Prédiction du revenu et de la note d'un film à partir de ses caractéristiques

Lorraine de Talhouet
lorraine.de-talhouet@student.ecp.fr

Hajar Khairallah
hajar.khairallah@student.ecp.fr

Abstract

L'objectif du projet est de prédire le revenu et la popularité d'un film à partir de multiples caractéristiques, tels que le genre, la société de production, la langue, le budget…
Une première approche consistera à prédire si un film est rentable ou pas. Puis, dans une deuxième temps, nous chercherons à classifier les films, tout d'abord en 4 classes selon leur niveau de rentabilité, et ensuite en 6 classes en fonction de la moyenne des notes attribuées par les spectateurs. Nous avons choisi le modèle des perceptrons à couches multiples (MLP) pour ce problème de classification, dont l'optimisation des paramètres sera effectuée par un Grid Search. Les performances seront mesurées avec l'accuracy.

La première étude de rentabilité binaire permet d'atteindre un taux de classification de 61.4%, résultat conséquent par rapport au hasard. Pour la deuxième expérience avec les 4 classes de rentabilité, nous avons obtenu un score de 37.9%. Notre algorithme restant peu performant en vue du nombre limité de données après 'nettoyage', nous nous sommes orientées vers la popularité des films qui est la note moyenne attribuée par les spectateurs, pour laquelle plus de données propres étaient disponibles. Nous atteignons alors un taux de classification de 27.4%. Comparé au choix aléatoire qui est à 17%, ce résultat reste intéressant.

La suit de notre projet compare les résultats obtenus par le modèle du perceptron à couches multiples, avec un autre modèle de machine learning Random Forest. Celui-ci permet légèrement d'avoir de meilleurs résultats que le MLP sur nos deux sujets de classification. Nous obtenons dans un premier temps un score de de 39.6% pour la classification de revenus et un score de 27.6% pour la classification des notes moyennes, selon les classes que nous avons explicitées précédemment.

Nous proposons enfin des améliorations sur notre modèle qui permettraient des prédictions de classe avec plus de précision.

Introduction

Récemment, des sites qui ne servaient à l'origine que de plateforme de visionnage de films et de séries, tels que Netflix, pivotent : elles se lancent dans la production de films/séries sous leur propre marque. Grâce aux données récoltées auprès des utilisateurs, ils ont accès à l'historique des succès de toutes les réalisations proposées sur leur site, et peuvent utiliser ces données pour anticiper le succès d'un film. La série House of Cards a par exemple été créée en voulant 'cocher' des critères de succès préétablis. Le même raisonnement est effectué dans les maisons de production avec les données en leur possession.

Le but de ce projet est de prévoir le succès d'un film en fonction de caractéristiques telles que le genre, le budget, la langue, ou s'il est la suite d'une collection par exemple. Nous avons décidé de prendre comme mesure de prédiction le revenu puis la note d'un film, obtenue en calculant la moyenne de tous les votes des spectateurs. Le potentiel objectif serait par exemple de permettre aux producteurs d'évaluer dès le départ leur retour sur investissement et le succès de leur création.

Etat de l'art

L'analyse des succès de films a fait l'objet de nombreuses études, effectuées autant par des étudiants que des leaders de l'industrie. L'étude de ces prédictions s'est souvent résolue par un problème de classification plutôt que de régression. Il s'agit de catégoriser les films et séries par des classes allant du 'flop' au 'grand succès', en se basant sur le chiffre d'affaires, auquel on soustrait le budget.

Une des études que nous avons trouvée sur le site www.sciencedirect.com, effectuée par Ramesh Sharda et Dursun Delen des chercheurs de l'université d'Oklahoma, présente justement un projet de classification financière. Il prend en compte sept critères indépendants : le rang du MPAA (Motion Picture Association of America), la situation concurrentielle lors de la sortie, la présence de stars du cinéma dans le casting, le genre, les effets techniques, le nombre de projections lors de la sortie et s'il s'agit d'une suite ou pas d'une collection. Un modèle 'Multi Layer Perceptron' (MLP) avec deux couches cachées, de respectivement 18 et 16 neurones, avec comme une fonction de transfert une sinusoïde, a été choisi. En entraînant le modèle avec une cross-validation en 10 parties, l'accuracy (ici la précision, soit le taux de classification correcte) s'élève à 36.9% pour l'ensemble et à 75.2% pour une seule classe, deux résultats bons. L'auteur de l'article conclut sur la pertinence du modèle par rapport à d'autres du type régression logistique ou arbre de régression et classification.

Il existe d'autres études effectuées avec un autre modèle, également très performant : les "convolutional neural networks", dits 'CNN'. Une application de ce modèle plus complexe est décrite dans un article de www.rd.springer.com écrit par Yao Zhou, Lei Zhan et Zhang Yi en août 2017. Il s'agit également de prédire le succès d'un film, mais en prenant en entrée les affiches des films, et d'en extraire des features, transformées en vecteurs. La particularité d'un réseau CNN résulte dans la présence d'au moins une couche qui utilise la convolution en place du produit matriciel. Les résultats sont excellents : dans l'article précédemment cité, l'accuracy s'élève à 51.45% pour l'ensemble et 86.44% pour une seule classe !

Néanmoins, l'extraction de features pour un réseau CNN nécessite plus de temps et plus de mémoire que le réseau MLP. Elle requiert également un espace de stockage conséquent pour les affiches des films. En vue du temps que nous avions et des ressources à disposition, nous avons préféré nous concentrer sur l'entraînement de modèles uniquement MLP.

Dans le cadre de notre projet, nous nous sommes tout d'abord intéressées à la classification des films en fonction du niveau de rentabilité (rentabilité supérieure ou inférieure à 1). N'étant pas satisfaites de nos résultats et souhaitant pousser notre recherche, nous avons souhaité appliquer le même principe à la classification des films par leur note moyenne.

Notre Approche

Notre approche repose sur deux concepts du machine learning : le perceptron à couches multiples et les random forests. Nous établissons une comparaison entre les deux concepts en termes de résultat en classification, comme un cas de régression.

Le MLP est un réseau de neurones connu pour sa fonction d'approximation pour les problèmes de régression et la classification. Si la structure et la taille du réseau sont optimisées, le MLP apprend à modéliser toute fonction non linéaire (Théorème d'approximation universelle). Le MLP nous a donc semblé être un bon candidat à l'approximation de la performance des films à partir de leurs diverses caractéristiques.

Nous avons opté pour un MLP à deux couches cachées, avec respectivement 18 et 16 neurones.

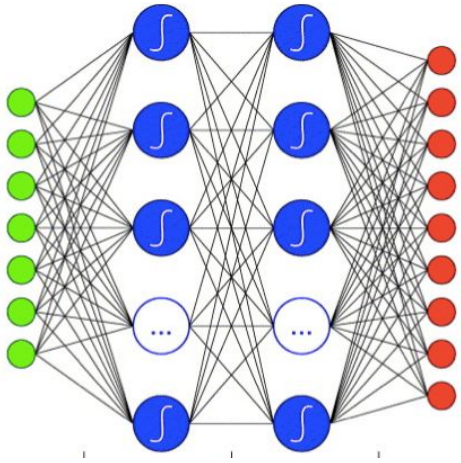

Fig 1 - *Schéma représentant un perceptron à deux couches*

Une fonction d'activation logistique a été utilisée. Nous avons initialement opté pour ces hyperparamètres sous les conseils de recherches précédemment effectuées sur le sujet. Ces hyperparamètres sont par la suite optimisés avec un Grid Search.

Pour évaluer le modèle, nous utilisons la validation croisée à 2 parties de façon à le valider puis à le tester sur l'ensemble du jeu de données tout, et en prenant un minimum de temps.

Ainsi, nous avons utilisé MLPClassifier de la bibliothèque sickit-learn facilement paramétrables pour approcher le sujet avec un réseau de neurones.

Le choix suivant a été de comparer notre premier travail avec le random forest à travers l'utilisation de *RandomForestClassifier* de scikit-learn. Cet algorithme est en effet simple d'utilisation et permet d'exploiter le potentiel de toutes nos caractéristiques. Le concept de tree bagging construit des arbres de décision à partir d'un ensemble d'apprentissage, aléatoirement généré par tirage avec remise. Le feature sampling restreint le choix à chaque noeud de l'arbre à quelques caractéristiques. L'algorithme calcule la moyenne des arbres obtenus, et diminue les biais et la variance des résultats.

Nos Expériences

1/ Environnement

Nous avons réalisé nos modèles expérimentaux avec les librairies Python Tensorflow et Scikit-learn et nous avons utilisé Jupyter Notebook comme interface.

2/ Données

Une étape préalable à l'exploitation de cette architecture est la construction d'un jeu de données pertinentes pour cette étude. Nous avons décidé d'utiliser les données disponibles sur le site kaggle, appelées "Full MovieLens Dataset". Ce dataset contient 20 millions de votes de 270 000 utilisateurs pour 45 000 films, dont voici quelques graphes d'aide à la compréhension de la composition.

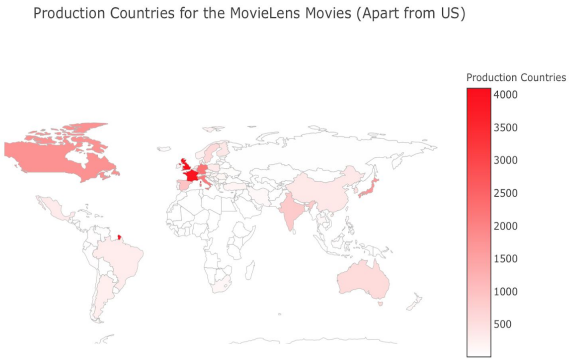

Fig 2 - *Répartition géographique des maisons de production*

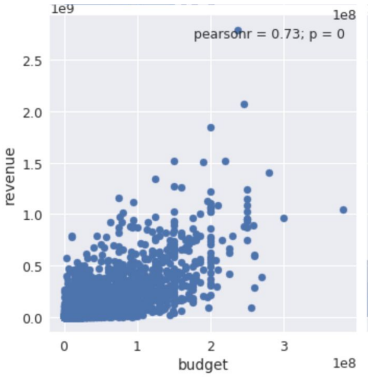

Fig 3 - *Répartition du revenu en fonction du budget*

Certains attributs ne nous semblent pas pertinents pour une première approche. En conséquence, nous supprimons le titre, l'identifiant, l'aperçu général, la 'tagline', l'image de l'affiche, le lien de l'image, la vidéo et l'attribut 'adulte'.

Nous devons ensuite opérer à des transformations pour labelliser certaines futures entrées du modèle, comme par exemple le genre, l'appartenance à une série, la maison de production, l'utilisation de la langue anglaise ou non… Il s'agit ici d'attribuer une valeur numérique (un 'classe' numérique) pour chaque entrée d'un attribut. Par ailleurs, nous homogénéisons également l'année, le mois, le jour et la durée du film en 'float'.
Nous obtenons ainsi une base de données exploitable avec laquelle il est possible d'entraîner notre modèle.

	0	1
belongs_to_collection	1.000000e+00	0.000000e+00
budget	3.000000e+07	6.500000e+07
genres	2.000000e+00	1.000000e+00
homepage	5.250000e+02	-1.000000e+00
original_language	1.000000e+01	1.000000e+01
popularity	2.194694e+01	1.701554e+01
production_companies	1.363000e+03	1.809000e+03
production_countries	7.400000e+01	7.400000e+01
release_date	1.681000e+03	1.693000e+03
revenue	3.735540e+08	2.627972e+08
runtime	8.100000e+01	1.040000e+02
spoken_languages	1.000000e+01	1.000000e+01
status	1.000000e+00	1.000000e+00
title	6.697000e+03	2.929000e+03
vote_average	7.700000e+00	6.900000e+00
vote_count	5.415000e+03	2.413000e+03
year	7.700000e+01	7.700000e+01
month	1.000000e+01	2.000000e+00
day	1.000000e+00	0.000000e+00

Fig 4 - *Aperçu des données post traitement et transformation en catégories*

3/ ***Expérience 1*** : Classification binaire de la rentabilité avec un réseau de neurones

Le problème est dans une première approche détournée vers une étude de rentabilité. Le modèle qui est un simple perceptron à couches multiples permettra de décider si un film est rentable ou pas. Nous remplissons une colonne "Rentabilité" définie par :

$$\frac{Revenue}{Budget} = Rentabilité$$

En utilisant une validation croisée et le MLP expliqué précédemment (deux couches cachées avec respectivement 18 et 16 neurones, et une fonction d'activation ReLu), nous obtenons la matrice de confusion suivante :

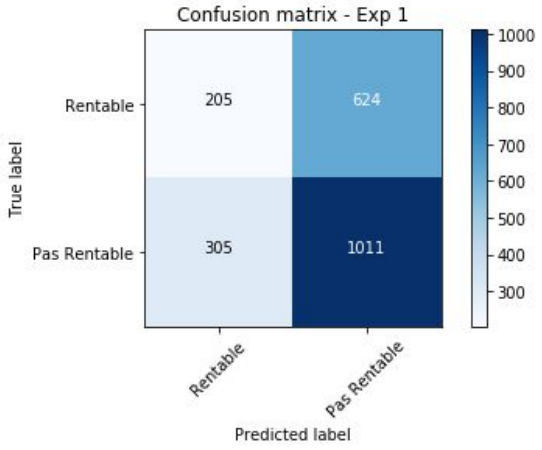

Fig 5 - *Matrice de confusion pour l'expérience 1*

Cette expérience permet d'obtenir un taux de classification de **56.7%** . Ce score est à mettre en contraste avec un tirage aléatoire ou un choix systématique de la classe la plus représentée. Pour le premier, il nous permet d'avoir un score de 50% (tirage aléatoire parmi deux classes) et pour le deuxième, en choisissant automatiquement la classe la plus représentée (ie la classe rentable), nous obtenons un score de 61%.

Face à cette performance intéressante mais faible, nous tentons tout d'abord d'harmoniser le

nombre d'éléments entre les deux classes et retirons les films de rentabilité supérieure à 3, pour obtenir deux classes équilibrées (2776 non rentables et 2174 rentables). Nous décidons d'optimiser nos hyperparamètres de façon à améliorer l'accuracy score.

Nous lançons un Grid Search sur les hyperparamètres suivants : le nombre d'époques (tout en se limitant à un nombre raisonnable pour éviter le surapprentissage), l'optimiseur (Stochastic Gradient Descent ou Adam Optimiser), la fonction d'activation (Relu ou Sigmoid), la taille d'un batch, et le nombre de neurones par couches. Notre accuracy score augmente alors à **61,44%**, ce qui est cette fois meilleur qu'un tirage aléatoire et que le choix automatique de la classe prépondérante (score de 56%).

Ce résultat reste faible, et peut s'expliquer notamment par le nombre limité de données (5000 lignes à peu près).

4/ ***Expérience 2*** : Classification des revenus en classes multiples avec un réseau de neurones

Le faible résultat de la première expérience s'explique tout d'abord par la répartition déséquilibrée des données sur les deux classes. Pour pallier à cela, nous avons pensé à répartir nous-mêmes les données sur plus de classes. Nous obtenons donc la répartition suivante:

Valeur de la rentabilité	Nombre de lignes
Return < 0.38	1647
Return < 1.6	1932
Return < 3.6	1745
Return >= 3.6	1815

Nous utilisons alors le MLP utilisant les mêmes paramètres que le précédent (avec les résultats de Grid Search) et obtenons un accuracy score de **29,5%** à comparer avec le hasard (score de 25%), et la catégorie prépondérante en termes de volume correspondant à "Return<1.6" (score de 27%).
Dans une optique d'optimisation des hyperparamètres, nous faisons un Grid Search sur les mêmes hyperparamètres que le premier exemple et utilisons le modèle en résultat. Le seul changement concerne la fonction d'activation qui était une sigmoïde et qui devient une fonction ReLu. Nous obtenons donc la matrice de confusion suivante :

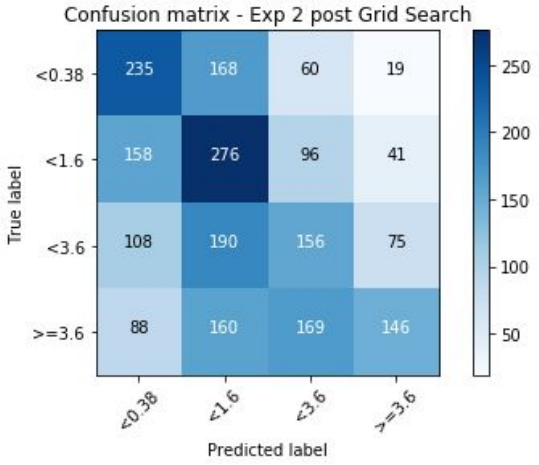

Fig 6 - *Matrice de confusion pour l'expérience 2*

Notre accuracy score est alors de **37,9%**.
Notre deuxième expérience permet donc de prédire si un film est dans l'une des quatre catégories de rentabilité avec une accuracy de 37,9%. C'est un score est relativement correct par rapport à un tirage aléatoire, mais il reste peu satisfaisant pour un producteur de film.

5/ ***Expérience 3*** : Classification des revenus en classes multiples avec Random Forest

Nous tentons une nouvelle expérience de classification, mais cette fois-ci avec un Random Forest à 500 arbres d'estimations. Le résultat est légèrement supérieur au MLP. Nous obtenons un score de **39,65%** vs 37,9% avec le MLP.

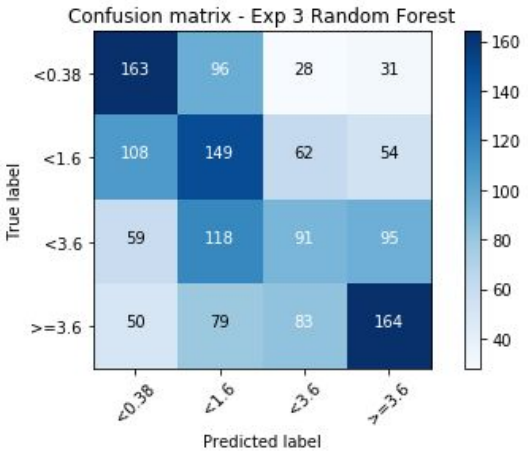

Fig 7 - *Matrice de confusion de l'expérience 3*

5/ ***Expérience 4*** : Classification de la popularité en classes multiples avec un réseau de neurones

Les résultats de classification de revenus précédemment obtenus avec un MLP, ne satisfont pas l'objectif business de notre étude. Comme précisé précédemment, ceci peut s'expliquer par le nombre relativement faible de données après suppression des lignes à revenu nul. En effet, après nettoyage, la base de données passe de 45 000 à 4 000 lignes environ. En revanche, si l'on s'intéresse à la note moyenne, le nombre de lignes après nettoyage et suppression des lignes nulles, reste aux environs de 45 000. Nous décidons alors d'entraîner un modèle, toujours selon la même méthode que les précédentes, mais sur cette nouvelle base de données plus conséquente en terme de volume - et donc potentiellement plus concluante pour des méthodes de Deep Learning. Grâce à cette taille initiale plus importante, nous pouvons désormais séparer nos données en trois ensembles (en plus de la cross-validation) : apprentissage, validation et test. C'est le point principal de différenciation; il permettra de donner plus de crédibilité à ce dernier modèle de classification.

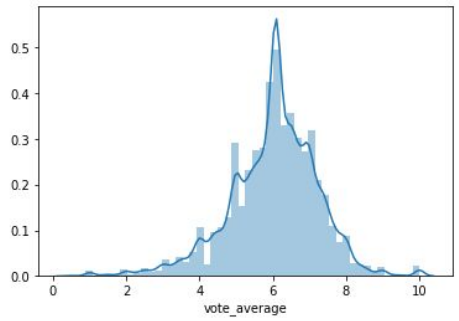

Fig 8 - *Répartition des films selon leur note*

Les données sont majoritairement condensées entre les notes 5 et 6, et nous optons pour la répartition des données sur les 6 catégories suivantes, en restant vigilantes à avoir environ le même volume pour chaque partie.

Note du film	Nombre de lignes
Note < 5	6890
Note < 5.5	5331
Note <= 6	8466
Note <= 6.4	8491
Note <= 7	8574
Note > 7	7714

Nous lançons un Grid Search sur l'ensemble d'apprentissage et de validation afin de sélectionner les hyper paramètres optimaux.
Le Grid Search nous propose un modèle avec une fonction d'activation sigmoïde, 20 époques, et un optimiseur Adam. L'accuracy sur l'ensemble de validation et de 26,5%. Cette valeur est largement supérieure à un tirage aléatoire, qui est de 16,7% ou encore de celle du choix de la classe prédominante qui est de 19% environ.
Enfin, il ne reste plus qu'à évaluer les performances sur l'ensemble "test", mis de côté depuis le début de notre expérience. Nous obtenons alors une accuracy de **27,4%** et la matrice de confusion suivante.

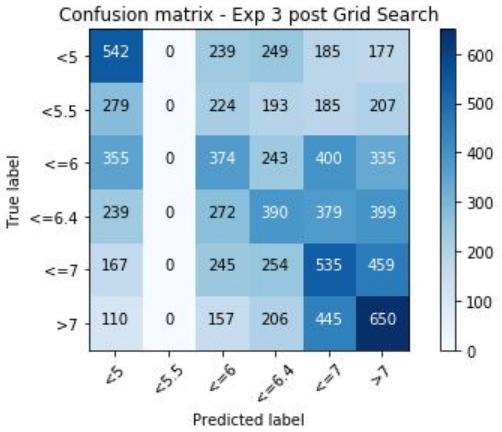

Fig 9 - *Matrice de confusion de l'expérience 4*

Nous remarquons que le modèle n'apprend pas à classer les données dans la deuxième classe (celle des films notés entre 5 et 5.5). Nous expliquons cela par le fait que c'est la classe avec le moins de volume de données.

6/ ***Expérience 5*** : Classification de la popularité en classes multiples avec Random Forest

Nous comparons la performance d'un apprentissage profond avec un concept de machine learning plus simple, comme le Random Forest. Nous obtenons le résultat suivant avec 500 arbres d'estimation.

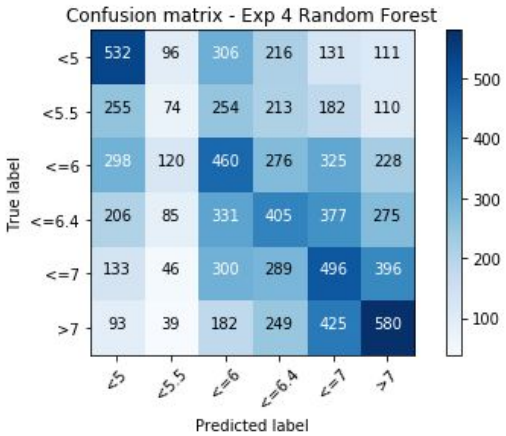

Fig 10 - *Matrice de confusion de l'expérience 5*

Nous remarquons que le modèle de Random Forest apprend sur la deuxième classe également, contrairement au MLP. Nous obtenons une accuracy de **27,6%** qui est donc légèrement meilleure que celle obtenu avec le modèle de deep learning.

Conclusion

Nous avons approché le problème de prédiction de revenus de films et de notes de films par la méthode d'apprentissage profond en utilisant deux modèles : celui de perceptron à couches multiples, et le concept de machine learning Random Forest. Après nos diverses expériences, nous avons conclu que le deuxième concept était plus performant pour notre base de données. Le deep learning nous a donc apparu comme ayant peu de valeur ajoutée pour notre projet de classification et pour le type de données dont nous disposions. Un approfondissement de notre étude pourrait commencer par l'enrichissement des données en terme volume et de caractéristiques (par exemple en ajoutant des colonnes avec les artistes principaux, le scénariste, le réalisateur, qui peuvent également être des éléments pertinents quant au succès d'un film). Ensuite, il serait possible d'utiliser une colonne que nous avons éliminée dès le départ : "poster path", qui regroupe les différentes affiches de films. Enfin, l'étape suivante serait d'essayer d'approcher le problème de prédiction des revenus des films et de leur popularité à travers un réseau de neurones convolutif comme proposé dans l'étude de Yao Zhou, Lei Zhang et Zhang Yi, précédemment mentionnée et qui a obtenue des résultats plus convaincants que les nôtres.

Bibliographie

[Yoo_11]

[Steven Yoo, Robert Kanter, David Cummings, Predicting Movie Revenue from IMDb Data, Stanford, USA, 2011]

[Ram_05]

[Ramesh Sharda, Dursun Delen, Predicting box-office success of motion pictures with neural networks, Oklahoma University, USA, 2005]

[Yao_17]

[Yao Zhou, Lei Zhang, Zhang Yi, Predicting movie box-office revenues using deep neural networks, Sichuan University, China 2017]