ÉCOLE NATIONALE SUPÉRIEURE DE TECHNIQUES AVANCÉES

ROB311 : Apprentissage pour la robotique

Parcours robotique

Année universitaire : 2019/2020

Classification par la méthode des K plus proches voisins

Auteurs :

M. AHMED YASSINE HAMMAMI

MME. HANIN HAMDI

Objectifs

L'objectif de ce travail est d'implémenter sur *Python* un algorithme de classification basé sur la méthode des *k plus proches voisins*. Lors de la classification, on utilise comme métrique la distance euclidienne afin de pouvoir déterminer la distance des éléments de la base de donnée les uns par rapport aux autres.

Afin de déterminer les performances de l'algorithme implémenté, on se base sur trois facteurs : l'erreur quadratique moyenne (MSE), la matrice de confusion et la précision.

L'algorithme des k plus proches voisins

L'algorithme K plus proches voisins (KNN) est un type d'algorithme ML supervisé qui peut être utilisé à la fois pour la classification et les problèmes de prédiction de régression. Toutefois, il est principalement utilisé pour les problèmes prédictifs de classification dans l'industrie.

L'algorithme du KNN utilise la *similitude des caractéristiques* pour prédire les valeurs des nouveaux points de données, ce qui signifie en outre que le nouveau point de données se verra attribuer une valeur en fonction de la proximité des points dans l'ensemble des informations d'entraînement.

On peut comprendre le fonctionnement de cet algorithme à partir des étapes suivantes :

- Pour implémenter n'importe quel algorithme, nous avons besoin d'un ensemble de données. Ainsi, pendant la première étape du KNN, nous devons charger les informations d'apprentissage ainsi que les données de test.
- Ensuite, nous devons choisir la valeur de K, c'est-à-dire les points de données les plus proches.
- Pour chaque point des données de test, on procède comme suit :
  - Calculer la distance entre les données de test et chaque ligne des données d'entraînement à l'aide de la distance euclidienne.
  - Maintenant, en fonction de la distance, on les trie par ordre croissant.
  - Ensuite, on choisit les k premières lignes de la liste triée.
  - Maintenant, on assigne une classe au point de test selon la classe la plus fréquente de ces lignes.
- Fin de l'algorithme.

Les données utilisées

Pour valider l'algorithme, on utilise deux bases de données *breast-cancer-wisconsin.data* et *haberman.data*.

Le premier ensemble de données représente des caractéristiques qui sont calculées à partir d'une image numérisée d'une fine ponction à l'aiguille (FNA) d'une masse mammaire. Ils décrivent les caractéristiques des noyaux cellulaires présents dans l'image. Les données sont représentées par 9 caractéristiques et sont classées selon 2 classe : "malignant" et "benign".

Le deuxième ensemble de données contient des cas tirés d'une étude menée sur la survie des patientes ayant subi une intervention chirurgicale pour un cancer du sein. Les données sont représentées par 3 caractéristiques et sont classée selon deux catégories : les patients qui survivent après 5 ans ou plus de l'opération et ceux qui meurent avant l'écoulement de 5 ans depuis leur opération.

La distance utilisée

Pour effectuer la classification, on utilise la distance euclidienne qui calcule la racine carrée de la somme des différences carrées entre les coordonnées de deux points :

$$D_e(x, y) = \sqrt{\sum(x_i - y_i)^2}$$

Influence de la valeur de k sur la classification

Le choix de la valeur K à utiliser pour effectuer une prédiction avec KNN varie en fonction du jeu de données. En règle générale, moins on utilisera de voisins plus on sera sujette au sous apprentissage. Par ailleurs, plus on utilise de voisins plus, sera fiable dans notre prédiction. Toutefois, si on utilise K nombre de voisins avec K=N et N étant le nombre d'observations, on risque d'avoir du overfitting.Pour cette raison, on a ajouté au code qu'on a implémenté une fonction qui calcule la meilleure valeur à choisir pour K en se basant sur la maximisation de la précision à chaque itération.

Analyse des performances de la prédiction

Pour analyser les performances de la classification, on s'est basé sur trois facteurs : la précision, l'erreur quadratique moyenne et la matrice de confusion.

Accuracy : Désigne la proportion des prédictions correctes effectuées par le modèle. Formellement, elle est définie ainsi :

$$\text{Accuracy} = \frac{N_c}{N_t}$$

Avec N_t : le nombre total des prédictions et N_c : le nombre des prédictions correctes.

L'erreur quadratique moyenne : On l'a utilisée pour saisir la qualité de la prédiction. Elle est représentée formellement par :

$$\text{MSE} = \frac{1}{N}\sum(Y_t - Y_p)^2$$

Avec :

- N : le nombre total des données test.
- Y_t : le vecteur de test qui caractérise l'attribution de classe pour chaque donnée.
- Y_p : le vecteur qui prédit Y_t.

La matrice de confusion : Une matrice de confusion est un résumé des résultats de prédictions sur un problème de classification. Les prédictions correctes et incorrectes sont mises en lumière et réparties par classe. La représentation de cette matrice nous a permit de comprendre de quelle façon le modèle de classification implémenté est confus.

Résultats de la classification

Erreur quadratique moyenne

L'erreur quadratique moyenne MSE fournit une indication par rapport à la dispersion ou la variabilité de la qualité de la prédiction. Cet indice peut être relié à la variance du modèle. Les valeurs de la MSE ne peuvent être interpréter qu'avec la comparaison avec la moyenne des prédictions.

Pour cela, on a calculé la moyenne de $Y\_predict$ qui vaut 2.842857142857143 dans la base de données *BREAST-cancer-wisconsin.data*. La valeur de la MSE est égale à 0.2.

On remarque que la variance du modèle correspond à seulement 7% de la moyenne des observations qui est faible. Notre prédiction donne de bonnes résultats.

Ces deux figures montrent la classifications des données dans le cas des données testées et leurs prédictions. On remarque que l'erreur est faible c'est à dire il n'existe que quelques fausses résultats.

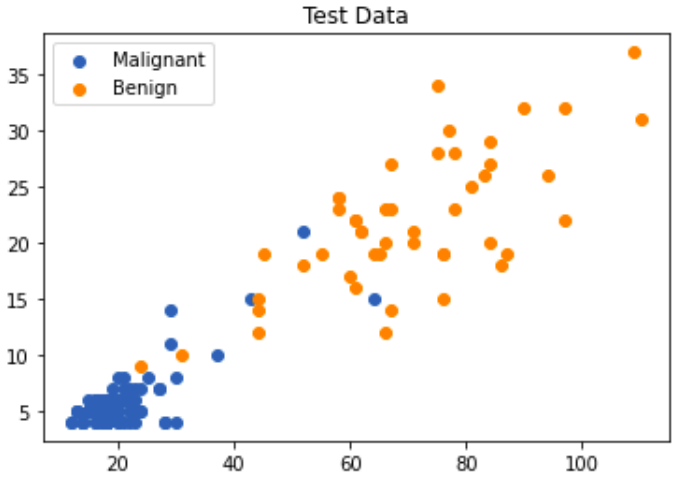

FIGURE 1 – Classification des données de test (*BREAST-cancer-wisconsin.data*)

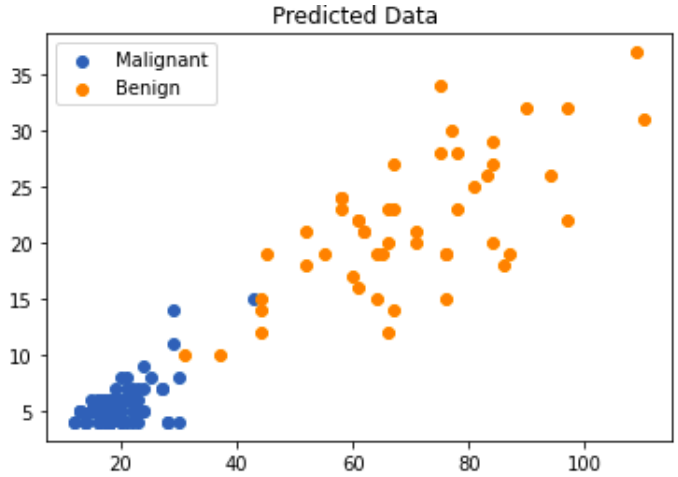

FIGURE 2 – Classification des données de prédiction (*BREAST-cancer-wisconsin.data*)

Dans le cas de la deuxième base de données, la moyenne est égale à 1.11 et la valeur

de MSE est égale à 0.2. La variance du modèle atteint ainsi 18%. Cette valeur est affirmée par ces deux figures :

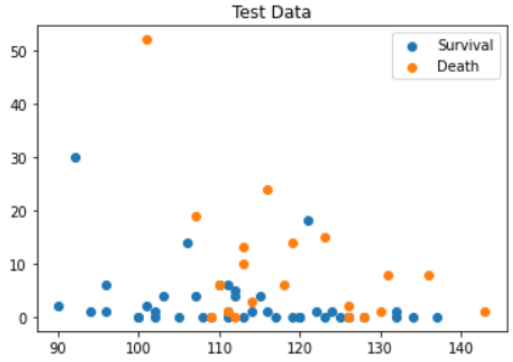

FIGURE 3 – Classification des données de test (*haberman.data*)

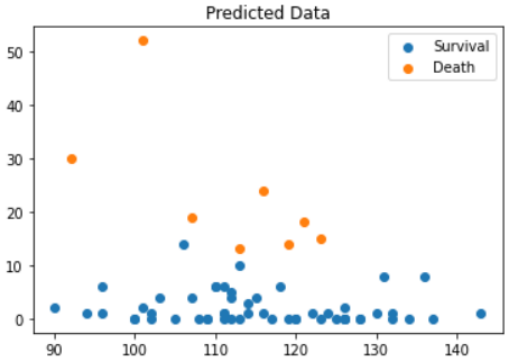

FIGURE 4 – Classification des données de prédiction (*haberman.data*)

Matrice de confusion

Avant d'interpréter le résultat de la matrice de confusion, il faut bien comprendre les quatre terminologies principales : TP, TN, FP et FN.

Voici la définition précise de chacun de ces termes :

— TP (True Positives) : les cas où la prédiction est positive, et où la valeur réelle est effectivement positive.

— TN (True Negatives) : les cas où la prédiction est négative, et où la valeur réelle est effectivement négative.

— FP (False Positive) : les cas où la prédiction est positive, mais où la valeur réelle est négative.

— FN (False Negative) : les cas où la prédiction est négative, mais où la valeur réelle est positive.

Dans le cas de la première base de données :

— TP = 79

— TN = 54

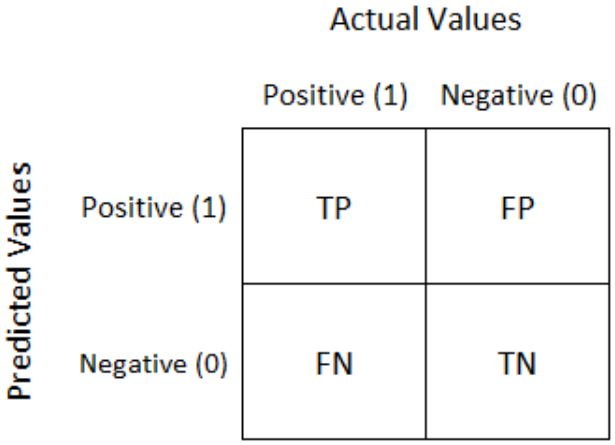

FIGURE 5 – Matrice de confusion

— FP = 5

— FN = 2

On remarque que la précision est égale 0.95. Notre prédiction donne de bonne résultat.

Dans le cas de la deuxième base de données :

— TP = 44

— TN = 5

— FP = 2

— FN = 11

La précision est égale 0.79. On doit choisir un meilleur nombre de voisin. (On a testé l'algorithme avec k=5 alors que la valeur optimale est pour k=3).

Conclusion

L'algorithme des k-plus proches voisins (KNN) est un algorithme d'apprentissage automatique simple et supervisé qui peut être utilisé pour résoudre des problèmes de classification et de régression. Il est facile à mettre en œuvre et à comprendre, mais cet algorithme nécessite beaucoup de calcul et demeure lent si la taille des données utilisées augmente.

Cet algorithme fonctionne en recherchant les distances entre une instance et tous les exemples dans les données, en sélectionnant le nombre de voisins les plus proches de l'instance.

Cette méthode présente un inconvénient qui se manifeste dans le choix du meilleur nombre k qui donne une précision optimale. Il faut donc chercher cette valeur ce qui nécessite un calcul coûteux surtout si on travaille avec des bases de données volumineuses.