

TP : Forêts aléatoires et bayésien naïf

1 Forêts aléatoires

Récupérer le jeu de données d'apprentissage habituel `synth_train.txt`. On a $Y \in \{1, 2\}$ et $X \in \mathbb{R}^2$. On dispose de 100 données d'apprentissage.

1. Charger le jeu de données dans R. Transformer la variable de sortie `y` en facteur.
2. Charger le package `randomForest` (ou l'installer si ce n'est pas déjà fait grâce à la commande : `install.packages("randomForest")`) et consulter l'aide de la fonction `randomForest`.

```
# install.packages("randomForest")
library(randomForest)

## randomForest 4.6-10
## Type rfNews() to see new features/changes/bug fixes.

help(randomForest)
```

3. Construire une forêts aléatoires, qu'on pourra appeler `rf`, à l'aide de la fonction `randomForest` (garder toutes les valeurs des paramètres par défaut pour l'instant). Faire afficher le résultat en tapant `rf`.
Que représente le "OOB estimate of error rate" ?
4. Calculer l'erreur d'apprentissage du prédicteur obtenu.
5. Charger le jeu de données test `synth_test.txt` puis calculer le taux d'erreur test.
Comparer ce taux d'erreur avec celui donné directement dans l'objet `rf`. Discuter.
6. Faire de même (construction de la forêt, taux d'erreurs d'apprentissage et test) en fixant le nombre de variables choisies aléatoirement à chaque noeud à 2 et le nombre d'arbres dans la forêt à 2000.
Avec ces valeurs de paramètres, de quelle méthode d'ensemble s'agit-il ? Comparer les taux d'erreur obtenus avec ceux de la forêt "par défaut" et commenter les résultats.
7. Re-faire une dernière fois la même chose en fixant maintenant le paramètre `maxnodes` à 4.
Quel est l'effet de ce paramètre sur les arbres de la forêt ? Comparer les taux d'erreur obtenus avec ceux de la forêt "par défaut" et commenter les résultats.

2 Bayésien naïf

On considère le jeu de données d'apprentissage habituel `synth_train.txt`. On a $Y \in \{1, 2\}$ et $X \in \mathbb{R}^2$ et on dispose de 100 données d'apprentissage. On note X^1 et X^2 les deux coordonnées de X .

On rappelle que le bayésien naïf est une approche générative où on fait l'hypothèse d'indépendance des variables d'entrée conditionnellement à Y :

$$\forall x \in \mathbb{R}^2, \forall k \in \{1, 2\} \quad f_k(x) = f_{k,1}(x^1)f_{k,2}(x^2)$$

où f_k est la densité conditionnelle de X sachant $\{Y = k\}$, et $f_{k,1}$ et $f_{k,2}$ sont les densités conditionnelles respectivement de X^1 et X^2 sachant $\{Y = k\}$.

De plus, on suppose que, pour tout $k \in \{1, 2\}$ et tout $j \in \{1, 2\}$ la loi de X^j sachant $\{Y = k\}$ est $\mathcal{N}(\mu_{k,j}, \sigma_{k,j}^2)$.

1. Donner les estimateurs du maximum de vraisemblance de tous les paramètres du bayésien naïf considéré.
2. Ecrire la règle de décision associée.
3. Charger le jeu de données dans R. Transformer la variable de sortie `y` en facteur.
4. Implémenter la méthode du bayésien naïf.
 - (a) On pourra commencer par créer une fonction `bn.estim` qui prend en argument les données et calcul les estimateurs des différents paramètres associés à la modélisation du bayésien naïf.
 - (b) Puis, on pourra écrire une fonction `bn.predict` permettant de prédire la classe associée à une observation x (cette fonction utilisera les paramètres estimés par la fonction précédente).
5. Tester les fonctions : appliquer la fonction `bn.estim` avec l'échantillon d'apprentissage, puis utiliser la fonction `bn.predict` pour prédire les points de coordonnées $(0, 1)$ et $(-2, 2)$.
6. Calculer le taux d'erreur d'apprentissage du bayésien naïf.
7. Charger le jeu de données test `synth_test.txt` puis calculer le taux d'erreur test du bayésien naïf.