Moore Tom
Spaci Nikola

Compte Rendu Machine Learning

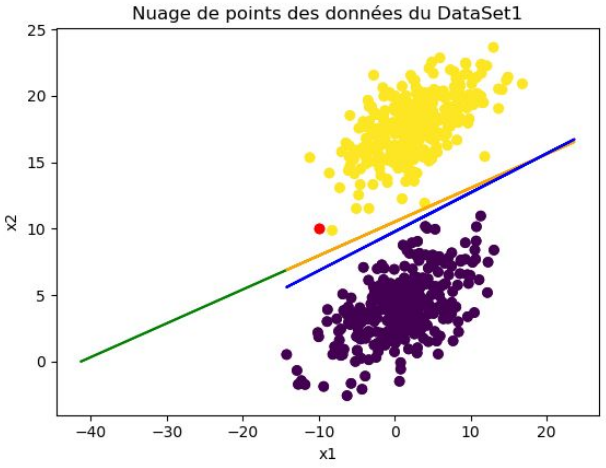
Partie 1 :

3)

```
#######DATASET1#######
S0 de notre ADL pour Xtest:  [[9.98843207]]
S1 de notre ADL pour Xtest:  [[17.01412777]]
Notre ADL prédit la classe 1
```

Afin de déterminer la classification du point Xtest nous avons calculé la règle d'affectation pour la classe 0 et la classe 1. Puis pour nous avons appliqué le théorème de bayes (arg max P(Y =k|X = x) avec k={0,1}). Ainsi notre classifieur prédit la classe 1 à Xtest.

4)



Voici la légende pour l'ensembles des graphiques du dossier :
Droite verte : Frontière de décision de l'ADL codée
Droite range: Frontière de décision de l'ADL de Sklearn
Droite Bleue : Frontière de décision de la régression logistique de SKlearn
Points violets : Observations de la classe 0
Points jaunes : Observations de la classe 1
Point rouge : Point Xtest à classifier

Nous obtenons la même frontière de décision pour le classifieur par ADL que nous avons codé et le classifieur par ADL de sklearn. Ainsi la classification des observations sera identique pour ces deux classifieurs. Sous la droite nous retrouvons toutes les observations de classe 0 et au dessus de la droite toutes les observations de la classe 1. Il n'y a aucune erreur de classification.
La frontière de décision de la régression logistique est quant à elle différente mais cela n'impact pas la classification des observations de ce jeu de données.

```
Lda de sklearn prédit la classe :  [1.]  pour Xtest
Classification logistique de sklearn prédit la classe :  [1.]  pour Xtest
```

La prédiction du point Xtest à l'aide des modèle de Sklearn est en phase avec la prédiction du modèle ADL que nous avons réalisé.

5)

```
Accuracy de ADL codée 100.0 %
Accuracy de ADL de Sklearn :  100.0 %
Accuracy de la classification logistique de SkLearn :  100.0 %
```

Ces résultats confirment les dires de la question précédente. En effet une accuracy de 100% indique que lors de la réalisation de la validation croisée, le classifieur a bien qualifié l'ensembles des données. Chacun des classifieurs ayant une accuracy de 100%, on peut dire qu'ils ont les mêmes performances sur ce jeu de données.

Partie 2 :

DATASET 2 :

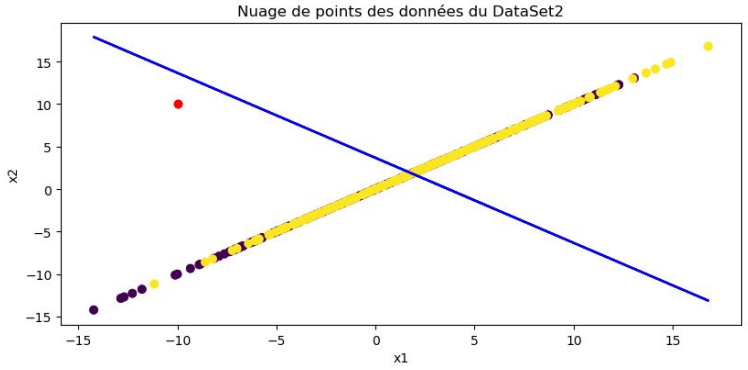
2.3)

Ici la matrice de covariance des variables a pour déterminant 0. Il s'agit d'une matrice singulière c'est-à-dire qu'elle n'est pas inversible. A première vue il n'est donc pas possible d'implémenter la frontière de décision et la règle d'affectation. Cependant ce problème peut être palier en calculant la pseudo inverse à l'aide de la fonction pinv() de numpy. Cette fonction retourne l'inverse de la matrice lorsqu'elle est disponible et la pseudo inverse quand elle ne l'est pas. Ainsi voici la prédiction pour le point Xtest.

```
#######DATASET2#######
S0 de notre ADL pour Xtest:  [[-0.70918465]]
S1 de notre ADL pour Xtest:  [[-0.85697341]]
Notre ADL prédit la classe 0
```

La valeur obtenue après le calcul de la règle d'affectation pour Xtest au sein de la classe 1 donne un résultat inférieur à la valeur obtenue après le calcul de la règle d'affectation pour Xtest dans la classe 0. D'après la règle de classification de bayes on prédit bien la classe 0 au point Xtest.

2.4)




```
Lda de sklearn prédit la classe :  [0.]  pour Xtest
Classification logistique de sklearn prédit la classe :  [0.]  pour Xtest
```

Nous sommes face à des variables colinéaires, avec une relation linéaire positive forte entre les variables on en déduit que les variables sont fortement corrélées.

Les 3 classifieurs étudiés ont la même frontière de décision représentées sur le graphe par 3 droites superposées. La prédiction Xtest de chaque classifieur semble correct au vu de la représentation graphique réalisée. En effet le point Xtest se situe sous la droite représentant les frontière de décision, on est donc bien dans la classe 0. A première vue, on peut supposer qu'aucun des trois modèles est meilleur qu'un autre. Cette supposition, sera confirmée après la réalisation d'une validation croisée.

2.5)

```
Accuracy de ADL codée 57.666666666666664 %

Accuracy de ADL de Sklearn :  57.666666666666664 %
Accuracy de la classification logistique de SkLearn :  57.666666666666664 %
```

La colinéarité des variables impacte les performances des classifieurs faisant ainsi diminuer fortement le taux de bonnes classifications des observations. Pour ce jeu de données les trois classifieurs ont un taux de bonnes classifications identiques : 57,66 %. Cela signifie que seulement 57,66% des observations sont bien classifiées par le modèle. On a un nombre de faux positifs et faux négatifs conséquents, ainsi le rappel négatif et positif des classifieurs ne sont pas optimal. Pour ce jeu de donnés, aucun des trois modèles n'est meilleur que les deux autres.

Ces 57,66% se rapprochent des 50% du hasard. En effet, on constate visuellement que les données des deux classes sont superposées. Il n'y a pas deux groupes distinct. Dans cette configuration la frontière de décision est difficilement identifiable et il y a quasiment autant de point de chaque classe de part et d'autres de la frontière de décision. Le classifieur a donc plus ou moins une chance/risque sur deux de prédire la bonne/mauvaise classe.

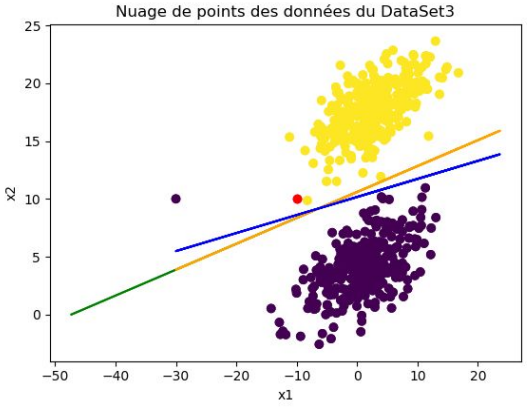
DATASET 3 :

2.3)

```
#######DATASET3#######
S0 de notre ADL pour Xtest:  [[9.05856804]]
S1 de notre ADL pour Xtest:  [[14.30349195]]
Notre ADL prédit la classe 1
```

Après avoir calculé la règle de décision pour Xtest pour la classe 0 et la classe 1. Nous avons appliqué le théorème de Bayes qui permet de prédire la classe 1 pour Xtest.

2.4)



Ici encore nous avons la même frontière décision pour le classifieur par ADL que nous avons codé et le classifieur par ADL de sklearn. De ce fait la classification des observations du jeu de données sont identiques pour les deux classifieurs. Quant à la frontière de décision de la régression logistique, celle ci est différente mais n'impact pas la prédiction des observations du jeu de données.

A l'aide du graphique nous constatons qu'une observation n'est pas bien classifié par les classifieurs. En effet, une observation ayant initialement une classe 0 se voit être classifié

dans la classe 1. Cela est visible car le point (l'observation) se situe au dessus de la frontière de décision et non en dessous.

```
Lda de sklearn prédit la classe :  [1.]  pour Xtest
Classification logistique de sklearn prédit la classe :  [1.]  pour Xtest
```

Les trois classifieurs prédisent la classe 1 pour le point Xtest.

2.2.5)

```
Accuracy de ADL codée 99.83333333333333 %
Accuracy de ADL de Sklearn :  99.83333333333333 %
Accuracy de la classification logistique de SkLearn :  99.83333333333333 %
```

Le taux de bonne classification des trois classifieurs est de 99,83%. Ils sont donc tout aussi performant sur ce jeu de données. Comme constaté sur le graphique précédent une observation n'as pas été correctement classifié par les classifieurs. Nous avons un faux positif qui diminue donc le taux de bonne classification des classifieurs (Accuracy).

Partie 3 :

1) En cette periode de fete de fin d'années, les achats sur internet sont bien plus conséquents. À l'aide d'un jeu de données récupéré sur kaggle (https://www.kaggle.com/roshansharma/online-shoppers-intention) correspondant à l'analyse de plus de 120000 sessions de navigation d'utilisateurs différents sur l'internet. Nous allons classifier le comportement d'un consommateur en fonction de différentes variables pour savoir si oui ou non il finira par effectuer un achat. Cela permettra à l'entreprise d'adapter ses stratégies marketing sur le web en fonction du comportement de l'individu pour potentiellement le pousser à acheter (choix de pub plus ciblés...). En d'autres terme, si le comportement d'un utilisateur (défini par ces variables) est labellisé comme "futur acheteur", alors l'entreprise pourra intensifier son action marketing sur cet utilisateur afin de ne pas manquer une vente.

Le jeu de données contient 18 variables :

Administrative : Il s'agit du nombre de page en lien avec la thématique de l'administration parcouru par l'utilisateur.
Administrative_Duration : Temps total en seconde passé sur des pages administratives.
Informational : Il s'agit du nombre de pages web d'informations parcourues par l'utilisateur.
Informational_Duration : Temps total en seconde passé sur des pages d'informations.
ProductRelated : Il s'agit du nombre de pages web en lien avec un produit qui ont été parcouru par l'utilisateur.
ProductRelated_Duration : Temps total en seconde, passé sur des pages de produits.
BounceRates : Sur google Analytic, il s'agit d'un taux qui mesure le pourcentage d'internautes ayant accéder à une page du site puis quitté le site sans avoir consulté d'autres pages.
ExitRates : Sur google Analytic, il s'agit du taux d'internautes qui après avoir visités plusieurs pages du site en arrivant sur l'une d'entre, décident de quitter le site.
PageValues : La valeur moyenne (chiffre d'affaire) d'une page que l'utilisateur a visité avant d'effectuer une transaction.
SpecialDay : Indicateur entre 0 et 1 indiquant la proximité de la visite du visiteur en fonction d'un événement spécial tel que noël ou la fête des mères.
Month : Mois de l'année à laquelle la session de navigation sur le web a été observée.
OperatingSystems : Système d'exploitation utilisé par l'individu.
Browser : Navigateur internet utilisé par l'individu.
Region : Région du visiteur.

TrafficType : Type de trafic.
VisitorType : Type de visiteur c'est à dire s'il s'agit d'un nouveau visiteur ou bien d'un ancien visiteur venant consulter le site.
Weekend : Indication pour savoir si l'observation a été observé le weekend ou non.
Revenue : Indication pour savoir si la visite des individus a mené à un achat ou non.
Nous avons donc réalisé une classification binaire par rapport à la variable revenue qui prend soit la valeur "true" si l'utilisateur effectue un achat soit la valeur "false" dans le cas contraire.

2) Afin de réaliser la classification des intentions d'achat d'un utilisateur nous avons utilisé le modèle d'analyse discriminante linéaire précédemment implémenté. La classification étant binaire, la formule pour la règle de prédiction ainsi que la frontière de décision sont identiques. Cependant nous utilisons un véritable jeu de données, il y a donc potentiellement des données erronés ou absentes. Nous avons donc procédé à un nettoyage du jeu de données pour supprimer notamment les observations contenant des valeurs Null (NAN).
Par ailleurs certaines variables étant qualitatives tels que le mois, le type de visiteurs il faut alors effectuer un codage afin de les rendre manipulable pouvoir et calculer l'ensemble des paramètres pour appliquer l'ADL. Pour cela nous avons convertis les variables qualitatives en un ensemble de variables muettes. Grâce à cela nous évitons d'accorder un poids plus fort à certaines valeurs plutôt qu'à d'autres. Après cela, l'ensemble des paramètres de l'ADL ont pu être déterminées. Afin de s'assurer de l'efficacité de la classification du modèle nous avons réalisé une validation croisée 80-20. Nous en avons ressorti les métriques suivantes :

```
Nombre de True positif :  438
Nombre de True negatif :  10249
Nombre de False positif :  159
Nombre de False negatif :  1470
Accuracy : 86.77330302046119  %
TPR :  22.955974842767297  %
TNR :  98.47232897770945  %
```

Avec plus de 86% de données bien classées le classifieur semble convenir. Cependant ce taux de bonne classification cache un problème: nous constatons un taux de faux négatif assez élevé.
De ce fait le classifieur identifie correctement 22,9% des utilisateurs qui vont acheter.
Le nombre élevé de faux négatif est beaucoup plus "grave" que le nombre de faux positifs. Notre modèle de ciblage ne sera juste pas tout à fait parfait et donc la stratégie marketing sera impacté et moins performante. Un faux négatif représente une vente ratée et donc un manque à gagner beaucoup plus tangible. Ce taux élevé de faux négatif s'explique par un déséquilibre dans le jeu de données avec un nombre total d'observations dans lesquels les individus n'ont pas acheté, 5 fois plus grand que le nombre total d'observations dans lesquels les individus ont acheté (10408 observations contre 1908). Le nombre de vrai négatif est quant à lui très satisfaisant on identifie bien un utilisateur qui ne va pas acheter avec 98,4 % de bonnes identifications. Afin d'améliorer les performances du classifieur il serait sans doute nécessaire d'augmenter le nombre d'observations dans lequel l'utilisateur va acheter.