Breast Cancer Classification With Machine Learning Algorithms

Directed by :
Younes Saouabeddine

Supervised by:

Aziz Khamjane

Table des matières

Résumé

L'idée principale du projet était de construire un système de classification du cancer du sein en utilisant des techniques d'apprentissage automatique. Le dataset utilisé provenait de Kaggle et contenait des informations médicales pertinentes pour la prédiction du cancer du sein.

Introduction

Le cancer du sein est un problème de santé majeur qui touche des millions de personnes dans le monde. Un diagnostic précoce et précis du cancer du sein est crucial pour une intervention rapide et des résultats améliorés pour les patients. Cependant, le processus de diagnostic du cancer du sein basé sur des images médicales et des données patient peut être complexe et difficile pour les professionnels de la santé.

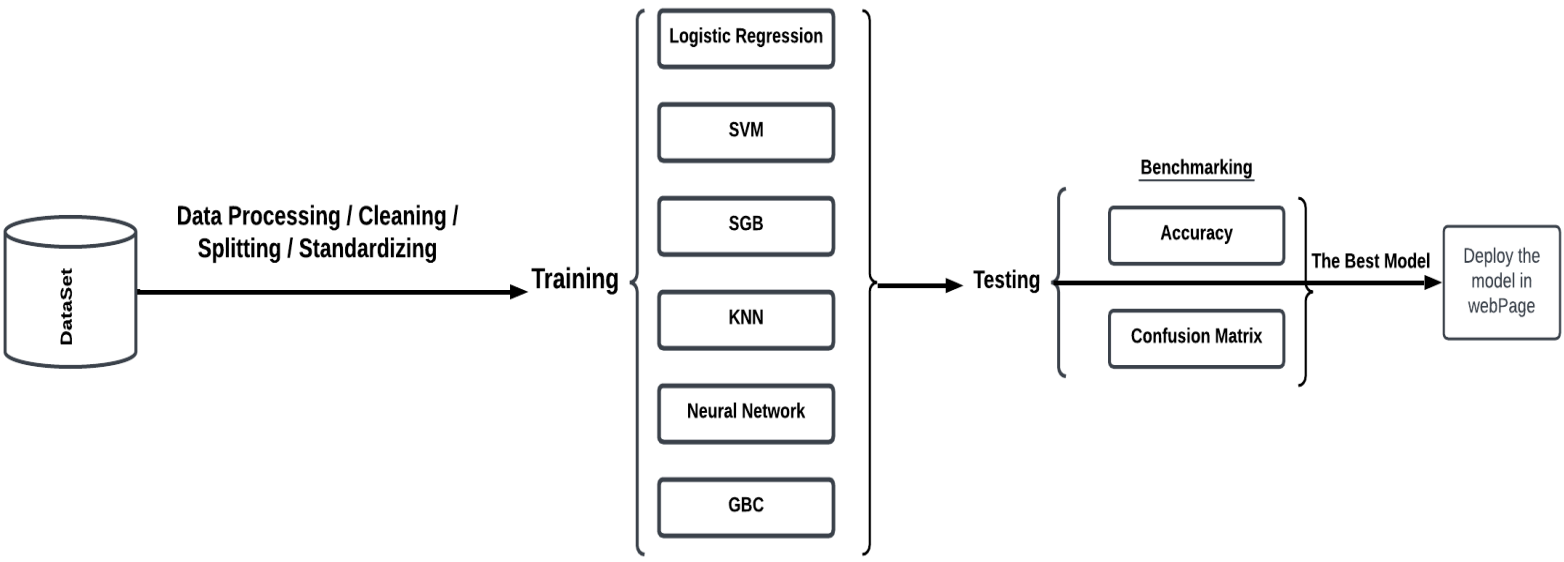
Dans ce rapport, nous abordons le problème du diagnostic du cancer du sein en utilisant des techniques d'apprentissage automatique. Notre objectif est de développer un modèle de classification fiable et précis capable d'aider les professionnels de la santé à identifier et à classer les cas de cancer du sein en fonction de données médicales pertinentes. En automatisant certaines parties du processus de diagnostic, nous visons à améliorer l'efficacité et la précision du diagnostic du cancer du sein, ce qui se traduirait par des soins et des résultats améliorés pour les patients.

À travers ce projet, nous cherchons à contribuer au domaine de la santé en démontrant le potentiel de l'apprentissage automatique pour aider les professionnels de la santé dans des tâches complexes de prise de décision. Notre rapport décrit les étapes suivies pour construire et évaluer le modèle de classification, ainsi que les conclusions et les implications de notre travail pour le diagnostic et le traitement du cancer du sein.

I. Nettoyage des données et Caractéristiques

Dans le cadre du projet de classification du cancer du sein, nous avons utilisé une Data Set qui contienne 32 colonnes, représentant diverses caractéristiques ou attributs liés aux données sur le cancer du sein, pui nous avons standardisé les caractéristiques dans le cadre de l'étape de prétraitement des données. La standardisation implique de transformer les données de telle sorte qu'elles aient une moyenne de 0 et un écart type de 1, ce qui peut être important pour certains algorithmes d'apprentissage automatique, notamment ceux sensibles à l'échelle des caractéristiques d'entrée. La standardisation des caractéristiques garantit qu'elles ont une échelle similaire et peut empêcher certaines caractéristiques de dominer le processus d'entraînement du modèle en raison de leur grande magnitude. Cette étape de prétraitement a probablement contribué à la stabilité et à l'efficacité de nos modèles d'apprentissage automatique.

II. L’architecture du projet



1. Architecture

Le processus de notre projet impliquait plusieurs étapes, notamment l'exploration et la préparation des données, la sélection et l'entraînement de différents modèles d'apprentissage automatique tels que la régression logistique, les k plus proches voisins (KNN), le SVM (SVC), les classificateurs de Gradient Boosting, le Gradient Boosting Stochastique et les réseaux neuronaux (NN). Une fois les modèles entraînés, leur performance a été évaluée à l'aide de mesures telles que la précision, le rappel, le F-score et la matrice de confusion.

L'objectif final était de sélectionner le modèle offrant les meilleures performances de prédiction pour le cancer du sein, en vue de son déploiement dans une application web utilisant Flask. Ce projet avait pour but d'améliorer la capacité de diagnostic du cancer du sein en fournissant un outil précis et fiable basé sur l'intelligence artificielle.

2. Les modèles utilisés

2.1 Régression Logistique :

La régression logistique utilise une fonction sigmoïde pour mapper les prédictions et leurs probabilités. Cette fonction convertit toute valeur réelle en une plage entre 0 et 1, créant ainsi une courbe en forme de S. Si la sortie de la fonction sigmoïde (probabilité estimée) est supérieure à un seuil prédéfini, le modèle prédit que l'instance appartient à une certaine classe ; sinon, il prédit qu'elle n'appartient pas à cette classe. Par exemple, si la sortie est supérieure à 0,5, elle est classée comme 1, et si elle est inférieure à 0,5, elle est classée comme 0. La sortie de la fonction sigmoïde représente la probabilité qu'un événement se produise, comme un lancer de pièce où une valeur de 0,65 indique une probabilité de 65 % que l'événement se produise. La fonction sigmoïde est la fonction d'activation de la régression logistique et est définie comme :

$$f(x) = \frac{1}{1 + e^{-x}}$$

L'équation suivante représente la régression logistique :

$$y = \frac{e^{(b_0 + b_1 X)}}{1 + e^{(b_0 + b_1 X)}}$$

Logistic Regression – Sigmoid Function

2.2 K-Nearest Neighbors (KNN) :

Un algorithme simple et intuitif utilisé pour les tâches de classification et de régression. Dans KNN, la prédiction pour une nouvelle instance est basée sur la classe majoritaire de ses k plus proches voisins dans l'espace des caractéristiques. KNN ne fait aucune hypothèse sur la distribution sous-jacente des données et peut capturer des frontières de décision complexes. Cependant, il peut être coûteux en termes de calcul, surtout avec de grands ensembles de données, car il nécessite le calcul des distances entre la nouvelle instance et toutes les instances existantes dans l'ensemble d'entraînement.

2.3 SVC (Support Vector Classifier):

SVC est une implémentation spécifique de l'algorithme des machines à vecteurs de support (SVM) conçue spécifiquement pour les tâches de classification. En d'autres termes, SVC est un SVM utilisé pour la classification. Il cherche à trouver l'hyperplan qui sépare au mieux les points de données en différentes classes. Les termes "SVC" et "SVM" sont parfois utilisés de manière interchangeable, mais lorsque quelqu'un fait référence à un "SVC", il fait généralement référence à la variante de l'algorithme utilisée pour la classification.

2.4 Classificateur par Gradient Boosting :

Le Gradient Boosting est une technique d'apprentissage ensembliste qui combine plusieurs faibles apprenants (souvent des arbres de décision) pour créer un modèle prédictif fort. Dans le gradient boosting, chaque nouveau modèle est entraîné pour corriger les erreurs des modèles précédents, en se concentrant sur les instances mal classées ou ayant de grandes résidus. Les classificateurs par gradient boosting sont connus pour leur grande précision prédictive et sont souvent utilisés dans les compétitions de machine learning. Cependant, ils peuvent être intensifs en calcul et nécessitent un réglage minutieux des hyperparamètres pour éviter le surajustement.

2.5 Boosting Stochastique :

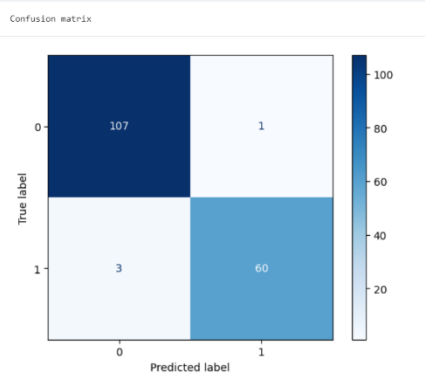
Le Boosting Stochastique est une extension du boosting qui introduit de l'aléatoire dans le processus d'entraînement du modèle. Au lieu d'utiliser l'ensemble de données d'entraînement complet pour chaque nouveau modèle, le boosting stochastique échantillonne un sous-ensemble des données (avec remplacement) pour chaque itération du modèle. Cette aléatoire peut conduire à des temps d'entraînement plus rapides et à une meilleure performance de généralisation, surtout dans les cas où l'ensemble de données est grand ou sujet au surajustement. Le boosting stochastique est particulièrement utile lorsqu'il s'agit de grands ensembles de données ou de données avec un grand nombre de caractéristiques.

2.6 Réseau de Neurones :

Les Réseaux de Neurones sont une classe de modèles inspirés par la structure et la fonction du cerveau humain. Ils sont constitués de nœuds interconnectés (neurones) organisés en couches, comprenant une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. Les réseaux de neurones sont capables d'apprendre des motifs complexes dans les données et sont largement utilisés pour des tâches telles que la reconnaissance d'images, le traitement du langage naturel, etc. Ils sont très flexibles et peuvent capturer des relations complexes dans les données. Cependant, l'entraînement des réseaux de neurones peut être coûteux en termes de calcul, surtout pour les grands réseaux ou ensembles de données, et ils nécessitent un réglage minutieux de l'architecture et des hyperparamètres pour obtenir des performances optimales.

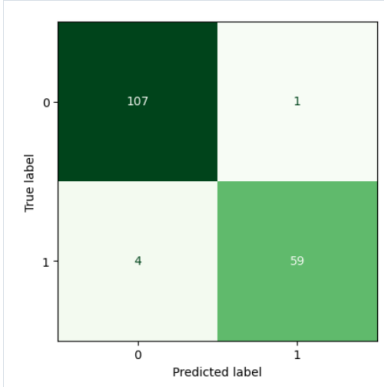
3. Benchmark

Logistic Regression



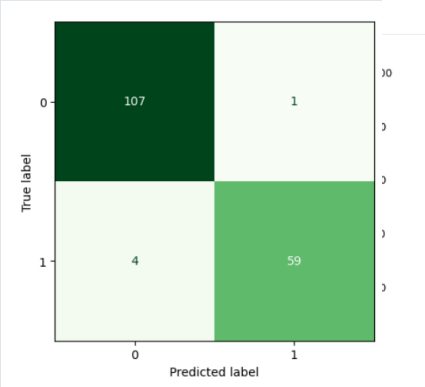
0.9766 accuracy

GBC



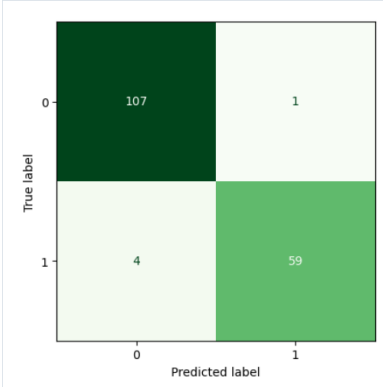
0.97 accuracy

SGB



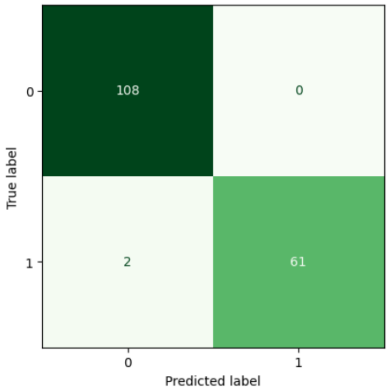
0.97 accuracy

KNN



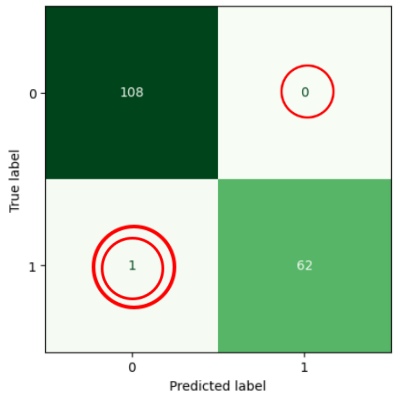
0.97 accuracy

SVC



0.988 accuracy

NN



0.9942 accuracy

Dans le cadre de l’analyse de la matrice de confusion j'ai constaté que le réseau de neurones présentait des performances remarquables, avec une précision, une sensibilité et une spécificité élevées. La précision indique la proportion de prédictions correctes parmi l'ensemble des prédictions du modèle, tandis que la sensibilité

mesure sa capacité à détecter correctement les cas de cancer du sein réels. D'autre part, la spécificité évalue sa capacité à identifier correctement les cas négatifs, c'est-à-dire ceux sans cancer du sein réel.

Ces résultats témoignent de la robustesse du réseau de neurones dans la classification du cancer du sein, mais ce qui est particulièrement intéressant, c'est que dans la matrice de confusion du réseau de neurones, le nombre de faux négatifs est faible. Les faux négatifs représentent les cas où le modèle prédit à tort qu'un patient n'a pas de cancer du sein alors qu'il en a effectivement. Avoir un faible nombre de faux négatifs est crucial dans le contexte du cancer du sein, car cela signifie que le modèle parvient à identifier efficacement la plupart des cas positifs.

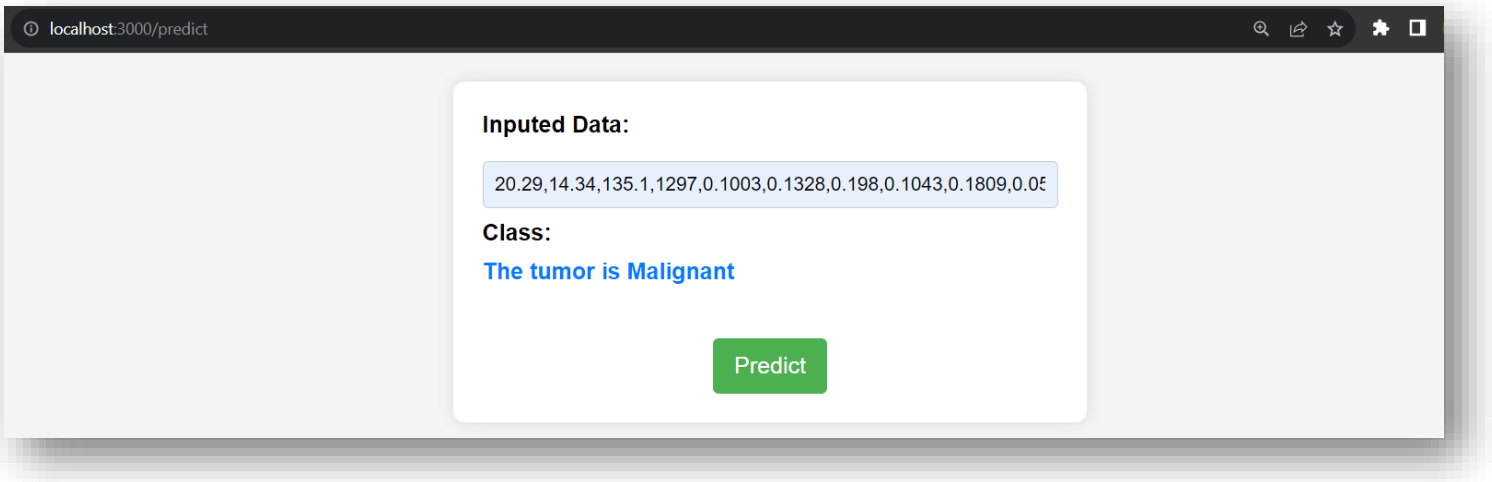
4. Déploiement de modèle

Pour le déploiement de mon modèle de réseau de neurones pour la classification du cancer du sein, j'ai utilisé Python avec les modules pickle pour la sérialisation et la désérialisation du modèle, ainsi que Flask pour créer une application web. J'ai également intégré HTML et CSS pour la partie frontend de l'application, offrant ainsi une interface conviviale aux utilisateurs pour interagir avec le modèle.

4.1 Démarrage

Pour exécuter le modèle de réseau de neurones que j'ai déployé pour la classification du cancer du sein, vous devez d'abord vous rendre à l'emplacement du projet dans le terminal, puis exécuter la commande suivante :

Cette commande spécifique démarre l'application Flask sur le port 3000, vous pourrez accéder à l'application à l'adresse http://localhost:3000 dans votre navigateur web une fois l'application démarrée.



5. Webographie

https://www.linkedin.com/pulse/svc-support-vector-classifier-dishant-salunke#:~:text=SVC%20(Support%20Vector%20Classifier)%3A%20SVC%20is%20a%20specific%20implementation,data%20points%20into%20different%20classes.

https://dl.acm.org/doi/abs/10.1145/3184066.3184080

https://link.springer.com/chapter/10.1007/978-981-15-7205-0_10

https://www.mdpi.com/2411-9660/2/2/13

https://pitch.com/v/Breast-Cancer-wfvft9