FLIGHT STATUS PREDICTOR

A Classification Machine Learning Project

By: Brian Treftz, MS Data Science Student

TABLE OF CONTENTS

01
INTRODUCTION

PROJECT MOTIVATION

Flight delays create logistical challenges for travelers, airlines, and airports, often resulting in cascading effects across the entire aviation system.

Advance notice of potential delays empowers stakeholders to take proactive actions instead of reacting to last-minute disruptions.

IMPACT ON KEY STAKEHOLDERS

TRAVELERS

- Decreased stress
- Less missed connections
- Reduced last-minute schedule adjustments
  - Ground transportation
  - Meeting schedules

AIRLINES

- Turnaround disruptions
- Increased fuel costs
- Scheduling inefficiencies

AIRPORTS

- Gate congestion
- Resource strain
- Operational delays

End User

SOLUTION

MACHINE LEARNING OF COURSE!

TWO SUPERVISED LEARNING APPROACHES:

REGRESSION MODEL – Quantify magnitude of the delay

CLASSIFICATION MODEL – Binary label (1 = Yes, 0 = No) for arrival delays >= 15min

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6777978 entries, 0 to 6777977
Data columns (total 43 columns):
 #   Column                    Dtype
---  ------                    -----
 0   Carrier_Name              object
 1   Year                      int64
 2   Quarter                   int64
 3   Month                     object
 4   Month_Day                 int64
 5   Week_Day                  object
 6   Flight_Date               object
 7   Airline_DOT_ID            int64
 8   Tail_Number               object
 9   Flight_Number             int64
 10  Origin_Airport_ID         int64
 11  Origin_City_Market_ID     int64
 12  Origin_IATA_Code          object
 13  Origin_City_State         object
 14  Destination_Airport_ID    int64
 15  Destination_City_Market_ID int64
 16  Destination_IATA_Code     object
 17  Destination_City_State    object
 18  Dep_Time_Block_Group      object
 19  Scheduled_Departure_Time  int64
 20  Actual_Dep_Time           int64
 21  Dep_Time_Offset           int64
```

02

DATA

AIRLINE ON-TIME PERFORMANCE DATA

WHAT: Monthly data on flights reported by US certified air carriers

CONTENTS: Flight details, performance data, and operational data

SOURCE: Bureau of Transportations Statistics

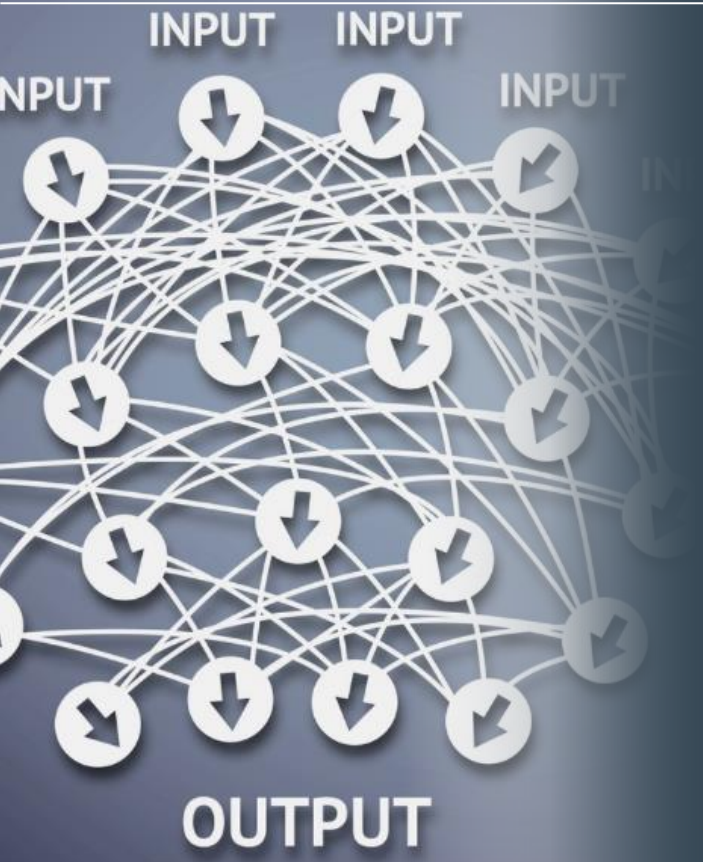
TIME FRAME: 01 Apr 2023 – 31 Mar 2024

Raw Data Example

A	B	C	D	E	F	G	H	I	J	K	L	M	N
Year	Quarter	Month	DayofMonth	DayOfWeek	FlightDate	Reporting_A	DOT_ID_Rep	IATA_CODE_	Tail_Numbe	Flight_Numb	OriginAirport	OriginAirport	OriginCityMa
2023	4	12	25	1	12/25/23	OO	20304	OO	N611SK	3173	11982	1198202	31982
2023	4	12	25	1	12/25/23	OO	20304	OO	N762SK	3178	12915	1291503	31205
2023	4	12	25	1	12/25/23	OO	20304	OO	N608SK	3205	16218	1621802	33785
2023	4	12	25	1	12/25/23	OO	20304	OO	N760SK	3232	12278	1227805	30928
2023	4	12	20	3	12/20/23	OO	20304	OO	N797SK	5772	10372	1037205	30372
2023	4	12	20	3	12/20/23	OO	20304	OO	N793SK	5776	10372	1037205	30372

BF	BG	BH	BI	BJ	BK	BL	BM	BN	BO	BP
WeatherDela	NASDelay	SecurityDela	LateAircraftD	FirstDepTime	TotalAddGTi	LongestAddG	DivAirportLa	DivReachedD	DivActualEla	DivArrDelay
							0			
							0			
							0			
							0			
							0			
							0			

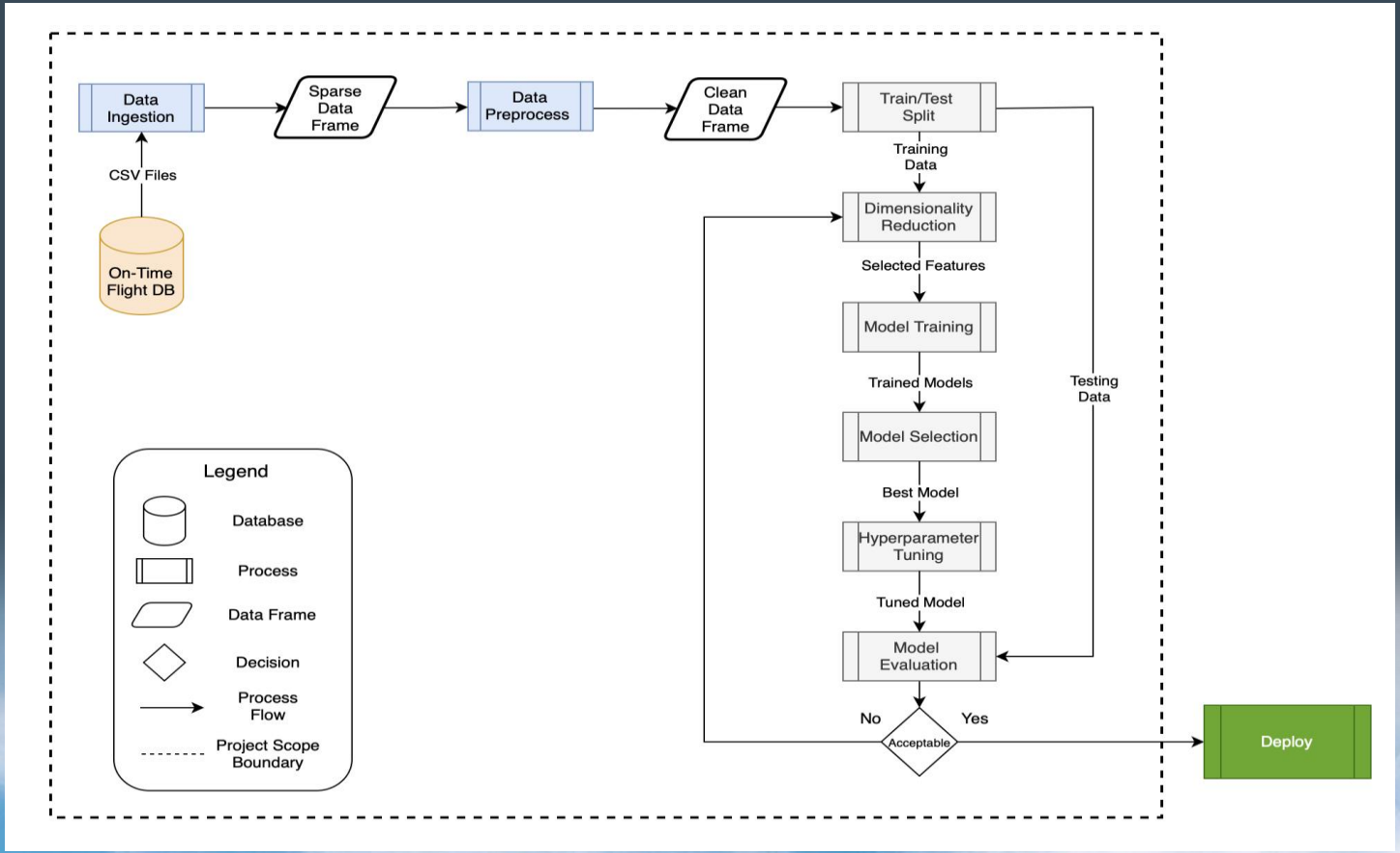
RAW DATA		DATA FOR PROJECT
6,884,250	Number of Flights (Rows)	6,777,978
111	Number of Features (Columns)	43
3.11 Gb	CSV File Size	1.48 Gb
11.0 Gb	Pandas DF Memory Usage	1.2 Gb



03

ML PIPELINE

DESIGN DIAGRAM



DIMENTIONALITY REDUCTION

EDA

- Distributions
- Correlations
- Summary statistics

TARGET ANALYSIS (TA)

- Quantifying delayed flights
- Temporal analysis
- Geographic Analysis

FEATURE ENGINEERING

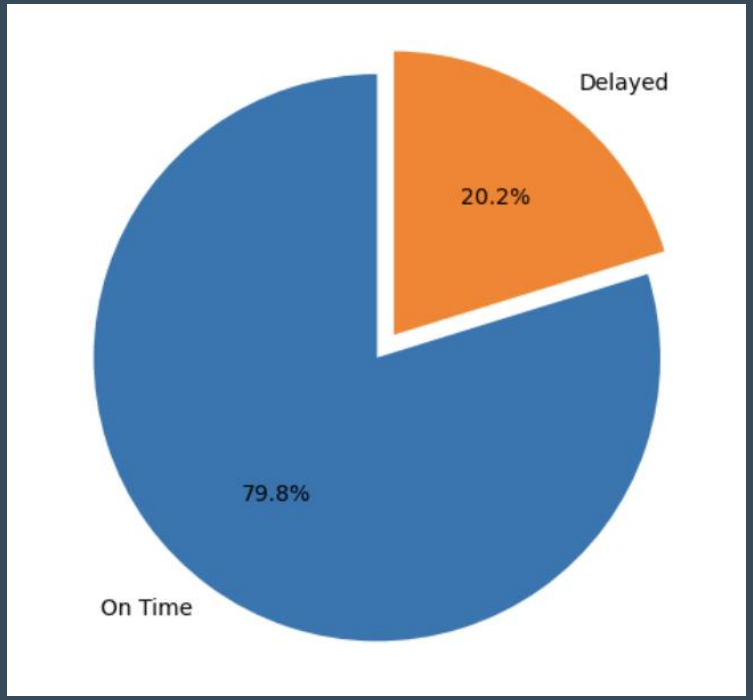
- Renaming features
- Binning
- Adding carrier names

FEATURE SELECTION

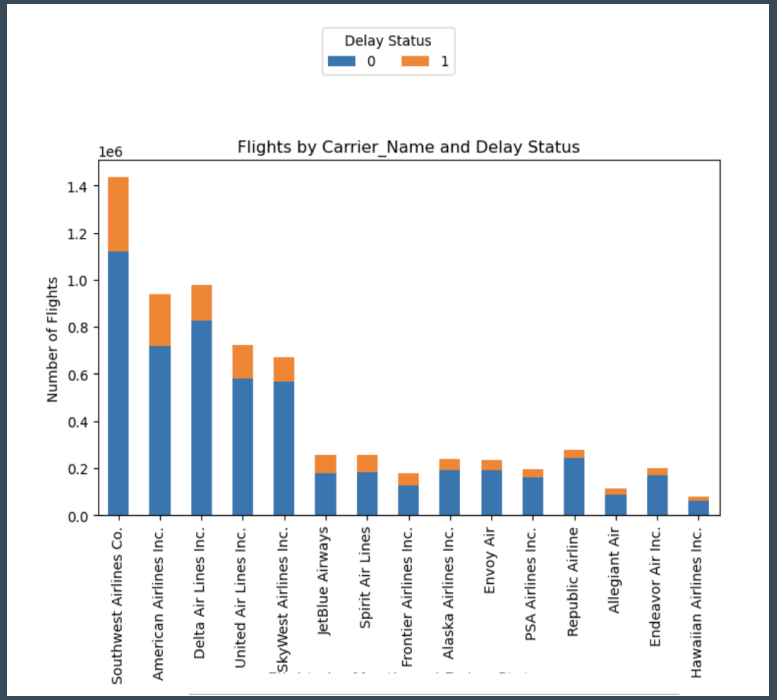
- Extracting Training and Testing datasets
- Down sampling to address imbalance in Target
- Feature importance
- Cramer's V score
- Pearson

EDA and TA

% Delayed of all the flights

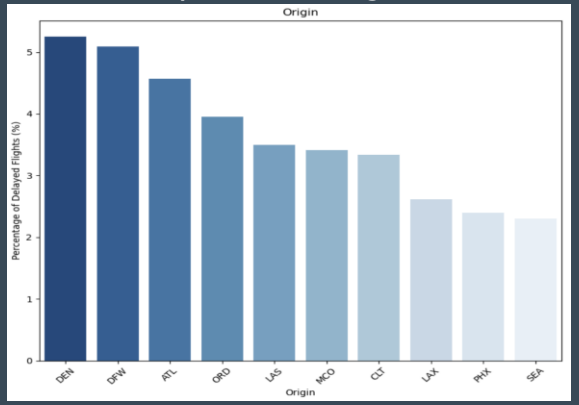


No. Flights Vs. Carrier

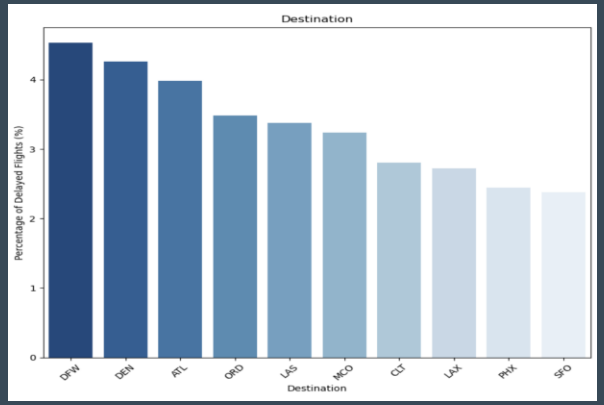


EDA and TA

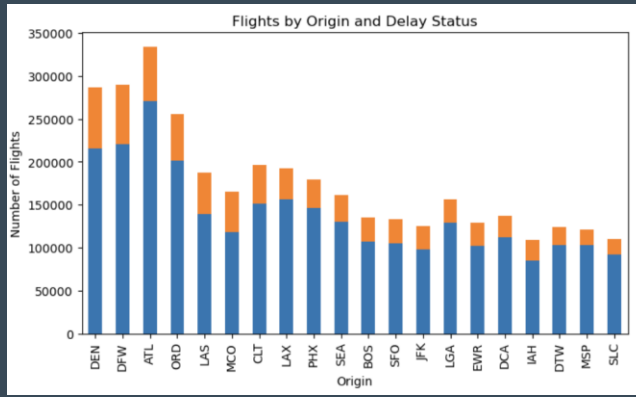
% Delayed VS Origin



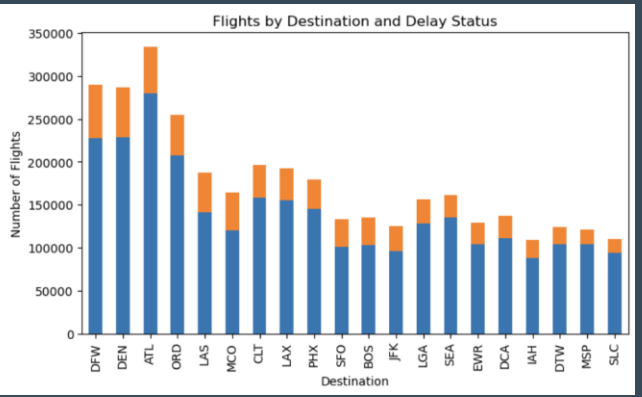
% Delayed VS Destination



No. of Flights vs Origin

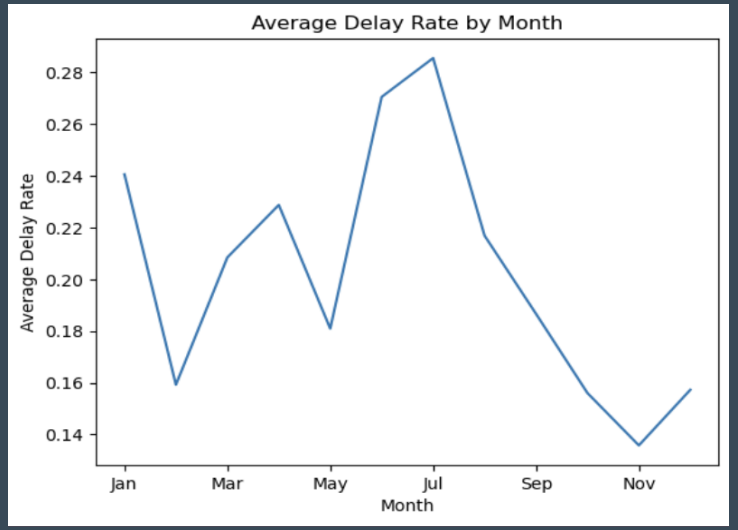


No. of Flights vs Destination

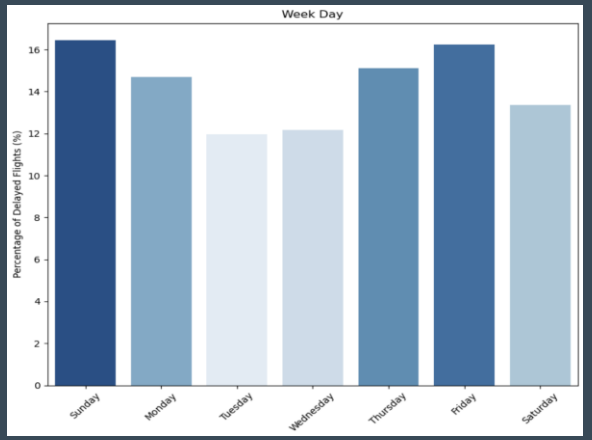


EDA and TA

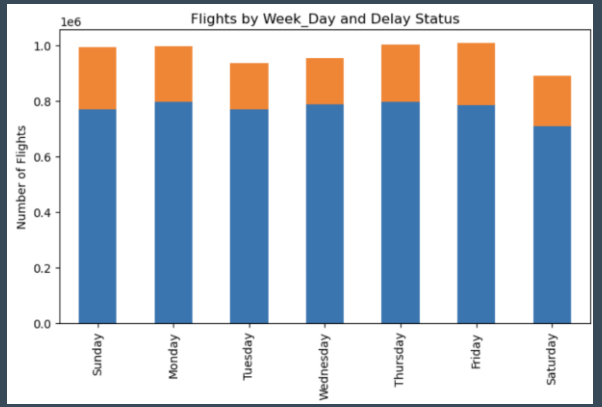
Delay Rate VS Month



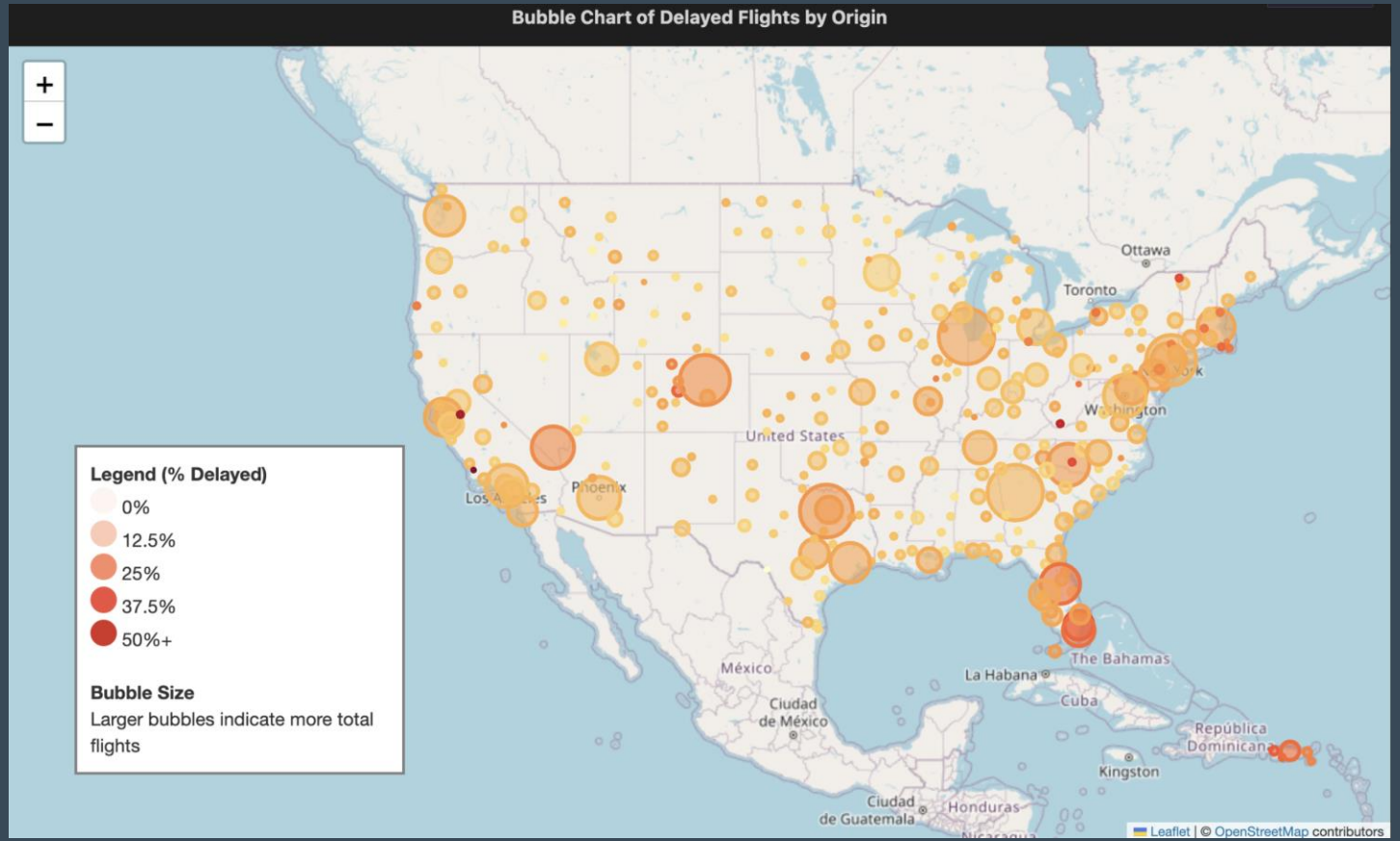
% Delay VS Weekday



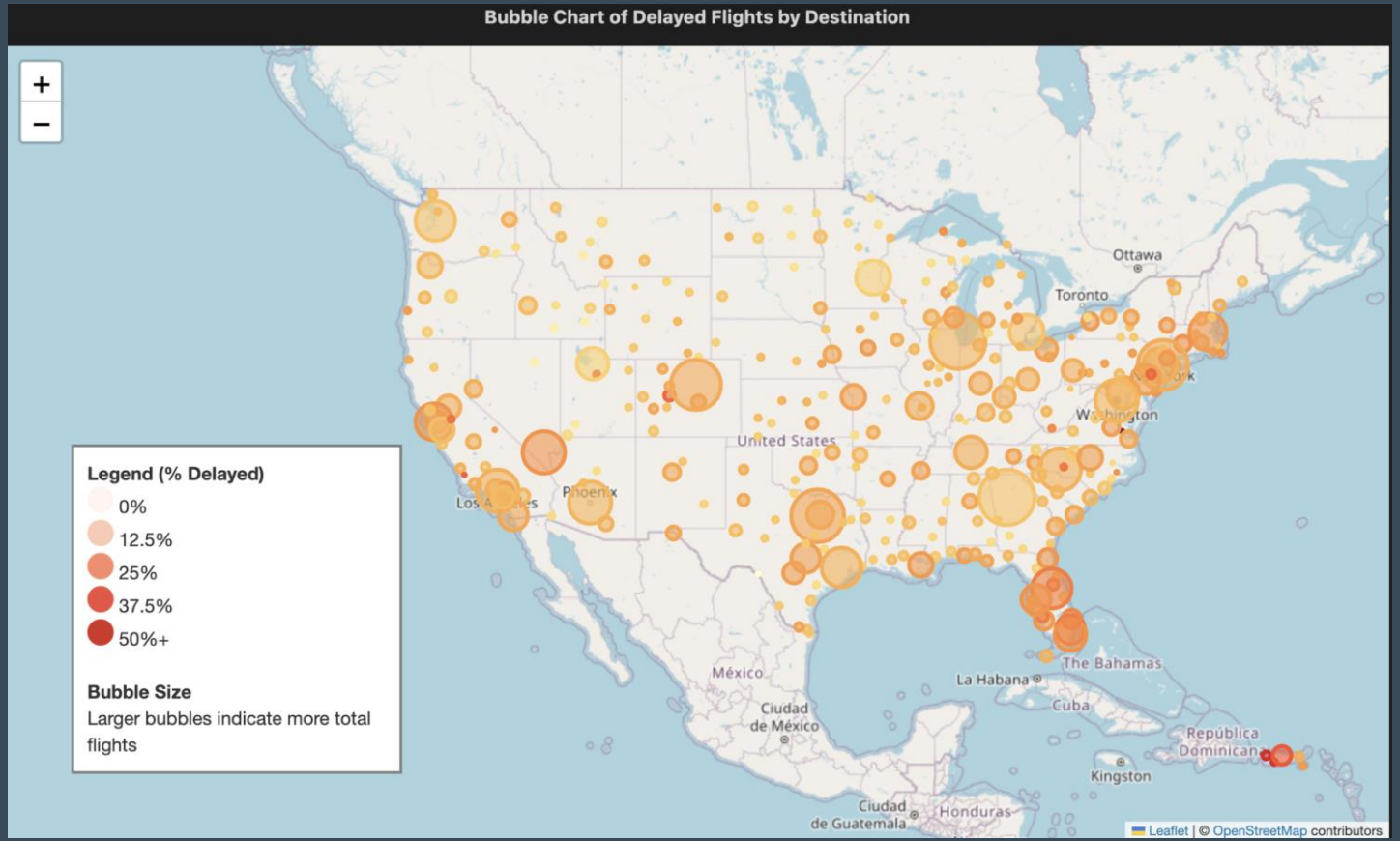
No. of Flights VS Weekday



EDA and TA


Bubble Chart of Delayed Flights by Origin
+
−
Legend (% Delayed)
0%
12.5%
25%
37.5%
50%+
Bubble Size
Larger bubbles indicate more total flights
Ottawa
Toronto
United States
Phoenix
México
Ciudad de México
La Habana
Cuba
The Bahamas
República Dominicana
Kingston
Ciudad de Guatemala
Honduras
Leaflet | © OpenStreetMap contributors

EDA and TA


Bubble Chart of Delayed Flights by Destination
+
−
Legend (% Delayed)
0%
12.5%
25%
37.5%
50%+
Bubble Size
Larger bubbles indicate more total flights
Ottawa
Toronto
United States
Phoenix
México
Ciudad de México
La Habana
The Bahamas
Cuba
República Dominican
Kingston
Ciudad de Guatemala
Honduras
Leaflet | © OpenStreetMap contributors

EDA and TA - Takeaways

- Model accuracy >= 80% is the starting benchmark
- Highest rate of delays occurs in July
- Largest % delays occur on Sundays and Fridays
- Delays are inevitable.
  - It’d be nice to know in advance.

FEATURE SELECTION

Down sampling to address imbalance

Before

```
# Display values before downsampling
print(training_df['Delayed'].value_counts())

# Display shape before downsampling
print(training_df.shape)
```

```
0    4183104
1    1072668
Name: Delayed, dtype: int64
(5255772, 18)
```

After

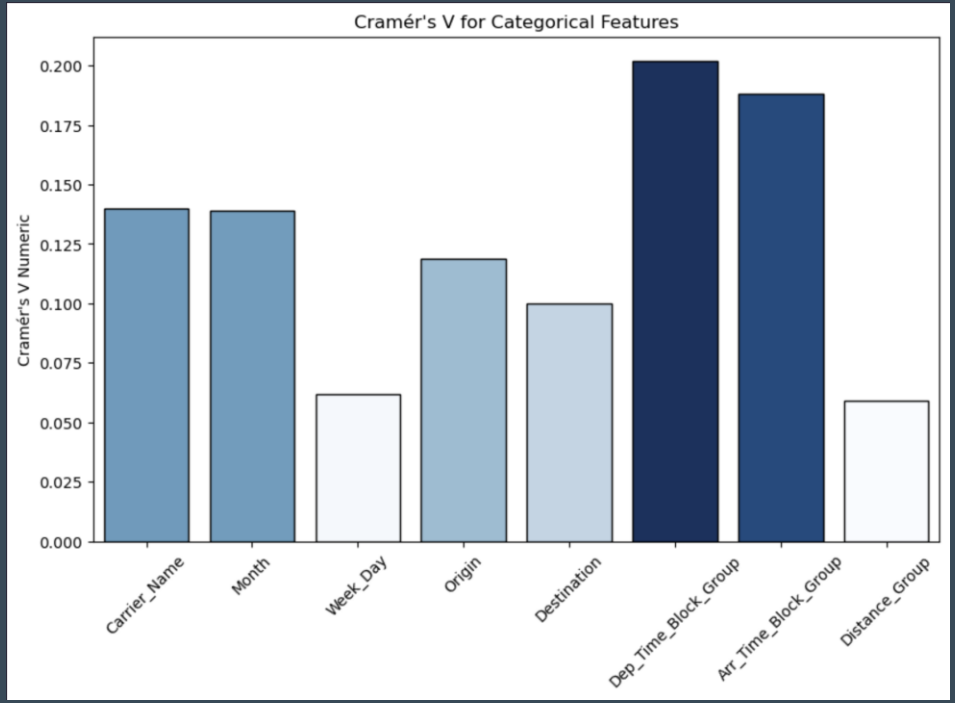
```
# Verify the new class distribution
print(training_df_ds['Delayed'].value_counts())

# Verify new dataframe shape
print(training_df_ds.shape)
```

```
1    1072668
0    1072668
Name: Delayed, dtype: int64
(2145336, 18)
```

FEATURE SELECTION

Quantifying effect size of categorical features. (Cramér's V)

	Chi2	p-value	Cramér's V
Carrier_Name	41816.6	0.0	0.140
Month	41426.3	0.0	0.139
Week_Day	8241.7	0.0	0.062
Origin	30172.1	0.0	0.119
Destination	21420.5	0.0	0.100
Dep_Time_Block_Group	87840.9	0.0	0.202
Arr_Time_Block_Group	75797.0	0.0	0.188
Distance_Group	7441.8	0.0	0.059



Pearson Correlation for numerical feature

```
Pearson correlation for 'Scheduled_Duration': 0.034
```

04
RESULTS

MODEL SELECTION

Model	Train Runtime	Predict Runtime	ROC AUC	Accuracy	F1 Score
XGBoost	**00:05.0**	**00:00.0**	**0.64**	**0.64**	**0.64**
CatBoost	01:00.0	00:00.0	0.64	0.64	0.64
LightGBM	00:19.0	00:02.0	0.64	0.64	0.641
Random Forest	00:48.0	00:00.0	0.63	0.63	0.634
Logistic Regression	00:10.0	00:00.0	0.61	0.61	0.623
SVM	11:49.0	00:17.0	0.51	0.51	0.434

HYPERPARAMETER TUNING

Method - GridSearchCV

Parameter	Default Value	Input Values	Output
Learning Rate	0.3	0.01, 0.05, 0.1, 0.2	0.2
Max Depth	6	3, 5, 7, 10	10
Subsample	0.9	0.7, 0.8, 0.9, 1.0	0.9

FINAL MODEL PERFORMANCE

Metric	Training Data		Test Data
	Before Tuning	After Tuning	Value
Accuracy	0.64	0.65	0.67 ↑
F1 Score	0.64	0.64	0.41 ↓
ROC AUC	0.64	0.65	0.64 ↓

05
CONCLUSION

What did I learn?

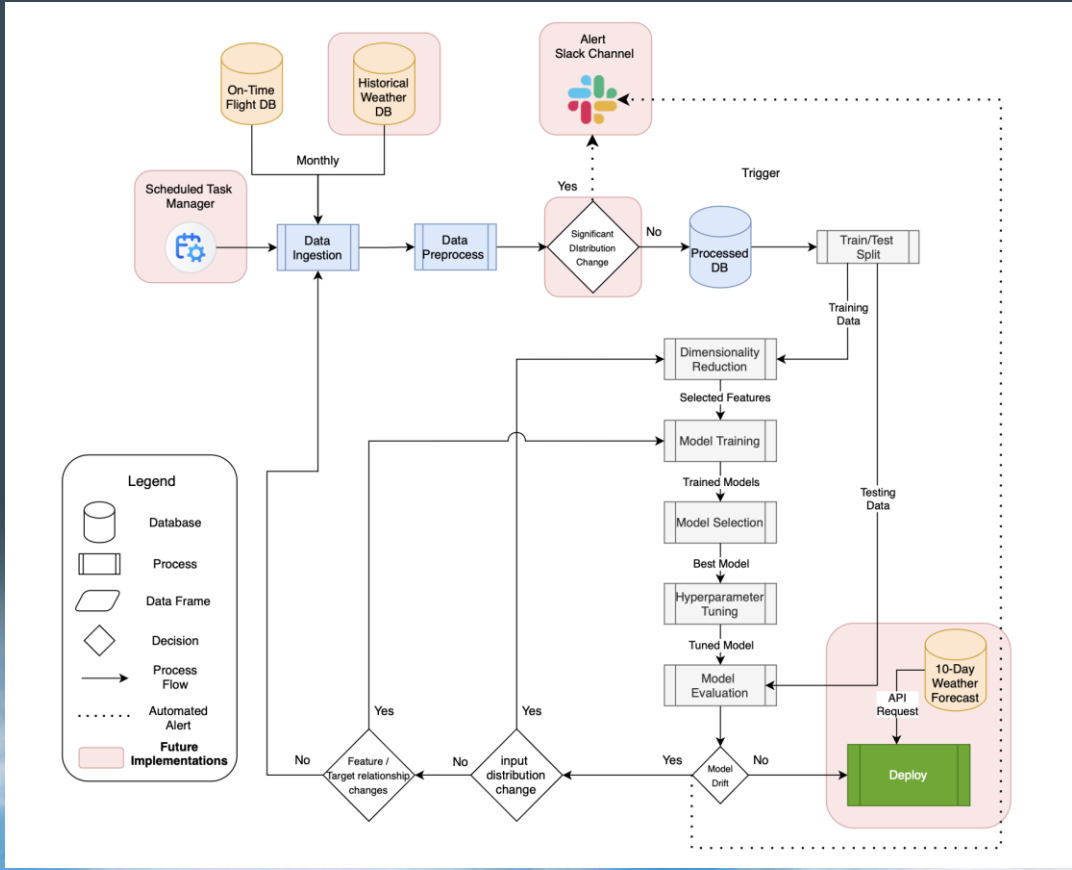
- I have more questions now than when I started (and that's a good thing!).

- Striving to build a robust and accurate model is important, and awesome.

- But even more important is being able to succinctly convey insights and ensure stakeholders see the value of the work.

- This is just the beginning of this project...

DESIGN DIAGRAM - V2



THANK YOU

Questions?
Please reach out to me at
brian.treftz@gmail.com