



---

# HAX907X - Apprentissage statistique

TP : Support Vector Machines (SVM)

---

Marine GERMAIN  
Coralie ROMANI DE VINCI

# Table des matières

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>Base de données Iris</b>                        | <b>2</b> |
| 1.1      | Classification avec noyau linéaire . . . . .       | 2        |
| 1.2      | Classification avec noyau polynomial . . . . .     | 2        |
| <b>2</b> | <b>SVM GUI</b>                                     | <b>3</b> |
| <b>3</b> | <b>Classification des visages</b>                  | <b>3</b> |
| 3.1      | Influence du paramètre de régularisation . . . . . | 3        |
| 3.2      | Variable de nuisances . . . . .                    | 3        |
| 3.3      | Réduction de dimensions . . . . .                  | 3        |

# 1 Base de données Iris

Dans cette section, nous étudierons la base de données Iris sur laquelle nous ferons une étude de classification sur les classes 1 et 2. Nous utiliserons d'abord le noyau linéaire puis le noyau polynomial et nous les comparerons. Nous avons séparé le dataset en deux parties : un ensemble d'entraînement (75% des données) et un ensemble de test (25% des données).

## 1.1 Classification avec noyau linéaire

La classification des deux premières variables avec un noyau linéaire à pour objectif de trouver une hyperplan qui sépare au mieux les deux classes en maximisant la marge entre les points les plus proches de chaque classe. Les scores obtenues pour cette méthode sont les suivants :

```
{'C': np.float64(0.013049019780144023), 'kernel': 'linear'}  
Generalization score for linear kernel: 0.6933333333333334, 0.64
```

Figure 1

Pour les données d'entraînement, le modèle a classifié correctement 69,3% des données, contre 64% pour les données de test.

## 1.2 Classification avec noyau polynomial

La classification des deux premières variables avec un noyau polynomial à pour objectif de trouver une frontière de décision non linéaire capable de séparer au mieux les deux classes. Nous souhaitons ici voir si l'emploi d'un noyau polynomial nous permet d'obtenir un meilleur classifieur que celui obtenu précédemment.

Les scores obtenues pour cette méthode sont les suivants :

```
{'C': np.float64(1.0), 'degree': np.int64(1), 'gamma': np.float64(10.0), 'kernel': 'poly'}  
Generalization score for polynomial kernel: 0.68, 0.68
```

Figure 2

Pour les données d'entraînement et de test, le modèle a classifié correctement 68% des données.

### Comparaison des deux méthodes :

- Nous disposons d'abord des scores obtenus en figure 1 et figure 2. On remarque que le score pour le noyau linéaire est très légèrement plus élevé que celui du noyau polynomial pour les données d'entraînements 0.693 contre 0.68. En revanche il est légèrement plus faible pour les données de test 0.64 contre 0.68. A noter que ces différences sont faibles et que ces deux méthodes semblent donc similaires.

- Ensuite, nous avons tracé graphiquement ces résultats à l'aide des frontières :

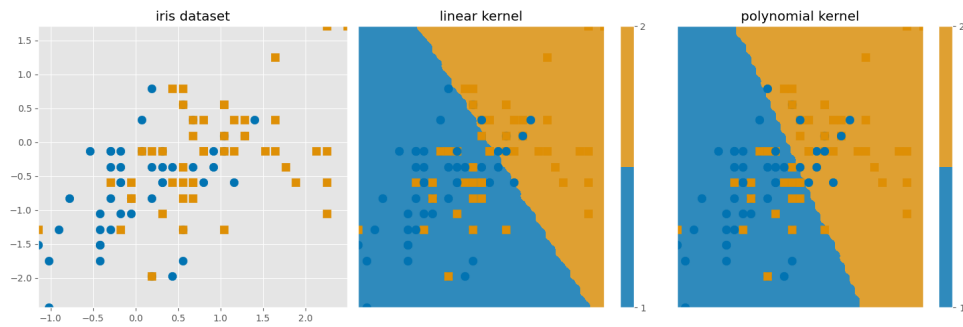


Figure 3

La figure 3 permet de visualiser les deux classifieurs sur l'ensemble des données du dataset Iris.

On observe que la frontière du noyau polynomial semble plutôt linéaire. La différence remarquable est que pour le noyau linéaire on observe qu'il y a moins de données "bleues" mal classifiées que pour le noyau polynomial. On peut faire l'observation inverse concernant les données "oranges". A REFORMULER + CONCLURE

## 2 SVM GUI

## 3 Classification des visages

### 3.1 Influence du paramètre de régularisation

### 3.2 Variable de nuisances

### 3.3 Réduction de dimensions