Rapport d'analyse et de prédiction

A QU'ELLE POINT PEUT-ON PREVOIR LE SCORE D'UN ANIME ?

Aymeric Demange

Sommaire

Table des matières

Data Visualisation

Mon jeu de donnée portera sur les différents anime sortie depuis 2007 à 2018. Celui-ci possède 20 colonnes et 1563 lignes.

Dataset statistics

Number of variables	20
Number of observations	1563
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	244.3 KiB
Average record size in memory	160.1 B

Un nombre assez grand de colonnes possède des informations non conformes malgré ce que dit le graphique ci-dessus (Missing cells = 0) car chaque cellule étant « non conformes » possède enfaite un « - » au lieu d'être vide.

Exemple :

First rows

	Title	Type	Episodes	Status	Start airing	End airing	Starting season	Broadcast time
0	Fullmetal Alchemist: Brotherhood	TV	64	Finished Airing	2009-4-5	2010-7-4	Spring	Sundays at 17:00 (JST)
1	Kimi no Na wa.	Movie	1	Finished Airing	2016-8-26	-	-	-
2	Gintama°	TV	51	Finished Airing	2015-4-8	2016-3-30	Spring	Wednesdays at 18:00 (JST)
3	Steins;Gate 0	TV	23	Currently Airing	2018-4-12	-	Spring	Thursdays at 01:35 (JST)
4	Steins;Gate	TV	24	Finished Airing	2011-4-6	2011-9-14	Spring	Wednesdays at 02:05 (JST)
5	Ginga Eiyuu Densetsu	OVA	110	Finished Airing	1988-1-8	1997-3-17	-	-

Comme précisé dans la partie « Machine learning » je souhaite obtenir seulement des int ou float afin de pouvoir utiliser des algorithmes dessus. Actuellement nous avons :

Variable types

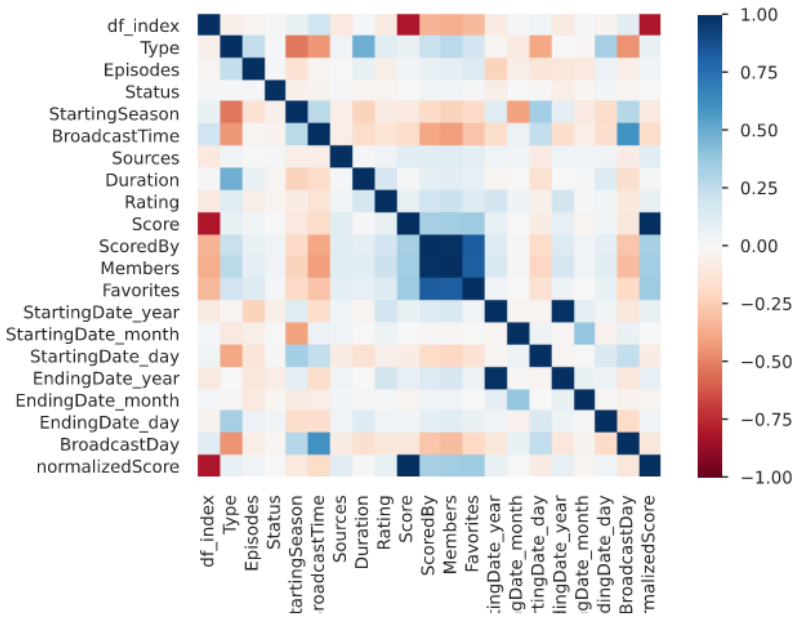
CAT	16
NUM	4

Il y aura donc 16 colonnes à traiter dans un premier temps, il faudra les convertir intelligemment en int ou float.

Voici le résultat du cleaning des données :

Type	Episodes	Status	StartingSeason	BroadcastTime	Sources	Duration	Rating	ScoredBy	Members	Favorites	StartingDate_year	StartingDate_month	StartingDate_day	EndingDate_year
5	64	1	1	59	6	1536	4	719706	1176368	105387	2009	4	5	2010
0	1	1	3	81	9	106	3	454969	705186	33936	2016	8	26	2016
5	51	1	1	63	6	1536	4	70279	194359	5597	2015	4	8	2016
5	24	1	1	26	12	1536	3	552791	990419	90365	2011	4	6	2011
3	110	1	3	81	8	1664	4	28452	121772	8370	1988	1	8	1997
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
5	12	1	2	81	6	1536	4	171506	296985	3576	2010	7	2	2010
3	1	1	3	81	6	1792	3	6062	12111	4	2013	8	6	2013
0	1	1	3	81	6	95	3	61505	104288	129	2009	8	1	2009
0	1	1	3	81	0	90	3	3054	12868	12	2012	6	9	2012
5	12	1	2	26	9	1536	4	46334	99299	330	2014	7	8	2014

Notre dataframe ne possède maintenant que des chiffres c'est parfait, il est possible de faire une matrice de corrélations afin de commencer à voir quelles colonnes affecte notre cible « Score »



On peut voir que ce qui affecte le plus notre cible « Score » ce sont les colonnes Favorites, Members et ScoredBy. Ce qui peut paraitre logique, lorsqu'une personne met en favoris un anime c'est qu'il souhaite le suivre ou l'a apprécié, ce qui indique un score plus ou moins élevé.

Data Modeling

Dans le data modeling on va souhaiter entrainer notre dataframe finalisé. C'est-à-dire qu'on va prendre un certain nombre de lignes, sur lesquels on va appliquer un ou plusieurs algorithmes de classification. Une fois entrainé il va tenter de fournir un Score qui ici est notre valeur cible. Plus il fournit un score proche de la vérité plus il sera précis, un algorithme précis est ce que l'on cherche, on essaie d'obtenir 100% de précision, cela voudrait dire que notre algorithme arrive à trouver à chaque fois le bon score.

J'ai utilisé dans mon étude trois algorithmes différents et ai gardé le plus performant :

- Random Forest Classifier

Chacun de ses algorithmes peut prendre plusieurs « paramètres » ce sont des informations fournies à l'algorithme afin qu'il soit le plus précis possible.

Voici les paramètres fournis pour l'algorithme : Random Forest Classifier

1) « n_estimators » : 35

2) « max_features » : 18

3) « max_depth » : 25

```
74.13793103448276 %
```

Évaluation

Dans l'évaluation on va vouloir utiliser des « metrics » nous permettant d'évaluer si nos algorithmes ont bien fonctionné.

1) Les métriques

Nous allons utiliser 3 métriques :

- Une matrice de confusion

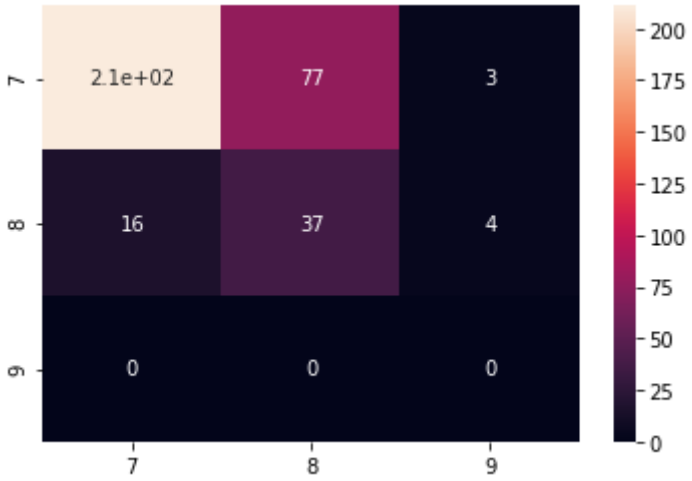
- Une courbe ROC

- Une courbe de validation

1.1) La matrice de confusion

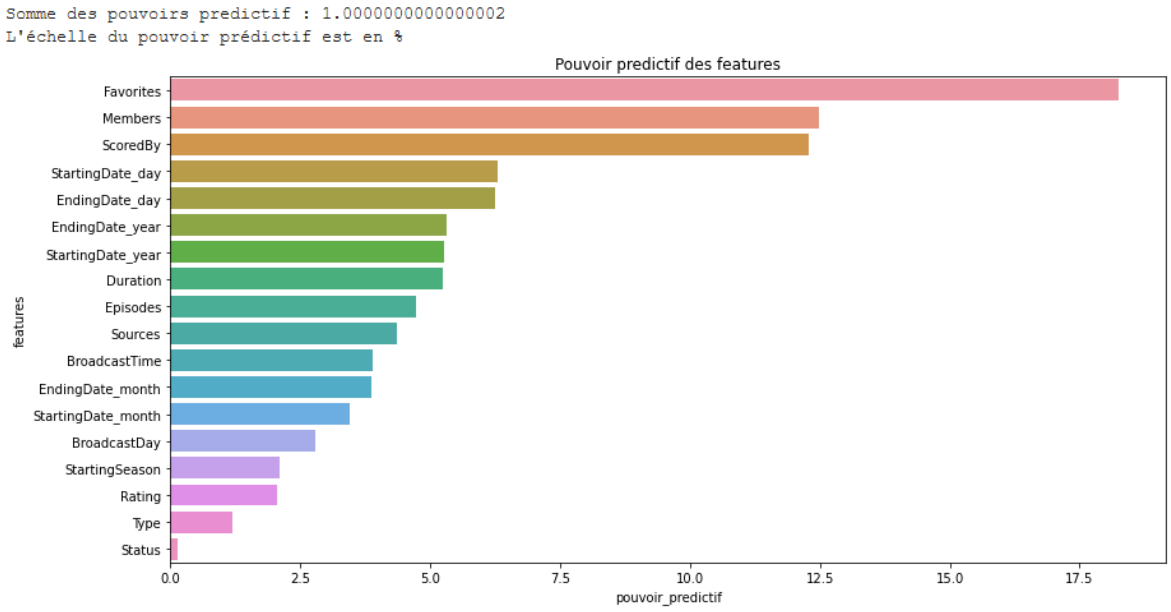
Une matrice de confusion va pouvoir nous montrer le nombre de faux positifs et faux négatifs présents dans notre prédiction.

Dans l'image ci-dessous on peut observer que notre algorithme SVC a 220 faux positifs en renvoyant le score 7, 90 faux positifs avec le score 8 et 4 faux positifs avec le score 9



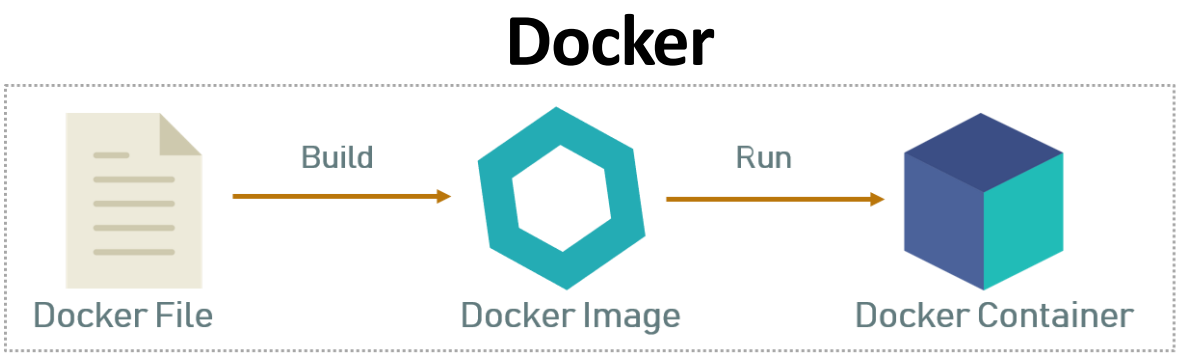
2) L'importance de nos colonnes

Il est possible de voir après avoir fait le data modeling à quel point une colonne aussi appelé « feature » influe sur notre précision. Voici par exemple comment nos features influe sur notre précision :



On peut donc voir que comme dit dans la partie « Data Preparation » lors de l'étude de la matrice de corrélation, nos colonnes les plus importantes sont bien Favorites, Members et ScoredBy.

En quoi est-ce important de regarder quelle colonne impacte notre précision c'est qu'il est possible par la suite de faire un choix entre obtenir une grande précision mais potentiellement un temps d'exécution plus long de notre code ou alors de supprimer les colonnes impactant le moins possibles notre précision et augmenter la vitesse d'exécution.



Le Docker file permet de choisir quelles vont être les paramètres de notre docker. Une fois l'image créer elle est prêtre à être « run » ce qui veut dire qu'elle va suivre un enchainement de commande afin de préparer un « container » qui lui vas être exécuté, lançant ainsi nos API, nos apps etc.

Exemple d'app lancer sur un container nommé « anime:2.3 »

A terme une url pourra être utilisé afin d'accéder à notre application.

Conclusion

Durant ce rapport j'ai pu faire l'analyse d'un dataset contenant différentes informations tournant autour des animes. Mon objectif était de réussir à prédire le score fourni par la communauté en fonction de toutes les autres colonnes fournies. La préparation fut longue pour transformer toutes les colonnes mais cela m'a donnée la possibilité d'atteindre une précision sur ma prédiction de 73% au maximum. Il est donc possible d'en conclure en disant que le jeu de donnée est surement trop petit pour pouvoir entrainer nos algorithmes a plus de 73%, de nouvelles features ou lignes pourrait améliorer la précision.

Il est quand même possible de prédire une note approximative qu'un nouvel anime aurait en sortant.

Bibliographie

Dataset anime : https://www.kaggle.com/canggih/anime-data-score-staff-synopsis-and-genre

Annexe