Intro. to Artificial Intelligence

Project 1: Kaggle Titanic Problem

Murathan Bakır
110180137

Code Link:
https://colab.research.google.com/drive/1nDeo8kGRB4K8DgWvaX5quVip_EwVengY?usp=sharing

Entrance

It is important to select the useful data to predict the result of a case. In order to choose useful data, causes and effects of the case should be determined precisely. In the Titanic example; sex, age, fee of the ticket, ticket class, companion status of the passenger and cabin number can be important and decisive to determine the final status of the traveller intuitively.

For example;

Women and children might be subjected to positive discrimination during the rescue operations rather than men and adults. Accompagnation of the individuals are also important. Maybe they had priority by their companions during the rescue operations etc.

Ticket class, fare and cabin numbers are related with economic opportunities of the passengers. People that have a specific ticket class or specific cabin might be close to emergency places or backup boats. The place of the cabins are playing decisive role on ticket fare after all. Expensive cabins might be in the upper floors, so maybe they have much more time to be rescued rather than lower floor cabins’ passengers etc.

Just the opposite; name, port of the embarkation, ticket number of the passengers are just dummy informations without any correlation with the recovery operations at all. I just eliminated these data from dataset.

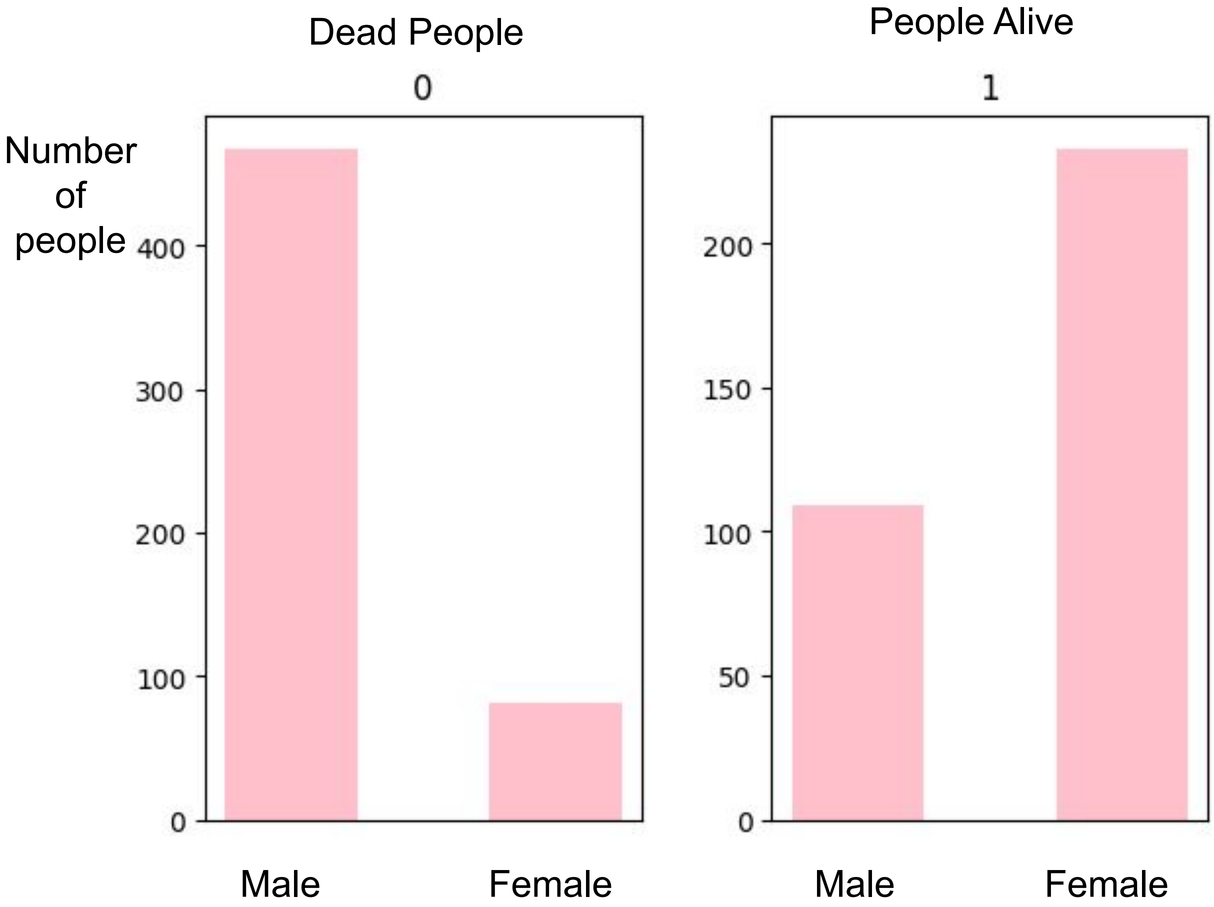
Preprocessing

- Age data has some deficiencies. 20% of the age data is missing. In order to take care of that, I replaced the NaN variables with average value of the whole data which is 29.7 years. 20% data filling might have some bias but it’s not that much harmful to overall result.
- Cabin data has much more gaps around 75%. We cannot handle these data by filling with such methods like taking mean or median etc. The best solution is just ignoring this information.
- It is important to determine which data are categorical or numerical for healthy results. Obviously, sex is one of the categorical data, we should consider this as a category. Then we can perform feature scaling successfully.

Statistical Plots

Here are some references to classification element by element. Details and interpretations made on the Google Colab Notebook texts.

Sex vs Survived

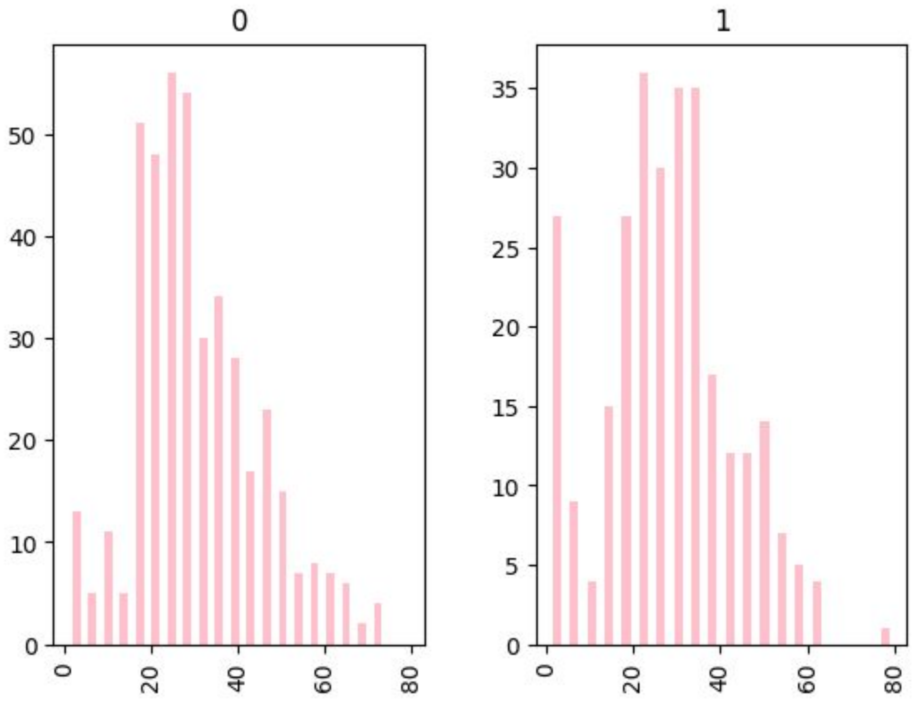


Age vs Survived

Dead People

People Alive

Number of people



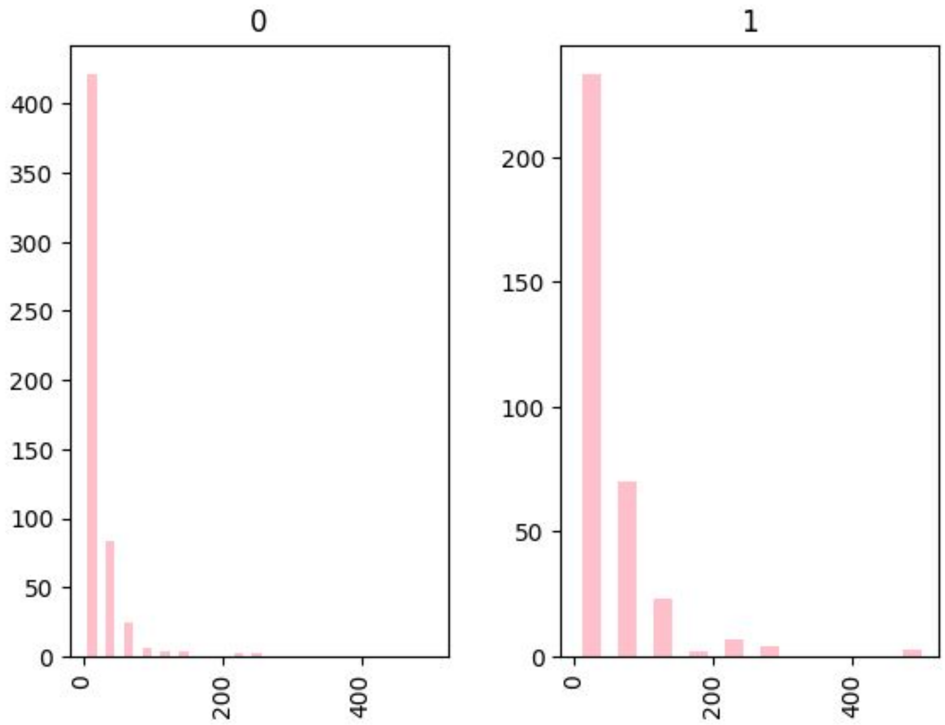
Ages of people

Ages of people

Fare vs Survived

Dead People

People Alive



Number of people

Ticket fare

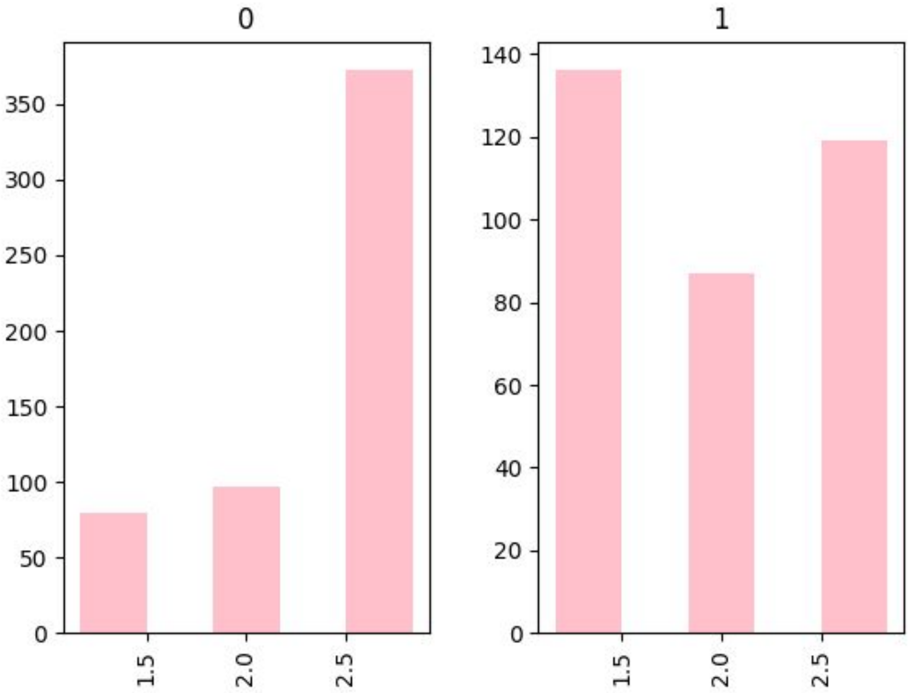
Ticket fare

Ticket Class vs Survived

Dead People

People Alive

Number of people



Ticket class

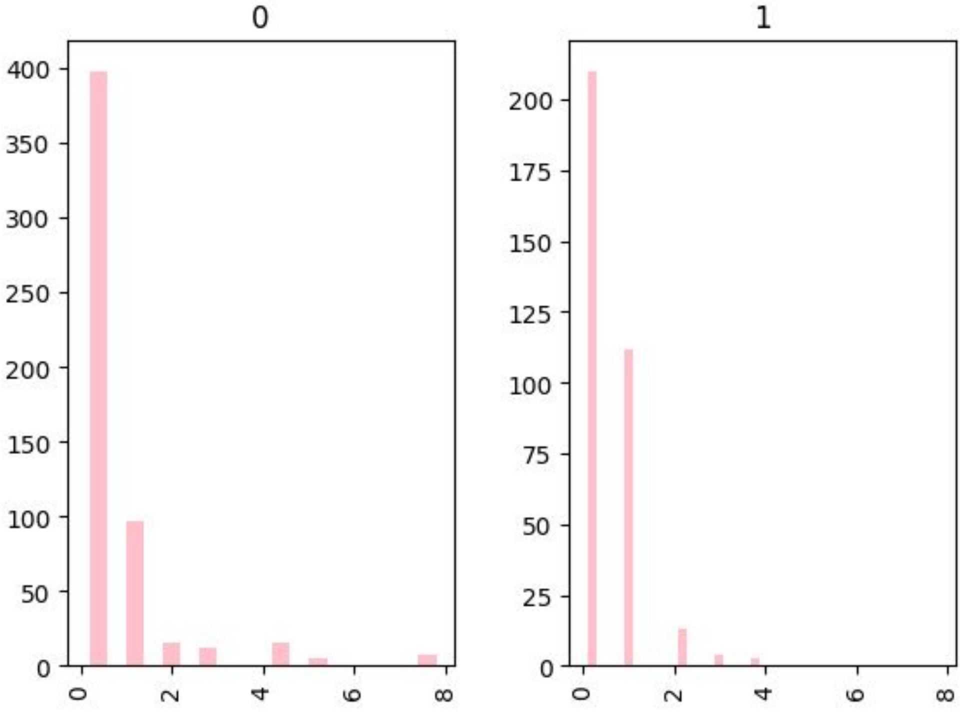
Ticket class

SibSp# vs Survived

Dead People

People Alive

Number of people



SibSp number

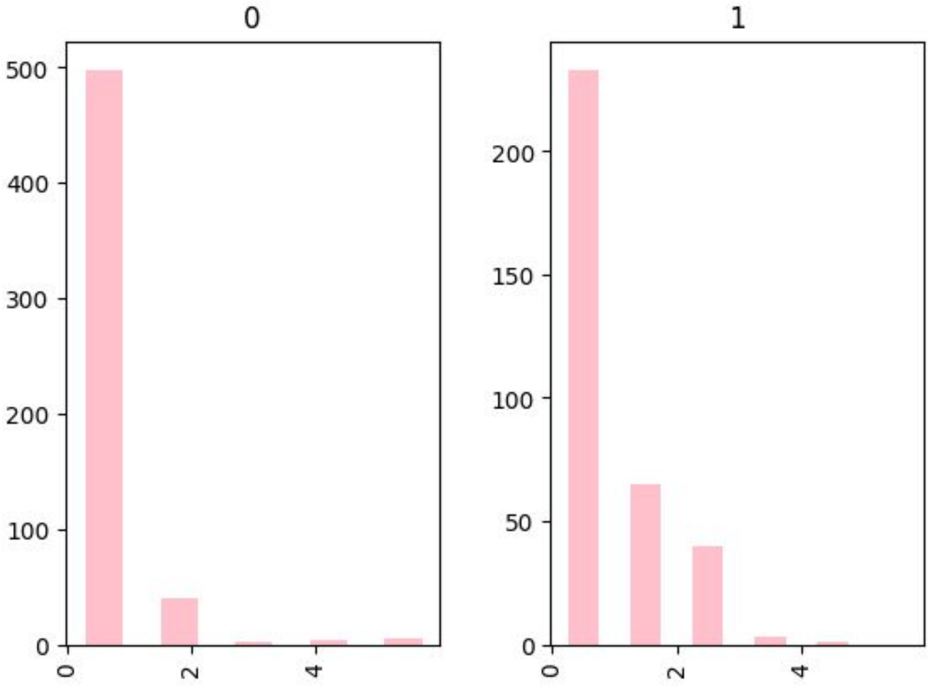
SibSp number

Parch# vs Survived

Dead People

People Alive

Number of people



Parch number

Parch number

Results

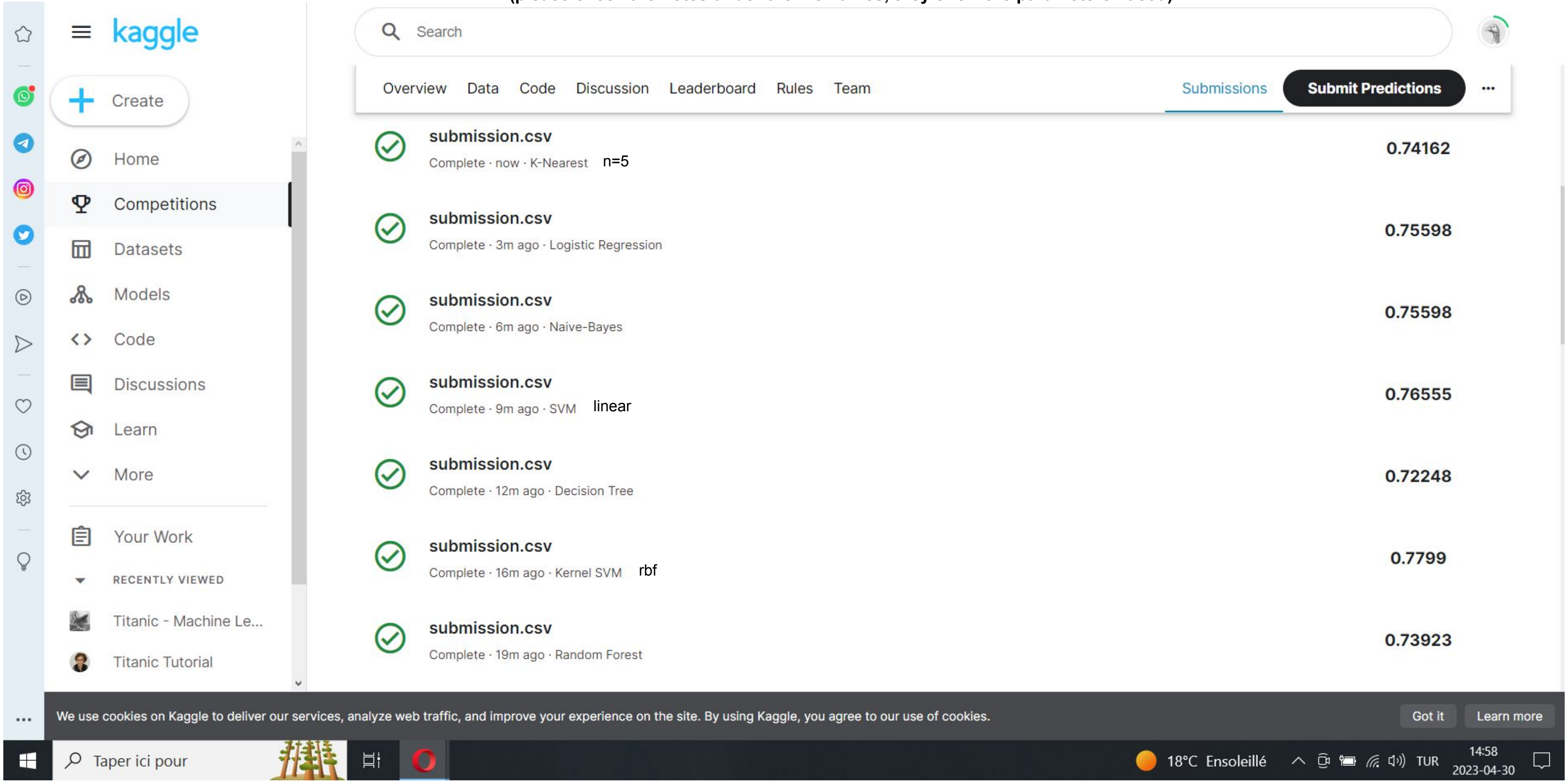
7 classification mode used separately. Success rate changes by changing the parameters of classifiers. It varies between 0.72 and 0.78.

In the first attempt, the best value came from Kernel SVM and the worst came from Decision Tree. I had changed the parameters of some methods and after tried again, then I saw better performance on Random Forest and K-Nearest but worse performance on Kernel SVM. Here are some meaningful changes by parameters:

K-Nearest : number of neighbors = 5 → number of neighbors = 20	**BETTER RESULT**
Random Forest: number of estimators = 10 → number of estimators = 100	**BETTER RESULT**
SVM: kernel = linear → kernel = polly	**BETTER RESULT**
Kernel SVM : kernel = rbf → kernel = linear	**WORSE RESULT**

First Results with Default Parameters

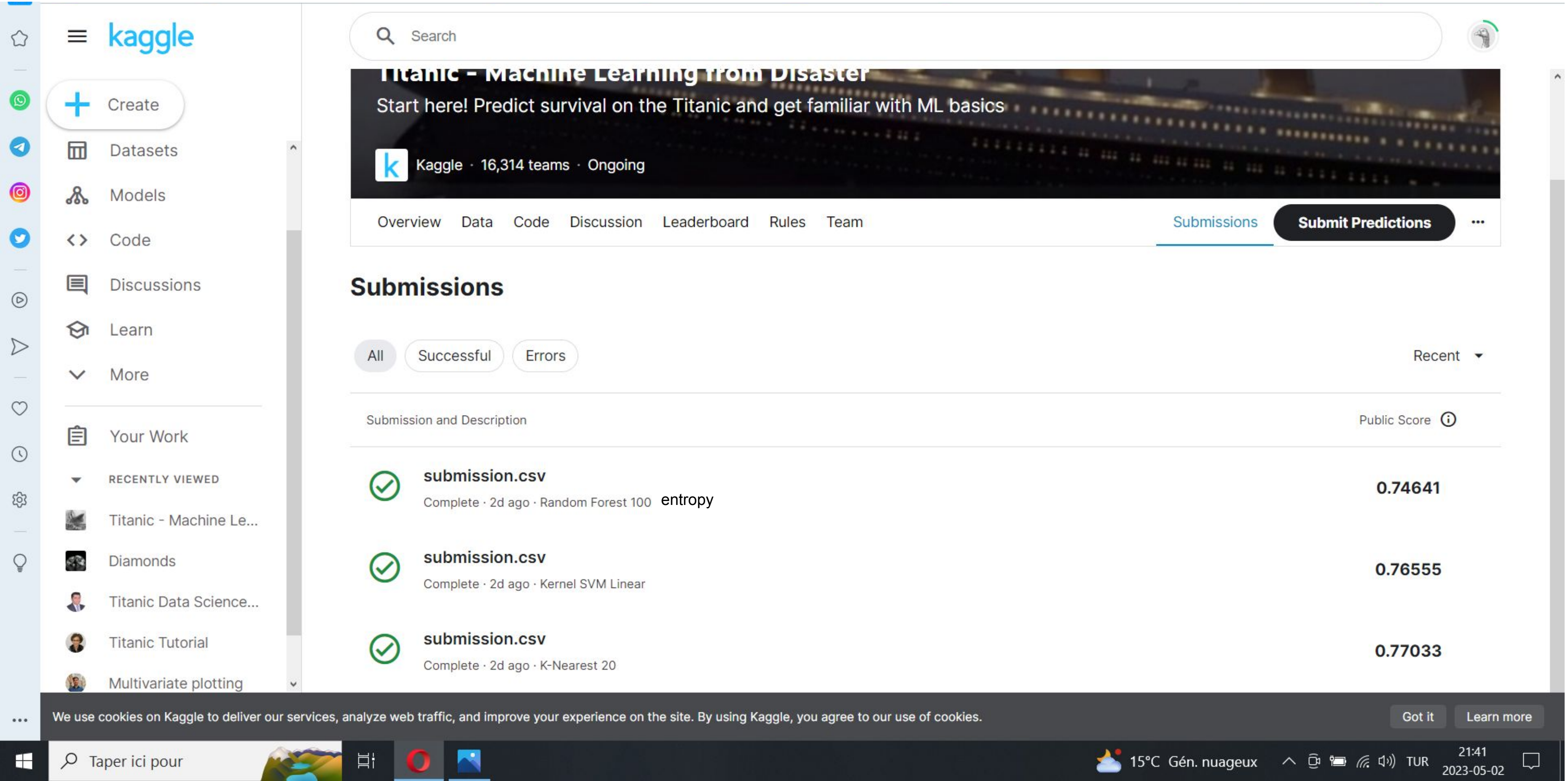
(please check the notes under the file names, they show the parameters I used)



submission.csv Complete · now · K-Nearest n=5	0.74162
submission.csv Complete · 3m ago · Logistic Regression	0.75598
submission.csv Complete · 6m ago · Naive-Bayes	0.75598
submission.csv Complete · 9m ago · SVM linear	0.76555
submission.csv Complete · 12m ago · Decision Tree	0.72248
submission.csv Complete · 16m ago · Kernel SVM rbf	0.7799
submission.csv Complete · 19m ago · Random Forest	0.73923

We use cookies on Kaggle to deliver our services, analyze web traffic, and improve your experience on the site. By using Kaggle, you agree to our use of cookies. Got it Learn more

Secondary Results with Different Parameters



Titanic - Machine Learning from Disaster

Start here! Predict survival on the Titanic and get familiar with ML basics

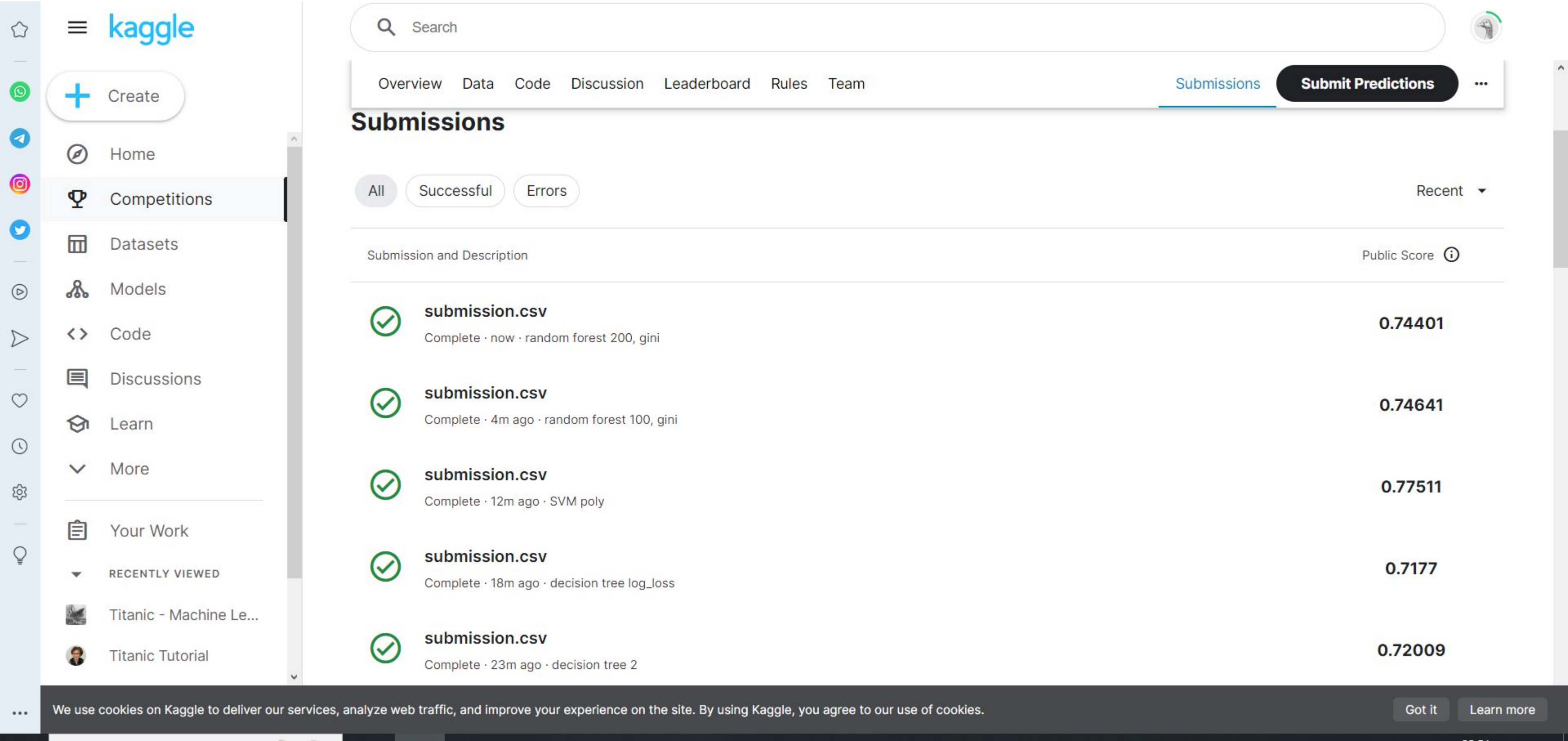
Kaggle · 16,314 teams · Ongoing

Submissions

Submission and Description	Public Score
submission.csv Complete · 2d ago · Random Forest 100 entropy	0.74641
submission.csv Complete · 2d ago · Kernel SVM Linear	0.76555
submission.csv Complete · 2d ago · K-Nearest 20	0.77033

We use cookies on Kaggle to deliver our services, analyze web traffic, and improve your experience on the site. By using Kaggle, you agree to our use of cookies.

Secondary Results with Different Parameters

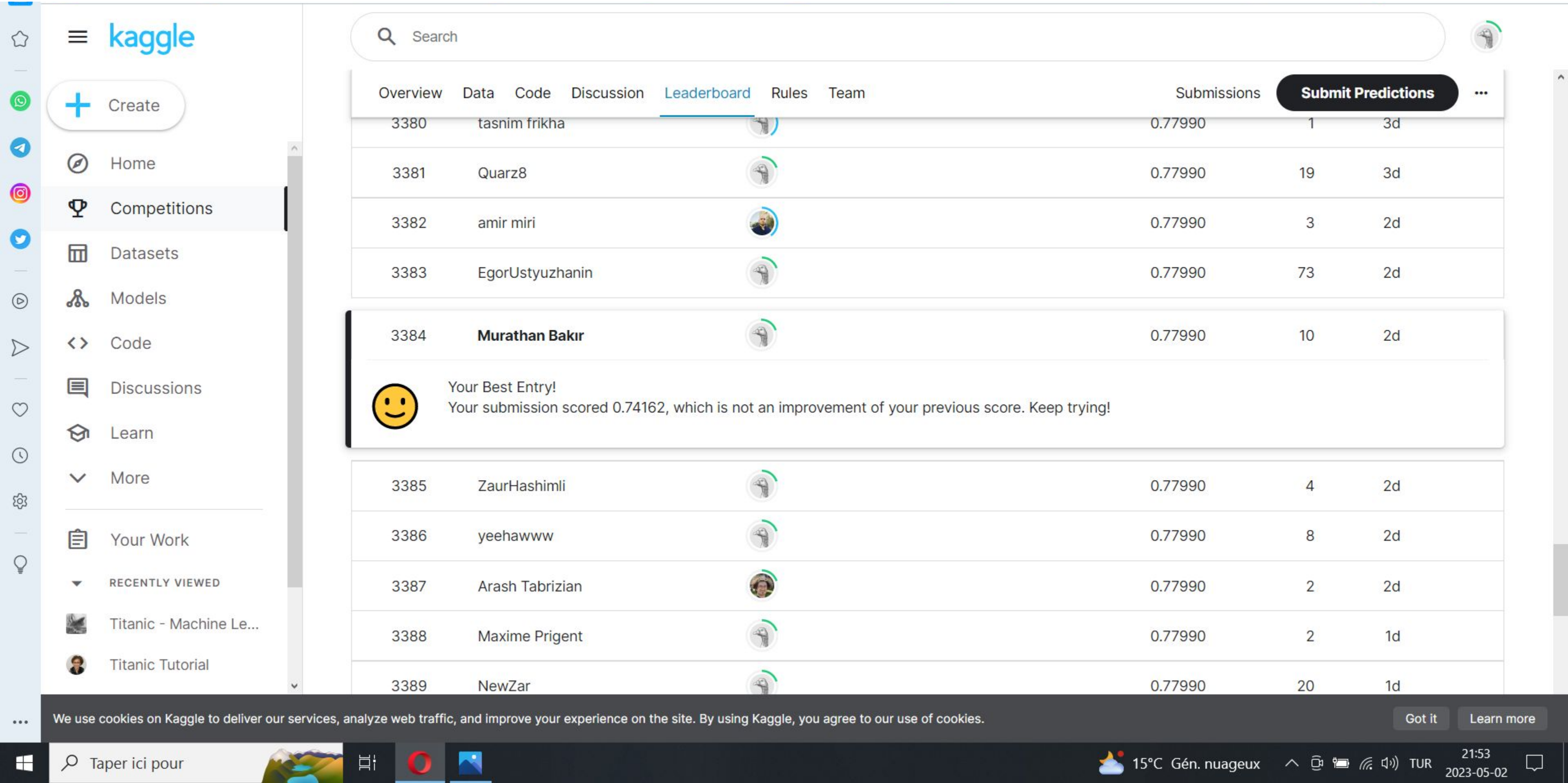


Submissions

Submission and Description	Public Score
submission.csv Complete · now · random forest 200, gini	0.74401
submission.csv Complete · 4m ago · random forest 100, gini	0.74641
submission.csv Complete · 12m ago · SVM poly	0.77511
submission.csv Complete · 18m ago · decision tree log_loss	0.7177
submission.csv Complete · 23m ago · decision tree 2	0.72009

We use cookies on Kaggle to deliver our services, analyze web traffic, and improve your experience on the site. By using Kaggle, you agree to our use of cookies.

Leaderboard



3380	tasnim frikha	0.77990	1	3d
3381	Quarz8	0.77990	19	3d
3382	amir miri	0.77990	3	2d
3383	EgorUstyuzhanin	0.77990	73	2d
3384	**Murathan Bakır**	0.77990	10	2d
3385	ZaurHashimli	0.77990	4	2d
3386	yeehawww	0.77990	8	2d
3387	Arash Tabrizian	0.77990	2	2d
3388	Maxime Prigent	0.77990	2	1d
3389	NewZar	0.77990	20	1d

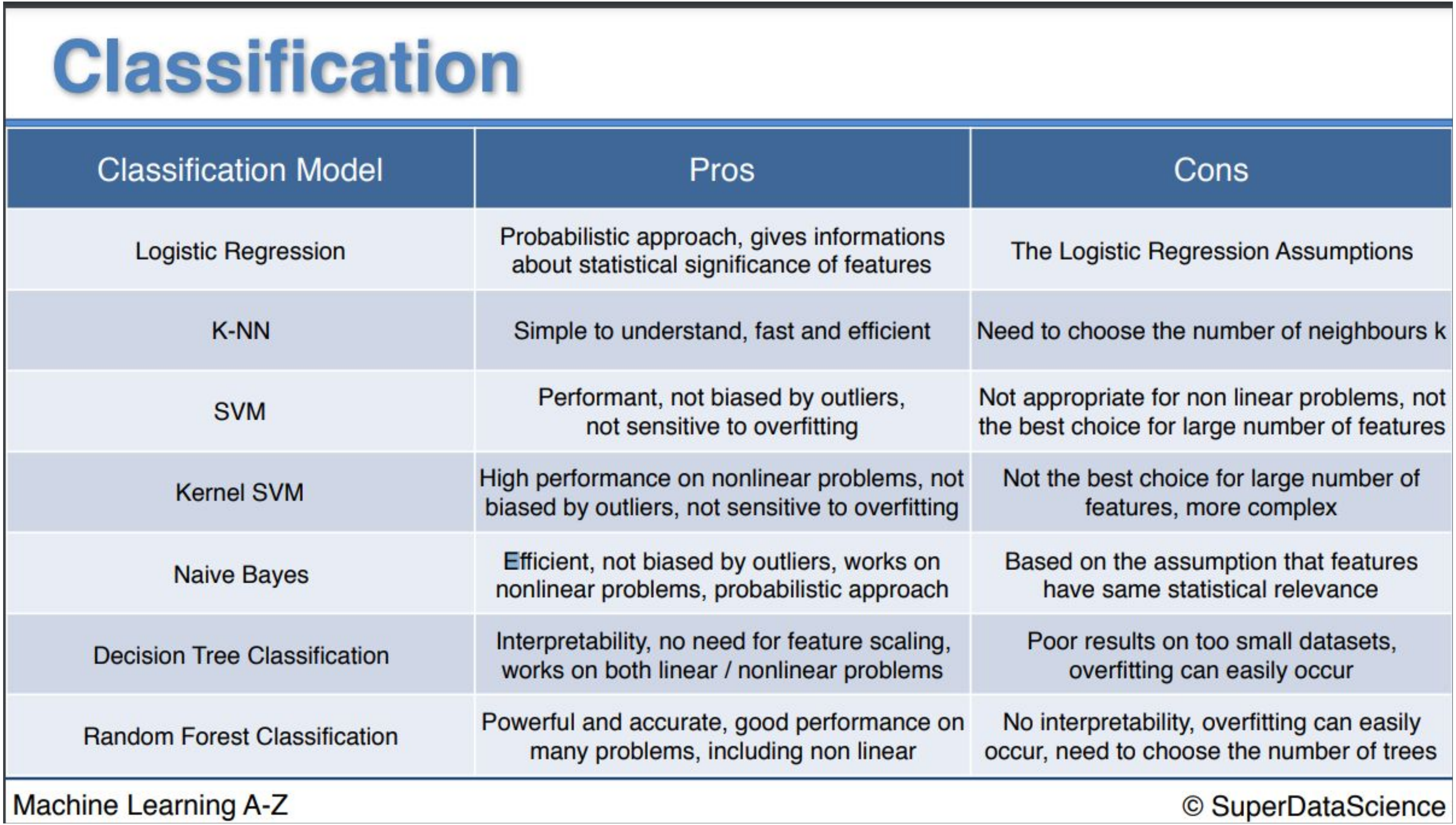
🙂 Your Best Entry!
Your submission scored 0.74162, which is not an improvement of your previous score. Keep trying!

We use cookies on Kaggle to deliver our services, analyze web traffic, and improve your experience on the site. By using Kaggle, you agree to our use of cookies.

Comment

I think the problem is non-linear. Firstly, the best performance came from Kernel SVM and SVM methods which are performant in non-linear problems. Also when I had changed the parameter of Kernel SVM from “rbf” to “linear”, the performance got worse. Inversely, I changed SVM’s kernel from linear to polly and result got better. My expectations for random forest and decision tree methods were very high but they literally disappointed me. I think I couldn’t handle the overfitting, then performance became worse than the others because they are sensible to overfitting. Due to reliability of Kernel SVM to overfitting, I got better result with it I guess.

On the other hand, maybe dataset can be small for random forest and decision tree methods. They work well with large amount of data rather than small data. However, Kernel SVM and SVM methods work better in small datasets according to chart on the next page that I took from my online AI course.



Classification

Classification Model	Pros	Cons
Logistic Regression	Probabilistic approach, gives informations about statistical significance of features	The Logistic Regression Assumptions
K-NN	Simple to understand, fast and efficient	Need to choose the number of neighbours k
SVM	Performant, not biased by outliers, not sensitive to overfitting	Not appropriate for non linear problems, not the best choice for large number of features
Kernel SVM	High performance on nonlinear problems, not biased by outliers, not sensitive to overfitting	Not the best choice for large number of features, more complex
Naive Bayes	Efficient, not biased by outliers, works on nonlinear problems, probabilistic approach	Based on the assumption that features have same statistical relevance
Decision Tree Classification	Interpretability, no need for feature scaling, works on both linear / nonlinear problems	Poor results on too small datasets, overfitting can easily occur
Random Forest Classification	Powerful and accurate, good performance on many problems, including non linear	No interpretability, overfitting can easily occur, need to choose the number of trees