Examen « Algorithmes Data Mining II »

Exercice 1 (QCM) : (3pts)

1- Laquelle des métriques suivantes n'est pas adaptée à l'évaluation dans le cas des données fortement imbalancées ?
   a. Area Under the ROC curve
   b. F-measure
   c. Precision and Recall
   d. Accuracy

2- Parmi les affirmations suivantes relatives à la sélection de features (*Feature Selection)*, lesquelles sont vraies ?
   a. La sélection de features peut réduire considérablement le biais d'un modèle SVM
   b. La régression Ridge élimine fréquemment certaines features.
   c. La sélection de features peut réduire l'overfitting
   d. Trouver le meilleur sous-ensemble de features prend un temps exponentiel.

3- Quelles stratégies peuvent aider à réduire l'overfitting dans les arbres de décisions ?
   a. Pruning
   b. S'assurer que chaque nœud feuille est une classe pure
   c. Fixer un nombre minimum d'observations (samples) dans les nœuds feuilles.
   d. Fixer une profondeur maximale pour l'arbre

4- Parmi les affirmations suivantes, lesquelles peuvent aider à réduire l'overfitting d'un classifieur SVM ?
   a. Utilisation des variables resort (*Slack variables*)
   b. Normalization des données
   c. Des variables polynomiales de haut degré
   d. Fixer un taux d'apprentissage (*learning rate*) très bas

5- L'entrainement d'un modèle « Random Forest » pour la classification du spam donne une performance trop faible pour l'ensemble de validation mais une très bonne performance sur l'ensemble de train. Quelle pourrait être la cause du problème ?
   a. Les arbres de décision dans le modèle sont trop profonds
   b. Au choix d'un split, un très grand nombre de features est aléatoirement choisi
   c. L'ensemble contient un nombre d'arbre très petit
   d. Dans l'implémentation du bagging, les observations sont tirées aléatoirement sans remise

6- Parmi les affirmations suivantes relatives au bagging, lesquelles sont vraies ?
   a. Dans le cas du bagging, des sous échantillons aléatoires de données sont tirés avec remise
   b. Le bagging avec des modèles de régression logistique est inefficace car les modèles apprennent la même frontière de décision
   c. L'objectif principal du bagging est de diminuer le biais des algorithmes d'apprentissage
   d. Si des arbres de décisions avec une seule observation (sample) par nœud feuille sont utilisés, le bagging ne donnera jamais une erreur d'apprentissage (training error) inférieure à celle d'un arbre de décision ordinaire

Exercice 2 :

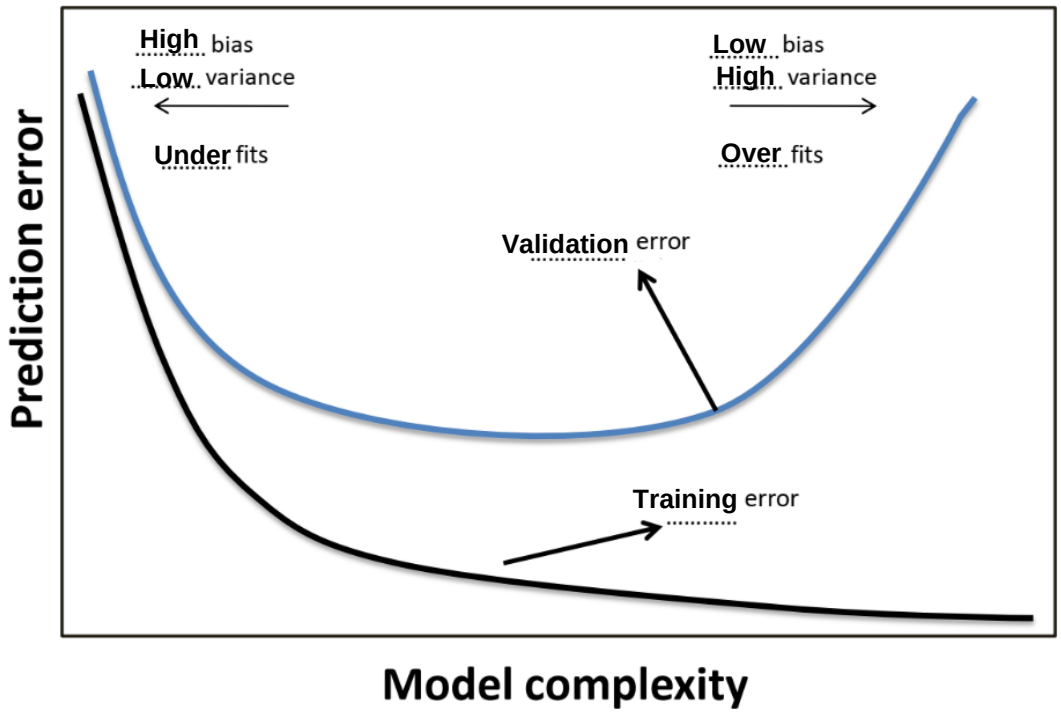
1- Décrire brièvement les étapes des deux méthodes de sélection de features : Forward Feature Selection, et Backward Feature Selection.
Voir cours

2- Les données manquantes sont un problème très fréquent quand on manipule des données réelles. Décrire les différentes méthodes permettant de traiter ce problème.

Voir cours

3- La figure ci-dessous représente les courbes des erreurs de train et de validation en fonction de la complexité du modèle. Remplissez la figure en indiquant :

- Quelles courbes représentent les « training error » et « validation error » ?
- Quelles parties du graphe correspondent au « high variance », « low variance », « high bias », « low bias »
- Quelle partie représente l'«overfitting» et quelle partie représente l'«underfitting »



4- La matrice de confusion ci-dessous représente le résultat d'évaluation d'un modèle pour la classification des spam :

Actual	Predicted		
	Spam	**Ad**	**Normal**
Spam	27	286	40
Ad	1	37	9
Normal	5	16	500

- Remplir le tableau (Classification report) suivant avec les valeurs des différentes métriques calculées
  Classe SPAM : TP=27, FP=6, FN=326
  Classe AD : TP=37, FP=302, FN=10
  Classe NORMAL: TP=500, FP=49, FN=21
  Voir cours pour les formules

	Precision	Recall	F1 score
Spam	27/(27+5+1)	27/(27+286+40)	
Ad	37/(37+286+16)	37/(37+1+9)	
Normal	500/(500+40+9)	500/(500+5+16)	
Micro Avg	(27+37+500) / (27+37+500+6+302+49)	(27+37+500) / (27+37+500+326+10+21)	2*Recall *Precision / (Recall+Precision)
Macro Avg	1/3*(precision(spam) + precision(ad) + precision(normal))	1/3*(recall(spam)+recall(ad) +recall(normal))	
Weighted avg	(353/921)*precision(spam) + (47/921)*precision(ad) + (521/921)*precision(normal)	(353/921)*recall(spam) + (47/921)*recall(ad) + (521/921)*recall(normal)	

Exercice 3 (Arbre de décision) :

Considérons la dataset (training dataset) ci-dessous pour la création d'un arbre de décision :

GPA	Studied	Passed
Low	No	No
Low	Yes	Yes
Medium	No	No
Medium	Yes	Yes
High	No	Yes
High	Yes	Yes

1) Calculer l'entropie des ensembles suivants :
   - E(Passed), E(Passed/GPA), E(Passed/Studied)

   $E(Passed)=-(1/3)*\log_2(1/3)-(2/3)*\log_2(2/3)=0.918$

   $E(Passed/GPA)=(1/3)*(E(Low)+E(Medium)+E(High))=2/3=0.66$

   $E(Low)=E(Medium)= -(1/2)*\log_2(1/2)-(1/2)*\log_2(1/2)=1$

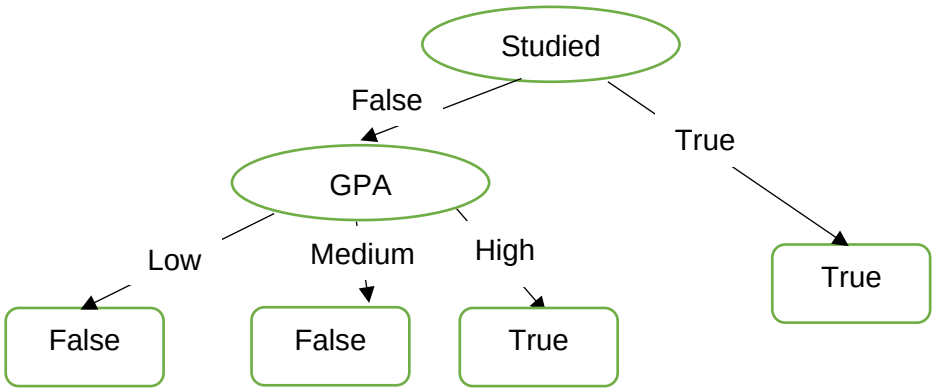
   $E(High)=-\log(1)=0$

   $E(Passed/Studied)=(1/2)*(E(Yes)+E(No))=0.459$

   $E(Yes) = 0$

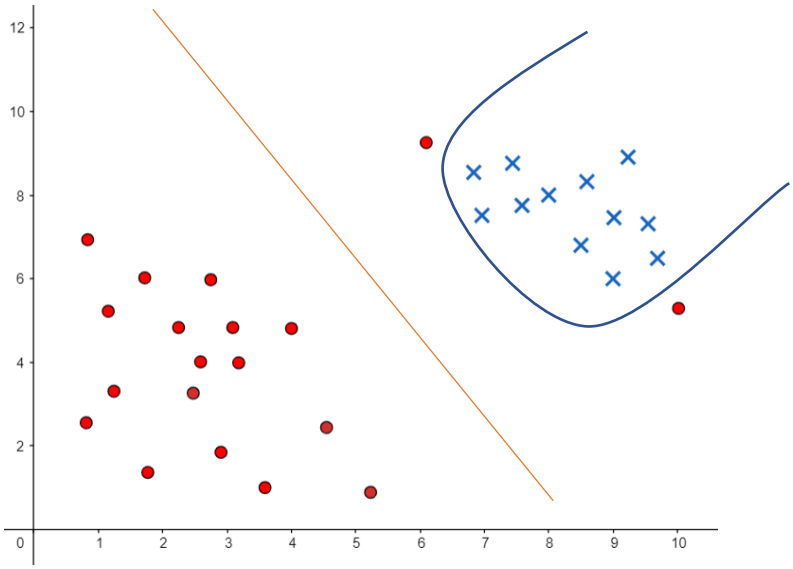
   $E(No) = -(1/3)*\log_2(1/3)-(2/3)*\log_2(2/3)=0.918$

2) Détaillez les différentes étapes de création de l'arbre de décision et donnez l'arbre résultat.



Exercice 4 (SVM) :

Nous utilisons la dataset ci-dessous pour l'entrainement d'un SVM. SVM est, à la base, un classifieur linéaire. Afin de l'adapter au cas non linéaire, la technique du kernel est utilisée. Pour cet exercice, nous utilisons un SVM avec un kernel quadratique (i.e. une fonction polynomiale de degré 2). Cela signifie que la frontière de décision peut avoir une forme parabolique. Le parabole séparant les classes est déterminé par la pénalité C (the slack variable).



1) Quelle sera la frontière de décision pour les grandes valeurs de C ? Justifier votre réponse puis dessiner la frontière de décision sur la figure ci-dessus.

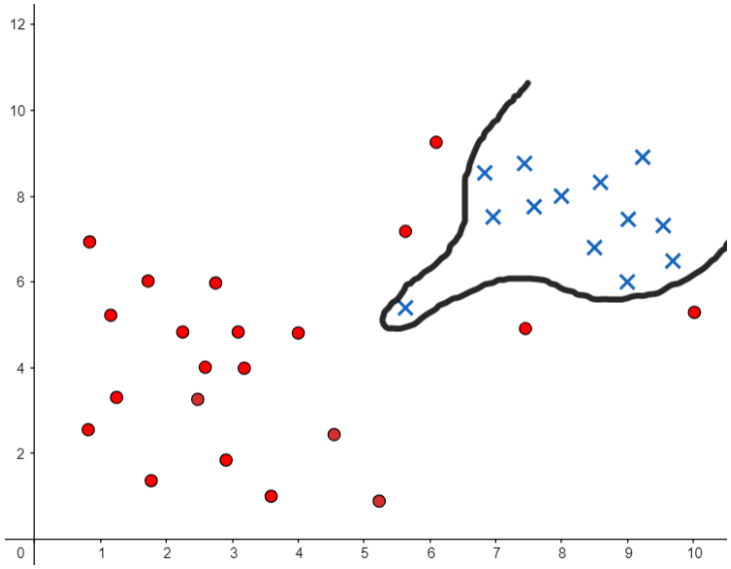
   Une valeur très grande de C signifie une grande pénalité des mauvaises classifications, et par conséquent une marge très petite

2) Quelle sera la frontière de décision pour C qui s'approche de 0 ? justifier votre réponse puis avec une autre couleur dessiner la frontière de décision sur la figure.

Une valeur très petite de C signifie une très petite pénalité des mauvaises classifications. La séparation sera linéaire ($x^2 = 0$)

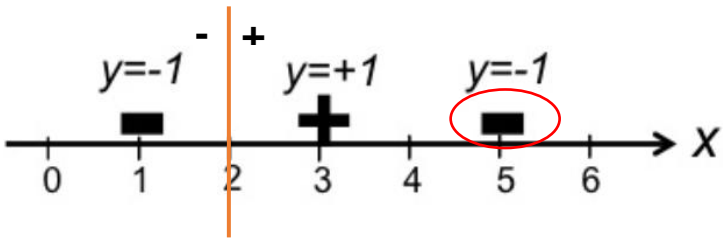
3) Supposant qu'on ajoute trois observations comme représenté dans la figure ci-dessous. Les données ne sont plus quadratiquement séparables, nous utilisons un kernel polynomial de degré 5. La séparation obtenue est représentée sur la figure ci-dessous. Le SVM obtenu souffrira très probablement d'un phénomène qui entrainera une mauvaise classification des nouvelles données. Expliquer ce phénomène.

Le modèle souffre de l'overfitting



Exercice 5 (Boosting) :

On souhaite étudier la performance d'un algorithme de boosting sur le simple problème de classification représenté dans la figure ci-dessous. Nous utilisons pour chaque apprenant faible (weak learner) le classifieur « Decision Stump ». Le « Decision Stump » est un simple arbre de décision très simple constitué d'un seul nœud (un arbre de profondeur 1). Ce classifier choisit une valeur constante c tels que : $y = \begin{cases} +1 \; si \; x > c \\ -1 \; si \; x \leq c \end{cases}$



1) Quel est le poids initial attribué à chaque observation ?

Le poids initial des observations est ($w_i^{(0)}$=1/3)

2) Dessiner sur la figure la frontière de décision du premier « decision stump », en indiquant le côté positif et le côté négatif de la frontière.

3) Entourer l'observation dont le poids augmente après la première itération du boosting puis donner le nouveau poids attribué à chaque observation.

Après la première itération :

Erreur du modèle r=1/3

Poids du modèle entrainé à la première itération $\alpha_1=\mu*\log(1-r/r)=\log(2)$ pour $\mu=1$

Le nouveau poids des observations :

Le poids de l'observation mal classée sera augmenté : $w_3^{(1)}= w_3^{(0)}*\exp(\alpha_1)$