[Airline Ticket Price Prediction]

Team number (7) SC

Team Members: -

Name	ID	Department	Section
اسامه عنتر محمد عفيفي	20191700091	SC	1
احمد محمد إبراهيم محمد	20191700059	SC	1
أدهم محمد توفيق محمد	20191700086	SC	1
احمد محمد علي عبد الرحمن	20191700068	SC	1
طارق أشرف محمود حسين	20191700322	SC	3

Milestone 1

When we saw the data

We made split data to (Futures and Goal)

- We make **preprocessing** to data:
  - we tried t **check the "NULL" values**, and for whom I did not find.
  - **Split** and **concate** to (Date, Price….)
  - **Encoding** to Strings Values.
  - Removing **outliers** in Y("Goal")
  - Drop the column that have the same correlation to another column
  - Convert some column to int
  - Drop that have the weak correlation ("**irrelevant**")

Then we completed the application in **the two model (Polynomial and Multi Variable)**:

- Then we make future selection
  - In the first model when we **calculated correlation > 0.05 (Polynomial),**

num_code	-0.213294
dep_time	0.033034
time_taken	0.026429
stop	0.118334
days	-0.001599
months	-0.090664
ch_code	0.310842
type	-0.937572
route	0.004336

- We use **another technique** that **select k best** columns to model.

- we completed the application in **the two model** regression techniques **(Polynomial and Multi Variable)**.
- **in Polynomial Model:**

MSE	32309651.094342146
Score “accuracy”	0.934894620311891 9

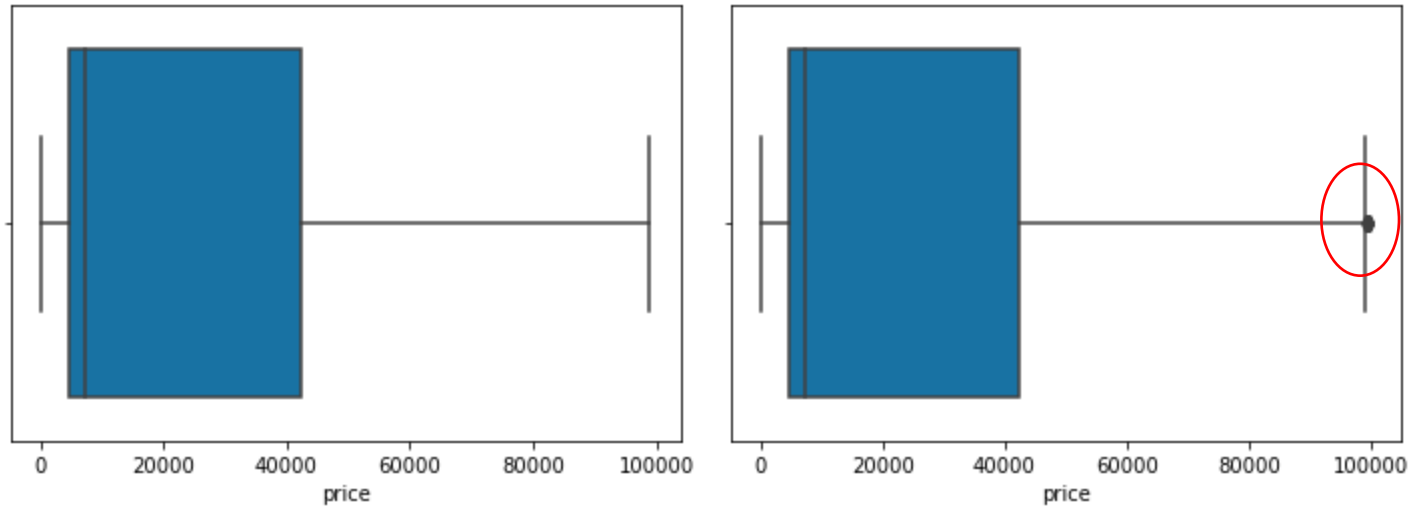
- **in Multi Variable Model:**

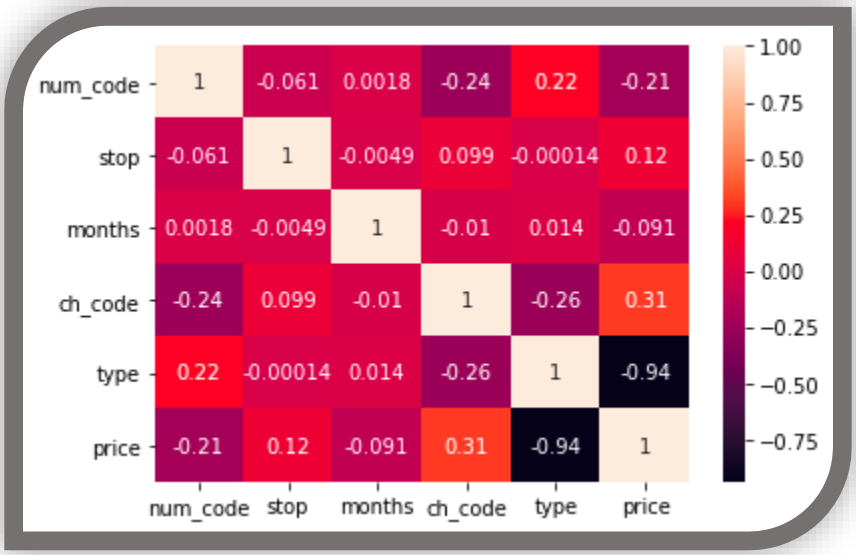
MSE	48541689.824759305
Score “accuracy”	0.9043692030723013

- **the future we do use** it:
  - [“num_code”], [“stop”], [“months”], [“ch_code”], [“type”]. In (polynomial)
  - All in multivariable regression.

- We use 80% training & 20% testing.

- We use **another technique** that **select k best** columns to model to **improve** the results.



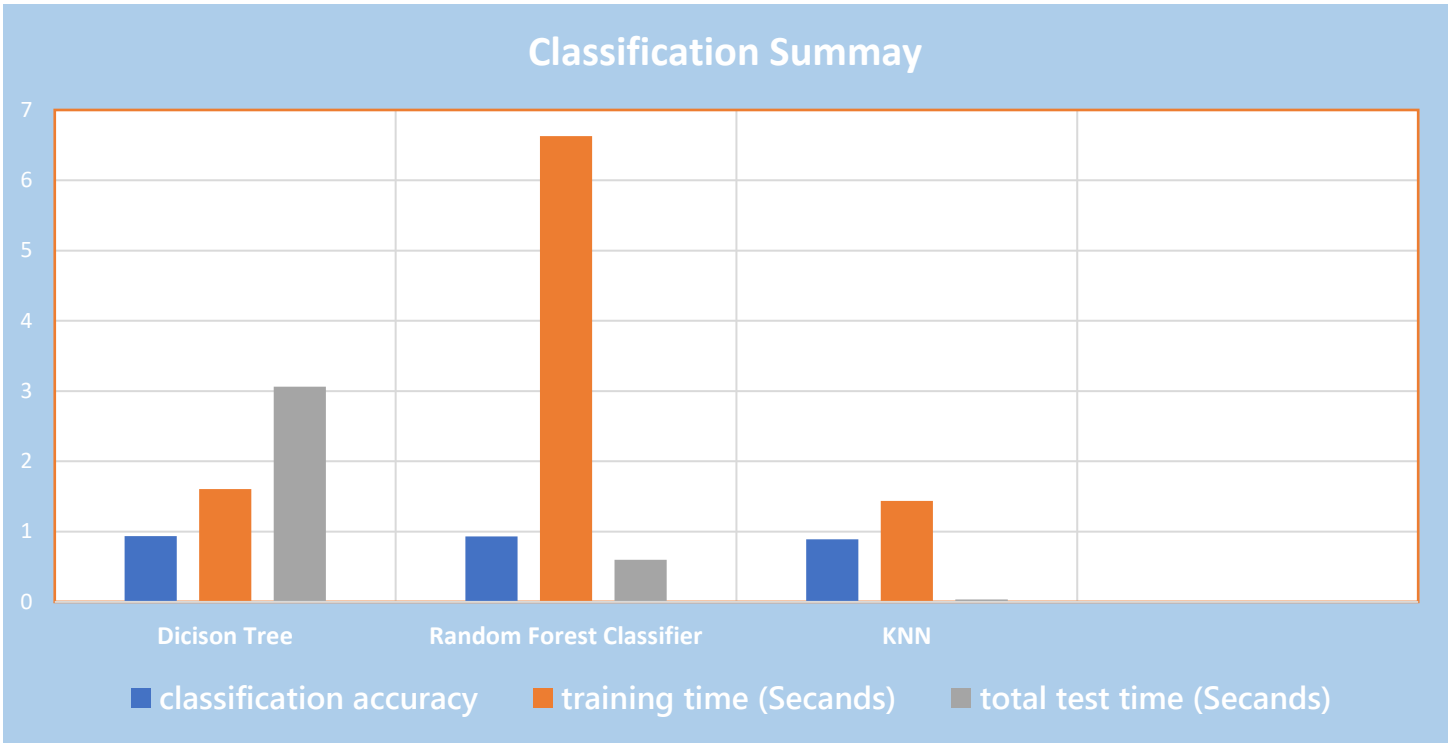


- **Conclusion:**

  I found many columns that have nothing to do with the goal, and we found that many of the Features that turned out to be important, but actually have nothing to do with the model and do not affect it, and therefore we used this model.

Milestone 2

❖ three bar graphs Summarize: -



❖ Feature Selection process: -

- In the Three model when we **calculated correlation > 0.03**

num_code	**-0.153206**
dep_time	**-0.009244**
time_taken	**0.041478**
stop	**0.247243**
days	**-0.003096**
months	**-0.257619**
ch_code	**0.313809**
type	**-0.477743**
route	**0.044292**

❖ hyperparameter:

- **Decision tree:**
  - **“Max_depth” = None**
    - Max Depth is how many subtrees taken to reach the end (from the parent to the last children).
      “None” was the best choice, as I didn’t need any constraints concerning number of decision levels.
  - **“Min_samples_split” = 3**
    - This hyperparameter decides the minimum number of splits per node.
      Min = 3, was better than binary split (min = 2) and from(3) it get worse as it increases. So, “3” is the best option.
  - **“Max_leaf_nodes” = None**
    - Maximum number of leaf nodes a decision tree can have.
      “None” is the most compatible with the (Max_depth = None).
      Also, it gave the best accuracy.

- **Random Forest:**
  - **“Max_depth” = 20**
    - Max Depth is how many subtrees are taken to reach the end (from the parent to the last children). “20” was the best choice, I needed some constraints concerning the levels, too many levels will cause overhead and overfitting, few will cause underfitting and less accuracy.
  - **“Min_samples_leaf” = 4**
    - This is how many leaves are there in the end of every decision tree. Choosing value “4” gave the best accuracy.
  - **“N_estimators” = 15**
    - This is the number of decision trees. “15” gave the best accuracy without overfitting.

- **K Nearest neighbors:**
  - **“Neighbors” = 10**
    - This is the number of neighbors taken of each node. “10” gave high accuracy and low overfitting as it doesn’t choose many nodes, and not also few.
  - **“Leaf_size” = 45**
    - Leaf size passed to Ball Tree or KD Tree (algorithms used in the knn to get the neighbors). This can affect the speed of the construction and query, as well as the memory required to store the tree.
  - **“P” = 2**
    - This is power parameter for the Minkowski metric. When p = 1, this is equivalent to using manhattan_distance, and euclidean_distance for p = 2.

- **Conclusion:**
  - My Intuition for the project was that I was going to use the same preprocessing as milestone one, but it turned out that I must change it according to the model.
  - After studying the data, I believed that the model with best accuracy is knn because it’s general and efficient. However, I found out that the random forest was the best because it combines the result of several decision tree.
  - Going through the data, you would think that certain features are very related and correlated, but that’s not the case after going through its value and studying its correlation. Some of these features are dropped due to their very low (almost zero) correlation.

Thank you ^-^