Diagnostic d'un cancer grâce au Machine Learning

Isaure Quétel

Introduction

Le diagnostic d'un cancer est essentiel pour mettre en place la meilleure stratégie thérapeutique pour chaque patient. Traditionnellement, ce diagnostic s'effectue manuellement par les médecins. Depuis quelques années, une méthode est recherchée pour pouvoir effectuer un diagnostic de cancer automatique. Cela est maintenant possible grâce au Machine Learning pour pouvoir améliorer les diagnostics effectués et permettre de gagner du temps. Le Machine Learning est une technique d'intelligence artificielle qui consiste à « apprendre » à une machine, à partir de données, à effectuer des prédictions. Cette méthode est utile dans de nombreuses situations comme pour la détection d'écriture manuelle, les recommandations de produits sur le net, la conduite autonome entre autre mais également à effectuer des diagnostics médicaux comme l'exemple abordé ici ou également la prédiction de crises cardiaques. Dans notre cas, on a l'expression de 20531 gènes de 801 patients porteurs de quatre différentes tumeurs. Ces quatre tumeurs différentes sont le cancer du sein (BRCA), l'adénocarcinome du côlon (COAD), le carcinome rénal à cellules claires du rein (KIRC), l'adénocarcinome pulmonaire (LUAD) et l'adénocarcinome de la prostate (PRAD). Les données proviennent de la base de données UCI Machine Learning Repository qui contient 559 sets de données permettant à la communauté du Machine Learning de pratiquer. Le but ici est de mettre en place un modèle permettant le diagnostic d'un cancer grâce à notre set de données.

Matériel et méthodes

Pour permettre cette méthode de Machine Learning, Python 3.8.5 a été utilisé, ainsi que de nombreuses librairies et fonctions. On retrouve les librairies Pandas 1.1.2, Scikit-learn 0.23.2 et Matplotlib 3.3.1. Pandas permet de manipuler les données et dans notre cas d'effectuer des matrices. Scikit-learn permet le Machine Learning c'est-à-dire permet à la machine d'apprendre et donc de faire des analyses prédictives des données. Dans notre cas c'est-à-dire donner le diagnostic du cancer. Cette librairie permet aussi de donner des informations sur la qualité du modèle choisi etc. Et enfin Matplotlib permet de créer et de visualiser les graphiques.
Tout d'abord un modèle simple a été effectué pour connaître les données basiques des données avec par exemple le nombre de patients ayant chaque cancer pris dans l'étape de test. Le but sera donc de se rapprocher le plus de ces données. Ensuite on effectue un underfitting pour vérifier si les données sont meilleures ou pas. Ce n'était pas le cas donc un overfitting a été effectué. Ensuite, les données ont été régularisées pour que le modèle soit le meilleur possible. Et enfin, une learning curve et un grid search ont été effectués.
Le dossier avec le script python est disponible au lien https://github.com/isaurequetel/projet_biostatistiques.

Résultats

Les données ont d'abord été scindées, on a d'un côté le type de cancer pour chaque patient (Y) et d'un autre côté toutes les expressions géniques pour chaque patient (X).

1. Simple model

pd.crosstab(y_test,z)

	BRCA	COAD	KIRC	LUAD	PRAD
BRCA	101	0	0	0	0
COAD	0	29	0	0	0
KIRC	0	0	47	0	0
LUAD	0	0	0	43	0
PRAD	0	0	0	0	45

Dans l'échantillon de test il y a une majorité de patients atteint du cancer du sain (BRAD), puis du carcinome rénal à cellules claires du rein (KIRC), de l'adénocarcinome de la prostate (PRAD), de l'adénocarcinome pulmonaire (LUAD) et enfin de l'adénocarcinome du côlon (COAD).

2. Underfitting

Accuracy score train the model :
y_test, X_test 0.34
y_train, X_train 1

pd.crosstab(y_test,Z)

	BRCA	COAD	KIRC	LUAD	PRAD
KIRC	0	0	46	0	0
PRAD	101	29	1	43	45

En underfitting on remarque que l'accuracy de test est mauvaise mais que l'accuracy de l'entrainement est parfaite.

3. Overfitting

Accuracy score train the model :
y_test, X_test 0.46
y_train, X_train 0.55

pd.crosstab(y_test,Z)

	BRCA	COAD	KIRC	LUAD	PRAD
BRCA	65	13	18	13	20
COAD	7	15	0	1	3
KIRC	9	0	21	4	1
LUAD	1	0	0	0	0
PRAD	19	1	8	25	21

En overfitting, on remarque que les deux valeurs d'accuracy, en test et en entrainement, sont mauvaises. Le modèle n'est donc pas assez bon.

4. Régularisation

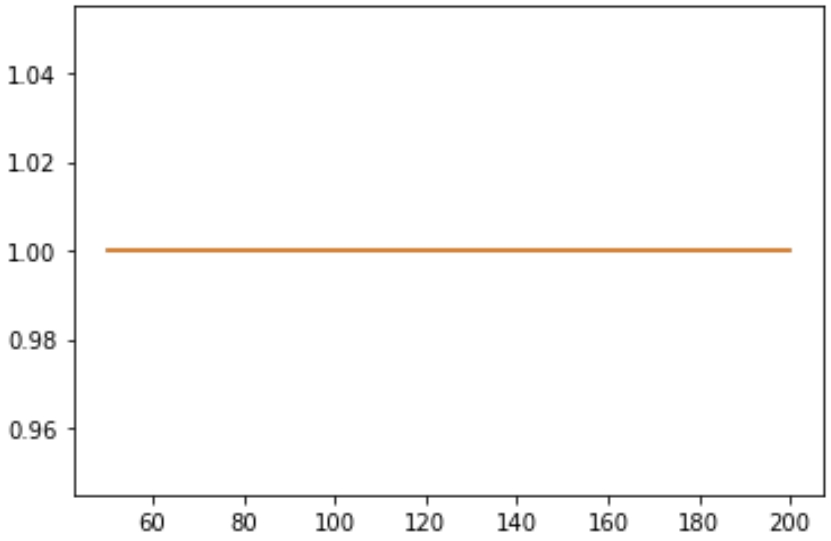
Accuracy score train the model :
y_test, X_test 1
y_train, X_train 1

	BRCA	COAD	KIRC	LUAD	PRAD
BRCA	101	0	0	0	0
COAD	0	29	0	0	0
KIRC	0	0	47	0	0
LUAD	0	0	0	43	0
PRAD	0	0	0	0	45

Avec la régularisation on remarque que les accuracy du test et de l'entrainement sont parfaites. Le modèle est donc parfait.

5. Learning curve

print(index) [50,100,150,200]
print(training_accuracy) [1,1,1,1]



6. Grid search

{'C' : 0.001}

Conclusion

En conclusion, le Machine Learning est une méthode très importante pour les diagnostics médicaux. Dans notre cas, il a permis le diagnostic de cancers chez des patients grâce aux expressions géniques. Cette méthode de détection couplée aux diagnostic d'un médecin peut éviter les erreurs et faire gagner du temps. Une meilleure stratégie thérapeutique peut donc être employé et des vies humaines peuvent potentiellement être sauvées. Le Machine Learning peut être imaginé pour d'autres utilisations, qu'elles soient biologiques ou non. C'est donc une méthode très importante pour le futur car elle permet une prédiction. Le but premier étant toujours d'améliorer ces prédictions.

Bibliographie

[1] Buitinck, Lars, et al. "API design for machine learning software: experiences from the scikit-learn project." *arXiv preprint arXiv:1309.0238* (2013).

[2] Danaee, Padideh, Reza Ghaeini, and David A. Hendrix. "A deep learning approach for cancer detection and relevant gene identification." *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2017*. 2017.

[3] Hunter, John D. "Matplotlib: A 2D graphics environment." *Computing in science & engineering* 9.3 (2007): 90-95.

[4] Hwang, Kyu-Baek, et al. "Applying machine learning techniques to analysis of gene expression data: cancer diagnosis." *Methods of microarray data analysis*. Springer, Boston, MA, 2002. 167-182.

[5]Kourou, Konstantina, et al. "Machine learning applications in cancer prognosis and prediction." *Computational and structural biotechnology journal* 13 (2015): 8-17.

[6] McKinney, Wes. "Data structures for statistical computing in python." *Proceedings of the 9th Python in Science Conference*. Vol. 445. 2010.