UNIVERSITÉ DE NANTES

IAE NANTES
ÉCONOMIE & MANAGEMENT

BIG DATA SOUS PYTHON

SVM et Réseaux de neurones

Naoufali Madi et Nibogora Darlene

Projet : Projet sur titanic

L'objectif de ce projet est de prédire si les voyageurs à bord du titanic ont survécu ou pas lors du naufrage

Nous allons dans un premier explorer les données pour connaitre les caractéristiques de nos données et dans un deuxième temps nous allons faire des prévisions à partir des deux modèles estimés par deux algorithmes de machine learning SVM et ANN sur la base de données d'apprentissage et enfin vérifier la performance des modèles en les appliquant sur une base de données de test.

Mais avant d'entrer dans le vif du sujet, nous débutons par une présentation de nos deux méthodes que nous allons utiliser dans ce travail d'étude.

Ces deux algorithmes sont appliqués pour résoudre les problèmes de classification et de régressions des fonctions linéaires et non linéaires.

1. SVM – Support Vector Machines

Cet algorithme consiste à chercher de frontières pour séparer les données en deux classes. Au cours de ce processus, une seule partie des échantillons d'étalonnage est utilisée réellement : ce sont des vecteurs supports qui délimitent les frontières.
« Les données sont transformées dans un nouvel espace, appelé noyau (kernel), qui permet de modéliser la non-linéarité. En étalonnage, cette matrice est de dimension NxN. Le noyau le plus courant est le noyau gaussien qui nécessite un paramètre d'optimisation de la largeur de la gaussienne (sigma) qui permet d'ajuster le degré de linéarité ».

Dans l'algorithme SVM, l'optimisation d'un paramètre de régularisation est très importante afin d'éviter le problème de sur-apprentissage.

Pour obtenir un modèle à la fois performant et robuste, le bon réglage de ces deux paramètres est très important.

Bien qu'au début de leurs créations, les SVM ont été introduit dans la résolution des problèmes de classification, ils ont été étendus à la régression.

https://agritrop.cirad.fr/547149/1/document_547149.pdf

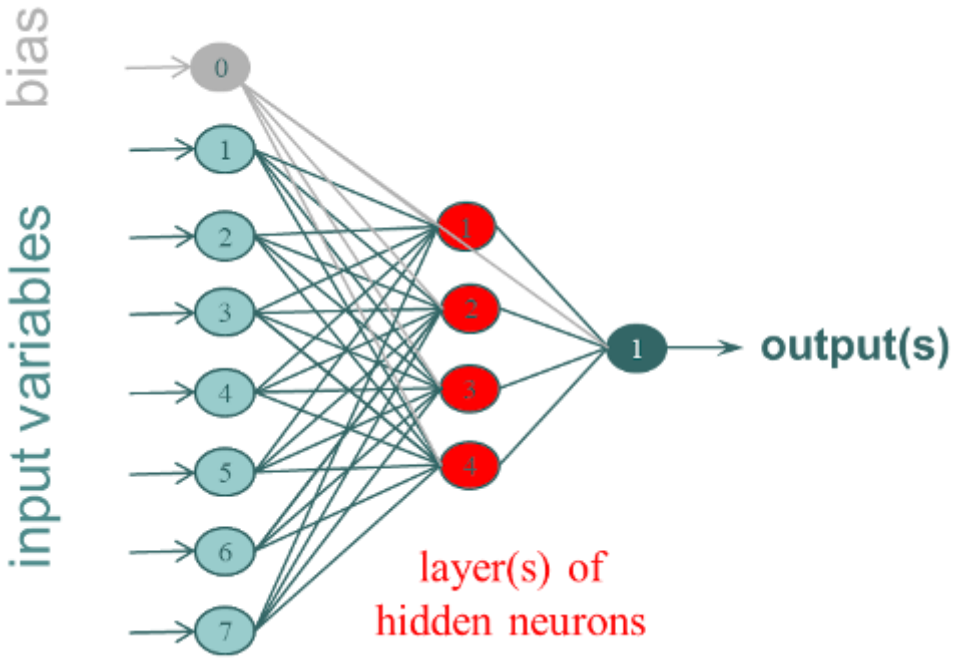
https://ondalys.fr/ressources-scientifiques/methodes-de-machine-learning/#MachineLearning

2. ANN (Artificial Neural Network)

ANN(Artificial Neural Network) également appelé réseau de Neurones Artificiels correspond à l'ensemble de méthodes de modélisation basées sur des modèles mathématiques simples du cerveau(neurones).

Le réseau de Neurones Artificiels le plus utilisé est le Multi-Layer Perceptron (MLP). Il est composé de couches de neurones connectés entre eux, avec au minimum 3 couches :

- 1 couche d'entrée qui correspond aux variables d'entrées (les inputs et le biais) :1 neurone par colonne.
- 1 ou plusieurs couche(s) cachée(s) de k neurones qui correspondent aux poids qu'il faudra entraîner pour réaliser le modèle.
- 1 couche de sortie qui correspond aux valeurs de sortie(output) : 1 neurone par colonne.



Les outputs peuvent être des valeurs quantitatives à prédire ou des classes selon le type de réseau de neurones développés. Les ANN correspondent aux « méthodes non-linéaires stochastiques », cela signifie que tous les processus de modélisations

https://agritrop.cirad.fr/547149/1/document_547149.pdf

https://ondalys.fr/ressources-scientifiques/methodes-de-machine-learning/#MachineLearning

fournissent des résultats différents, d'où il est plus important d'effectuer plusieurs itérations.

Les fonctions d'activation à chaque sortie d'un neurone de la couche cachée sont utilisées afin de gérer les problèmes de non-linéarités. Ces fonctions d'activation sont de natures différentes : tangente, sigmoïde, etc.

L'ajustement des poids est effectué en parcourant chaque échantillon de la base d'étalonnage plusieurs fois. Pour éviter un problème de sur-apprentissage, un critère d'arrêt s'avère nécessaire.

Il faut donc utiliser ces méthodes avec précaution, mais il existe des astuces de modélisations qui permettent d'estimer des modèles robustes.

Analyse statistique :

Dans un premier temps nous avons fait une analyse statistique afin d'avoir une aperçus de la composition de notre base données .

De plus nous avons au préalable découper notre base en deux. Une base pour entrainer notre jeu de données (80% de la base de données) et une autre pour tester les modèles pour vérifier sa robustesse(20% de la base de données).

La base d'entrainement est composée de 891 individus et de 12 variables et la base de test est quant à elle composée de 418 individus et de 11 variables (demander pourquoi 11 variables).

Par ailleurs nous avons constaté que notre base de données comportaient des valeurs manquantes (NaN) pour les variables Age, Sexe, et Pclass, Fare nous avons ainsi décidé de remplacer ces valeurs par leur médianes.

De plus la variable qualitative Cabin a été transformé en variable binaire 0 et 1.

Modèles SVM :

https://agritrop.cirad.fr/547149/1/document_547149.pdf

https://ondalys.fr/ressources-scientifiques/methodes-de-machine-learning/#MachineLearning

Concernant la méthode de classification SVM, nous avons dans un premier temps effectué un modèle de base sur notre jeu de test qui nous a donné des résultats peu satisfaisants.

Comme on peut le voir ci-dessous sur les deux matrices de confusions les deux modèles classes assez bien les True Positive mais ce n'est pas le cas pour les False Positive.

1) Modèle de base

Matrice de confusion avec toutes les variables	
True Positive = 117	**True Negative = 11**
False Negative = 69	**False Positive = 26**

2) Modèle de base avec variables Pclass, male, Age, Fare, Cabin

Matrice de confusion avec quelques variables	
True Positive = 112	**True Negative = 16**
False Négative = 67	**False Positive = 28**

Étant donné que nous faisons face à un problème linaire c'est à dire que notre échantillon n'est pas linéairement séparable nous avons donc eu recours à une fonction kernel (Noyau) qui correspond au produit scalaire du nouvelle espace.

Ainsi deux noyau ont été utilisé, un noyau linéaire et polynomiale.

Les modèles avec les deux noyau ont donnés des résultats peu fameux, nous avons décidé de retenir le modèle qui nous semblait être le moins pire des deux qui est le modèle polynomiale.

3) Modèle SVM polynomiale

Matrice de confusion avec toutes les variables	
True Positive = 115	**True Negative = 13**
False Negative = 85	**False Positive = 10**

https://agritrop.cirad.fr/547149/1/document_547149.pdf

https://ondalys.fr/ressources-scientifiques/methodes-de-machine-learning/#MachineLearning

Modèle Réseaux de neurones :

Le réseaux de neurones quant à lui nous a donné un résultats assez satisfaisant, en effet la matrice de confusion du modèle place assez bien les individu dans la partie True Positive et la partie False Positive.

1) Modèle réseaux de neurones

Matrice de confusion avec toutes les variables	
True Positive = 112	**True Négative = 16**
False Négative = 31	**False Positive = 64**

De plus le taux de prédiction sr l'échantillon d'entrainement et l'échantillon test sont très proches, en effet on a respectivement 0.821 pour le train et 0.789 pour le test.

Et à l'aide de la cross validation nous avons pu voir que l'on avait un taux de prédiction moyen de 0.719.

https://agritrop.cirad.fr/547149/1/document_547149.pdf

https://ondalys.fr/ressources-scientifiques/methodes-de-machine-learning/#MachineLearning