Cohort 28 Capstone Project for the Certificate of Data Science

Domain [Industry]: Aviation Analytics

- Jennifer Bamanya [project coordinator]
- Eva Gross

GitHub: https://github.com/georgetown-analytics/team-3

GEORGETOWN UNIVERSITY

Table of contents

Project topic

- Prediction of the operational efficiency of the overall aviation system in the US
    - Prediction of which flights will be delayed

Problem Statement and Background

The Enterprise Information Management (EIM) strategy is centered on using data more effectively to support the FAA's mission and driving innovation through improved access to data.
By analyzing data about weather, the location of aircraft, and other conditions throughout the National Airspace System, we aim to enhance the value of data to derive new insights in order to help the FAA make better decisions on how to improve the operational efficiency and better use of available airspace and airport capacity.

Flight delays, tarmac times, mishandled baggage, mishandled wheelchairs/scooters, denied boardings are among other factors affecting operational efficiency.

Factors affecting operational efficiency at the FAA may include the following

Cause	**Description**
Air Carrier [CarrierDelay]	The cause of the cancellation or delay was due to circumstances within the airline's control (e.g. maintenance or crew problems, mechanical issues, aircraft cleaning, baggage loading, fueling, etc.)
Extreme Weather [WeatherDelay]	Significant meteorological conditions (actual or forecasted) that, in the judgment of the carrier, delays or prevents the operation of a flight such as tornado, blizzard or hurricane.
National Airspace System (NAS) [NASDelay]	Delays and cancellations attributed to the national airspace system that refer to a broad set of conditions, such as non-extreme weather conditions, airport operations, heavy traffic volume, and air traffic control
Late-arriving aircraft [LateAircraftDelay]	A previous flight with the same aircraft arrived late, causing the present flight to depart late.

Security [SecurityDelay]	Delays or cancellations caused by evacuation of a terminal or concourse, re-boarding of aircraft because of security breach, inoperative screening equipment and/or long lines in excess of 29 minutes at screening areas.

Average aircraft delay can be referred to as an indication of airport capacity.
We shall evaluate the relationship between these factors to understand how these impact the operational efficiency of the National Airspace System (NAS)

Hypothesis

Assumptions:

- A flight is counted as "on time" if it operated less than 15 minutes later than the scheduled time shown in the carriers' Computerized Reservations Systems (CRS).
- A flight is counted as delayed anytime an aircraft departs at least 15 minutes later than scheduled
- A flight is counted as canceled when a scheduled flight will not depart at all from the airport.
- Departure performance is based on departure from the gate.
- Operational efficiency is measured by the number of flights operated on time
- The average weather patterns for an area over a long period of time (20 - 1,000,000 years) are determined by rainfall and temperature which are influenced by latitude, elevation and ocean currents

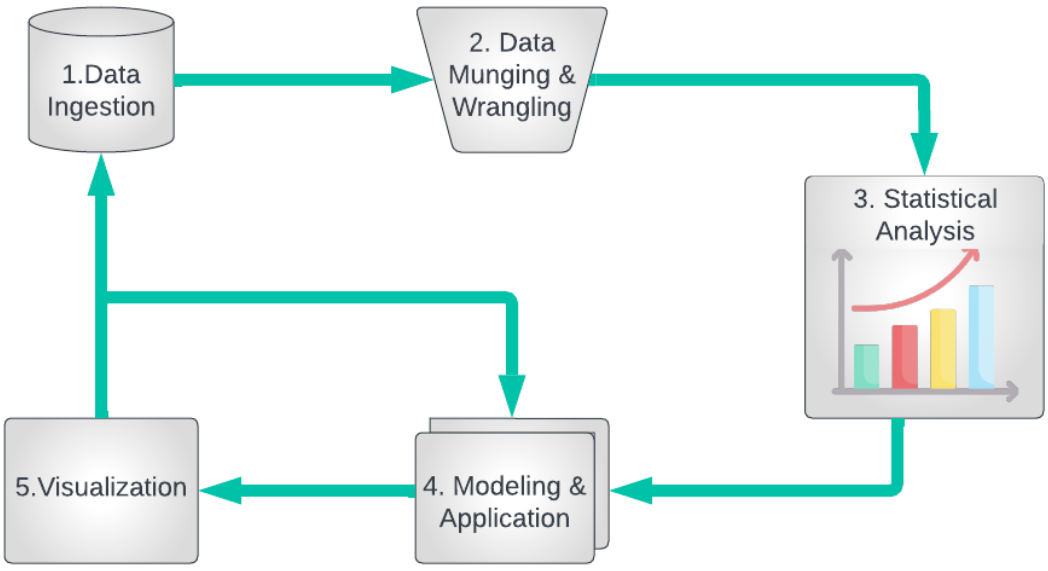
Goals of Analysis

The goal is to build a model to predict which flights will be delayed and to leveraging data analytics to

- derive insights by identifying and connecting complementary datasets
- improve air traffic monitoring, optimize operations, and increase overall profitability
- streamline maintenance, tracking performance, boosting customer experience, and assessing risk—often reducing costs along the way.

Solution Approach

Architecture & Design



Development Tools

Phase	Python Libraries
IDE	Jupyter, Visual Studio Code
Data Manipulation	Pandas, NumPy
Modeling	Scikit-Learn
Visualization	Matplotlib, Seaborn, Yellowbrick, Plotly, Django

Phase 1: Data Acquisition

Flight Data	
Source	Bureau of Transportation Statistics https://www.transtats.bts.gov/Tables.asp?QO_VQ=EFD&QO_anzr=Nv4yv0r%FDb0-gvzr%FDcr4s14zn0pr%FDQn6n&QO_fu146_anzr=b0-gvzr
Scope	January 2018 - August 2022
Target Variable	DepartureDelayMinutes
Size	(745540, 45)

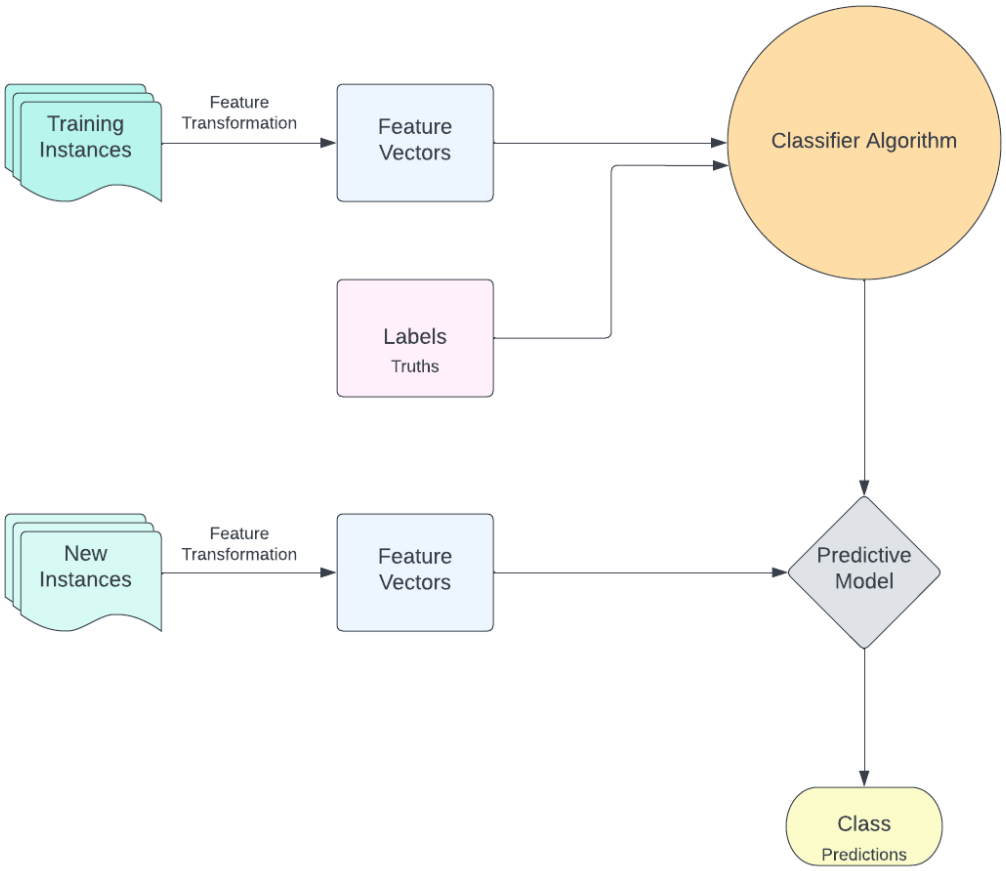
Phase 2: Data Preprocessing

During this phase, the raw data or input is transformed into another format to prepare it for the next phase [Model Training]

Stages	**Challenges & Actions performed**
Exploratory Data Analysis (EDA)	- Cleaning data - Handling missing or null values - Renaming columns - Dropping irrelevant columns
Feature Engineering	Feature creation Handling Ordinal Categories - Encoding Categorical Variables Handling imbalance of classes [imbalanced classification] - Tweaked threshold Feature transformation using scikit-learn transformer - StandardScaler - features transformed using z-score normalization (μ =0, σ = 1)
Feature Selection	Techniques used: - Rank 2D [Pearson] - Correlation Matrix with Heatmap - Feature Importance [Embedded Method] - Using RandomForestClassifier
Statistical Analysis	Understanding the data - Flight data of year 2020 was not consistent due to the effects of covid-19, and for this reason it will be excluded from from the model training phase

Phase 3: Model Building

During this phase we train/test with several models, learn some parameters, fine tune them and pick the one that gave us the best performance



Machine Learning Process

Stages	Description
Model Selection	Model Selection will be based on the accuracy level Candidate models: - DecisionTreeClassifier - RandomForestClassifier [Bagging] - XGBoost (eXtreme Gradient Boost) [Boosting]

	- KNeighborsClassifier - Gradient [Boosting] - Adaboost [Boosting] - Support Vector Machine (SVM)

Model Evaluation	- Cross Validation
	  - K Fold
	  - Stratified K-fold technique is used to overcome imbalance of classes in the training data

Hyperparameter Tuning	To increase the accuracy of our model, we use the following techniques:
	- RandomizedSearchCV
	  - Used to narrow down our results
	- GridSearchCV
	  - Used to select best hyperparameters
	
	Hyper Parameters of selected models
	
	● RandomForest:
	  - {'criterion': 'entropy', 'max_depth': 560, 'min_samples_split': 4, 'n_estimators': 400}
	● XGBoost:
	  - {'learning_rate': 0.1, 'max_depth': 9, 'n_estimators': 180}

Phase 4: Performance assessment

Models	Precision	Recall	Accuracy	F1 Score
SVC*				
SVC**				
DecisionTreeClassifier*	0.87955	0.879465	0.879465	0.879505
DecisionTreeClassifier**				
RandomForestClassifier*	0.93911	0.936063	0.936063	0.934905
RandomForestClassifier**	0.940897	0.937948	0.937948	0.936848
KNeighborsClassifier*	0.683911	0.695753	0.695753	0.68007
KNeighborsClassifier**				
XGBoostClassifier*	0.981388	0.981013	0.981013	0.980915
XGBoostClassifier**	0.987335	0.987154	0.987154	0.98711
GradientBoostingClassifier*	0.831404	0.791953	0.791953	0.767572
GradientBoostingClassifier**				
AdaBoostClassifier*	0.743733	0.71183	0.71183	0.660087
AdaBoostClassifier**				

* Default Parameters ** Hyper Parameters

Phase 4: Model Selection

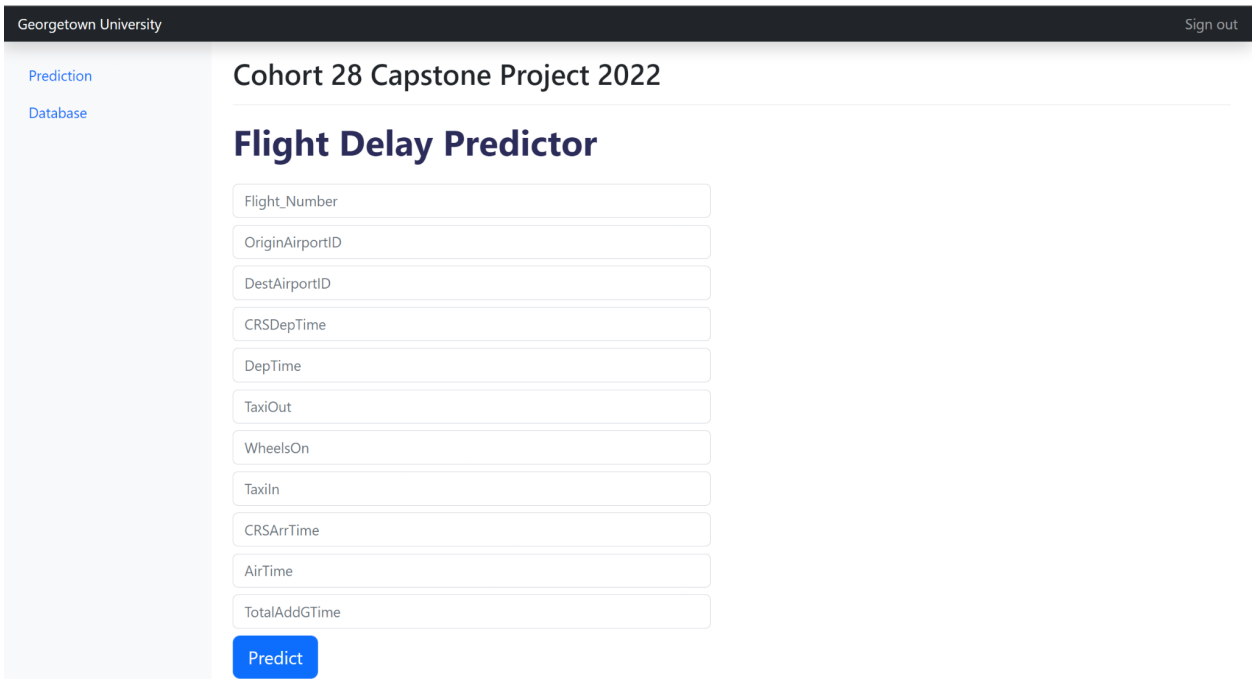
Based on the above results we picked the models with the best performance results

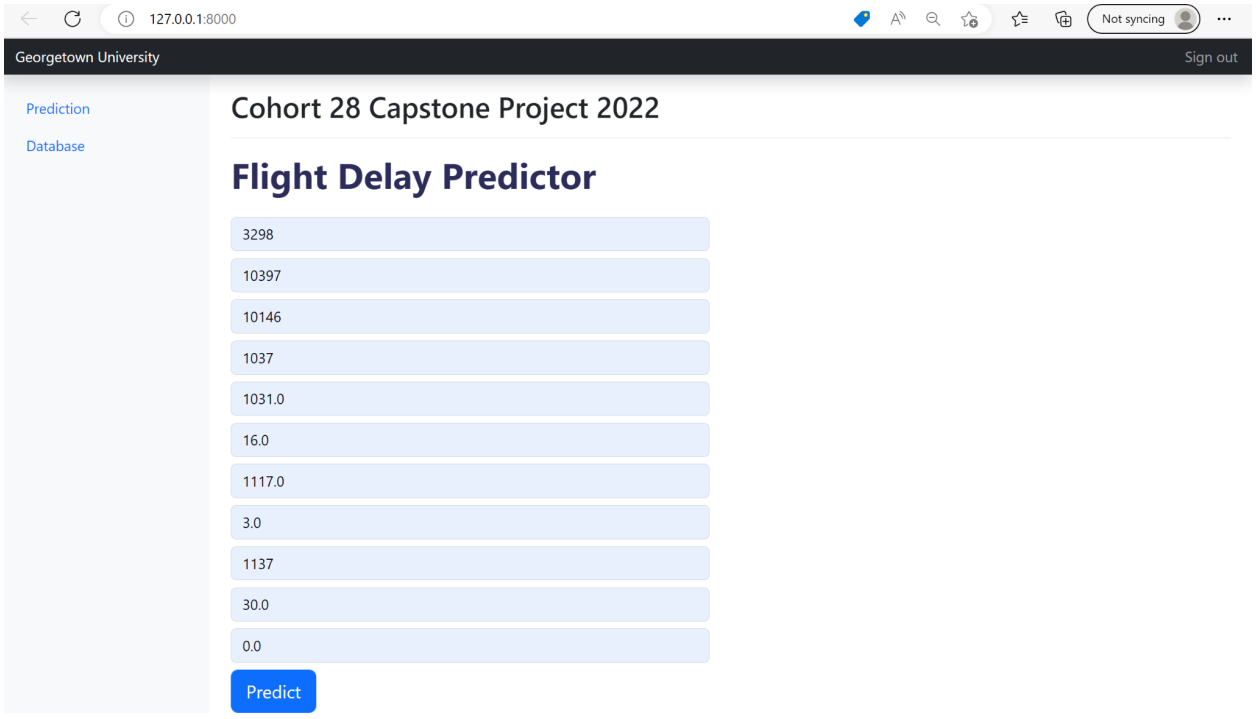
Models	Precision	Recall	Accuracy	F1 Score
RandomForestClassifier*	0.93911	0.936063	0.936063	0.934905
RandomForestClassifier**	0.940897	0.937948	0.937948	0.936848
XGBoostClassifier*	0.981388	0.981013	0.981013	0.980915
XGBoostClassifier**	0.987335	0.987154	0.987154	0.98711

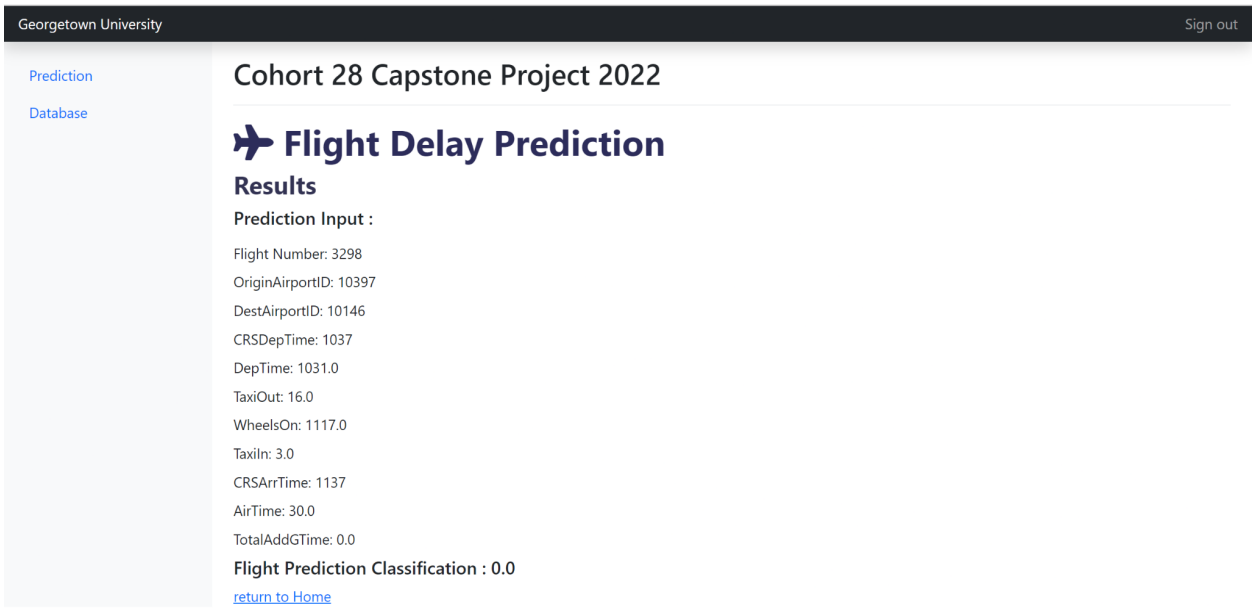
* Default Parameters ** Hyper Parameters

Phase 5: Modeling Deployment

During this phase we create a pickle file which is a serialized file containing our model, and we use this in a Django application to display our predictions







Future Work

- Build a regression model
- Automate data pipeline