HACETTEPE UNIVERSITY
COMPUTER ENGINEERING DEPARTMENT

UNDERGRADUATE PROJECT FINAL REPORT

Project Name	Report Date
Decision Support System for Predicting Flight Delays	08.01.2019

Student Number(s)	Student Name(s)
21427454 21427291	Güler Ece TAVASLI Ali Peker
Supervisor(s)	**Company Representative(s)**
Engin Demir	-

Project Coordinator	Report Approval
Date: _____	☐ Yes ☐ No If no, rational of rejection: _____

A. TECHNICAL RESULTS

ABSTRACT

Flight delay estimation is a well-studied problem for knowing the delays in advance may help save significant amounts of resources and time for all the parties, namely the passengers, airlines and airports. In this project, our goal was to develop a flight decision support system, which will estimate the **take-off delay** of an upcoming flight and project the estimation over a webpage. We used Machine Learning techniques to model the problem, essentially by building a multiple linear regression model in order to predict how long a delay will be.

Keywords: flight delay estimation, machine learning, data analysis, data science

I. INTRODUCTION

Flight delay prediction is a significant problem that has been modeled in several ways, such as Statistical Analysis, Probabilistic Models, Operational Research and Machine Learning.[1] These methods are used with various parameters about the flight, plane or environmental factors, such as weather, in order to estimate delays. Proper selection of these parameters is crucial to obtain correct estimations.

Aside from the method used, flight delay prediction is seen to be approached mostly as a classification problem.[1] Yet, besides estimating whether there will be a delay or not, an estimation of the duration of a delay would allow passengers, airlines and airports to take precaution accordingly and save significant amounts of time, money and other resources. In this project, we aimed to supply this information by developing a Flight Decision Support System that estimates **take-off delay** durations and serves this information through a web interface. To do the estimations, we adopted a Machine Learning approach and we modelled the problem using multiple linear regression. In addition to the parameters mentioned in literature, we enriched our parameter set with additional parameters to improve our prediction accuracy.

In the first part (A.TECHNICAL RESULTS) of this report, technical details about the project method, implementation and validation can be found, as well as the project context and a brief view on the literature. In the second half (B.PROJECT RESULTS), organizational details about the project execution, practical plan of the development and the results of the project are provided.

II. BACKGROUND

Data

To train and test our model, we used a dataset containing flight and weather information of the every plane took off from Esenboğa Airport between dates 01-01-2014 , 31-12-2014.

Multiple Linear Regression

Multiple linear regression is a predictive analysis method that attempts to model the relationship between two or more explanatory, independent variables and a dependent, so-called target variable, by fitting a linear equation to observed data. It can be used to identify the strength of the effect that the independent variables have on a dependent variable, in our case, the effect of each parameter in our dataset to the duration of delay.

To be able to apply multiple linear regression on a dataset, one must ensure the dataset satisfies several assumptions. To bring the data in correct form, various pre-processing methods are used.

A multiple linear regression model can be implemented in various ways. We chose using the modules provided by scikit-learn, a machine learning library for the Python programming language which provides practical tools for both pre-processing and model generation.

Web Technologies

Angular is a platform that makes it easy to build web applications. It includes various services to communicate with other servers.

Java Spring is a framework for dependency-injection which is a pattern that allows to build very decoupled systems.

Python Falcon is a reliable, high-performance Python web framework.

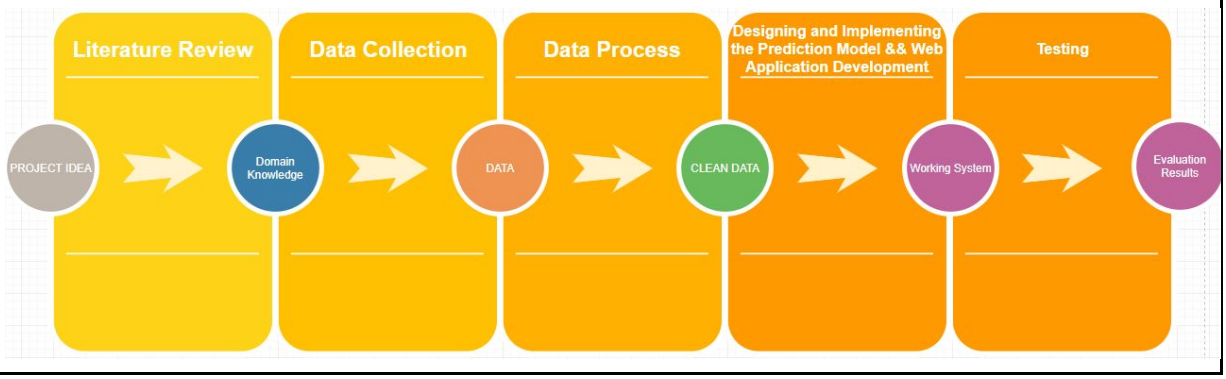
III. RELATED WORK

There have been predictive modeling and simulation attempts to forecast delay in advance.

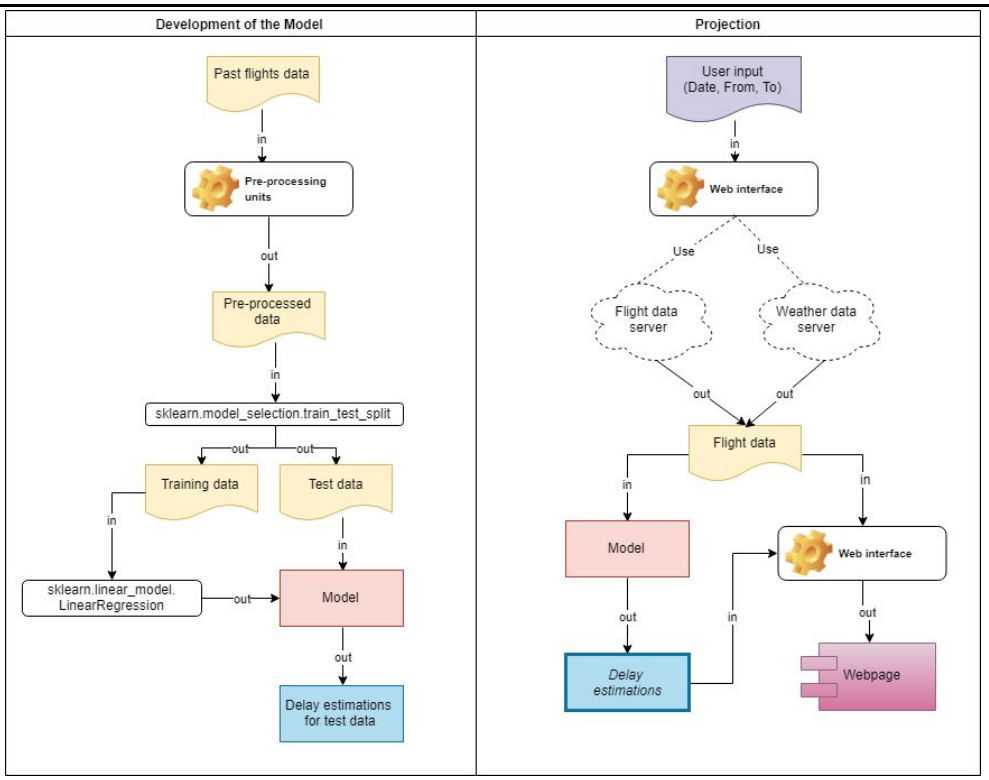
- Juan Jose Rebollo and Hamsa Balakrishnan [2] summarized the results of different classification and regression models based on 100 origin-destination pairs (OD pairs). The study reveals that amongst all the methods used, random forest was found to have superior performance. However, the predictability might vary due to factors such as forecast horizon and the number of origin-destination pairs.
- Dominique Burgauer and Jacob Peters[3] develop a multiple regression model and show that factors such as distance, day and scheduled departure play a significant role in flight delay. While the model gives the significant factors, the prediction accuracy was found to be poor. In addition, the model is limited to only one flight route, namely Los Angeles to San Francisco.
- In another attempt to analyze flight data, Q. L. Qin and H. Yu[4] investigate the overall airline data. A comparison of the K means clustering and Fourier fit model yield that Fourier fit model gave a thorough analysis of the JFK airport in different aspects and could predict the delay trend with a high precision. It is found that the two methods used, work well for a single airport and are not suitable for multiple airport analysis. Similarly, Eric R. Mueller and Gano B. Chatterji[5] summarize that departure delay is modeled better using a Poisson distribution. The study reveals improvement in delay forecast over tools like Enhanced Traffic Management System(ETMS), Collaborative Routing Coordination Tools (CRCT) and NASA Future ATM Concept Evaluation Tool (FACET).

However, the predictability varies based on factors like time frame and the number of airports considered. From the search of literature [2-5], one may conclude that, to better predict flight delays irrespective of the route, number of days, forecasting horizon and number of airports, operational factors must be modeled.

IV. METHOD

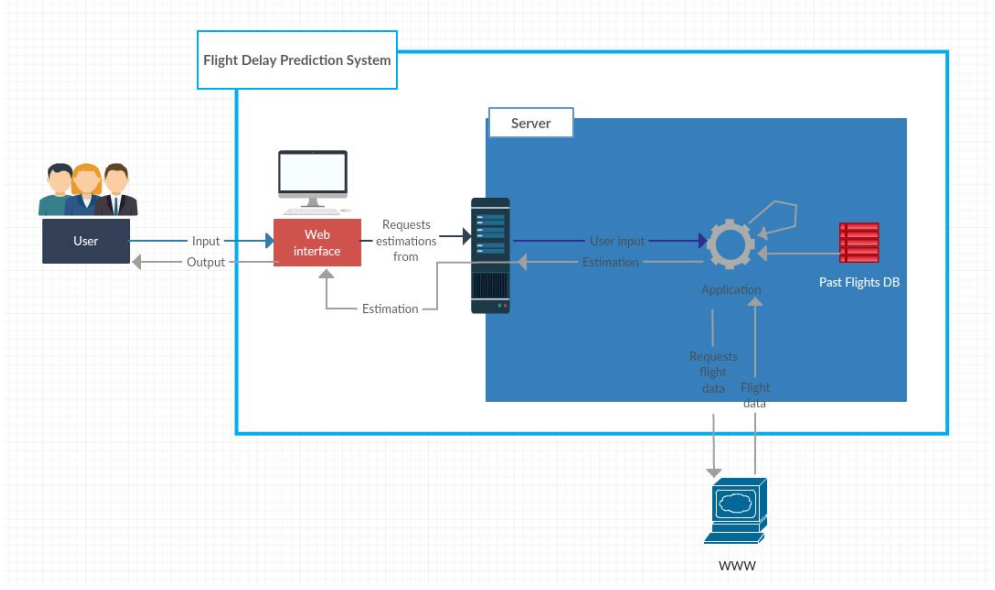


V. TECHNICAL DESIGN AND CONFIGURATION



A common operational scenario:

- User accesses the Flight Decision Support webpage, chooses
  - Origin city
  - Destination city
  - Flight date
- User clicks “Search” button
- Angular sends Origin city, Destination city and Flight date to Java Spring, and requests delay
- Java Spring obtains
  - flight information for the given date, origin city and destination city from another website (ucuzabilet.com)
  - weather information of the origin city for the given flight day and time
  - plane name from another website (flightradar24.com)
- Java Spring obtains information about the plane (weight and number of seats) from plane_info.csv in our local storage
- Java Spring combines these data and sends it to ‘falcon’ module running in our Python server
- Python server receives the data, reads prediction model from the disk and obtains a delay estimation by inputting the data to the model
- Python server sends delay estimation to Java Spring
- Java Spring combines flight data and delay estimation and sends it to Angular
- Flight data and delay estimation are loaded on the webpage



VI. PROJECT IMPLEMENTATION

A. Obtaining Predictor Model

1. Cleaning Data

- Read excel file with data into a Pandas dataframe
  - `Dataframe dimensions: (92212, 28)`
- Extract take-offs
  - `Take-offs: (45479, 28)`
- Remove not-to-be-used columns and keep only flights with flight code 110 (scheduled flights)
- Delete rows with missing fields:
  - `Prev. dataframe dimensions: (40118, 18)`
  - `Updated dimensions: (28861, 18)`
- Insert new parameters:
  - `Airline`
  - `Season`
  - `Month`
  - `Month day`
  - `Weekday`

- Save dataframe as cleaned_data.csv

2. Encoding Categorical Parameters

- Read cleaned_data.csv into Pandas dataframe
- Extract categorical parameters and number of unique values each
- Apply one-hot-encoding to each parameter (using pandas.get_dummies)
- Remove old columns of categorical variables from dataframe and add new, one-hot encoded versions
  - Old dataframe size: `(28861, 19)`
  - New dataframe size: `(28861, 92)`
- Save new data frame as onehot_cleaned_data.csv

3. Regression

- Read onehot_cleaned_data.csv to Pandas dataframe
- Remove 'slot' and 'fark' columns and keep this in dataframe “df”, which will be the feature matrix
- Cross-validation: *Use 'K-fold'* method (used by sklearn *cross_val_predict ()* and *cross_val_score ()* methods) to choose best polynomial degree for fit

We've observed that the error always grew higher after the polynomial degree n=10, therefore we checked error for range(1,15).
We've calculated all possible combinations of number-of-folds and poly-degree and picked the pair with minimum error. The best results on the test set were obtained by:
> number of folds = 190
> polynomial degree = 7
> error = 6.66 minutes

➢ We built the model based on the specifications above, using sklearn library's functions.

➢ We've saved the model as a .sav file using library "joblib". This way, we are able to load the model from the disk and re-use it.

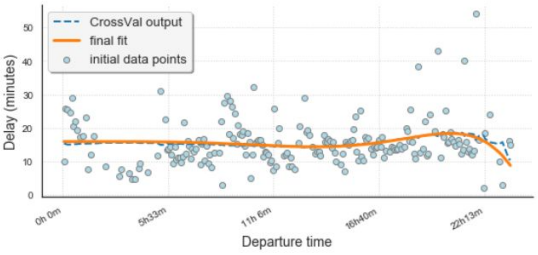
B. Web interface

- First, we used angular 6, html and css for our interface. According to the data entered from the interface, the interface makes a request from the java.
- Next, Java Spring is able to obtain weather data from the date and time of the flight and the flight weight and number of seats of the flight. This data is transferred to the python falcon.
- Python calculates the estimated delay of the flight with this data and sends it to Java. Java Spring sends the relevant information to the interface.
- Finally, the interface displays the relevant information on the user screen.

VALIDATION AND RESULTS

A. Model

- Dataset was divided into a random “test set” and “train set” using **sklearn.model_selection.train_test_split**, which splits matrices into **random train and test subsets.**
- After training the model with “train set”, the model was tested with “test set”.
- Fitted model can be seen below, as the orange polynome.
- When tested with ‘test set’, model obtained a 6.63 min. average error.



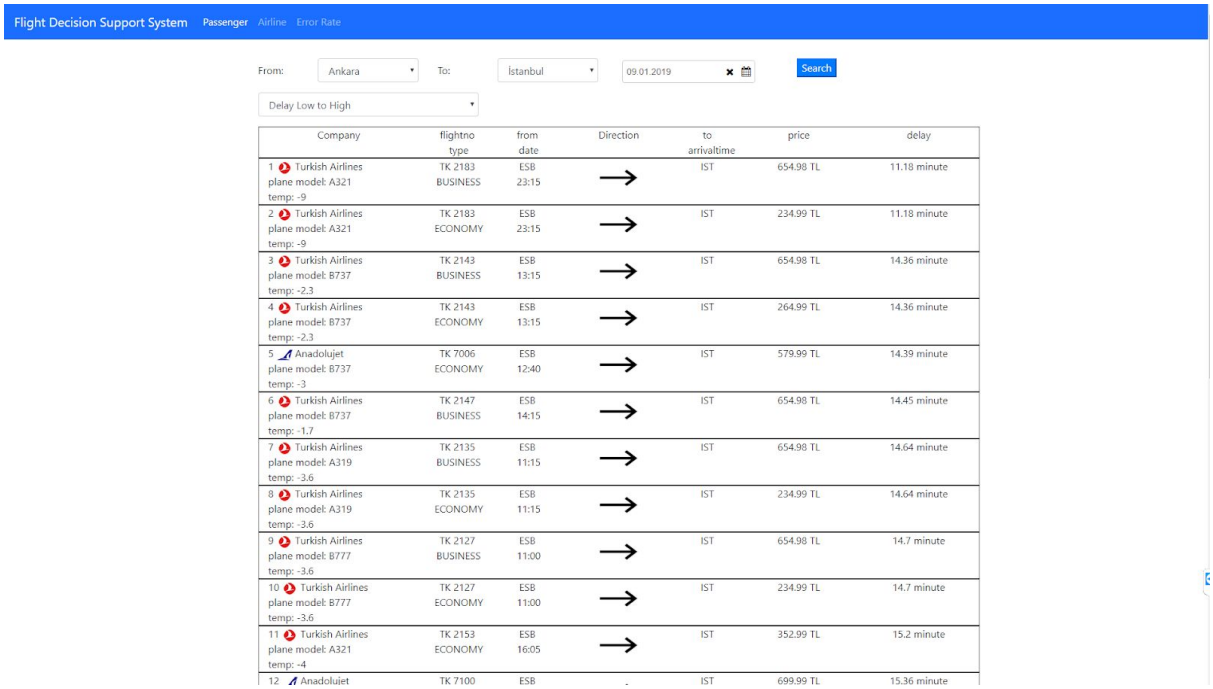

```
In [82]: score = metrics.mean_squared_error(fit.result, fit2.Y)
Out[82]: 43.894510763832436

In [83]: 'Ecart = {:.2f} min'.format(np.sqrt(score))
Out[83]: 'Ecart = 6.63 min'
```

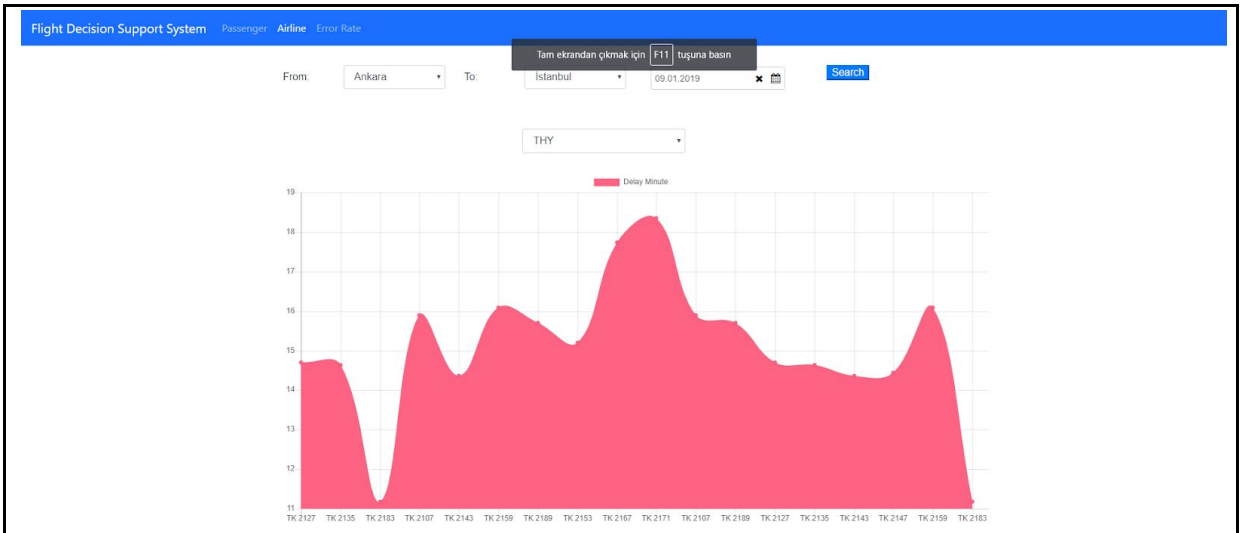
Even though this result looks good right now, there is no guarantee that it will perform as well for the real data, namely the flights for 2018. We don’t know whether the weights of the parameters have changed as the years passed. Also, more evaluation on the features and the dataset was needed, but due to limited time, we might not have accomplished this very well.

B. Web interface

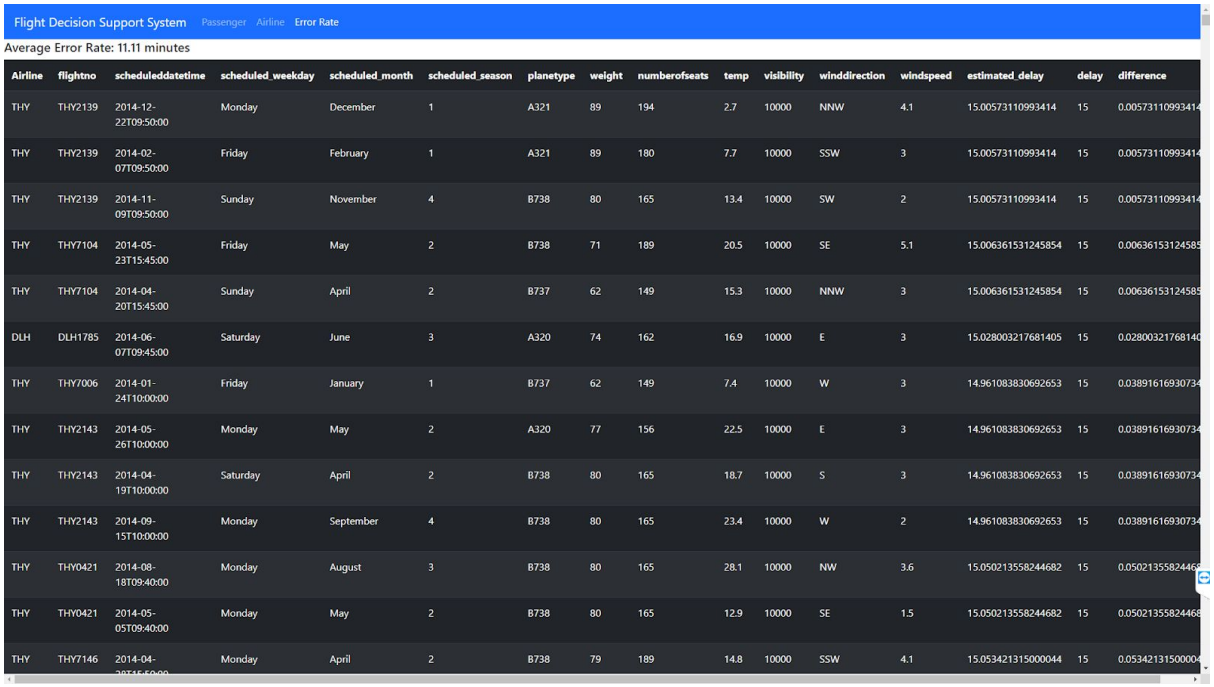
Search page for passengers. All flights and calculated delays on the date requested by the passenger.



A chart showing the delay of the airlines flights.



Flight delays that we have already calculated in our test data and actual delays and the parameters we use in calculate delay.



Airline	flightno	scheduleddatetime	scheduled_weekday	scheduled_month	scheduled_season	planetype	weight	numberofseats	temp	visibility	winddirection	windspeed	estimated_delay	delay	difference
THY	THY2139	2014-12-22T09:50:00	Monday	December	1	A321	89	194	2.7	10000	NNW	4.1	15.00573110993414	15	0.00573110993414
THY	THY2139	2014-02-07T09:50:00	Friday	February	1	A321	89	180	7.7	10000	SSW	3	15.00573110993414	15	0.00573110993414
THY	THY2139	2014-11-09T09:50:00	Sunday	November	4	B738	80	165	13.4	10000	SW	2	15.00573110993414	15	0.00573110993414
THY	THY7104	2014-05-23T15:45:00	Friday	May	2	B738	71	189	20.5	10000	SE	5.1	15.006361531245854	15	0.006361531245854
THY	THY7104	2014-04-20T15:45:00	Sunday	April	2	B737	62	149	15.3	10000	NNW	3	15.006361531245854	15	0.006361531245854
DLH	DLH1785	2014-06-07T09:45:00	Saturday	June	3	A320	74	162	16.9	10000	E	3	15.028003217681405	15	0.028003217681405
THY	THY7006	2014-01-24T10:00:00	Friday	January	1	B737	62	149	7.4	10000	W	3	14.961083830692653	15	0.038916169307347
THY	THY2143	2014-05-26T10:00:00	Monday	May	2	A320	77	156	22.5	10000	E	3	14.961083830692653	15	0.038916169307347
THY	THY2143	2014-04-19T10:00:00	Saturday	April	2	B738	80	165	18.7	10000	S	3	14.961083830692653	15	0.038916169307347
THY	THY2143	2014-09-15T10:00:00	Monday	September	4	B738	80	165	23.4	10000	W	2	14.961083830692653	15	0.038916169307347
THY	THY0421	2014-08-18T09:40:00	Monday	August	3	B738	80	165	28.1	10000	NW	3.6	15.050213558244682	15	0.050213558244682
THY	THY0421	2014-05-05T09:40:00	Monday	May	2	B738	80	165	12.9	10000	SE	1.5	15.050213558244682	15	0.050213558244682
THY	THY7146	2014-04-	Monday	April	2	B738	79	189	14.8	10000	SSW	4.1	15.053421315000044	15	0.053421315000044

CONTRIBUTION(S) TO INDUSTRY AND ECONOMY

Even though there are visual interfaces projecting whether there's going to be a flight delay or not, we haven't came across to any other visual interface projecting the duration of estimated delay like ours, apart from the airport's information screens, which provides this information only shortly before the flight, relying on operational issues.

In addition to be a possible solution to problems such as improving the tactical and operational decisions of airports and airlines, allowing passengers to take precautions about possibility of flight delays or innovative applications such as flight recommendation for passengers with connecting flights or very urgent transportation needs, our product can allow a quicker decision making with the help of visualizations it will provide.

INNOVATIVE ASPECTS

- We provide the "duration" of the delay instead of providing only whether there will be a delay or not.
- Unlike many of the other solutions, we included **flight-specific features** in our set of features (like types of the aircraft, the order and the frequency they take off, and external weather features such as local weather, and sensor data) to obtain more accurate estimations for the individual scheduled flights.
- Since we cannot directly use the date DD/MM/YYYY as a parameter, we extracted new features out of the features we already had:
    - Month, Season, Weekday, Month-day from "flight date"
    - Airline company from "flight number"
- Open data collected from flightradar24.com and weather channel is utilized.
- We created a Flight Decision Support web interface such that, one can easily plug a new prediction model in.
- We provide graphical representations of the estimations in an easily interpretable way for airline companies, that can help them with their organizational issues

REFERENCES

1. Sternberg, A., Soares, J., Carvalho, D., Ogasawara, E., A Review on Flight Delay Prediction, 2017
2. Juan Jose Rebollo and Hamsa Balakrishnan (2014).'Characterization and Prediction of Air Traffic Delays',Massachusetts Institute of Technology.
3. Diminique Burgauer and Jacob Peters(2000). 'Airline Flight Delays and Flight Schedule Padding', University Of Pennsylvania.
4. Q. L. Qin and H. Yu (2014), A statistical analysis on the periodicity of flight delay rate of the airports in the US, Advances in Transportation Studies an international Journal 2014 Special Issue, Vol. 3.
5. Eric R. Mueller and Gano B. Chatterji. 'Analysis of Aircraft arrival and departure delay characteristics',NASA Ames Research Center, Moffett Field, CA 94035-1000

B. PROJECT RESULTS

I. CHANGES TO PROJECT PLAN

II. PROJECT MILESTONES AND OBJECTIVES

Milestone #	Primary Objective	Due Date	Project Deliverable (if any)	Milestone Achieved?
1	Data ready for use on ML algorithms	03.11.2018	Data	Yes
2	Web interface design is ready	10.11.2018	Web interface design	Yes
3	Model ready	25.12.2018	Model	Yes
4	Web interface ready	25.12.2018	Web interface	Yes
5	Model integrated with web interface	30.12.2018	Final product	Yes

III. PROJECT PRACTICES AND MEASURES

Task #	Task Description	Responsibility	Start Date	Finish Date	Success Criteria	Task Succeeded?
1	Literature review, collection and preprocessing of data	All teammates	27.10.2018	10.11.2018	Information collected for measures of accuracy, feature selection and previously applied methods. Data is ready and formatted in a suitable to be fed into ML algorithms.	Yes
2	Design the web interface	All teammates	04.11.2018	10.11.2018	The elements, abstract layout is prepared and features/functioning of the web interface is determined.	Yes
3	Feature extraction & selection	All teammates	04.11.2018	17.11.2018	Features are obtained from data and literature review and are supported with sufficient reasonings	Yes
4	Implement the web interface	Ali Peker	11.11.2018	10.12.2018	Web interface is ready to be used by an	Yes

					end-user, capable of retrieving real-time flight data, allowing users to make query search, yet retrieving arbitrary values in the fields allocated for estimations until the integration with the model is made	
5	Training and choosing the model	Ece Tavaslı	18.11.2018	27.11.2018	A model is selected taking into account the expressiveness (which is helpful while explaining the processes to domain experts later) and bias-variance trade-off	Yes
6	Decide on representative ways to visualize estimations	All teammates	22.11.2018	25.11.2018	Visualization methods enabling quick interpretation and decision making are determined	Yes
7	Testing and improving the model	Ece Tavaslı	28.11.2018	10.12.2018	The model is systematically tested and required improvements are determined. Model has a better accuracy	Yes
8	Integrate the model with the web interface and test the final product	All teammates	10.12.2018	22.12.2018	Web interface is integrated with the model and now has real delay estimations obtained from the model. The final product is tested systematically, major errors are documented and fixed; minor errors are noted to be fixed in the next step.	Yes
9	Do the last edits	All teammates	23.12.2018	24.12.2018	Project is completed and meets its goals.	Yes

Team Member	**Task # Under Responsibility**	**Description of the Work Done**
Ece Tavaslı	1,2,6	● Reviewed the literature to get to know about earlier approaches to the problem of flight delay prediction. ● Written Python tools for the evaluation and cleaning of the dataset. ● Visualized data to get a superficial idea about the contribution of parameters to the recorded flight delays. ● Contributed to the design of web interface and choosing representative ways to visualize estimations.

Ali Peker	1,2,4,6	• Reviewed the literature to get to know about earlier approaches to the problem of flight delay prediction. • Graphics to be shown to the airline company was determined. • The passenger screen is designed. Implementation has started. • The architecture to be used throughout the project was determined.

IV. PROJECT BUDGET

Item #	Description of Income	Date of Income	Planned Amount	Actual Amount	Amount Difference

Item #	Description of Expense	Date of Expense	Planned Amount	Actual Amount	Amount Difference

Overall Balance	Planned Amount	Actual Amount	Amount Difference
Income			
Expense			
Total			

V. PROJECT RISKS

Risk Item #	Description	Probability	Effect	Did It Happen?	How did you handle its occurrence if happened? (Plan-B)
1	We may not finish dependent tasks on time due to unexpected incidents.	High	We may not keep up with the milestones.	Yes	We redefined and rescheduled our tasks and milestones with respect to the time and resources each teammate has.
2	We may have a data shortage.	High	Our model may have less accurate estimations	Yes	For now, we used the available dataset. As we proceed with the construction of model, we will seek for other sources of data.
3	There may be an argument among teammates	Medium	Our progress	No	We may consult our supervisor if we cannot solve it ourselves.

			may slow down		
4	Our data may not be clean or expressive enough	Medium	Our model will have inaccurate predictions.	.No	Apply suitable methods to balance bias-variance as much as possible.
5	Our Prediction model may not work as intended	Medium	Our model will have inaccurate predictions.	No	We will consult our supervisor. 1. If the collected data does not provide a good asset, we can opt for simulation. 2. If any of the single method is not robust to predict the delay, an ensemble of techniques can be deployed 3. We will make weekly revision of the progress to determine and eliminate any source of delay and revise our solution scheme by considering an available open source toolkit to resolve the task.
6	We may have trouble delivering the delay estimations from server to the web interface	Medium	Our product will not meet its requirements.	No	We will consult the experts on the topic that we can access to.

VI. SELF EVALUATION

- Machine learning was way harder than we thought it was.
- We underestimated that incidents may occur that can cause us to fall very behind our schedule.
- Before starting the project, we were familiar with web technologies. But we had no experience in web scraping.
- Our future goal in the project is to establish a system that is constantly fed with data. We would have had a continuous data set for the most frequently used flights by continuously pulling the data through the calls made by the users. Unfortunately, we don't have enough time for doing this.

VII. LESSONS LEARNED

We learned that communication and teamwork are important for the project.

We've learned how to use several Python libraries to implement a machine learning model.

We've learned and implemented how servers communicate with each other.

We learned how to do web scraping and also, how regression works.

We've learned the logic of api's work. And we used api.