UVSQ

université PARIS-SACLAY

04/11/2021

TP 1 de Machine learning

Algorithme de KNN

FERHATI Amina
M2 DATASCALE

Table des matières

Introduction

En général, une méthode de validation est toujours nécessaire lorsqu'on veut déterminer les paramètres optimaux d'un modèle. Dans ce cas, on veut trouver la meilleure valeur pour le paramètre k de l'algorithme KNN en utilisant deux stratégies de validation afin d'établir une comparaison entre ces deux dernières et avoir le k optimal.
Tout en prenant en compte une nouvelle mesure d'erreur qui permet de pénaliser (coût *5) le fait de prédire une absence d'attaque cardiaque lorsqu'en réalité cette attaque cardiaque a lieu :
Erreur= FP + 5* FN / (FP+FN+TN+TP)

Analyse statique descriptive

- Les valeurs nulles :

Il n'y a aucune valeur nulle dans le jeu de données heart.

- L'équilibre des classes :

La répartition de la classe "label" : La label 2 veut dire qu'il y a une présence de maladie cardiaque et le label 1 veut dire l'absence de cette dernière. On voit bien que ces deux classes ne sont pas très bien équilibrées.

- Selon la boîte à moustaches :

On voit clairement que la majorité des instances de la variable THAL ont le label 2
On voit clairement que la majorité des instances de la variable EIA ont le label 2

- Selon le graphe de corrélation, on peut distinguer :

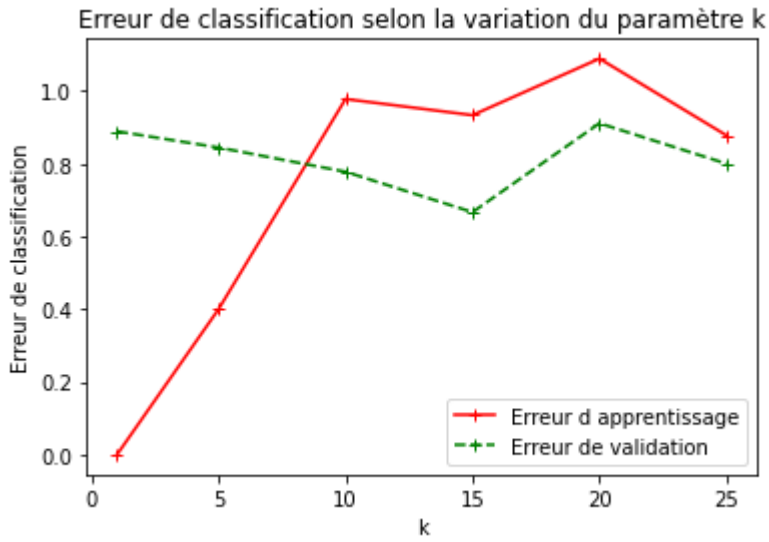
Les variables qui influencent le plus sur le label sont CPT (chest pain type), THAL et EIA (exercise induced angina).
Les variables les plus corrélées entre elles sont (OP : oldpeak et SPST : the slope of the peak exercise ST segment).

Première approche :

Principe de l'approche

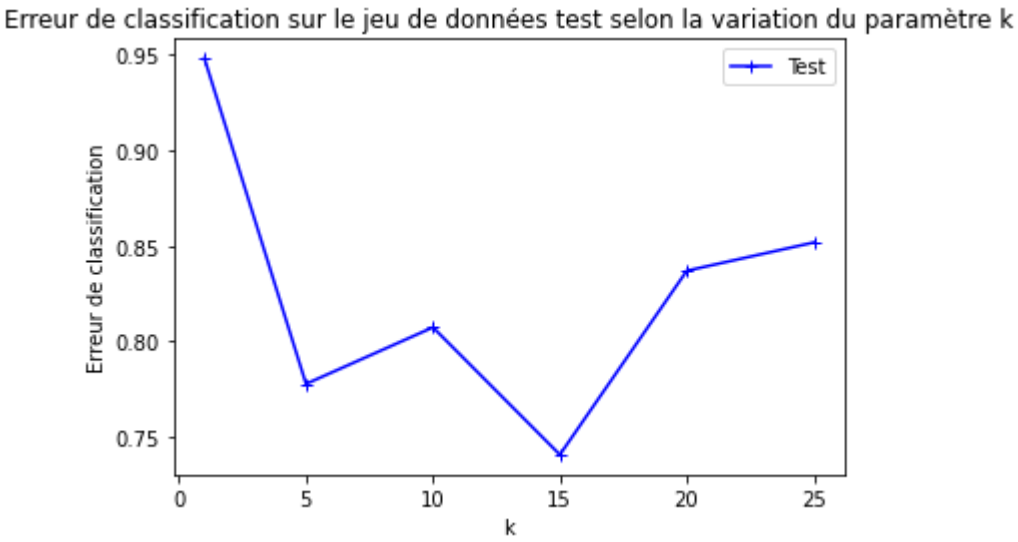
Le modèle est initialement ajusté sur l'ensemble d'apprentissage (1/3 de l'ensemble initial).. Successivement, le modèle ajusté est utilisé pour prédire les réponses pour les observations dans un deuxième jeu de données appelé validation (1/6 de l'ensemble initial) qui fournit une évaluation impartiale de l'ajustement du modèle sur le jeu de données d'apprentissage. Enfin, le jeu de données de test (1/ 2 de l'ensemble initial) est utilisé pour fournir une évaluation finale du modèle.

Résultats obtenus sur le jeu de données apprentissage et validation :



La courbe rouge qui représente l'erreur d'apprentissage augmente, car le modèle est en train d'apprendre.
La courbe verte qui représente l'erreur de validation qui baisse jusqu'à avoir la meilleure valeur de k qui est égale à 15 puis elle augmente encore une fois.

Résultat obtenu sur le jeu de données test :



On voit clairement dans ce graphique que la meilleure valeur de k c'est bien celle qu'on a choisi à partir de l'erreur de validation (k= 15), car avec cette valeur l'algorithme de knn donne le moins d'erreurs.
L'évaluation du meilleur modèle (pour k=15) sur le jeu de données test est 0.948.

Avantages/Limites de la stratégie

- Cette technique est efficace, sauf si les données sont limitées car il peut alors manquer certaines informations sur les données qui n'ont pas été utilisées pour l'entraînement, et les résultats peuvent donc être hautement biaisés.
+ Si l'ensemble de données est vaste et que la distribution est égale entre les deux échantillons, cette approche convient tout à fait.

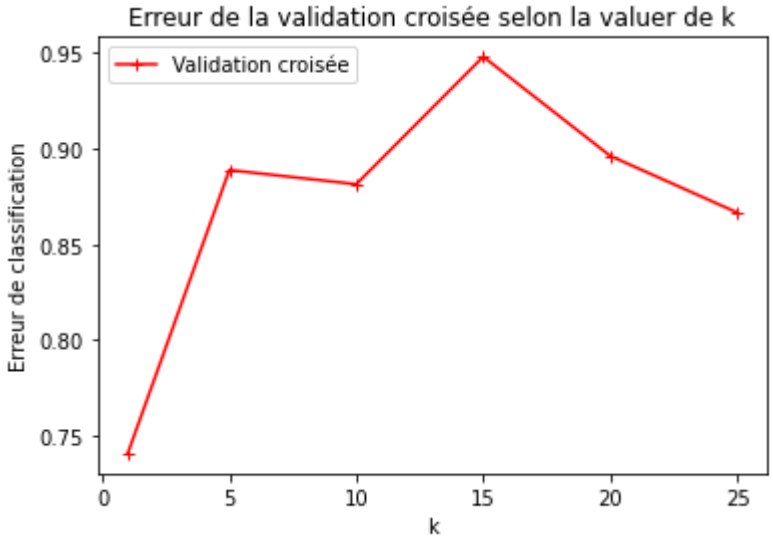
Deuxième stratégie

Principe de la stratégie

La cross validation est une technique très utilisée dans le domaine du ML, dans notre cas, pour garder l'équilibre des classes, on va utiliser StratifiedKFold comme implémentation pour la technique de validation croisée.

On commence tout d'abord par séparer l'ensemble d'apprentissage (X_av) en K= 5 parties. On ajuste ensuite le modèle en utilisant les parties K-1. Le modèle est validé en utilisant le K-fold restant. La moyenne des scores enregistrés est la métrique de performance du modèle.

Résultats de la stratégie sur le jeu de données apprentissage



La courbe rouge représente l'erreur de la validation croisée, comme on peut bien le voir, cette erreur augmente avec l'augmentation de la valeur de k.

Selon cette courbe, la meilleure valeur de k est 1.

Résultats de la stratégie sur le jeu de données test

L'évaluation de l'erreur en utilisant le meilleur modèle (pour k=1) sur le jeu de données test est 0.896.

Avantages/Limites de la stratégie

- **+** En cas de données limitées, il s'agit donc de l'une des meilleures approches, car elle permet d'assurer que toutes les observations de l'ensemble de données original aient la chance d'apparaître dans l'ensemble d'entraînement et dans l'ensemble de validation (l'utilisation totale des données).
- **+** Cette technique est utilisée pour éviter le surajustement et estimer la compétence du modèle sur de nouvelles données.
- **-** Le choix de la valeur K peut être difficile des fois car elle ne doit être ni trop basse ni trop haute, et on choisit généralement une valeur comprise entre 5 et 10 en fonction de l'envergure du dataset.
- **-** C'est un processus très lent et coûteux surtout si on a un grand nombre de données.

Conclusion

Durant ce TP, on a appris l'algorithme K-Nearest Neighbor ; son fonctionnement, la construction du modèle et l'évaluation sur un ensemble de données parlant des maladies cardiaques en utilisant le paquet Python Scikit-learn.
Pour de meilleurs résultats, la normalisation des données à la même échelle est fortement recommandée.
La méthode la plus performante est bien celle de la validation croisée, car cette méthode permet l'utilisation totale des données ce qui permet au modèle de bien apprendre et choisir le bon cas.

Améliorations possibles :

- On pouvait bien améliorer la première stratégie en réduisant la taille du jeu de données test (par exemple 30% ou bien 25 %) afin d'avoir plus de lignes dans le modèle d'apprentissage.
- Il est utile d'utiliser le Stratified K-folds à la place de cross validation afin d'établir un équilibre lors de la répartition du jeu de données en K-folds.
- Augmenter le paramètre de parallélisme dans le CrossValidator, qui définit le nombre de threads à utiliser lors de l'exécution d'algorithmes parallèles afin de rendre le processus plus performant.

Alternatives possibles :

- Méthodes bootstrap

Elle tire au hasard des ensembles de données à partir de l'échantillon original. La taille de chaque échantillon est égale à la taille d'apprentissage originale. Chaque modèle, avec ses hyperparamètres particuliers, est ajusté en utilisant les échantillons bootstrapés. Le ou les modèles sont examinés avec les données hors-sac, c'est-à-dire les données non sélectionnées dans chaque bootstrap.

- Leave one out

C'est un type de validation croisée K-fold où l'on ne retient qu'un seul échantillon à chaque fois. C'est un bon moyen de validation, mais il nécessite un temps de calcul élevé.