### Projet de traitement de données massives

Équipe no 14

Maxime Mainardi (536942625)

Cédric Fontaine (536983535)

Kenza Bellebouir (537198197)

Techniques avancées en intelligence artificielle IFT-4102/IFT-7025

Travail présenté à Brahim Chaib-draa



Table de matière

[Techniques d’apprentissage automatique à développer 3](#_Toc185688392)

[1. Précision 3](#_Toc185688393)

[2. Rappel 3](#_Toc185688394)

[3. F1-Score 3](#_Toc185688395)

[4. Matrice de Confusion (Confusion Matrix) 3](#_Toc185688396)

[Arbre de décision 5](#_Toc185688397)

[1. Évaluation des performances sur l’ensemble de test 6](#_Toc185688398)

[2. Comparaison avec l'implémentation Scikit-learn 7](#_Toc185688399)

[1. Évaluation des performances sur l’ensemble de test 8](#_Toc185688400)

[2. Comparaison avec l'implémentation Scikit-learn 8](#_Toc185688401)

[Discussion 9](#_Toc185688402)

[Comparaison entre les quatres techniques sur l’ensemble de test des jeux de données 10](#_Toc185688403)

[Conclusion 11](#_Toc185688404)

[ANNEXE 12](#_Toc185688405)

# Techniques d’apprentissage automatique à développer

## Précision

La précision est une mesure de la qualité des prédictions positives. Elle représente la proportion des prédictions positives qui sont effectivement correctes. Par exemple, si un modèle prédit qu'un certain nombre d'instances appartiennent à une classe donnée, la précision correspond au pourcentage des prédictions correctes parmi celles-ci.

### Formule :

## Rappel

Vrai Positifs (VP)

𝑃𝑟é𝑐𝑖𝑠𝑖𝑜𝑛 = Vrai Positifs (VP) + Faux Positifs (FP)

Le rappel est une mesure de la capacité du modèle à détecter toutes les instances positives. Il représente la proportion des vrais positifs qui ont été correctement identifiés par rapport au nombre total de vrais positifs dans l'ensemble des données.

### Formule :

## F1-Score

Vrai Positifs (VP)

𝑅𝑎𝑝𝑝𝑒𝑙 = Vrai Positifs (VP) + Faux Négatifs (FN)

Le F1-score est une mesure qui combine la précision et le rappel. Il s'agit de la moyenne harmonique de ces deux métriques, et il est utile lorsque l'on souhaite équilibrer la capacité d'un modèle à identifier correctement les exemples positifs (rappel) et à minimiser les fausses alertes (précision).

### Formule :

𝐹1 − 𝑆𝑐𝑜𝑟𝑒 = 2 ×

Précision × Rappel Précision + Rappel

## Matrice de Confusion (Confusion Matrix)

La matrice de confusion est un tableau qui résume les performances d'un modèle de classification. Elle permet de voir comment les prédictions sont réparties entre les vraies classes et les classes prédites. Dans une classification binaire, elle contient quatre éléments :

* + **Vrai Positifs (VP)**: Nombre de prédictions positives correctes.
  + **Faux Positifs (FP)**: Nombre de prédictions positives incorrectes.
  + **Faux Négatifs (FN)**: Nombre de prédictions négatives incorrectes.
  + **Vrai Négatifs (VN)**: Nombre de prédictions négatives correctes. À partir de cette matrice, on peut extraire les différentes métriques :
  + **Exactitude** : Mesure de la proportion d'exemples bien classifiés sur le total.

VP + VN

𝐸𝑥𝑎𝑐𝑡𝑖𝑡𝑢𝑑𝑒 =

VP + FP + FN + VN

* + **Précision** et **Rappel** sont extraits respectivement en utilisant les vrais positifs, faux positifs et faux négatifs.
  + **F1-Score** est ensuite calculé en fonction de la précision et du rappel.

Dans le cas où la classification n'est pas binaire, les métriques sont généralement calculées pour chaque classe en utilisant une approche "un-contre-tous", puis moyennées pour obtenir une vue globale des performances.

# Arbre de décision

## Évaluation des performances sur l’ensemble de test

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Jeu de données** | **Classe** | **Exactitude** | **Précision** | **Rappel** | **F1-score** | **Matrice de confusion** |
| **Iris** | Classe 0 |  |  |  |  | [[15, 0, 0],  [0, 16, 0],  [0, 0, 14]] |
| Classe 1 |  |  |  |  |
| Classe 2 |  |  |  |  |
| **Abalone** | Classe 0 |  |  |  |  | [[117, 22, 1],  [67, 835, 49],  [0, 71, 92]] |
| Classe 1 |  |  |  |  |
| Classe 2 |  |  |  |  |
| **Wine** | Classe 0 |  |  |  |  | [[431, 48],  [107, 225]] |
| Classe 1 |  |  |  |  |

## Comparaison avec l'implémentation Scikit-learn

Réseau de neurones

## Évaluation des performances sur l’ensemble de test

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Jeu de données** | **Classe** | **Exactitude** | **Précision** | **Rappel** | **F1-score** | **Matrice de confusion** |
| **Iris** | Classe 0 |  |  |  |  | [[15, 0, 0],  [0, 15, 1],  [0, 1, 13]] |
| Classe 1 |  |  |  |  |
| Classe 2 |  |  |  |  |
| **Abalone** | Classe 0 |  |  |  |  | [[121, 19, 0], [141,  490, 320], [1, 60,  102]] |
| Classe 1 |  |  |  |  |
| Classe 2 |  |  |  |  |
| **Wine** | Classe 0 |  |  |  |  | [[366, 113],  [64, 268]] |
| Classe 1 |  |  |  |  |

## Comparaison avec l'implémentation Scikit-learn

# Discussion

### Analyse par jeu de données Iris

* + **Caractéristiques des données** : Ce jeu de donnée est composé de 150 instances réparties uniformément entre trois classes (Iris-setosa, Iris-versicolour, Iris-virginica) et présente à la fois des classes linéairement séparables et non séparables.

### Résultats Arbre de décision :

### Résultats Réseau de neurones :

### Abalone

* + **Caractéristiques des données** : Ce jeu de donnée, avec 4177 instances et 3 classes représentant des intervalles d’âges d’ormeaux, est très déséquilibré, avec une majorité d’exemples pour la classe intermédiaire.

### Résultats Arbre de décision :

### Résultats Réseau de neurones :

### Wine

* + **Caractéristiques des données** : Avec 2700 instances et deux classes binaires (bon ou mauvais vin), ce jeu de donnée présente des caractéristiques fortement corrélées (ex. : acidité, densité).

### Résultats Arbre de décision :

### Résultats Réseau de neurones :

# Comparaison entre les quatres techniques sur l’ensemble de test des jeux de données

### Résultat KNN :

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Jeu de données** | **Exactitude** | **Précision** | **Rappel** | **Temps\* (ms)** | **Best K**  **(2, 3, 5, 7, 9)** | **Métrique (euclidean, manhattan,**  **chebyshev, minkowski (p=3))** |
| **Iris** | - | - | - | 1.002 | 3 | chebyshev |
| **Classe 0** | 1.0 | 1.0 | 1.0 |  |  |  |
| **Classe 1** | 0.9333 | 0.9444 | 0.8947 |  |  |  |
| **Classe 2** | 0.9333 | 0.8571 | 0.9231 |  |  |  |
| **Wine** | - | - | - | 79.9994 | 2 | manhattan |
| **Classe 0** | 0.8089 | 0.8011 | 0.8998 |  |  |  |
| **Classe 1** | 0.8089 | 0.8242 | 0.6777 |  |  |  |
| **Abalones** | - | - | - | 174.0124 | 2 | manhattan |
| **Classe 0** | 0.9282 | 0.6359 | 0.8357 |  |  |  |
| **Classe 1** | 0.8333 | 0.8998 | 0.8780 |  |  |  |
| **Classe 2** | 0.9035 | 0.6479 | 0.5644 |  |  |  |

### Résultat Naïve Bayes :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Jeu de données** | **Exactitude** | **Précision** | **Rappel** | **Temps\***  **(ms)** |
| **Iris** | - | - | - | 1.9996 |
| **Classe 0** | 1.0 | 1.0 | 1.0 |  |
| **Classe 1** | 0.9333 | 0.9444 | 0.8947 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Jeu de données** | **Exactitude** | **Précision** | **Rappel** | **Temps\***  **(ms)** |
| **Classe 2** | 0.9333 | 0.8571 | 0.9231 |  |
| **Wine** | - | - | - | 16.0007 |
| **Classe 0** | 0.7817 | 0.8512 | 0.7641 |  |
| **Classe 1** | 0.7817 | 0.7034 | 0.8072 |  |
| **Abalone** | - | - | - | 34.9994 |
| **Classe 0** | 0.8716 | 0.4601 | 0.8643 |  |
| **Classe 1** | 0.5694 | 0.8612 | 0.5152 |  |
| **Classe 2** | 0.6962 | 0.2417 | 0.6258 |  |

### Temps d'exécution :

### Exactitude (Accuracy) :

### Précision (Precision) :

### Rappel (Recall) :

# Conclusion

# ANNEXE