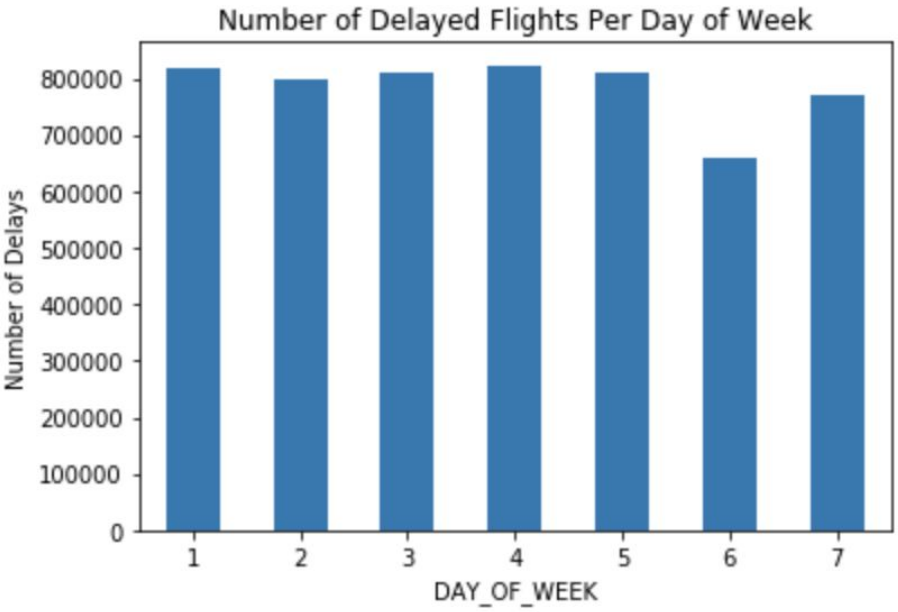
5 - Thursday

6 - Friday

7 - Saturday

4	872521
1	865543
5	862209
3	855897
2	844600
7	817764
6	700545

Delays per Day of Week



1 - Sunday

2 - Monday

3 - Tuesday

4 - Wednesday

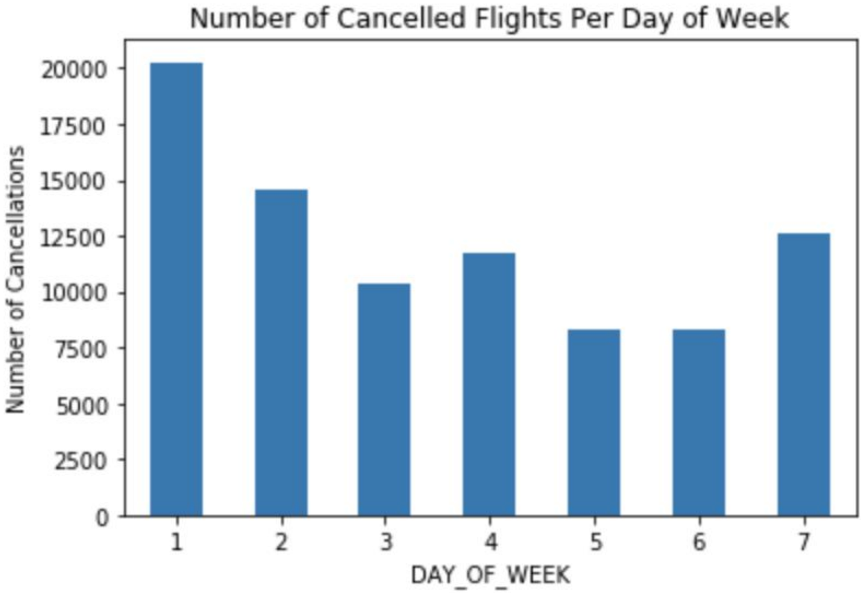
5 - Thursday

6 - Friday

7 - Saturday

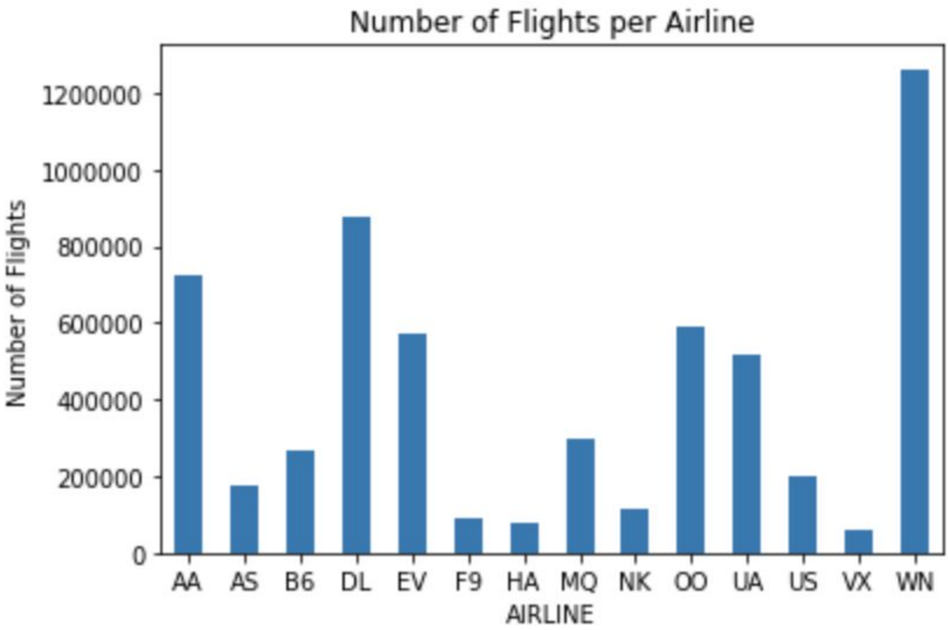
4	822663
1	816401
5	811637
3	808415
2	798613
7	771144
6	660846

Cancellations per Day of Week



1 - Sunday	1	20255
2 - Monday	2	14609
3 - Tuesday	7	12617
4 - Wednesday	4	11741
5 - Thursday	3	10314
6 - Friday	5	8325
7 - Saturday	6	8292

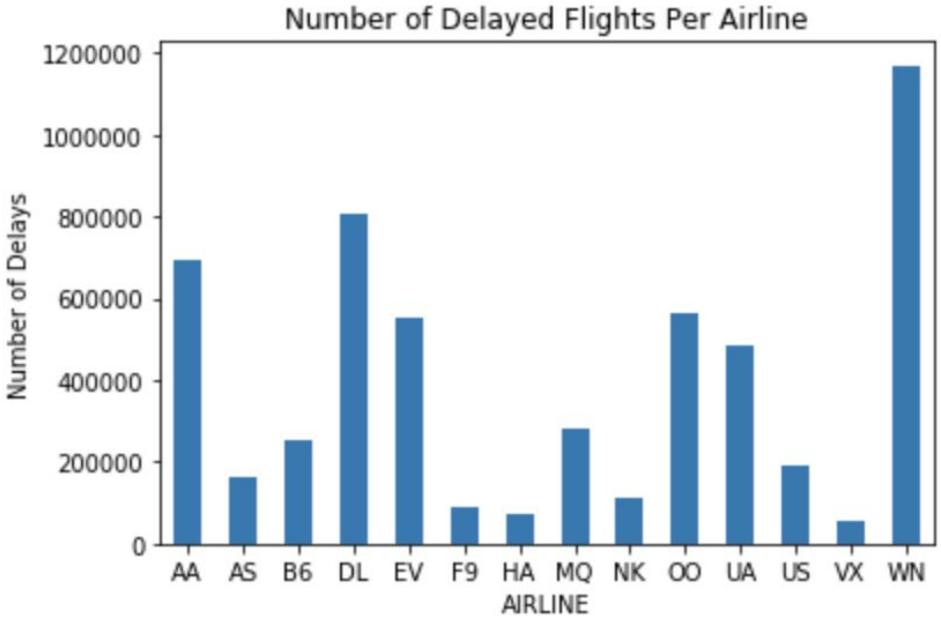
Flights per Airline



IATA_CODE	AIRLINE
UA	United Air Lines Inc.
AA	American Airlines Inc.
US	US Airways Inc.
F9	Frontier Airlines Inc.
B6	JetBlue Airways
OO	Skywest Airlines Inc.
AS	Alaska Airlines Inc.
NK	Spirit Air Lines
WN	Southwest Airlines Co.
DL	Delta Air Lines Inc.
EV	Atlantic Southeast Airlines
HA	Hawaiian Airlines Inc.
MQ	American Eagle Airlines Inc.
VX	Virgin America

WN	1261855
DL	875881
AA	725984
OO	588353
EV	571977
UA	515723
MQ	294632
B6	267048
US	198715
AS	172521
NK	117379
F9	90836
HA	76272
VX	61903

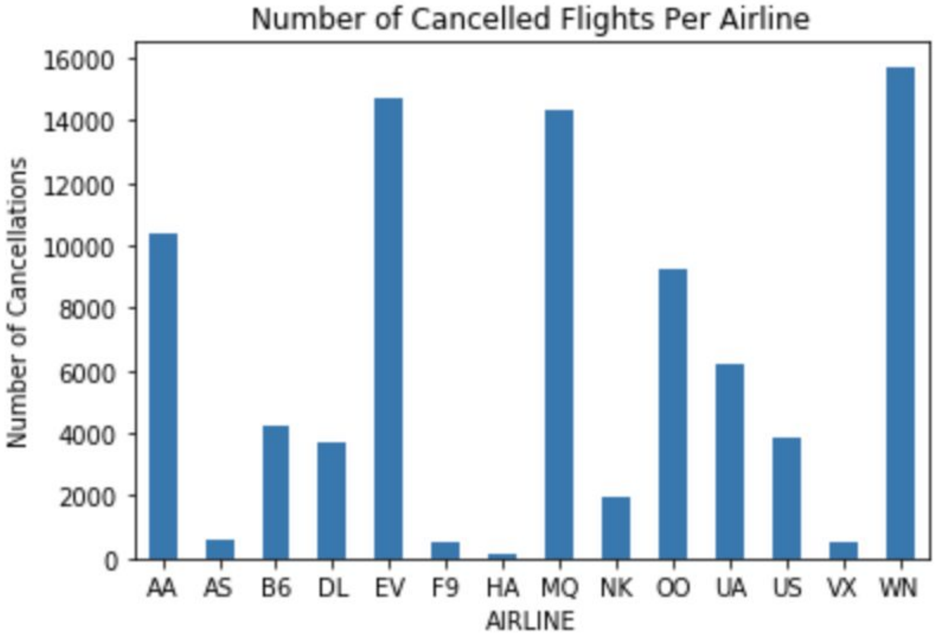
Delays per Airline



IATA_CODE	AIRLINE
UA	United Air Lines Inc.
AA	American Airlines Inc.
US	US Airways Inc.
F9	Frontier Airlines Inc.
B6	JetBlue Airways
OO	Skywest Airlines Inc.
AS	Alaska Airlines Inc.
NK	Spirit Air Lines
WN	Southwest Airlines Co.
DL	Delta Air Lines Inc.
EV	Atlantic Southeast Airlines
HA	Hawaiian Airlines Inc.
MQ	American Eagle Airlines Inc.
VX	Virgin America

WN	1170316
DL	808748
AA	692329
OO	561209
EV	550173
UA	487593
MQ	279588
B6	253426
US	189487
AS	165430
NK	112677
F9	87583
HA	73140
VX	58020

Cancellations per Airline



IATA_CODE	AIRLINE
UA	United Air Lines Inc.
AA	American Airlines Inc.
US	US Airways Inc.
F9	Frontier Airlines Inc.
B6	JetBlue Airways
OO	Skywest Airlines Inc.
AS	Alaska Airlines Inc.
NK	Spirit Air Lines
WN	Southwest Airlines Co.
DL	Delta Air Lines Inc.
EV	Atlantic Southeast Airlines
HA	Hawaiian Airlines Inc.
MQ	American Eagle Airlines Inc.
VX	Virgin America

WN	15726
EV	14683
MQ	14350
AA	10386
OO	9267
UA	6189
B6	4205
US	3890
DL	3704
NK	1925
AS	611
F9	546
VX	518
HA	153

Problems with Data or Possible Improvements

- Contains unnecessary data for our particular project
- Could provide more recent data

Modeling the Data and Machine Learning

What do we want to learn from our data?

Based off of:

-DISTANCE, DEPARTURE_DELAY, SCHEDULED_TIME, ELAPSED_TIME, AIR_TIME, DIVERTED

```
# Select only desired columns from data
flights = flights[['DISTANCE', 'DEPARTURE_DELAY', 'ARRIVAL_DELAY',
                   'SCHEDULED_TIME', 'ELAPSED_TIME', 'AIR_TIME', 'DIVERTED']]
flights = flights.dropna()
```

We are going to predict:

- ARRIVAL_DELAY

```
# Select which data to predict (y)
X = flights.drop('ARRIVAL_DELAY',axis = 1)
y = flights['ARRIVAL_DELAY']
```

Training and Splitting the Data

```
from sklearn.model_selection import train_test_split
```

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.3,random_state = 2)
```

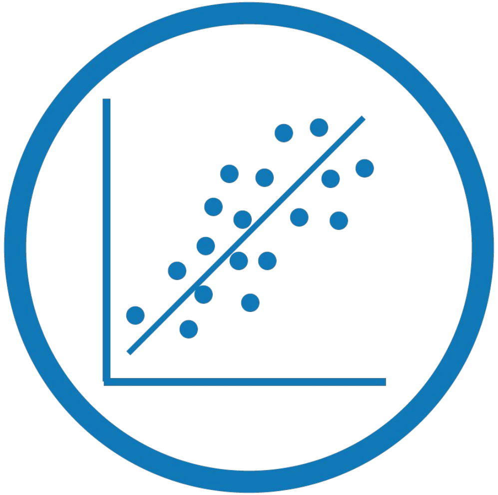
The data was trained and split using sklearn's model_selection tool. A test size of 0.3 was used since the data was so large there was no need to use a larger portion. The "random_state" parameter is used for hyper-parameter-tuning and making sure the data set uses the same random seed.

Choosing a Machine Learning Model

Out of the many available machine learning models provided in sklearn(K-nearest-neighbors, linear regression, lasso, etc.), we chose linear regression as our main model. A decision tree regressor was chosen as our comparative model.

```
from sklearn.linear_model import LinearRegression
```

```
from sklearn.tree import DecisionTreeRegressor
```

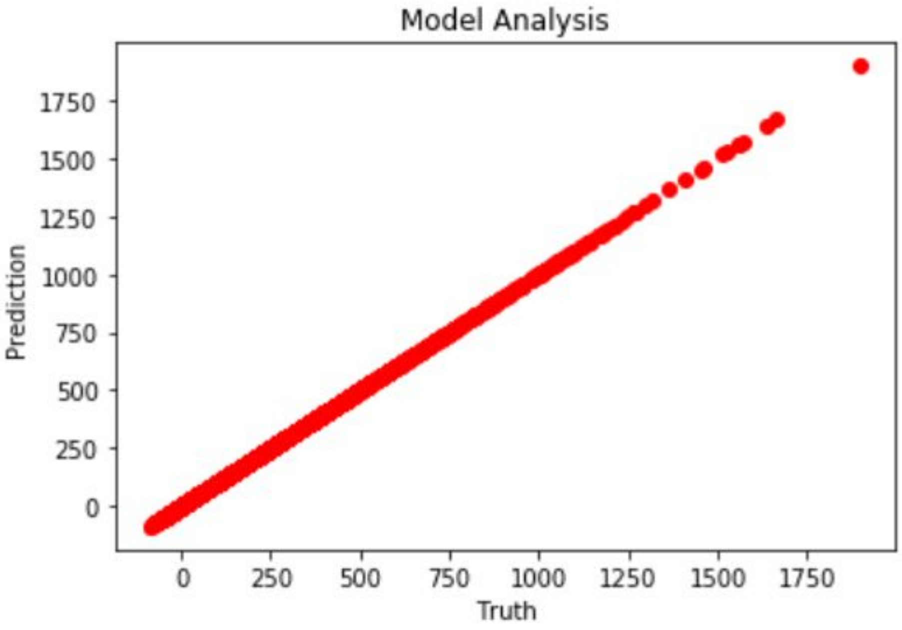


Linear Regression

The reason we chose this Machine learning model is because:

1. Works very well for big data since it is easy and efficient to train
2. Linear regression models are fairly intuitive to use so an untrained eye would be able to understand the data
3. Linear regression works well when you have easily separable data (most of our data points are integers)
4. Best for predicting cause and effect data

Linear Regression Modeled



Accuracy (R) & Error:

```
Linear Regression
Error: 1.166724019783705e-06
R: 0.9999999984709158
```

Decision Tree Regressor

The reason we chose Decision Tree Regressor as a comparative model is because:

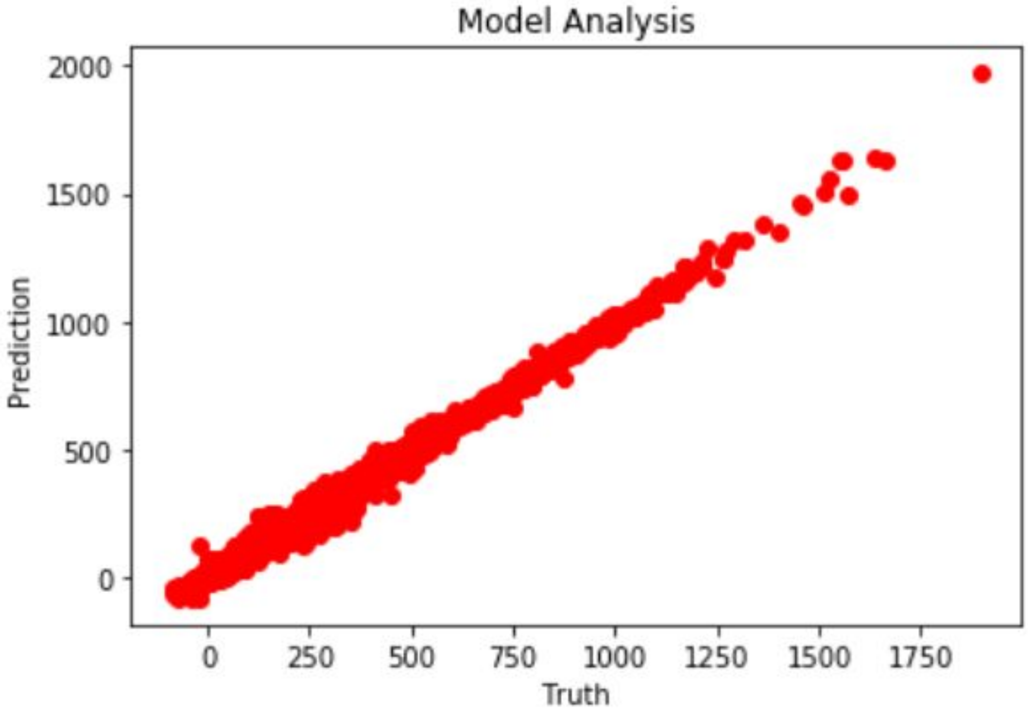
1. Struggles with many columns (branches of data)

2. Harder to predict cause and effect

3. Using nodes to make decisions provides a drastic difference from linear regression models

4. Hyper parameter tuning

Decision Tree Regression Model



Accuracy (R) & Error:

```
Decision Tree
Error: 0.3483000554776768
R: 0.9986301932940718
```

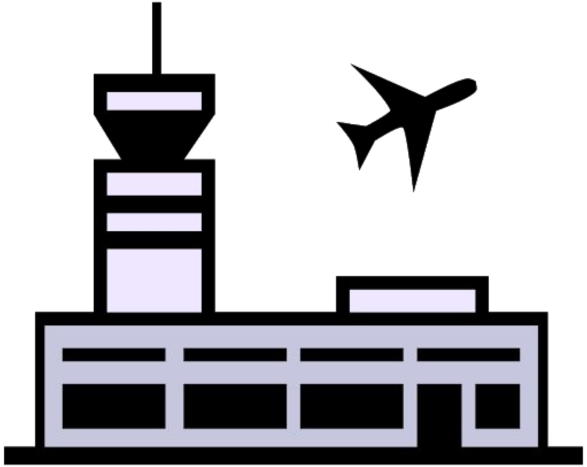
Alternate Methods to Try

- Decision Tree Regressor with 3 parameters
- Linear regression with a test_size close to 1
- Using a classifier

Prescriptive Analysis

What Can We Do With This Information?

- Know when the best time to fly is
- Choose the right airline
- Be prepared for the possibility of a delay or cancellation



Thank You :)