

Présentation des techniques de classification supervisée utilisées

Classification Paramétrique vs Non-Paramétrique

En classification supervisée, les algorithmes peuvent être divisés en **méthodes paramétriques** et **méthodes non-paramétriques**. Cette distinction repose sur la manière dont les modèles apprennent et généralisent les données.

1. Méthodes Paramétriques

Les modèles paramétriques supposent l'existence d'une distribution sous-jacente aux données et estiment ses paramètres à partir des données d'entraînement.

Classifieurs paramétriques utilisés :

- Régression Logistique (OVA et OVO)
- Régression Multinomiale (Softmax Regression)
- Analyse Discriminante Linéaire (LDA)
- Analyse Discriminante Quadratique (QDA)
- Classifieur Bayésien Naïf

2. Méthodes Non-Paramétriques

Les modèles non-paramétriques ne supposent pas l'existence d'une distribution sous-jacente aux données. Ils estiment les limites entre les classes de façon "gloutonne" en maximisant les différences inter-classes et minimisant les différences intra-classes sur les données d'entraînement.

Classifieurs non-paramétriques utilisés :

- **Arbre de Décision (CART)**
 - **Forêt Aléatoire (Random Forest)**
 - **K-Nearest Neighbors (KNN)**
 - **Machines à Vecteurs de Support (SVM - OVA et OVO)**
 - **Réseau de Neurones (MLP avec sortie Softmax)**
-

Classifieurs Multiclasses Natifs vs Classifieurs Binaires Adaptés

Certains classifieurs sont conçus pour gérer directement plusieurs classes (**classifieurs multiclassés natifs**), tandis que d'autres sont conçus pour distinguer uniquement deux classes et doivent être adaptés pour des problèmes multiclassés.

1. Classifieurs Multiclasses Natifs

Ces classifieurs peuvent traiter directement un problème à plusieurs classes sans nécessiter de transformation :

- **Régression Multinomiale (Softmax Regression)**
- **Analyse Discriminante Linéaire (LDA)**
- **Analyse Discriminante Quadratique (QDA)**
- **Classifieur Bayésien Naïf**
- **Arbre de Décision (CART)**
- **Forêt Aléatoire (Random Forest)**
- **Réseau de Neurones (MLP avec sortie Softmax)**
- **K-Nearest Neighbors (KNN)**

2. Classifieurs Binaires Adaptés

Les classifieurs binaires doivent être transformés pour gérer plusieurs classes en utilisant des approches comme **One-Versus-All (OVA)** ou **One-Versus-One (OVO)** :

- **Régression Logistique** (adaptée en OVA et OVO)
 - **Support Vector Machines (SVM)** (adaptées en OVA et OVO)
-

One-Versus-All (OVA) vs One-Versus-One (OVO)

Les approches **OVA** et **OVO** permettent d'adapter des classifieurs binaires aux problèmes multiclassés.

1. One-Versus-All (OVA)

Avec cette approche, un modèle binaire est entraîné pour chaque classe en la comparant à toutes les autres classes regroupées. Pour une classification à **N classes**, **N modèles binaires** sont entraînés.

Avantages : - Moins de modèles à entraîner (**N** contre $N(N-1)/2$ en OVO), soit 7 modèles pour 7 classes. - Plus efficace pour les jeux de données avec un grand nombre de classes.

Inconvénients : - Les classes déséquilibrées peuvent poser problème car le modèle doit distinguer une classe contre un ensemble de classes plus nombreuses. - Peut générer des scores de confiance biaisés si les classes sont très déséquilibrées.

Exemples de classifieurs utilisant OVA :

- Régression Logistique (OVA)
- SVM (OVA)

2. One-Versus-One (OVO)

Avec l'approche OVO, un modèle est entraîné pour chaque paire de classes. Pour une classification à **N classes**, $N(N-1)/2$ modèles binaires sont entraînés, soit 21 modèles pour 7 classes.

Avantages : - Meilleure séparation entre classes si elles sont bien distinctes. - Moins sensible aux classes déséquilibrées car chaque modèle compare seulement deux classes à la fois.

Inconvénients : - Temps d'entraînement plus long à cause du grand nombre de modèles. - Peut nécessiter plus de ressources pour l'inférence.

Exemples de classifieurs utilisant OVO :

- Régression Logistique (OVO)
- SVM (OVO)