

L'impact des études en distanciel

Une IA pour aider les étudiants en difficultés

Miléna Kostov, Sami Joudet

L3 Informatique : Méthodes de simulation
Valrose - Université Nice Côte d'Azur

Mai, 2022



① Introduction

② Méthodes

③ Expériences

④ Résultats

⑤ Références

1 Introduction

2 Méthodes

3 Expériences

4 Résultats

5 Références

Présentation de la base de données

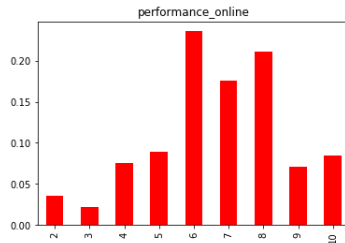
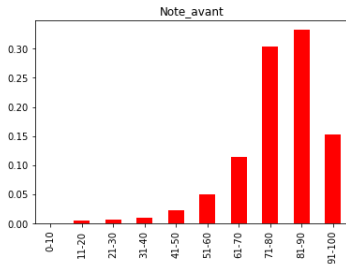
	Gender	Home Location	Level of Education	Age(Years)	Number of Subjects	Device type used to attend classes	Economic status	Family size	Internet facility in your locality	Are you involved in any sports?
0	Male	Urban	Under Graduate	18	11	Laptop	Middle Class	4	5	No
1	Male	Urban	Under Graduate	19	7	Laptop	Middle Class	4	1	Yes

Nom de la base : Online Education System - Review (lien)

23 variables dont des continues et des catégorielles

Taille de l'échantillon : 1033 élèves et étudiants

Une simple observation



1 Introduction

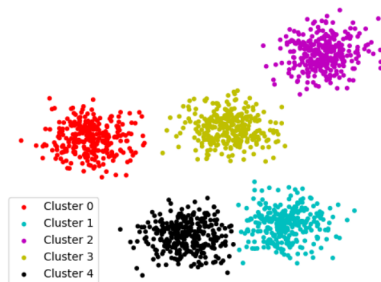
2 Méthodes

3 Expériences

4 Résultats

5 Références

Apprentissage non supervisé : clustering



$$\text{Inertia} = \sum_{i=1}^k \sum_{x_j \in C_i} \|x_j - \mu_i\|^2 \quad (1)$$

Apprentissage supervisé : Support Vector Machine (SVM)

But : séparer les données selon une classe y en maximisant la marge entre les classes

$$\arg \max_{\omega, b} \left\{ \min_i d(x_i, H) = \frac{1}{\|\omega\|} \min_i y_i (\omega^T x + b) \right\} \quad (2)$$

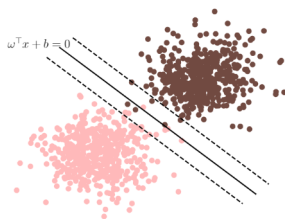
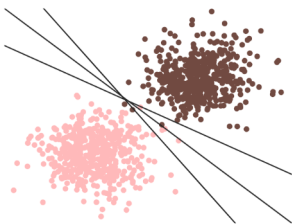
3 méthodes :

- Classification binaire
- Classification multi-classe
- Noyau RBF

SVM avec une *classification binaire*

Deux classes : positive et négative

Séparation de la base pour les tests : 50% entraînement, 25% validation, 25% test

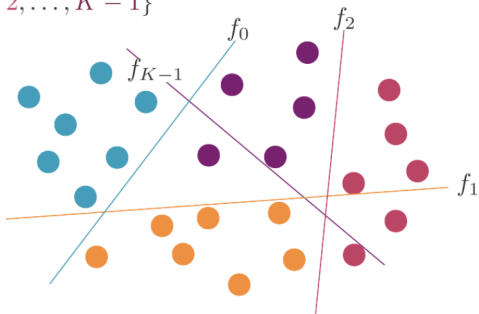


SVM avec une *classification multi-classe*

Plusieurs classes : $y \in \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$

Séparation de la base pour les tests : 50% entraînement, 25% validation, 25% test

$$y \in \{0, 1, 2, \dots, K-1\}$$



SVM avec un *noyau RBF* (*Radial Basis Function*)

But : Calculer la similitude entre deux points X_1 et X_2

$$K(X_1, X_2) = \exp\left(-\frac{\|X_1 - X_2\|^2}{2\sigma^2}\right) \quad (3)$$

Deux critères importants :

- Gamma : mesure de similitude entre deux points.
- C : paramètre de régularisation.

Séparation de la base pour les tests : 70% entraînement, 30% validation

1 Introduction

2 Méthodes

3 Expériences

4 Résultats

5 Références

K_mean

```
1 for i in range(1, 50):
2     kmeans = KMeans(n_clusters=i,
3                     init='k-means++', max_iter=300,
4                     n_init=10, random_state=0
5                 )
6     kmeans.fit(X)
7     wcss.append(kmeans.inertia_)
8
9     init_centers=np1.array(
10         [[0]*29,[1]*29,[2]*29, [3]*29,
11          [4]*29,[5]*29,[6]*29,[7]*29]
12     )
13     kmeans = KMeans(n_clusters=8,init=init_centers)
14     pred_y = kmeans.fit_predict(X)
```

SVM à noyau linéaire

```
1 def repeated_random(n_iter,df_1) :  
2     C_i=[0.001,1,100,1000]  
3     for i in range (n_iter) :  
4         for i in C_i:  
5             svm_clf = SVC(  
6                 kernel="linear", C=C_i[i]  
7             )  
8             svm_clf.fit(  
9                 X_syn_training, y_syn_training  
10            )  
11 matrice = np1.array(repeated_random(10,df_1))
```

SVM à noyau RBF (Radial Basis Function)

```
1 param_grid = {  
2     "C": loguniform(1e2, 1e5),  
3     "gamma": loguniform(1e-4, 1e-1),  
4 }  
5 clf = RandomizedSearchCV(  
6     SVC(kernel="rbf", class_weight="balanced"),  
7     param_grid, n_iter=10  
8 )  
9 clf = clf.fit(X_training, y_training)
```

1 Introduction

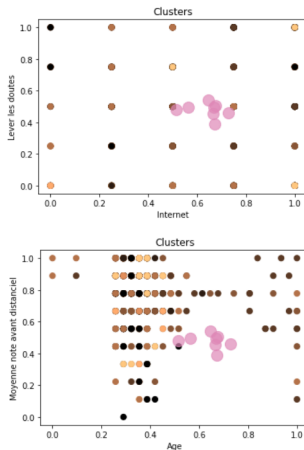
2 Méthodes

3 Expériences

4 Résultats

5 Références

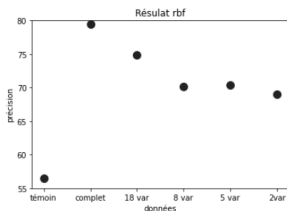
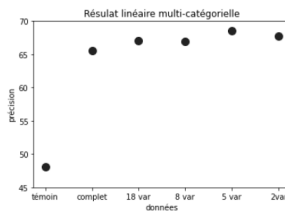
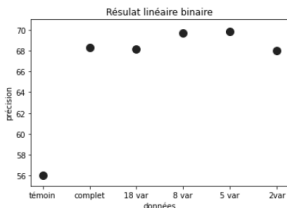
Clustering : invalidation



Observations avec un tableau de corrélation

[illegible]

SVM : les différents modèles



- Meilleur résultat: 79.49% de précision avec RBF.

1 Introduction

2 Méthodes

3 Expériences

4 Résultats

5 Références

Cours de cette année : <https://github.com/mesin-cours/methodes-simulation-informatique>
<https://www.kaggle.com/datasets/sujaradha/online-education-system-review>
https://scikit-learn.org/stable/auto_examples/applications...
[openclassrooms](#)
https://chrisalbon.com/code/machine_learning/support_vector_machines/svc_parameters_using_rbf_kernel/

Merci pour votre attention