Flight Foresight: Predicting Delays with ML


Andy Guinto


Irene Na


Muthumayan Madhayyan


Rahul Chugh

Project Abstract

OBJECTIVE

Forecast domestic flight departure delays 2 hours ahead of schedule

DATA

- Flight Data from US Department of Transportation
- Weather Data from National Oceanic and Atmospheric Administration

From 2015 through 2019

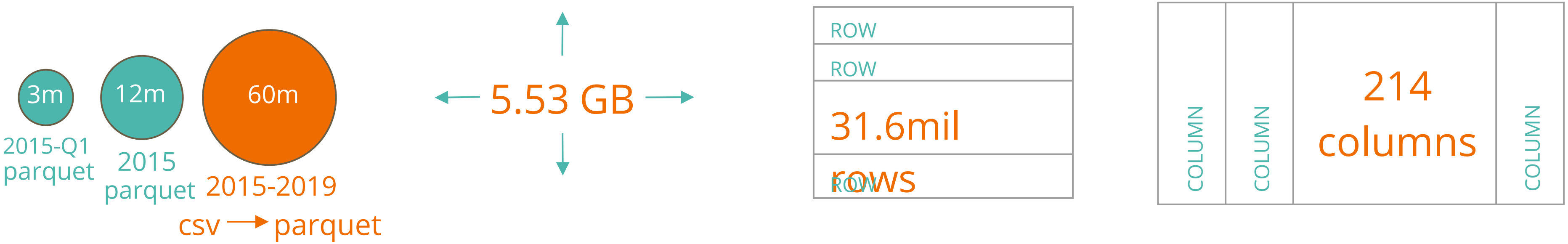
TARGET CONSUMER

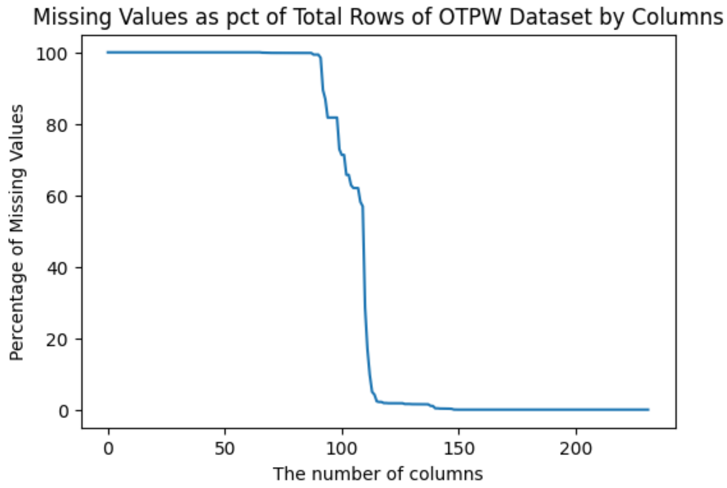
Airlines

PREDICTION SCOPE

- Limited to data at hand
- Not exploring other datasets like IT outages, personnel shortages, etc

Data Description and Missing Values Treatment





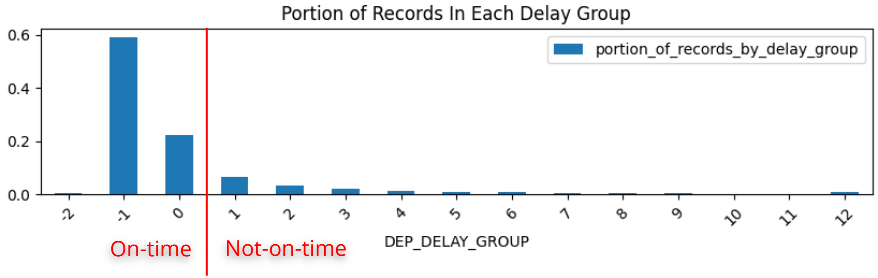
MISSING VALUES TREATMENT

- Dropped 113 columns >20% missing values
- Imputed data for 16 numerical columns
  (YEAR, MONTH, ORIGIN_AIRPORT_ID) or (YEAR, MONTH, DEST_AIRPORT_ID)
- Dropped rows for cancelled flights
  Created a new feature for cancelled flights in a rolling time window

Target variable: DEP_DELAY_GROUP

ORIGINAL

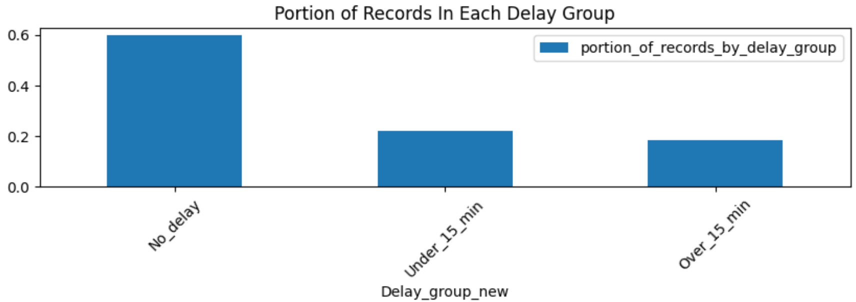
- 14 groups
- 15 minute increments



High Label Imbalance (on-time group accounting for 80% of the records)

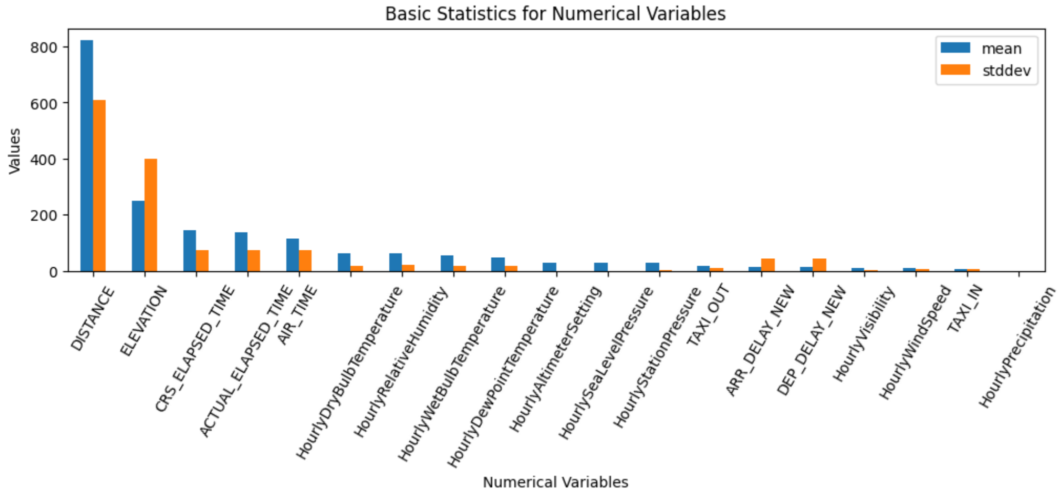
TRANSFORMED

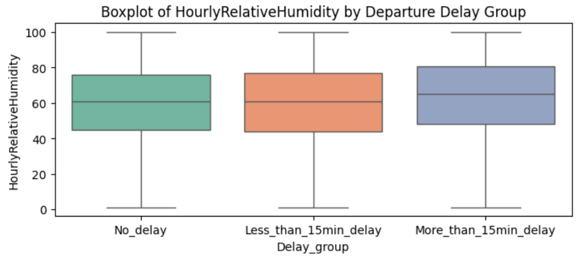
- 3 groups
- No delay, <15 min delay, >15 min delay

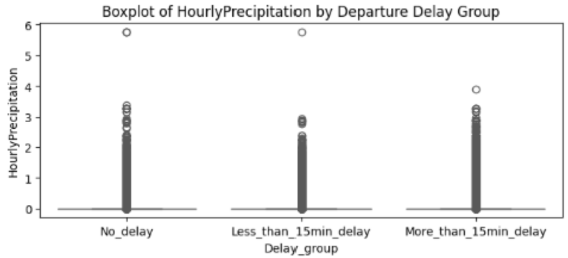


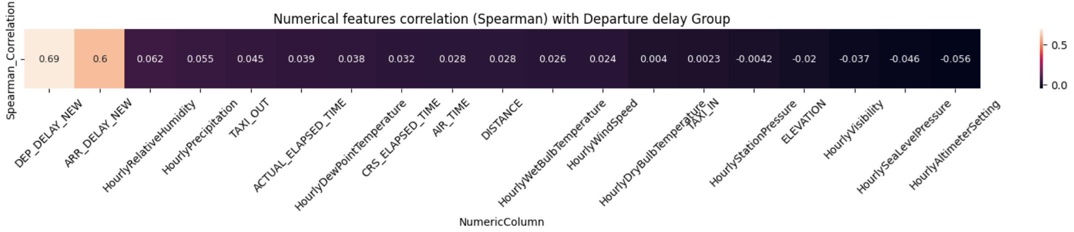
Better balance of distribution without losing business meaning

5 Year - Exploratory Data Analysis - Numerical Variables





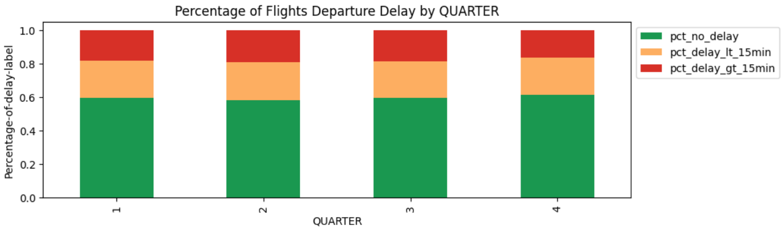




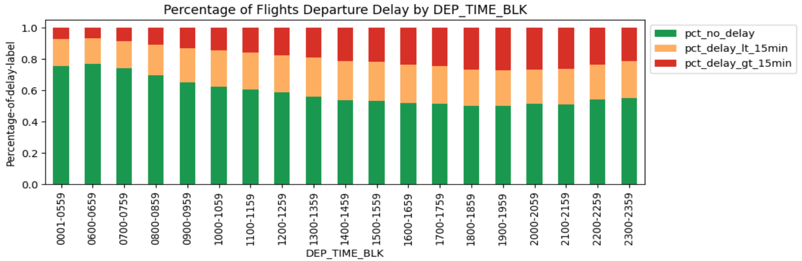
- ~20 numerical variables, with the rest as categorical.
- The numerical variables have very different ranges, and some have very narrow span.
- Spearman correlation between DEP_DELAY_GROUP and each numerical variable show that, apart from those not ex-ante available variables (e.g. departure delay time, taxi-out), none of them has high correlation with target.
- Hourly humidity, precipitation have higher correlation with target, and somewhat different distribution across delay groups.

5 Year - Exploratory Data Analysis-Categorical Variables

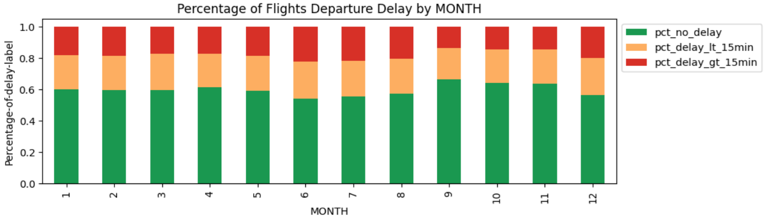
Quarter



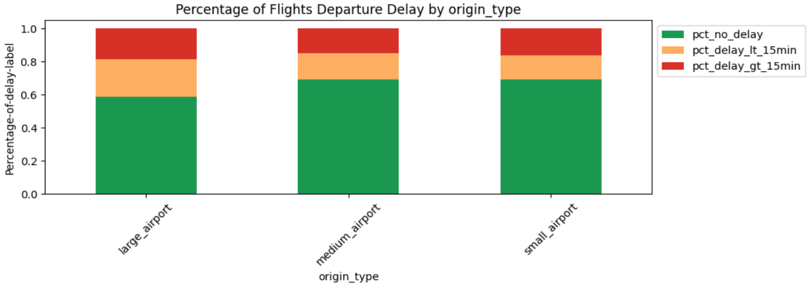
Time of Day (Departure, Arrival)



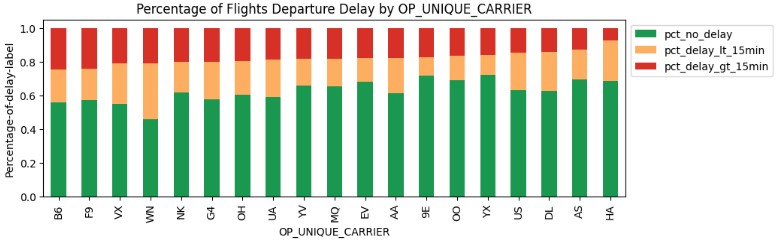
Month of the Year



Airport Size (Departure, Arrival)

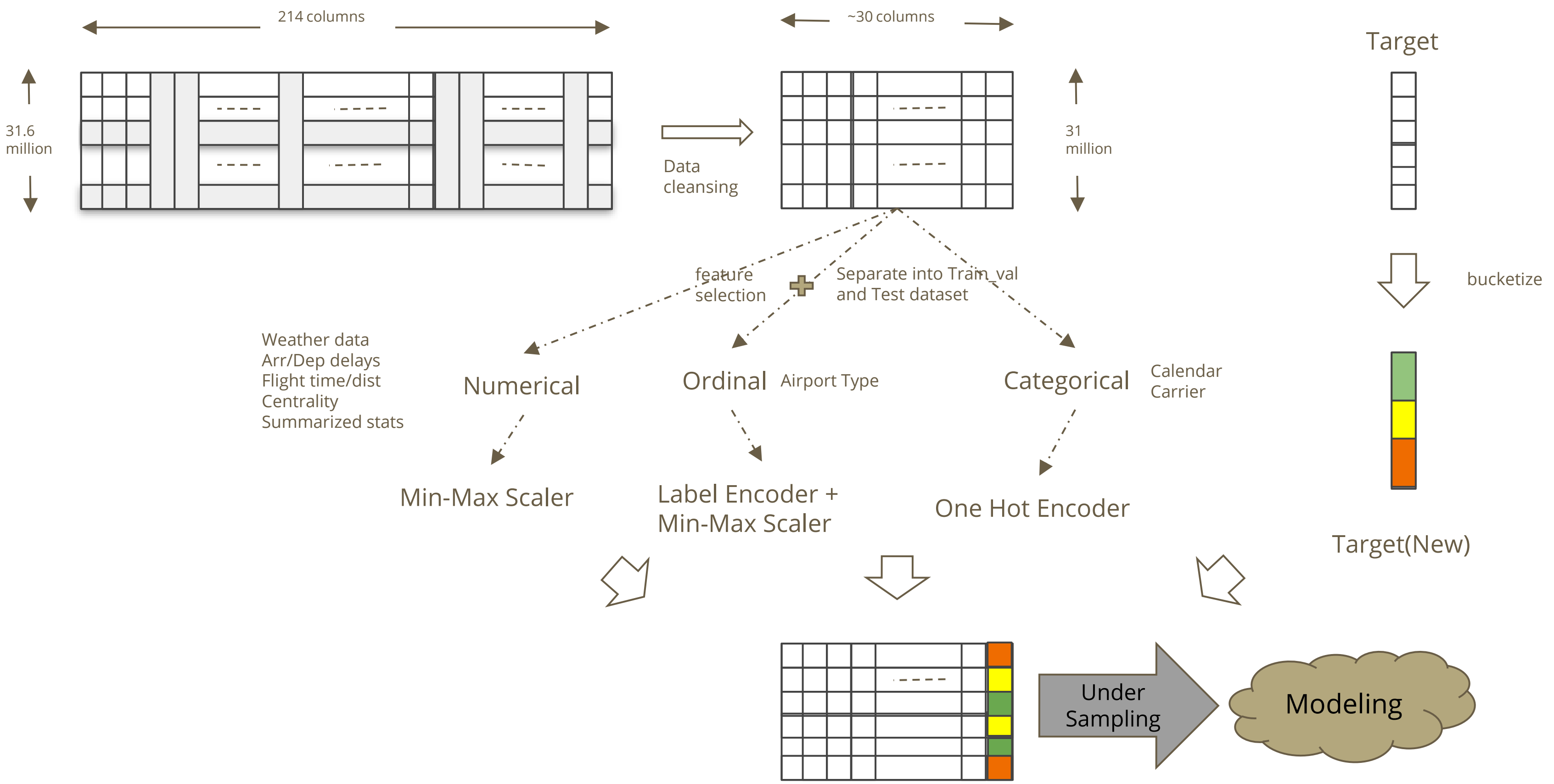


By Carrier

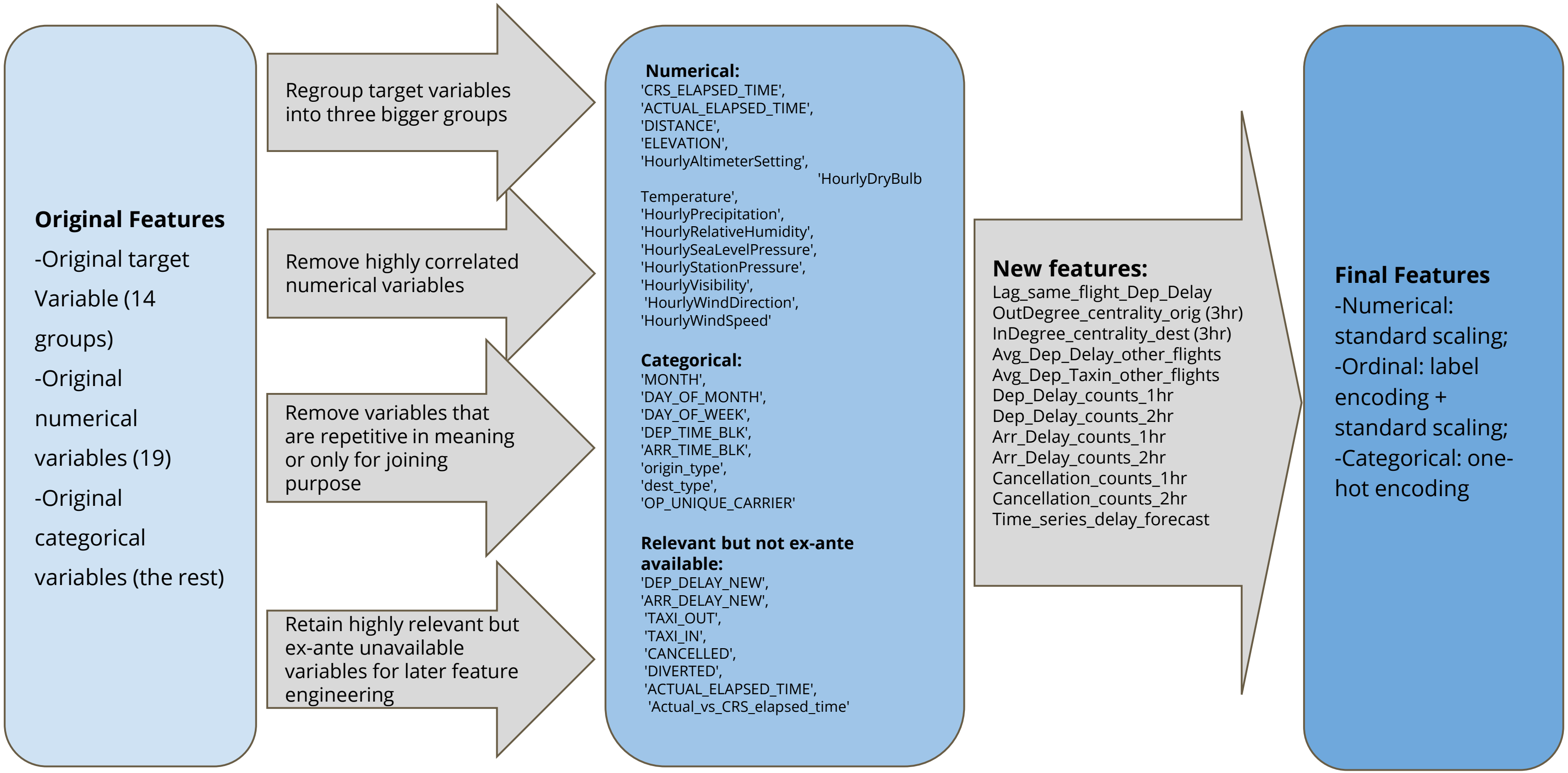


We examined relevant categorical variables, and noted several categorical variables whose class-wise distribution (count percentage) of delay groups show notable variation. This indicates their informational power in predicting delays. Examples include scheduled departure / arrival time of the day, quarter, month of the year, airport size, or a particular carrier.

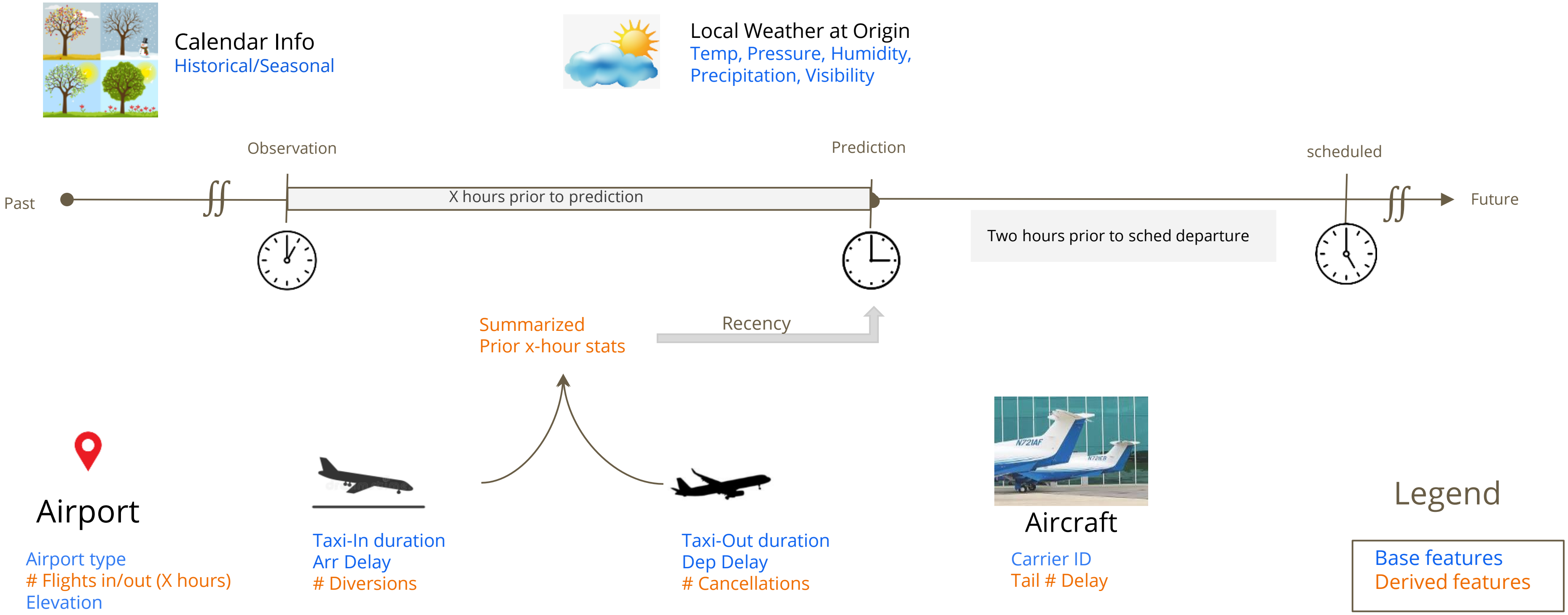
Feature Segregation and Encoding



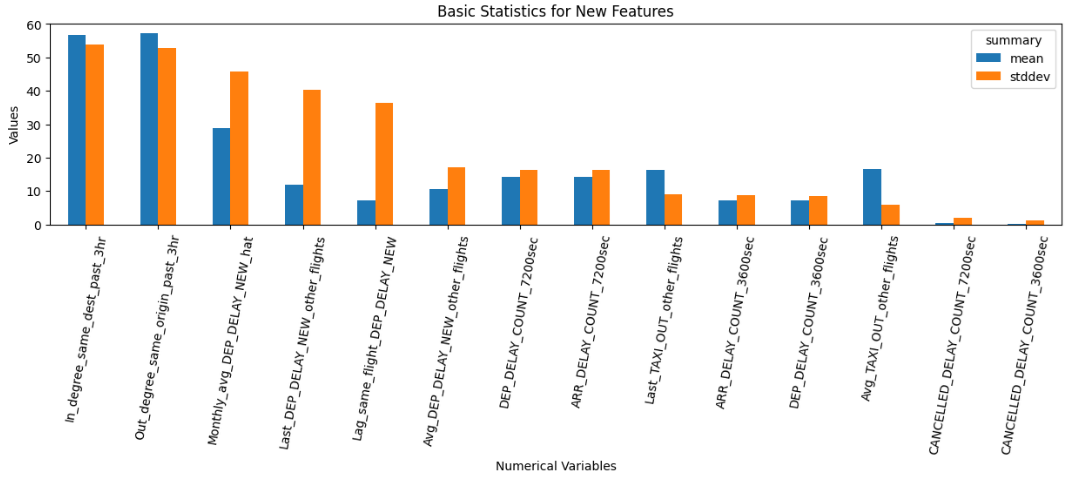
Feature Selection Flowchart

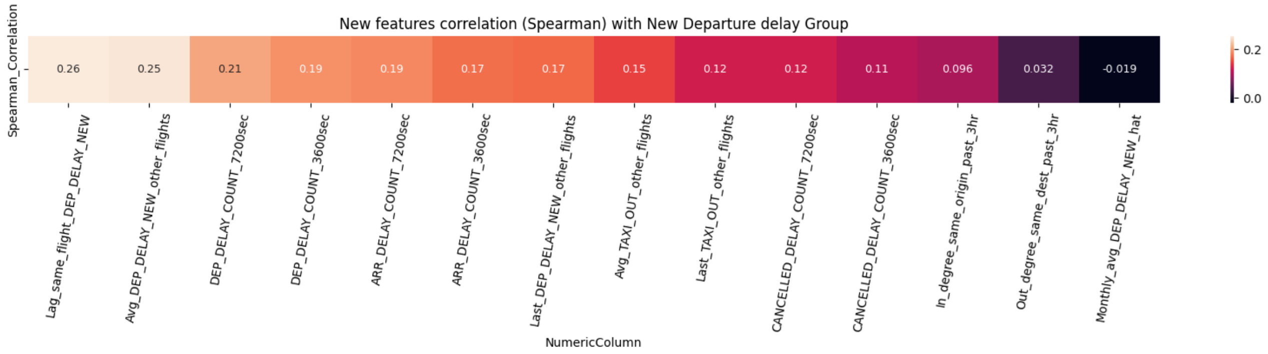


Derived features



New Features EDA

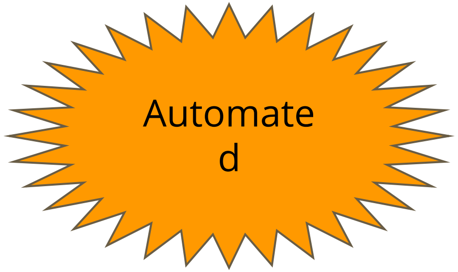


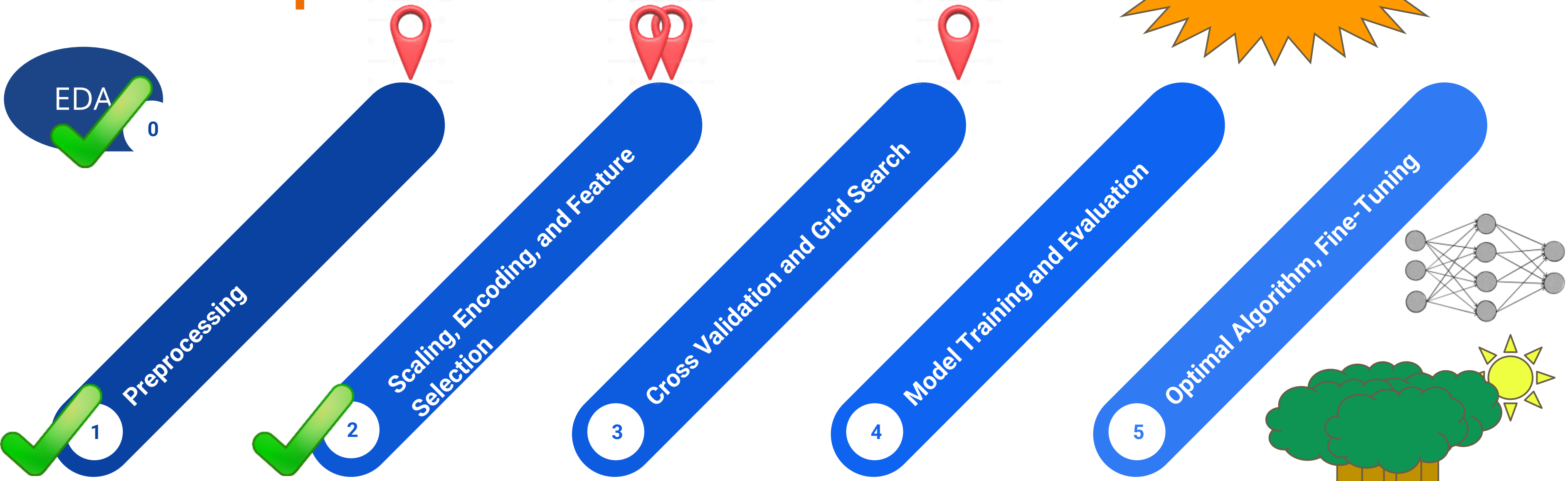


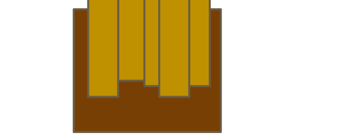
Key Takeaways:

- New features are of different magnitude and dispersion, which warrants the needs of scaling (min-max)
- The spearman correlation of new features with target variables are overall much higher than the most correlated original feature (without leakage)

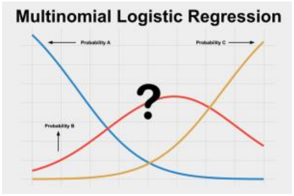
Model Pipeline - Overview





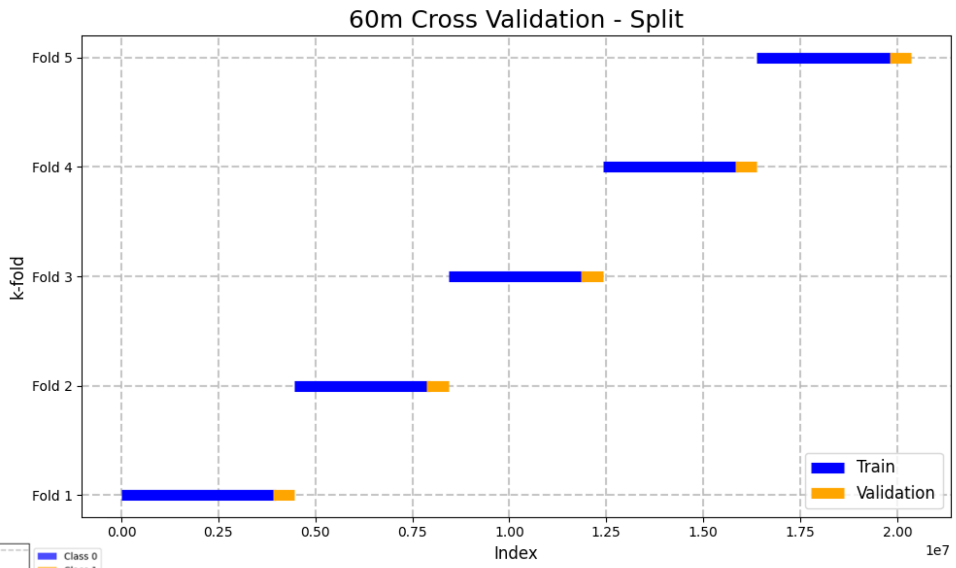


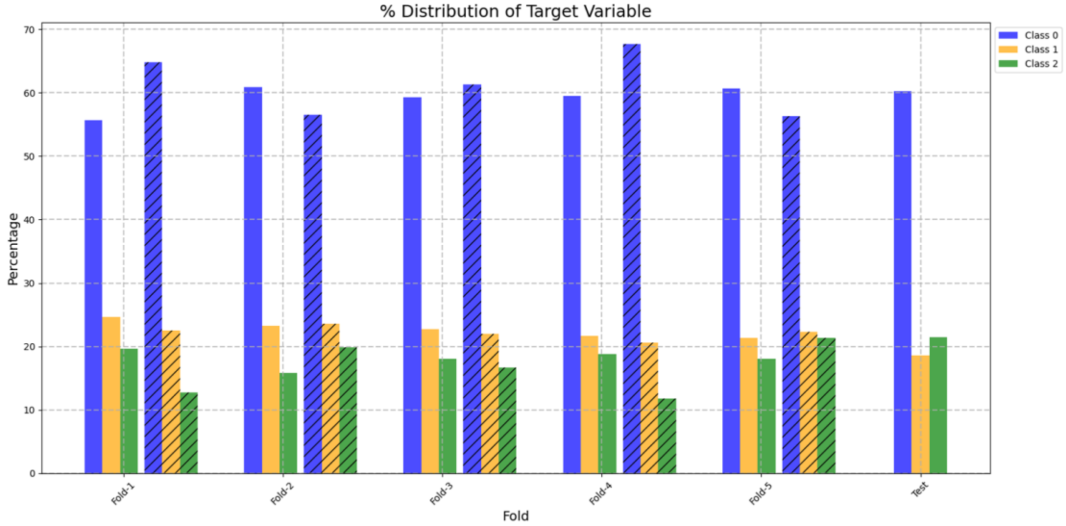
- **Baseline:** *Multinomial Logistic Regression*
- **Loss Function:** *Categorical Cross Entropy*
- **Metric: F2**
  - **Positive Class:** *Delayed*
  - **Negative Class:** *Not Delayed*

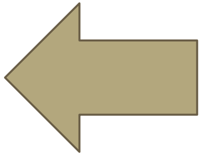


Cross Validation Strategy

- Train/Val/Test Split
- Checkpointed each split







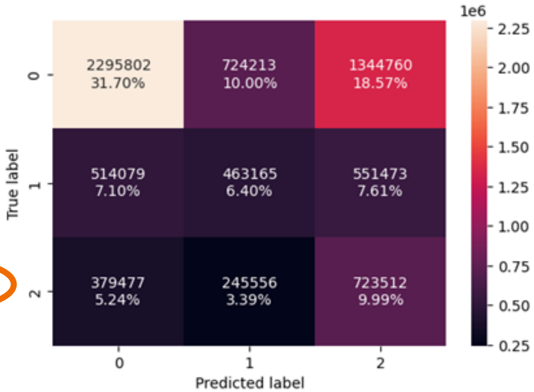
Uniform Dist.

Baseline Models (with and without features)

Logistic Regression Without New Features

- Minimal Regularization is used based on cross validation
- Achieves better recall than precision

	Train	Test
F2.0 score	0.500649	0.451409
Class 0 F2.0-Measure	0.487617	0.555926
CLass 0 precision	0.477258	0.719832
Class 0 recall	0.490277	0.525984
Class 1 F2.0-Measure	0.371835	0.306821
CLass 1 precision	0.437269	0.323228
Class 1 recall	0.358426	0.302976
Class 2 F2.0-Measure	0.50064	0.451409
CLass 2 precision	0.447501	0.276176
Class 2 recall	0.515969	0.536513

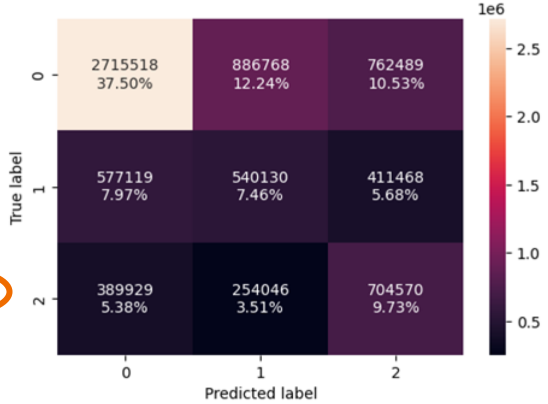


Class 0: No Delay
Class 1: < 15 min Delay
Class 2: > 15 min Delay

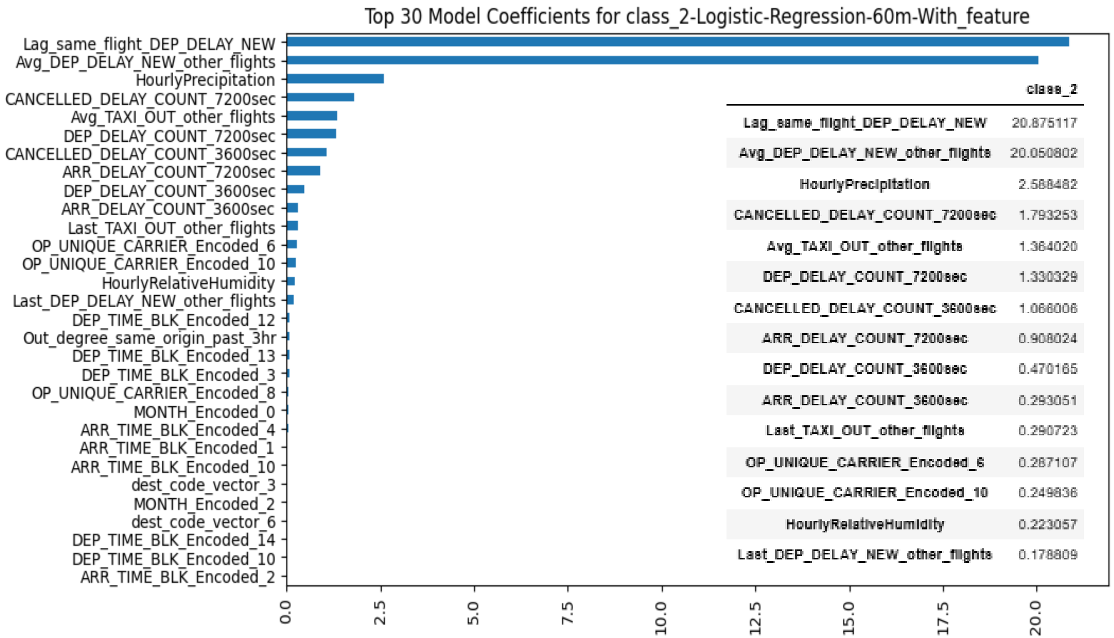
Logistic Regression With New Features

- **New features add 3% F2 for Class 2 in LR**
- Achieves better recall than precision

	Train	Test
F2.0 score	0.508744	0.484393
Class 0 F2.0-Measure	0.582746	0.642219
CLass 0 precision	0.491940	0.737398
Class 0 recall	0.610939	0.622144
Class 1 F2.0-Measure	0.404280	0.346423
CLass 1 precision	0.455555	0.321325
Class 1 recall	0.393215	0.353322
Class 2 F2.0-Measure	0.508744	0.484393
CLass 2 precision	0.556567	0.375065
Class 2 recall	0.498046	0.522467



Baseline Models - Feature Importance (with new features)



	class_2
Lag_same_flight_DEP_DELAY_NEW	20.875117
Avg_DEP_DELAY_NEW_other_flights	20.050802
HourlyPrecipitation	2.588482
CANCELLED_DELAY_COUNT_7200sec	1.793253
Avg_TAXI_OUT_other_flights	1.364020
DEP_DELAY_COUNT_7200sec	1.330329
CANCELLED_DELAY_COUNT_3600sec	1.068006
ARR_DELAY_COUNT_7200sec	0.908024
DEP_DELAY_COUNT_3600sec	0.470165
ARR_DELAY_COUNT_3600sec	0.293051
Last_TAXI_OUT_other_flights	0.290723
OP_UNIQUE_CARRIER_Encoded_6	0.287107
OP_UNIQUE_CARRIER_Encoded_10	0.249836
HourlyRelativeHumidity	0.223057
Last_DEP_DELAY_NEW_other_flights	0.178809

Key Takeaways:

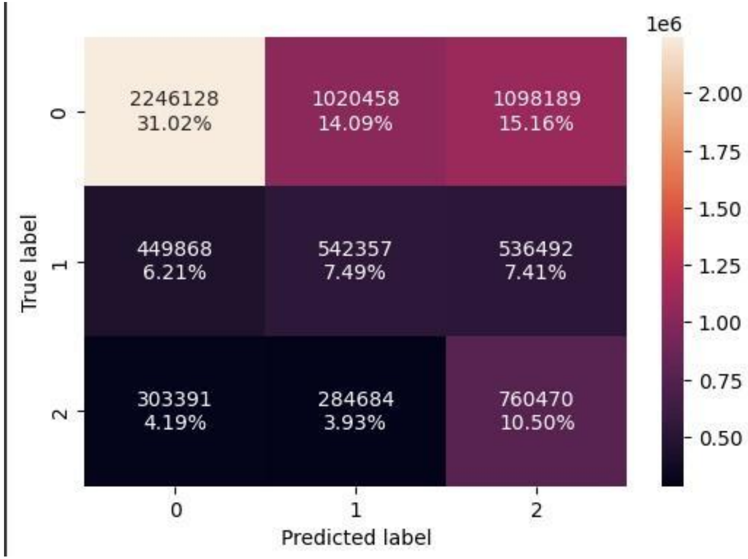
- Logistic Regression With New Features flags lag_same_flight_dep_delay and Avg_Dep_Delay_other_flights as top two features
- It also flags most of our new features as Top 10 features, including Cancelled_delay_count_7200sec, Departure_delay_count_7200sec, Avg_tax_out_other_flights, etc.

[160, 8, 8, 3]
l-bfgs

	Train	Test
F2 score	0.575255	0.548948
Class 0 F2	0.540691	0.610016
Precision 0	0.491474	0.748862
Recall 0	0.600862	0.514603
Class 1 F2	0.398104	0.321281
Precision 1	0.444985	0.293563
Recall 1	0.360159	0.354779
Class 2 F2	0.509146	0.406267
Precision 2	0.517579	0.317504
Recall 2	0.500983	0.563919

Multilayer Perceptron Classifier (MPC)

- Lack of hyperparameters
- Wider and Deeper Networks
- Narrow and Shallow Networks
- Lack of flexibility
  - Only fully connected layers
  - No GPU support out of the box



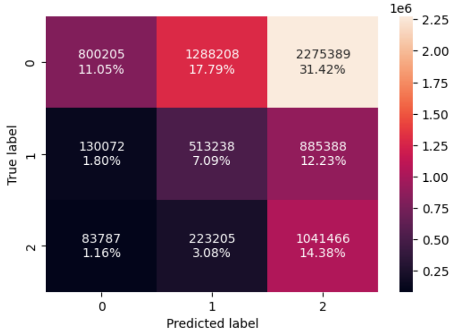
	EMA F2	layers	solver	blockSize	stepSize	epochs	Training Time (s)	Fold-0 Train Time (s)	Fold-1 Train Time (s)	Fold-2 Train Time (s)	Fold-3 Train Time (s)	Fold-4 Train Time (s)
0	0.498330	[160, 8, 8, 4, 3]	gd	32	0.001	50	565.195893	135.279693	96.451866	109.762344	113.899849	109.802135
1	0.408242	[160, 3, 3]	l-bfgs	32	0.001	50	297.840303	73.452858	51.999708	55.364368	58.588688	58.434675
2	0.408242	[160, 3, 3]	l-bfgs	32	0.001	50	298.066462	73.564465	51.917611	55.416429	58.646824	58.521127
3	0.406815	[160, 160, 64, 8, 8, 3]	l-bfgs	32	0.001	50	10853.416550	2513.591729	1983.874367	1614.906628	2379.455491	2361.588327
4	0.406815	[160, 160, 64, 8, 8, 3]	l-bfgs	32	0.001	50	10133.920616	2236.871659	1673.124784	2066.904745	1904.877495	2252.141926
5	0.392358	[160, 8, 8, 3]	l-bfgs	32	0.001	50	556.757080	135.217397	91.277122	106.676224	111.285250	112.301079
6	0.386381	[160, 10, 10, 3]	l-bfgs	32	0.001	50	641.682976	156.715210	105.403404	128.300383	131.681844	119.582130
7	0.382084	[160, 8, 8, 4, 3]	l-bfgs	32	0.001	50	687.850525	175.317339	117.248723	131.509670	134.574828	129.199957
8	0.382084	[160, 8, 8, 4, 3]	l-bfgs	32	0.001	50	689.202822	175.564271	117.480313	131.789813	134.875048	129.493370
9	0.367649	[160, 4, 4, 3]	l-bfgs	32	0.001	50	888.987086	218.732579	158.006508	165.241241	171.214697	175.792054
10	0.367649	[160, 16, 16, 3]	l-bfgs	32	0.001	50	888.689387	218.333938	157.850456	165.047561	171.233042	176.224383
11	0.367464	[160, 8, 3, 3]	l-bfgs	32	0.001	50	473.087354	112.354304	79.963029	95.523544	91.240550	94.005922
12	0.364155	[160, 12, 12, 3]	l-bfgs	32	0.001	50	847.244021	214.740585	140.437548	155.285935	160.776265	176.003683
13	0.000000	[160, 8, 3, 3]	gd	32	0.001	50	419.738375	100.675426	71.959141	81.065878	84.641517	81.396407
14	0.000000	[160, 16, 16, 3]	gd	32	0.001	50	783.328395	187.620962	133.739990	152.474692	158.004630	151.488115
15	0.000000	[160, 160, 64, 8, 8, 3]	gd	32	0.001	50	8396.656472	1952.079107	1259.210449	1752.337119	1970.643686	1462.386104
16	0.000000	[160, 8, 64, 8, 3]	gd	32	0.001	50	1486.436466	356.276275	252.634256	287.651898	299.831332	290.042700
17	0.000000	[160, 16, 16, 3]	gd	32	0.001	50	782.516460	187.782360	133.416055	152.143484	158.336419	150.838136
18	0.000000	[160, 4, 4, 3]	gd	32	0.001	50	782.566204	187.921757	133.397714	152.535010	157.838119	150.873598
19	0.000000	[160, 16, 16, 3]	gd	32	0.001	50	782.852146	187.958369	133.189114	152.560590	158.320090	150.823976
20	0.000000	[160, 8, 64, 8, 3]	gd	32	0.001	50	1485.221421	356.058995	252.305789	287.221585	300.158449	289.476598
21	0.000000	[160, 8, 3, 3]	gd	32	0.001	50	419.501984	100.484623	71.505780	81.753873	84.315862	81.441839
22	0.000000	[160, 12, 12, 3]	gd	32	0.001	50	626.894920	151.920338	106.688658	121.060088	125.695304	121.530525
23	0.000000	[160, 3, 3]	gd	32	0.001	50	281.074852	67.546574	48.131046	54.492408	56.544292	54.360526

Neural Networks: Extra Models

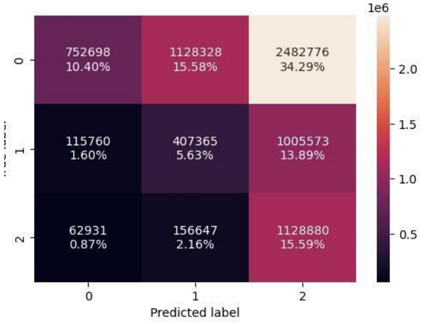
- Custom Callback - Penalized Class 2 more
- Data isn't balanced, so monitoring loss does not accurately reflect our goal to reduce the F2 score
- TCN
    - 2 convolutional stacks was the best
    - Kernel Size of 3 was optimal. Raising it to 5 made it worse
    - Added higher penalization for mispredicting class 2. Improved score of class 2, but decreased others
        - Adding higher penalties to class 1 and 2 improved dramatically
    - Increasing nb_stacks to 3 improved score
    - Increasing dilation made it worse
- GRU
    - Just like MPC Narrower and Shallower hidden layers performed best. However, the gates implementation made balanced the classes a little better
- Overall, the models learned to prioritize Class 2 as it would cost airlines more money.

Class 0: No Delay
Class 1: < 15 min Delay
Class 2: > 15 min Delay

GRU



TCN



NN Architecture	Train [Class 0, Class 1, Class 2]	Test [Class 0, Class 1, Class 2]
Temporal Convolutional Network (TCN)	[0.3772327 , 0.43125454, 0.646092]	[0.2046866 , 0.26089278, 0.56381637]
Gated Recurrent Unit	[0.22143069, 0.38430917, 0.645049]	[0.21663143, 0.3152783, 0.5426521]

XGBoost Multiclass Classifier - CV Results

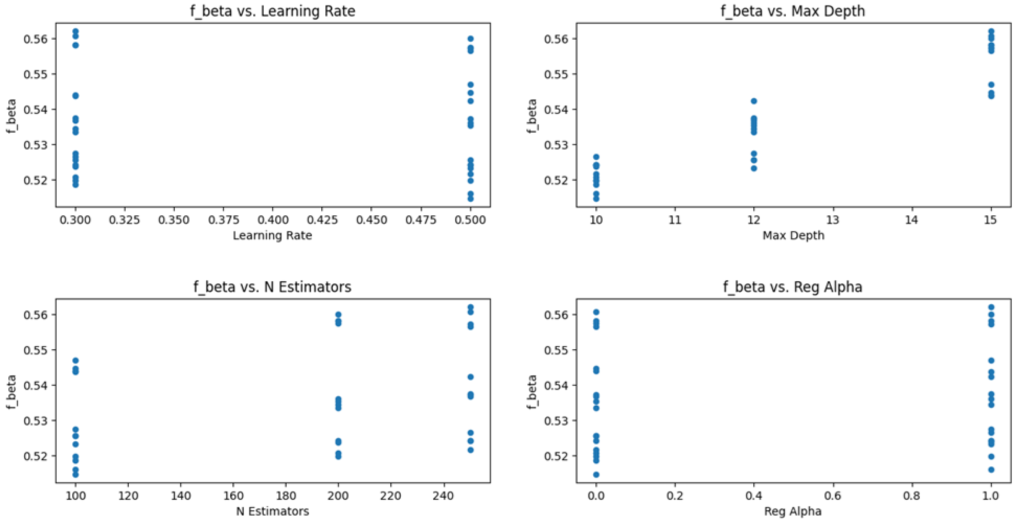
	Learning Rate	Max Depth	Gamma	Colsample Bytree	Colsample Bylevel	N Estimators	Reg Alpha	f_beta
0	0.3	15	0	0.8	0.9	250	1	0.5621
1	0.3	15	0	0.8	0.9	250	0	0.560671
2	0.5	15	0	0.8	0.9	200	1	0.559867
3	0.3	15	0	0.8	0.9	200	1	0.558225
4	0.3	15	0	0.8	0.9	200	0	0.558182
5	0.5	15	0	0.8	0.9	200	0	0.557499
6	0.5	15	0	0.8	0.9	250	1	0.557121
7	0.5	15	0	0.8	0.9	250	0	0.556606
8	0.5	15	0	0.8	0.9	100	1	0.547019
9	0.5	15	0	0.8	0.9	100	0	0.544729
10	0.3	15	0	0.8	0.9	100	0	0.544019
11	0.3	15	0	0.8	0.9	100	1	0.5436
12	0.5	12	0	0.8	0.9	250	1	0.542298
13	0.3	12	0	0.8	0.9	250	1	0.537402
14	0.5	12	0	0.8	0.9	250	0	0.5372
15	0.3	12	0	0.8	0.9	250	0	0.536801
16	0.5	12	0	0.8	0.9	200	1	0.5361
17	0.5	12	0	0.8	0.9	200	0	0.53535
18	0.3	12	0	0.8	0.9	200	1	0.534302
19	0.3	12	0	0.8	0.9	200	0	0.5334
20	0.3	12	0	0.8	0.9	100	1	0.5274
21	0.3	10	0	0.8	0.9	250	1	0.526376
22	0.5	12	0	0.8	0.9	100	0	0.5256
23	0.3	12	0	0.8	0.9	100	0	0.525476
24	0.5	10	0	0.8	0.9	200	1	0.5242
25	0.5	10	0	0.8	0.9	250	1	0.52414
26	0.3	10	0	0.8	0.9	250	0	0.524024
27	0.3	10	0	0.8	0.9	200	1	0.523747
28	0.5	12	0	0.8	0.9	100	1	0.5232
29	0.5	10	0	0.8	0.9	250	0	0.5216
30	0.3	10	0	0.8	0.9	200	0	0.520606
31	0.3	10	0	0.8	0.9	100	1	0.519744
32	0.5	10	0	0.8	0.9	200	0	0.51962
33	0.3	10	0	0.8	0.9	100	0	0.5186
34	0.5	10	0	0.8	0.9	100	1	0.51606
35	0.5	10	0	0.8	0.9	100	0	0.5147

	Best_param
Learning Rate	0.3
Max Depth	15.0
Gamma	0.0
Colsample Bytree	0.8
Colsample Bylevel	0.9
N Estimators	250.0
Reg Alpha	1.0

Key Takeaways:

- More depth helps meaningfully
- More n_estimator helps as well in general
- 0.3 vs 0.5 learning rate depends on the depth
- Although not obviously, model prefers L1 regularization than not
- To curb overfitting we use 0.8 for Colsample by Tree, and 0.9 for Colsample by Level (instead of 1)

f_beta vs. parameters



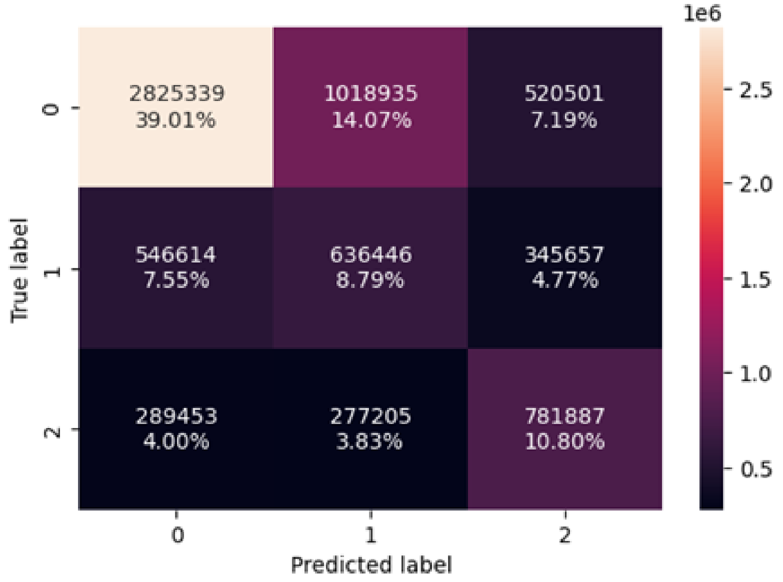
XGBoost Multiclass Classifier - Final Model Results

	Train	Test
F2.0 score	0.574676	0.555142
Class 0 F2.0-Measure	0.635475	0.668862
CLass 0 precision	0.541354	0.771654
Class 0 recall	0.664351	0.647305
Class 1 F2.0-Measure	0.448867	0.395433
CLass 1 precision	0.486584	0.329324
Class 1 recall	0.440334	0.416327
Class 2 F2.0-Measure	0.574676	0.555142
CLass 2 precision	0.644736	0.474433
Class 2 recall	0.559477	0.579800

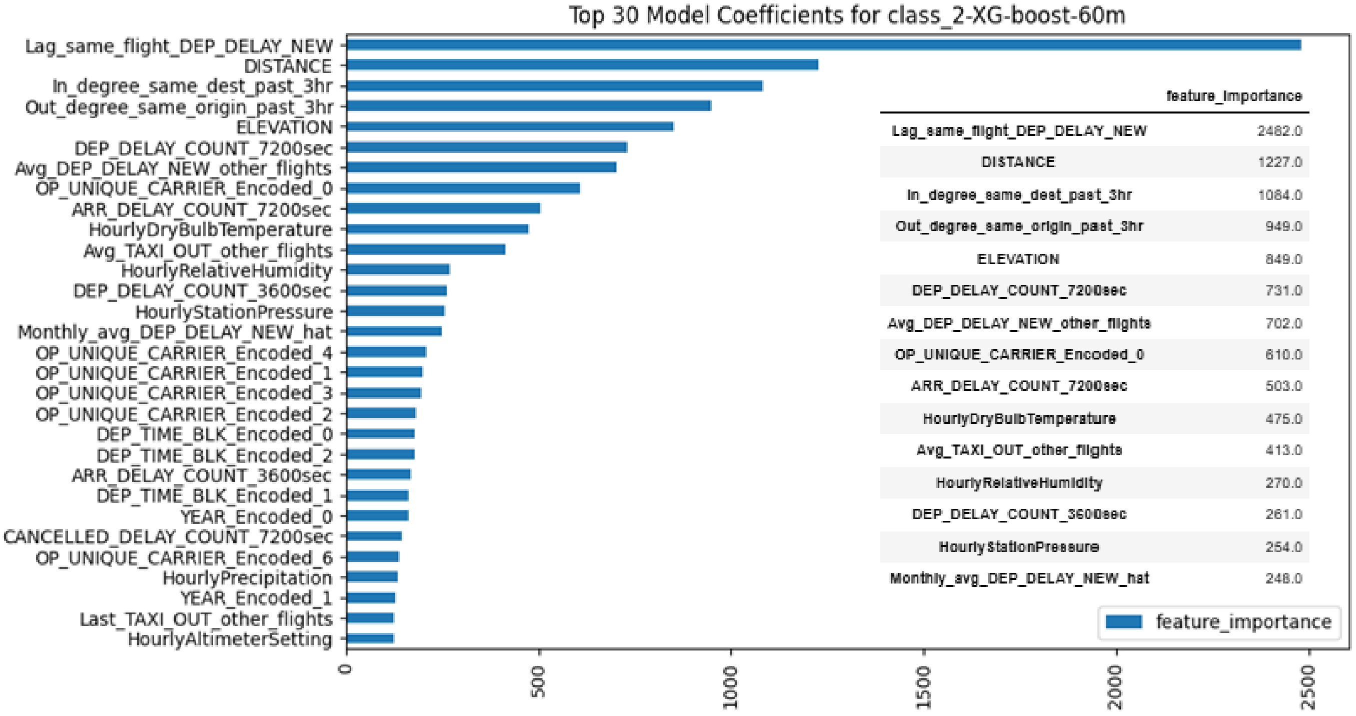
Class 0: No Delay
Class 1: < 15 min Delay
Class 2: > 15 min Delay

Key Takeaways:

- Test results line up with train and cross validation results very well (56% vs 57% F2), indicates successful cross validation strategy
- Improves baseline on both recall and precision, especially precision
- Test did slightly better in recall (58%) than train, and weaker in precision
- XGBoost model parallels very well with big data set. Final model only takes 5.4 minutes



XGBoost Multiclass Classifier - Feature Importance

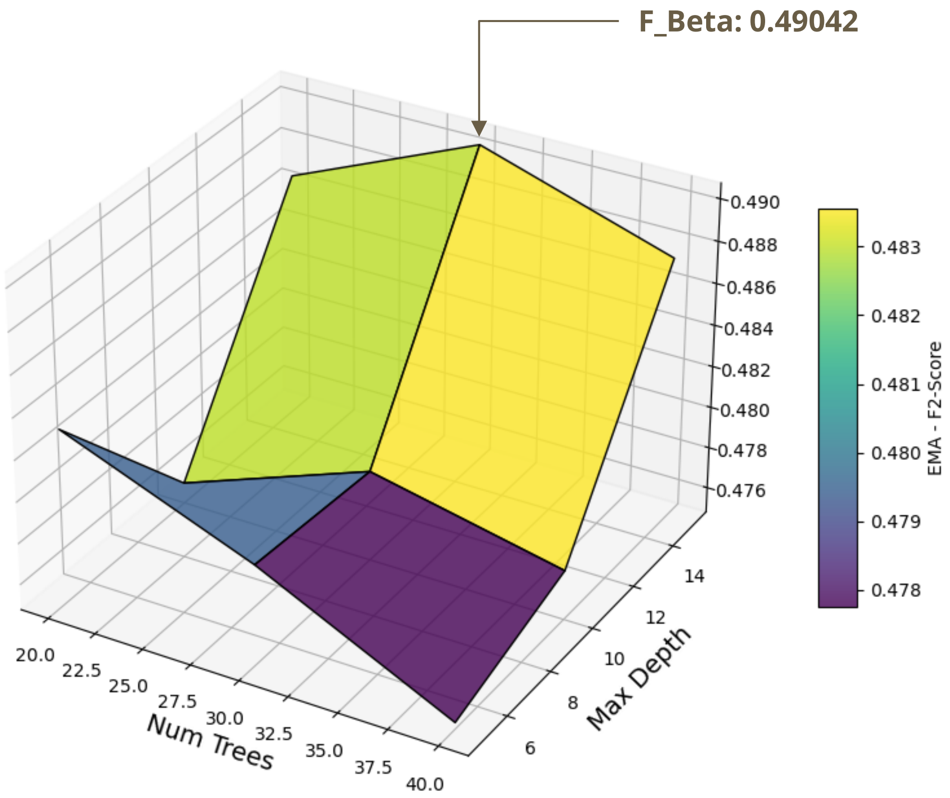


	feature_importance
Lag_same_flight_DEP_DELAY_NEW	2482.0
DISTANCE	1227.0
In_degree_same_dest_past_3hr	1084.0
Out_degree_same_origin_past_3hr	949.0
ELEVATION	849.0
DEP_DELAY_COUNT_7200sec	731.0
Avg_DEP_DELAY_NEW_other_flights	702.0
OP_UNIQUE_CARRIER_Encoded_0	610.0
ARR_DELAY_COUNT_7200sec	503.0
HourlyDryBulbTemperature	475.0
Avg_TAXI_OUT_other_flights	413.0
HourlyRelativeHumidity	270.0
DEP_DELAY_COUNT_3600sec	261.0
HourlyStationPressure	254.0
Monthly_avg_DEP_DELAY_NEW_hat	248.0

Key Takeaways:

- Lag_same_flight_dep_delay is again flagged as most important
- Different from logistic regression, XGBoost flags degree centrality, distance, elevation as top features, besides the more linear features such as count and average delay of other flights
- This is consistent with XGBoost's nonlinear modeling nature

Random Forest Multiclass Classifier - CV Results



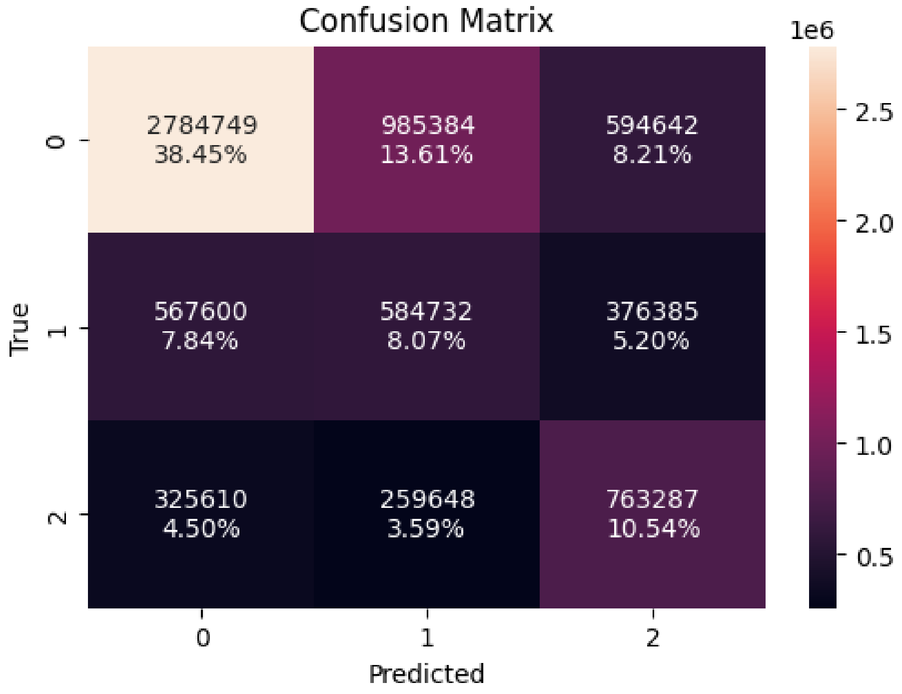
MaxDepth	5	10	15
NumTrees			
20	0.4832	0.47578	0.48657
30	0.4796	0.47915	0.49042
40	0.4751	0.47717	0.48747

Note: MaxDepth > 15 results in out of memory

Random Forest Multiclass Classifier

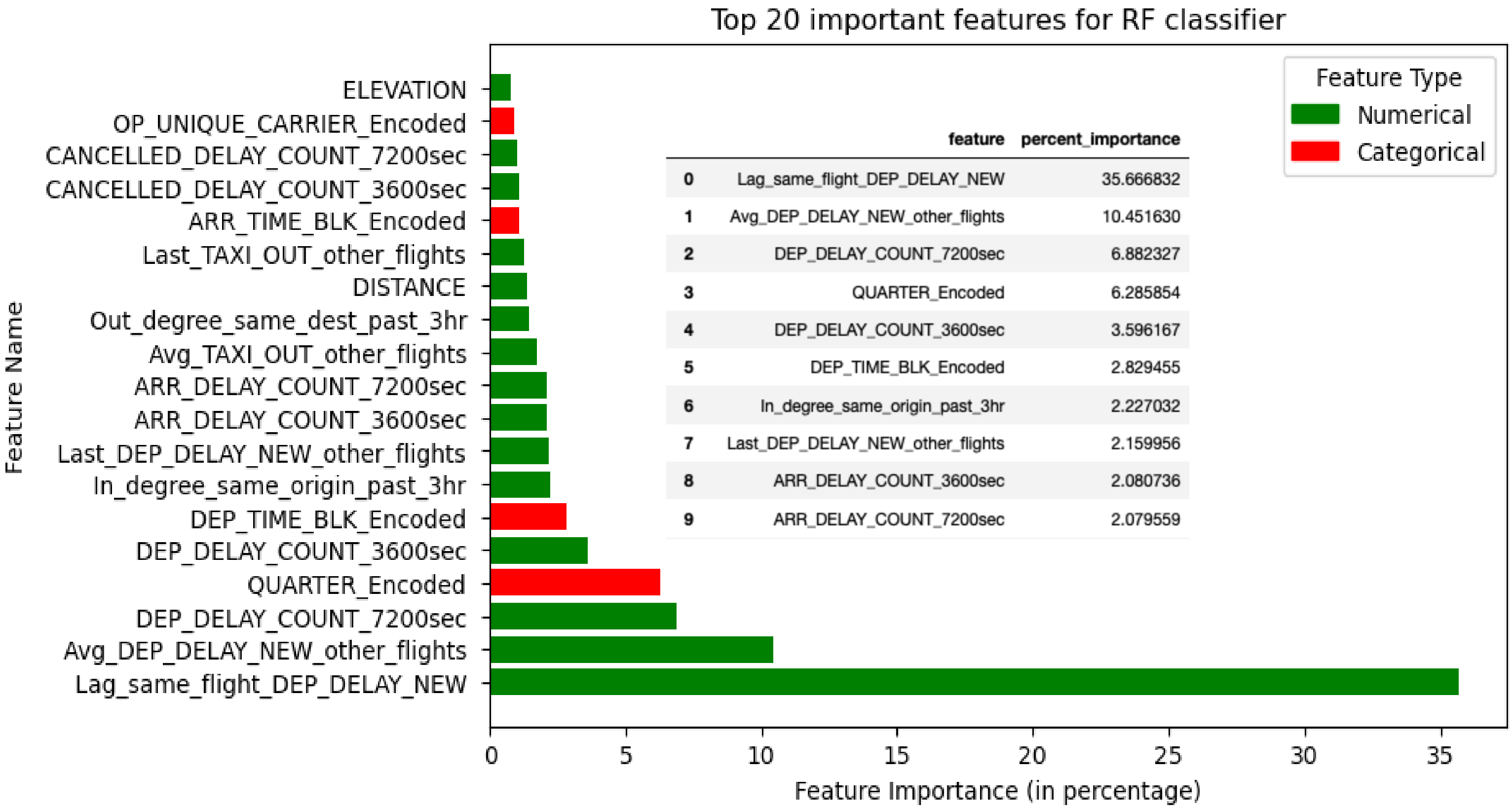
Test Class 2.0 F2.0-Measure = 0.5353774584084662
Test Class 2.0 precision = 0.4401088845503179
Test Class 2.0 recall = 0.566007808415737

	Train	Test
F2.0 score	0.522786	0.535377
Class 0 F2.0-Measure	0.686184	0.658736
CLass 0 precision	0.754128	0.757145
Class 0 recall	0.671069	0.638005
Class 1 F2.0-Measure	0.391187	0.368004
CLass 1 precision	0.342762	0.319567
Class 1 recall	0.405510	0.382499
Class 2 F2.0-Measure	0.522786	0.535377
CLass 2 precision	0.472817	0.440109
Class 2 recall	0.536974	0.566008



Class 0: No Delay
Class 1: < 15 min Delay
Class 2: > 15 min Delay

Random Forest Multiclass Classifier - Feature Importance



	feature	percent_importance
0	Lag_same_flight_DEP_DELAY_NEW	35.666832
1	Avg_DEP_DELAY_NEW_other_flights	10.451630
2	DEP_DELAY_COUNT_7200sec	6.882327
3	QUARTER_Encoded	6.285854
4	DEP_DELAY_COUNT_3600sec	3.596167
5	DEP_TIME_BLK_Encoded	2.829455
6	In_degree_same_origin_past_3hr	2.227032
7	Last_DEP_DELAY_NEW_other_flights	2.159956
8	ARR_DELAY_COUNT_3600sec	2.080736
9	ARR_DELAY_COUNT_7200sec	2.079559

Key Takeaways

About 80% of importance explained by just 10 features (of total 150)

8 out of those 10 are synthesized features

Key Takeaways and Model Recommendations

Delay Prediction	Weather Conditions	Recent Delays	Airport specific derived info	Maintenance Emergencies	Technical Failures	Other Factors
	Considered for modeling	Considered for modeling	Considered for modeling	Not considered for modeling	Not considered for modeling	Not considered for modeling

	Model	F2 Score for >15 min delay	Training Time	Suitable Application Scenario
BASELINE	**Logistic Regression**	48.4%	2.3 mins	If you have limited resources but need descent recall
	XGBoost	55.5%	5.4 mins	If you are able to parallelize but with limited computing power, and need decent F2 (both decent precision and recall)
	Random Forest	53.5%	14.8 mins	If you need interpretable results with minimal feature engineering, and need accuracy and F2 above average
BEST	**TCN**	56.4%	652 mins	Great for capturing long-range dependencies in time-serie data efficiently and can afford the higher computational cost
	MPC	40.6%	21.2 mins	Budget friendly NN, where you only need fully connected layers, simple hyperparameters, and you don't need spatial relationships between features. Great for an NN baseline.
	GRU	54.3%	478 mins	A lightweight alternative to LSTMs for time-series data to capture long-term dependencies

Thank You!

Andy Guinto

Muthumayan Madhayyan

Irene Na

Rahul Chugh