MLWEEK - Formation Machine Learning

Cours 3 : Forêts Aléatoire Et Machine à Vecteur Support (Random Forest & Support Vector Machine)

Jeff Abrahamson
François-Marie Giraud

6 Mars 2019

https ://www.ml-week.com/

Machine Learning

Cours 3 : Forêts Aléatoire Et Machine à Vecteur Support
(Random Forest & Support Vector Machine)

Machine Learning

Rappels

Rappels

Apprentissage supervisé

Prédire une valeur numérique ou l'appartenance à une classe
Données d'entrainement **annotées** !

Ex : prédiction CAC40, classification d'image/texte/...

À l'intérieur du **Modèle** :

- **Algèbre linéaire**
- Théorie de l'Optimisation
- Calcul différentiel
- Probabilités
- Statistiques

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} ax + by \\ cx + dy \end{bmatrix}$$

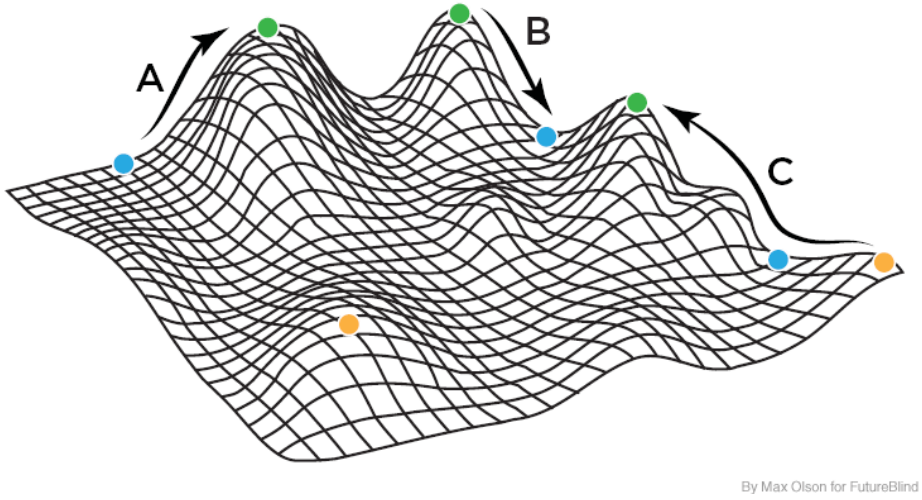
$$\begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} ax + by + c \\ dx + ey + f \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} ax + by + cz \\ dx + ey + fz \\ gx + hy + iz \end{bmatrix}$$

$$\begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} ax + by + cz + d \\ ex + fy + gz + h \\ ix + jy + kz + l \\ 1 \end{bmatrix}$$

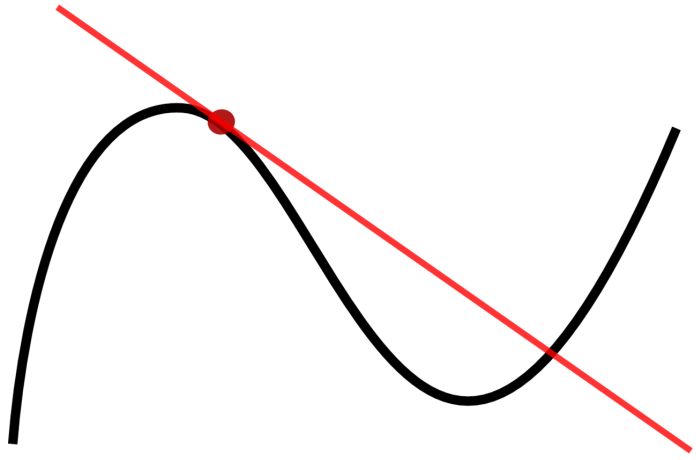
À l'intérieur du **Modèle** :

- Algèbre linéaire
- **Théorie de l'Optimisation**
- Calcul différentiel
- Probabilités
- Statistiques



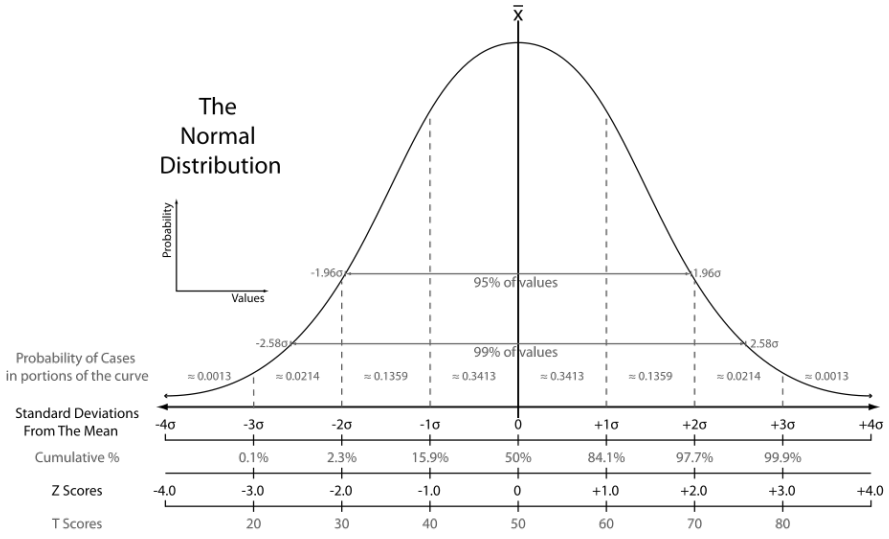
À l'intérieur du **Modèle** :

- Algèbre linéaire
- Théorie de l'Optimisation
- **Calcul différentiel**
- Probabilités
- Statistiques

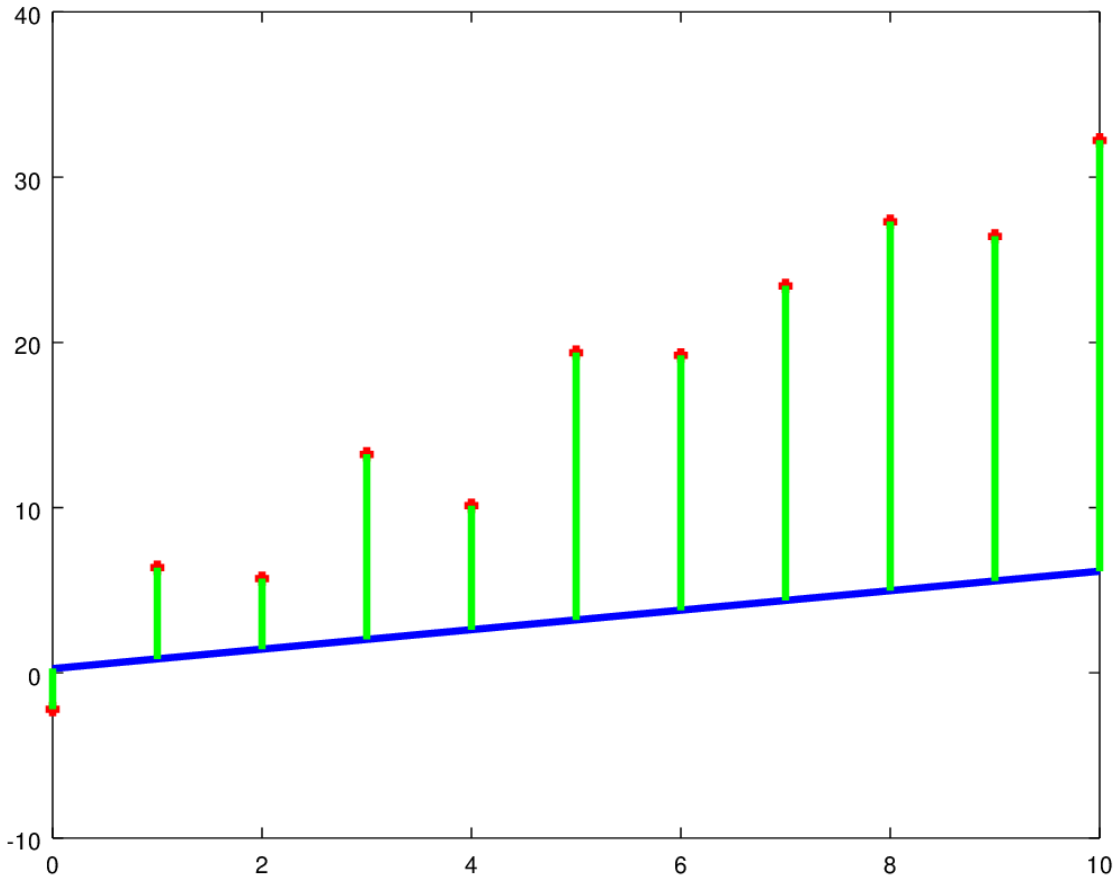


À l'intérieur du **Modèle** :

- Algèbre linéaire
- Théorie de l'Optimisation
- Calcul différentiel
- **Probabilités**
- **Statistiques**



Initiation à la descente de gradient :

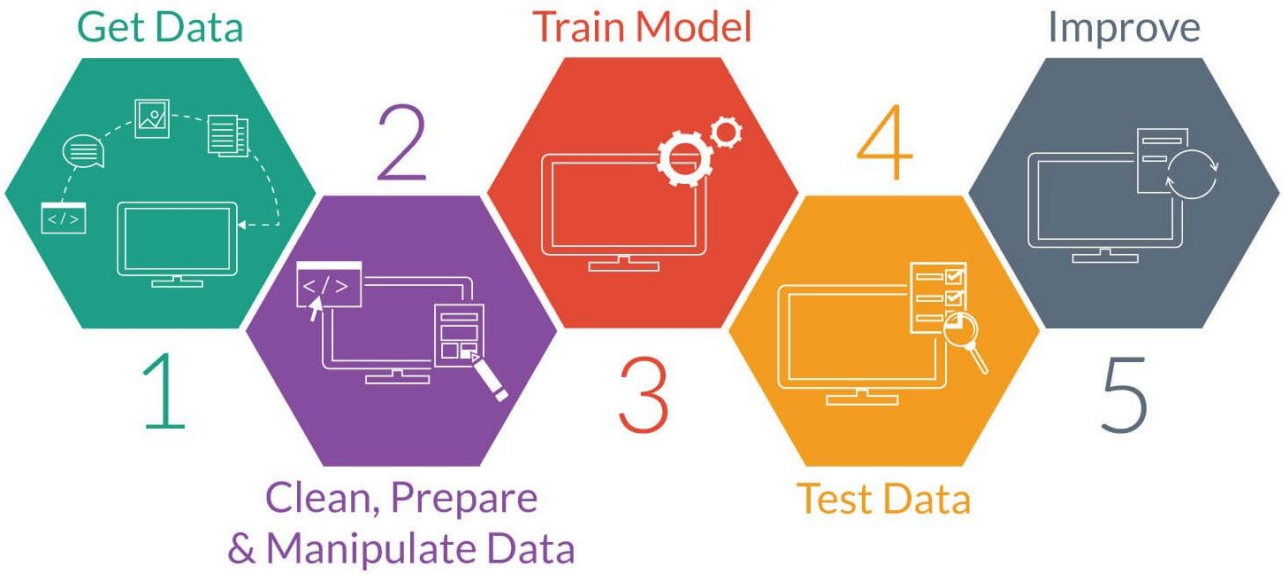


REVIEW
&
QUESTIONS ?

Mettre en place un transition IA

Développer un projet en Machine Learning

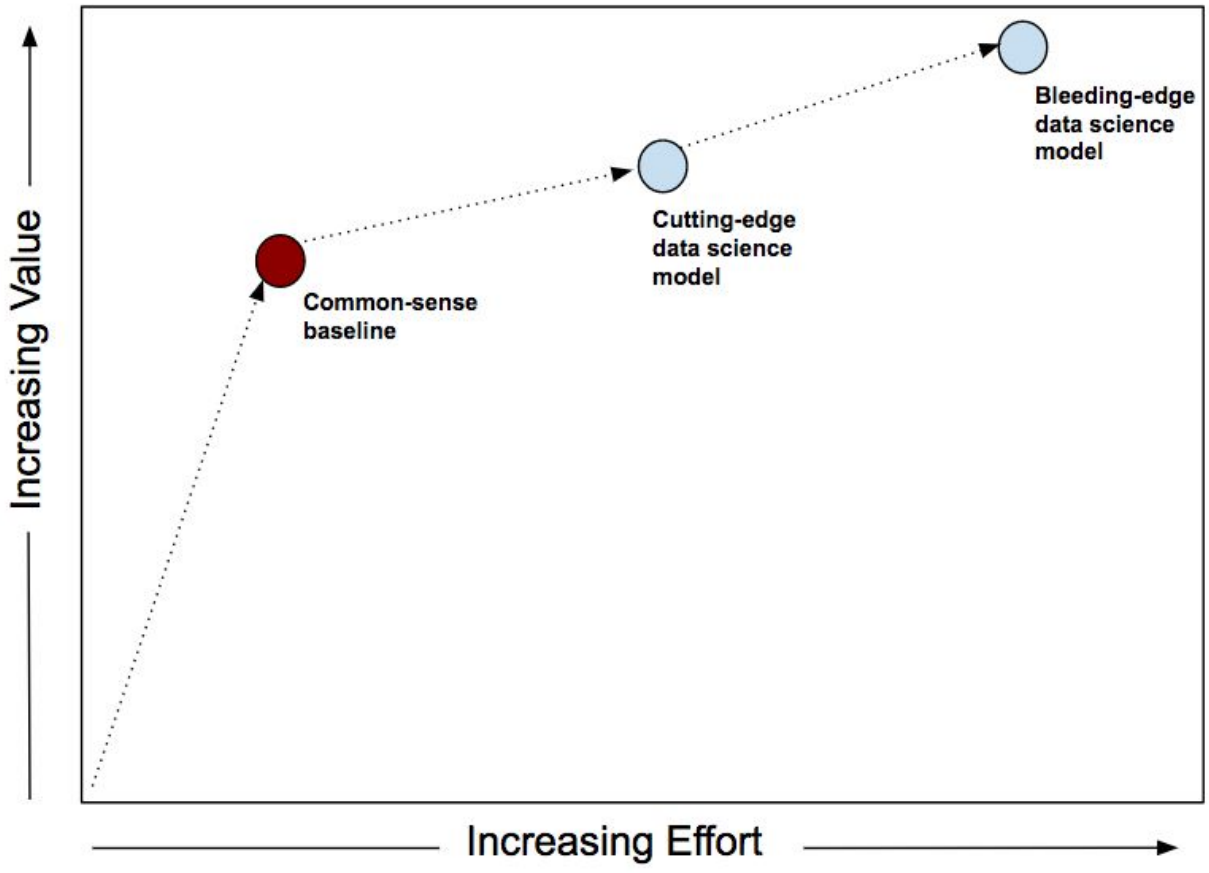
Développer un projet en Machine Learning



Développer un projet en Machine Learning

- Séparer les données en TRAIN/VALIDATION/TEST (i.e 60/20/20)
- Apprendre sur **TRAIN**
- Optimizer les hyperparamètres sur **VALIDATION**
- Observer la performance finale sur **TEST**

Développer un projet en Machine Learning



Développer un projet en Machine Learning

Projet académiques Vs Industriels

- $\neq$ Développement logiciel
- $\neq$ Infrastructure
- $\neq$ Performances

Développer un projet en Machine Learning

Développement logiciel

Académique	Industriel
▪ Pile de scripts	▪ Code hiérarchisé et déployable en production
▪ Peu de documentation	▪ Documentation
▪ Fonctionne le temps de l'expérience	▪ Code maintenable et robuste
▪ "Fair use"	▪ Galaxies de licences à respecter

Développer un projet en Machine Learning

Infrastructure

Académique	Industriel
▪ Données = un fichier	▪ Données = cloud
▪ Hardware limité	▪ Cloud computing
▪ Performance = Précision	▪ Performance = Plus-value

Développer un projet en Machine Learning

Un problème d'ingénérie avant d'être un problème de machine learning :

Données et prétraitements de qualité $>$ algorithme de qualité

Développer un projet en Machine Learning

Une approche en 4 étapes :

- Créer un pipeline robuste de bout en bout (sans ML)
- Intégrer du ML simple
- Ajouter des caractéristiques sensées
- Conserver un pipeline robuste

Développer un projet en Machine Learning

Créer un pipeline robuste de bout en bout (sans ML) :

- Une baseline avec une heuristique
- Mettre en place des statistiques d'évaluation

Développer un projet en Machine Learning

Intégrer du ML simple :

1. Obtenir des données
2. Définir UNE métrique d'évaluation facile à observer
3. Définir des caractéristiques sensées et faciles à obtenir
4. Considérer les heuristiques comme des caractéristiques
5. Documenter TOUTES les caractéristiques utilisées

Développer un projet en Machine Learning

Intégrer du ML simple :

6. Apprendre un modèle tous les n-jours
7. Évaluer la dégradation des performances en fonction de l'âge du modèle
8. Vérifier les performances en test avant de déployer en production
9. Modèle appris sur des données jusqu'au jour N, tester sur les données après le jour N
10. Mesurer la différence entre performance en apprentissage et test
11. Plateau de performance $\Rightarrow$ trouver des nouvelles caractéristiques/augmenter la puissance du modèle
12. Supprimer des caractéristiques pas déterminantes

Développer un projet en Machine Learning

Ajouter des caractérstiques sensées :

- Beaucoup de caractéristiques simples $>$ peu de caractéristiques complexes
- Des caractéristiques répandues plutôt que rares
- Regarder les erreurs pour imaginer les caractéristiques qui aideraient
- Communiquer avec les experts métiers

Développer un projet en Machine Learning

Des questions à garder en tête :

- Ajouter des statistiques d'évaluation ?
- Revoir/Complexifier la métrique d'évaluation ?
- Les données sont-elle "stables" ?

Machine Learning

Les premiers modèles de machine learning à tester

Les premiers modèles de machine learning à tester

Intégrer du ML simple :

1. Obtenir des données
2. Définir UNE métrique d'évaluation facile à observer
3. Définir des caractéristiques sensées et faciles à obtenir
4. Considérer les heuristiques comme des caractéristiques
5. Documenter TOUTES les caractéristiques utilisées

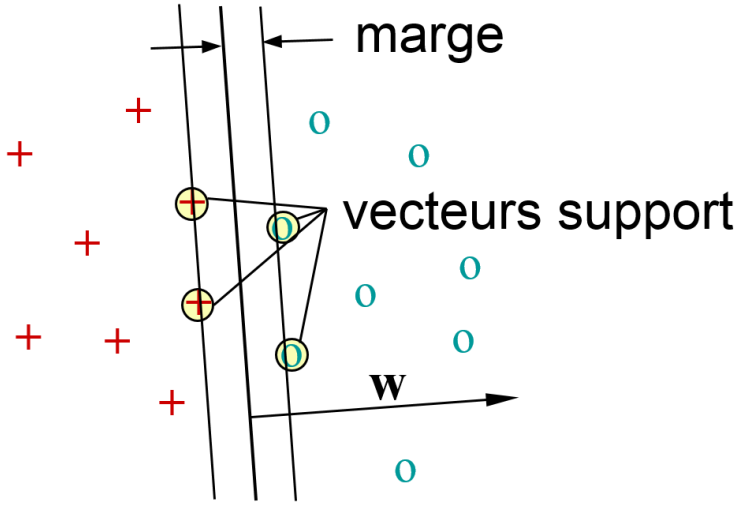
Les premiers modèles de machine learning à tester

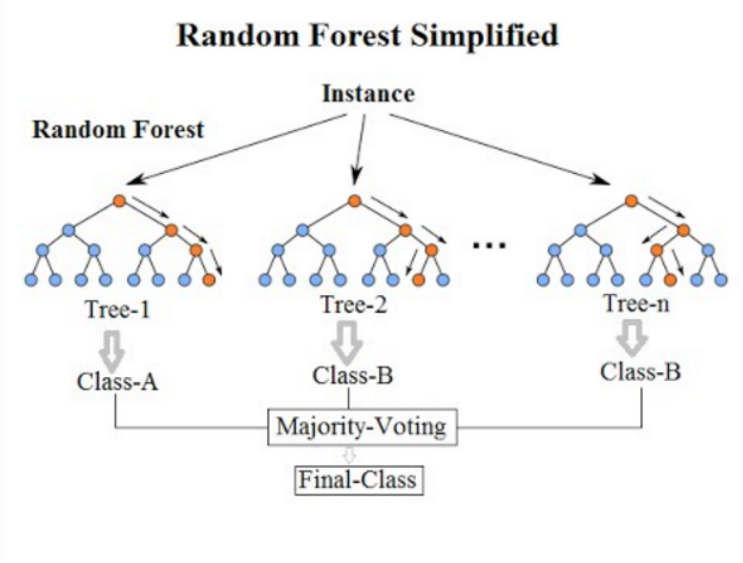
Intégrer du ML simple :

6. **Apprendre un modèle tous les n-jours**
7. Évaluer la dégradation des performances en fonction de l'âge du modèle
8. Vérifier les performances en test avant de déployer en production
9. Modèle appris sur des données jusqu'au jour N, tester sur les données après le jour N
10. Mesurer la différence entre performance en apprentissage et test
11. Plateau de performance $\Rightarrow$ trouver des nouvelles caractéristiques/augmenter la puissance du modèle
12. Supprimer des caractéristiques pas déterminantes

Les premiers modèles de machine learning à tester

Support Vector Machine

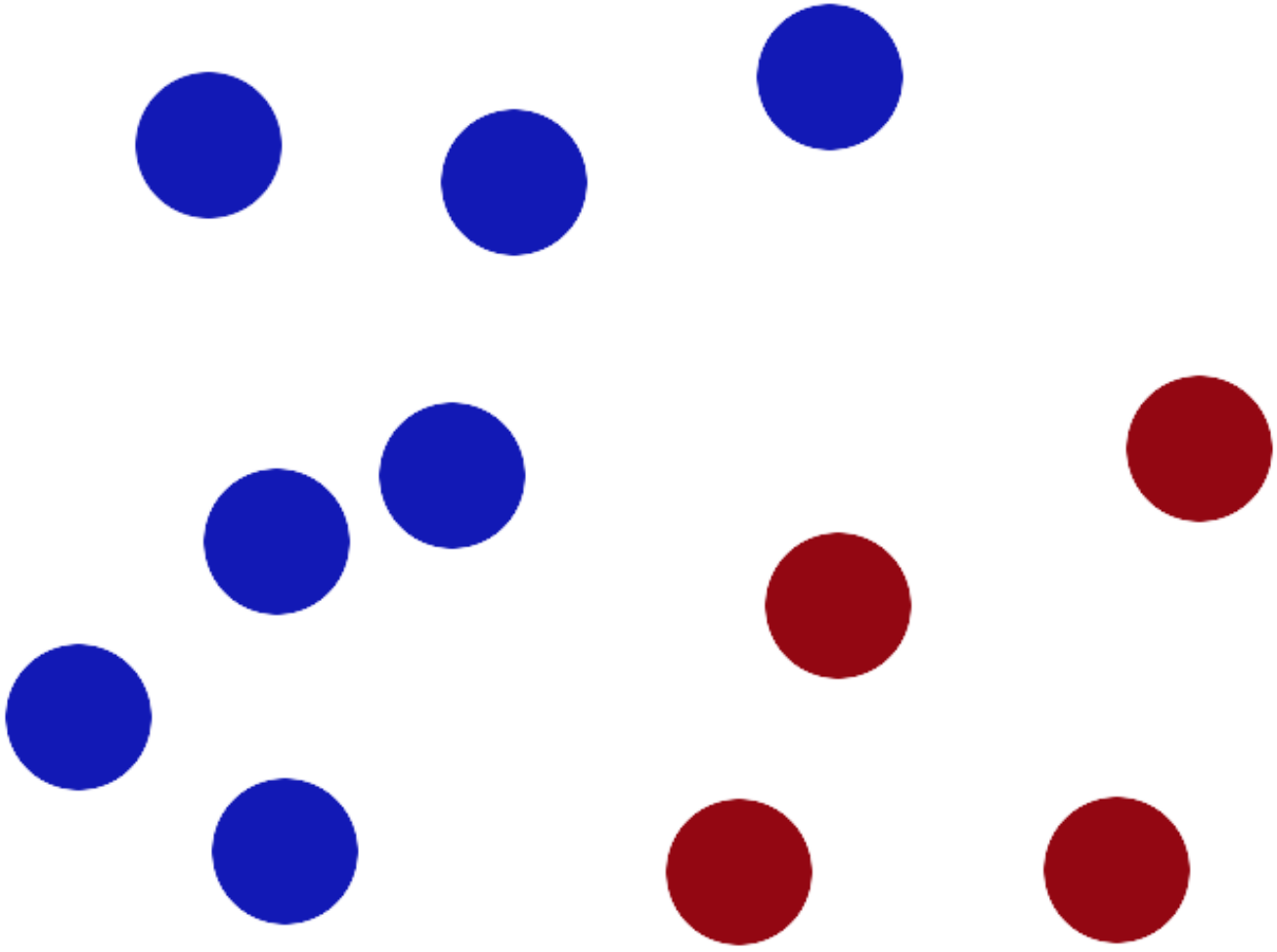




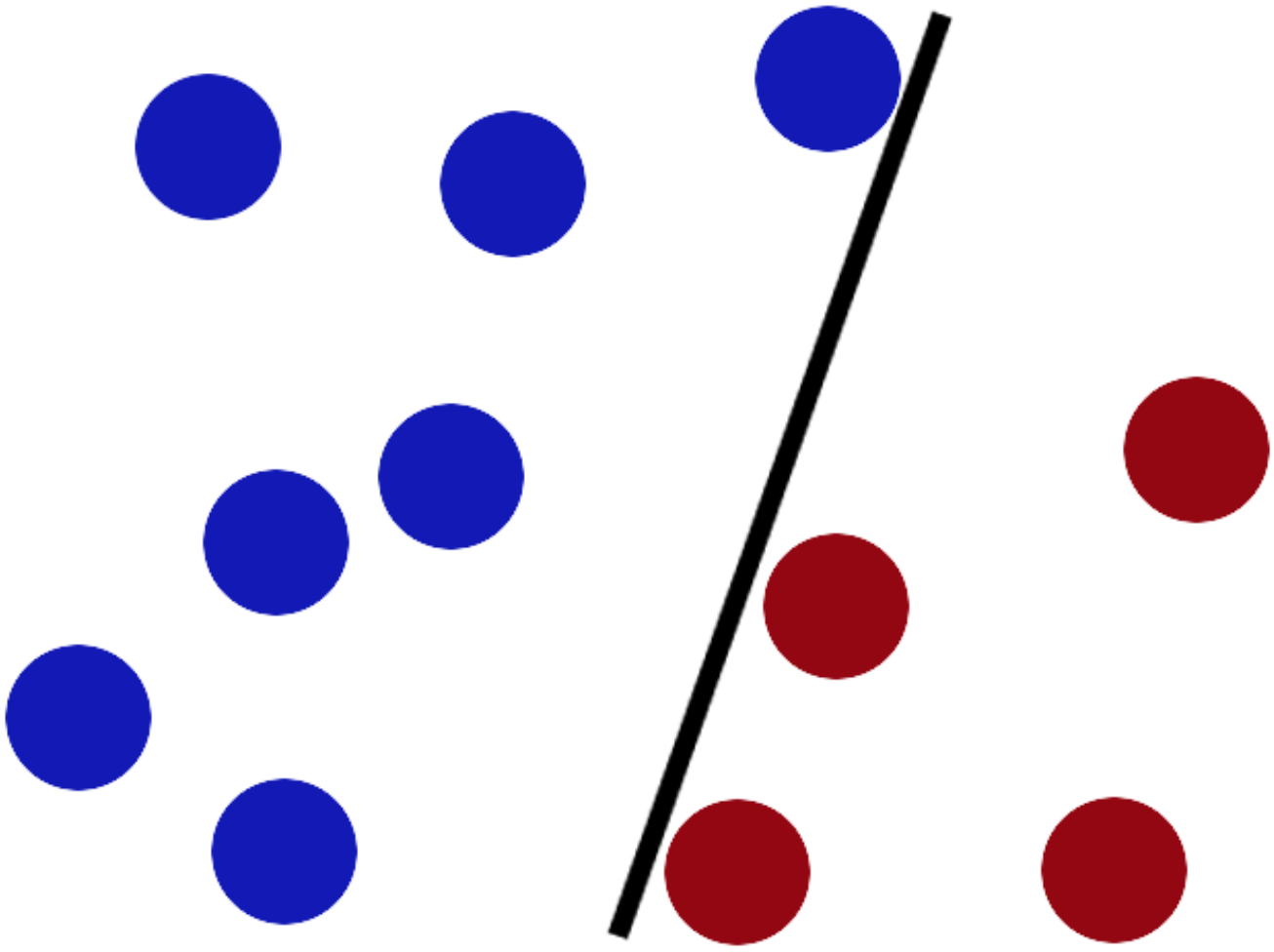
Machine Learning

Support Vector Machine

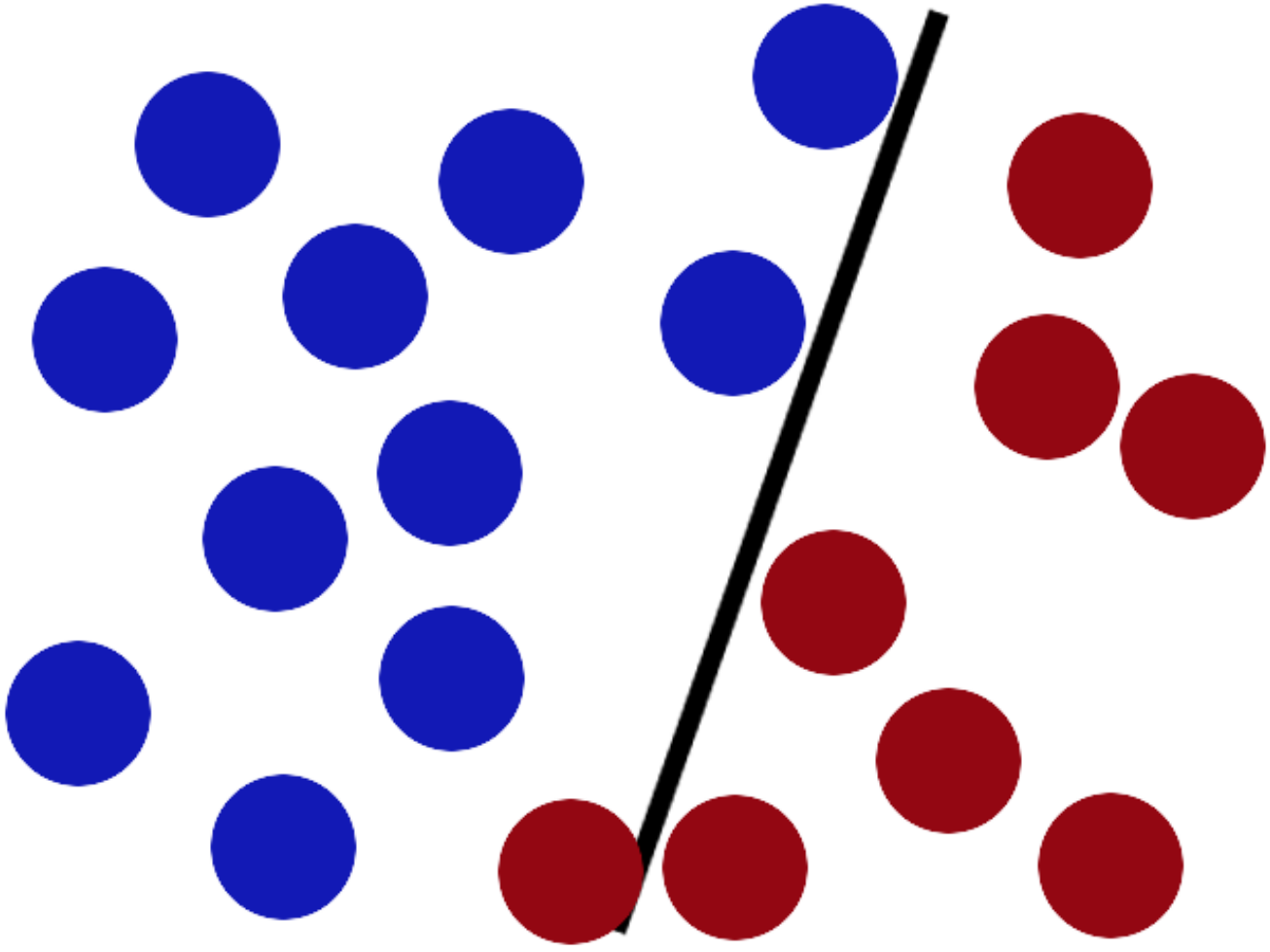
Support Vector Machine



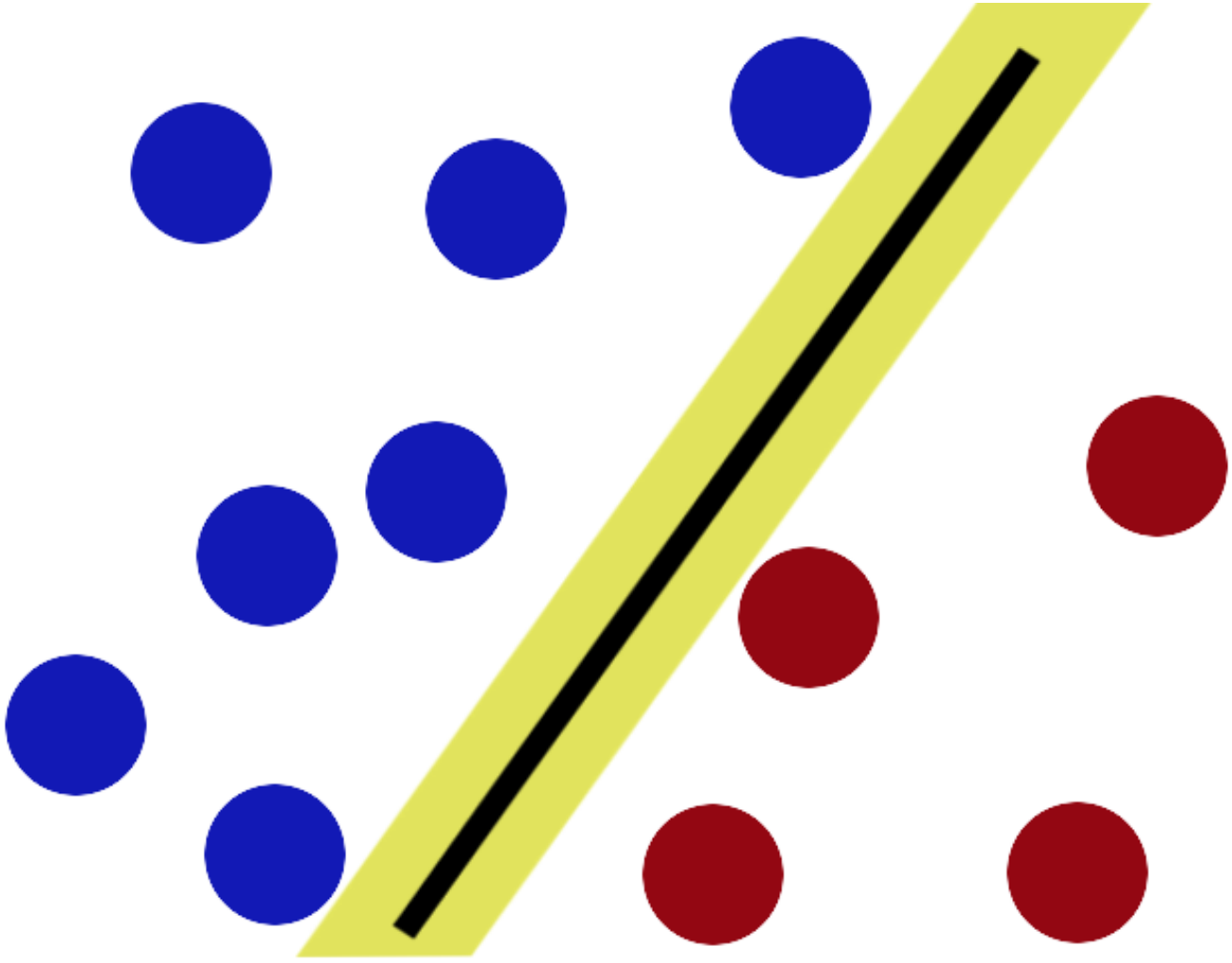
Support Vector Machine



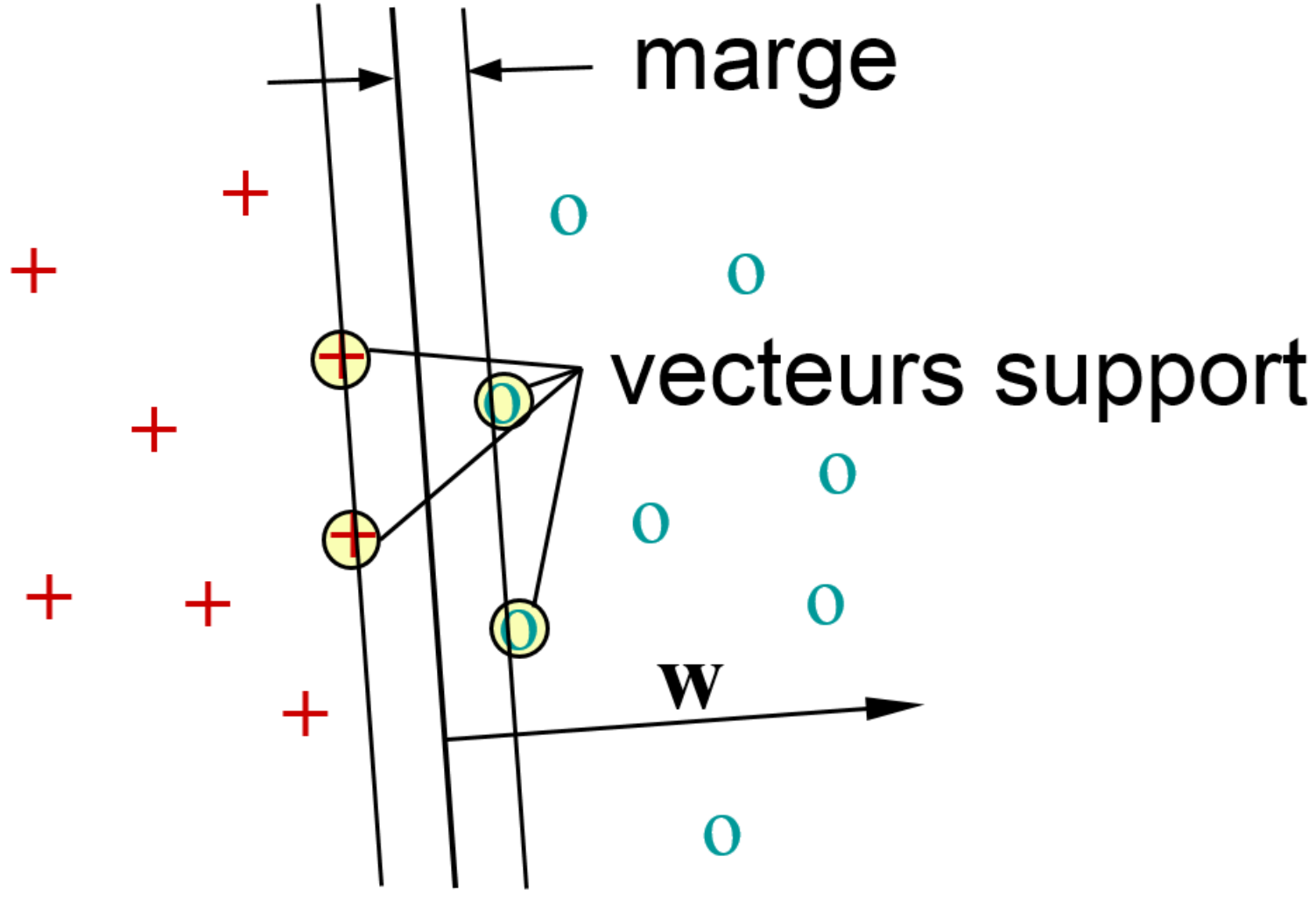
Support Vector Machine



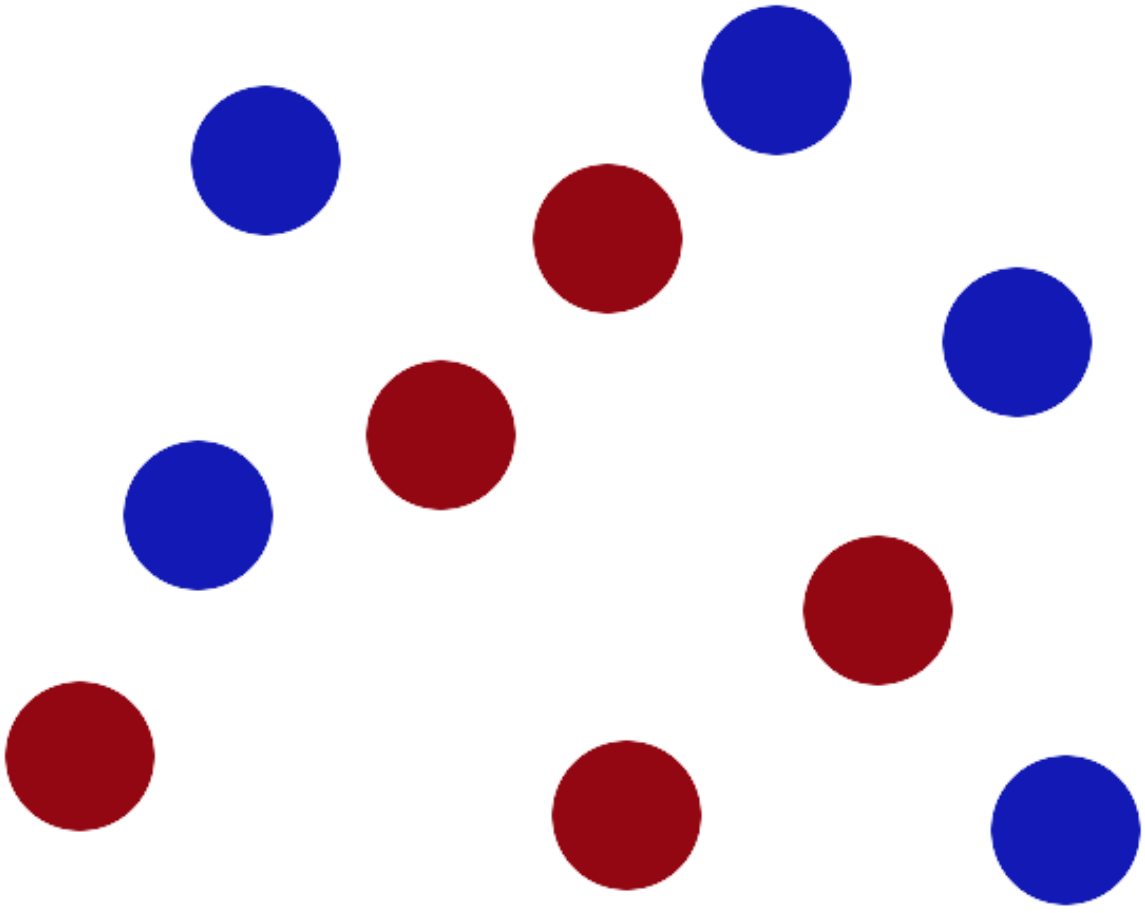
Support Vector Machine



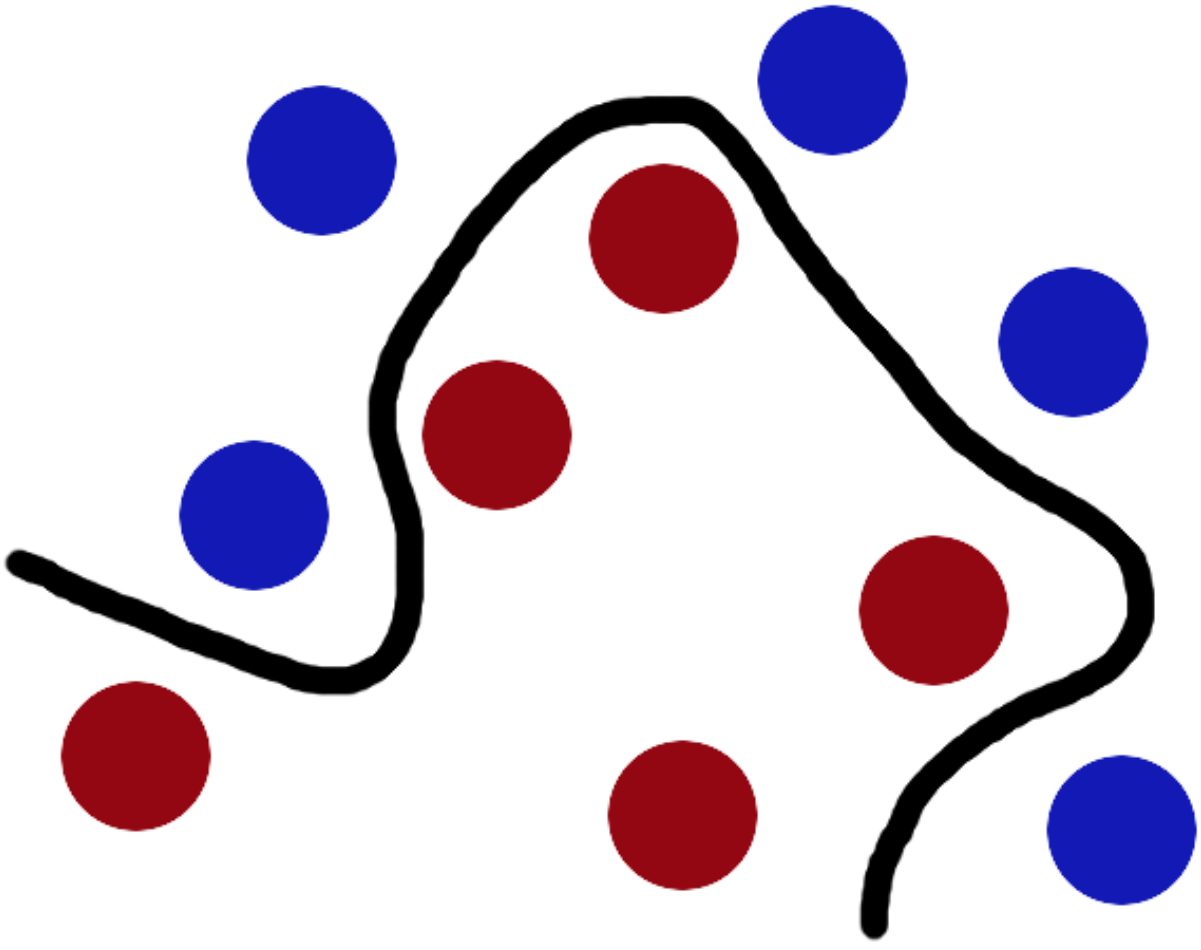
Support Vector Machine



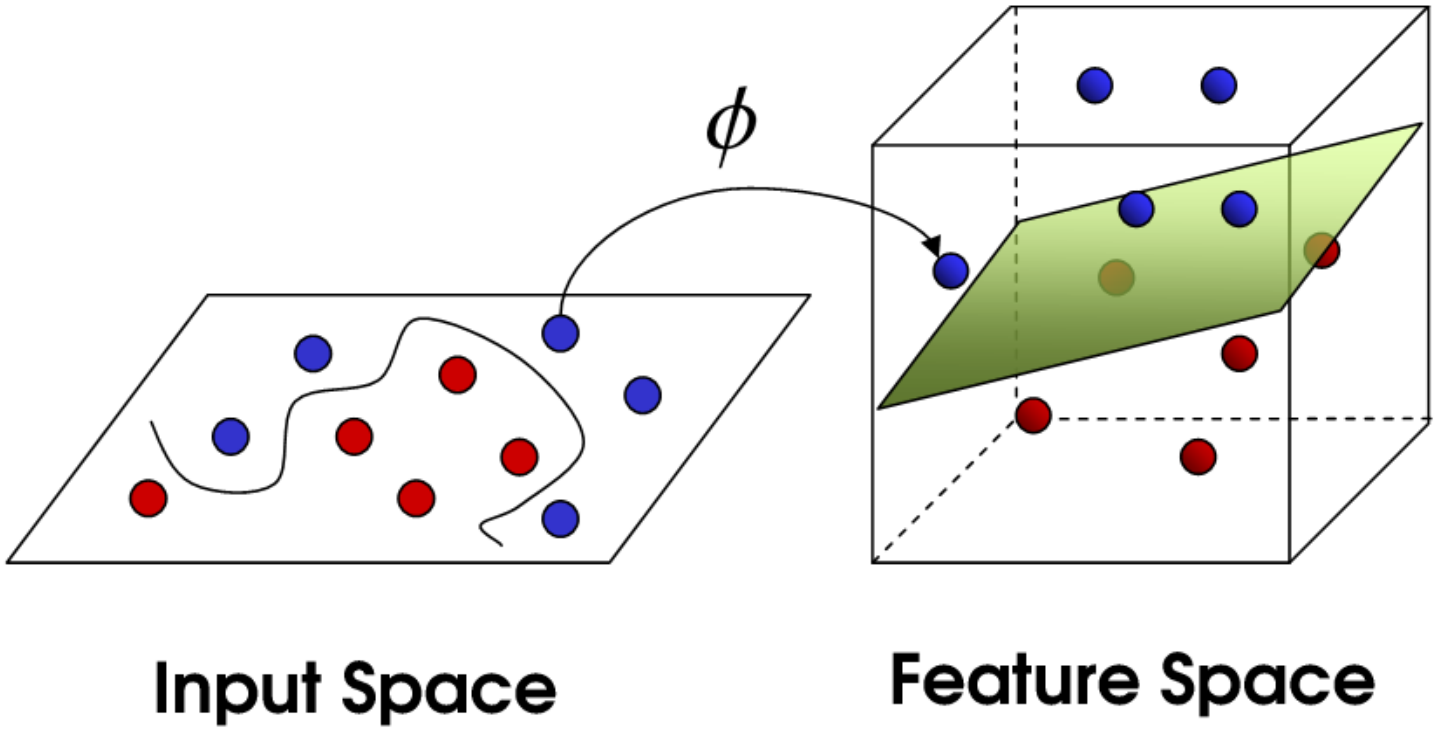
Support Vector Machine



Support Vector Machine



Support Vector Machine



Support Vector Machine

Petite vidéo d'explication des méthodes à noyaux (Kernel SVM)

Support Vector Machine

Généralisation à un problème de régression logistique à $K > 2$ classes :

- One Vs All : K modèles. Agréagation par meilleur score.
- One Vs One : $\frac{K(K-1)}{2}$ modèles. Vote majoritaire.

Support Vector Machine

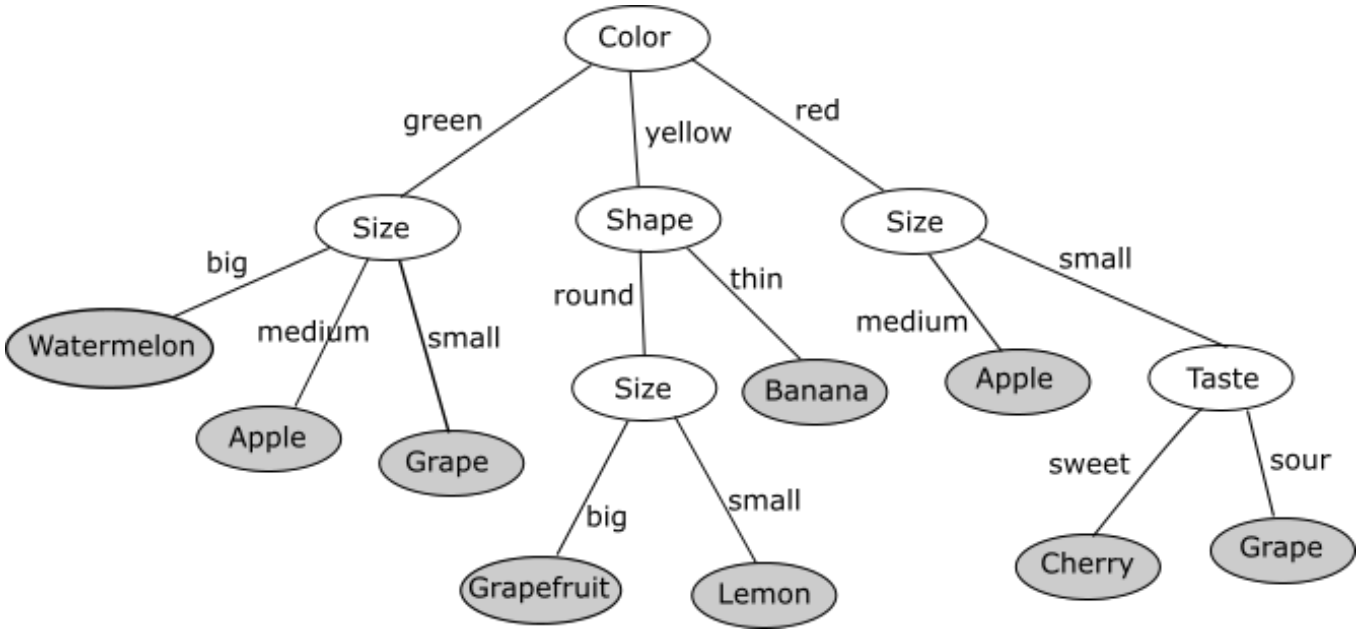
QUESTIONS ?

Machine Learning

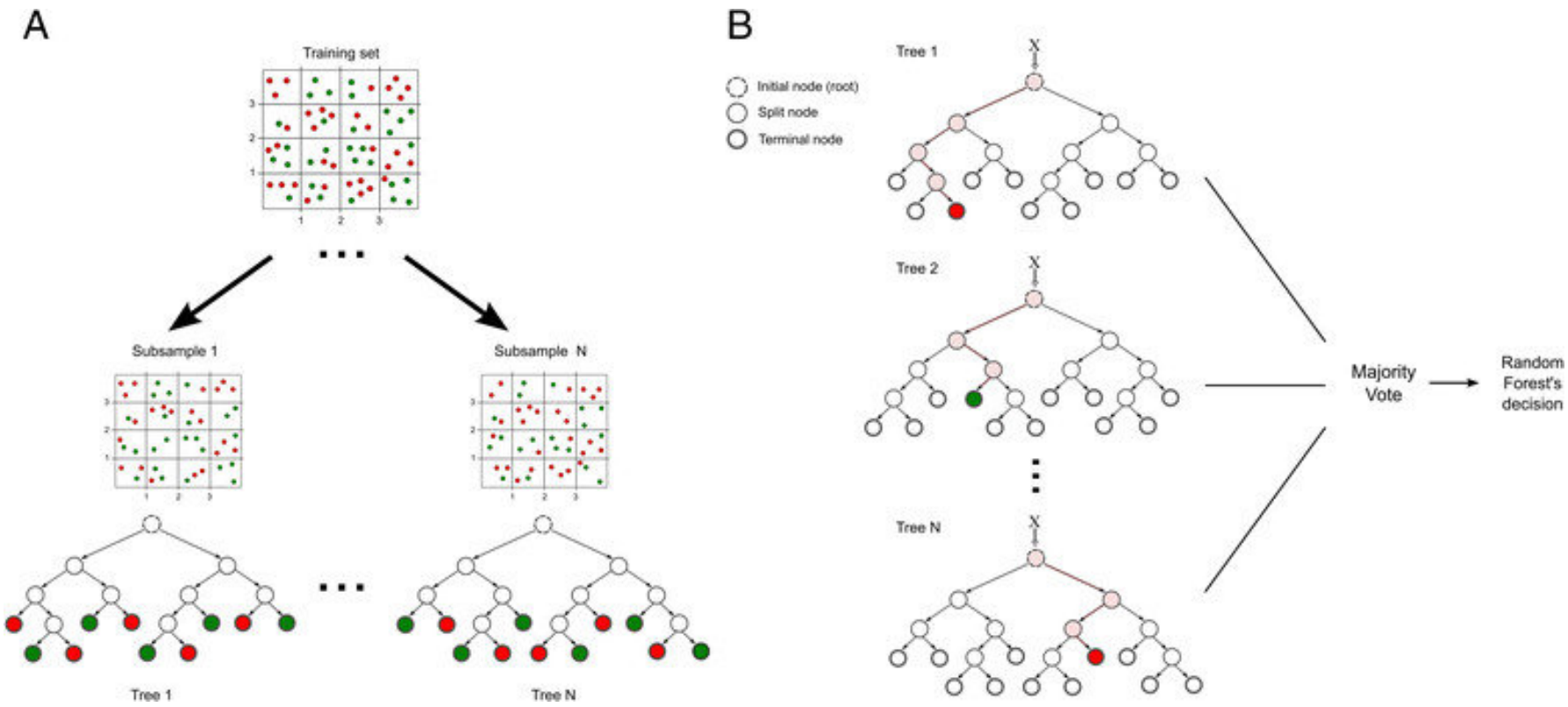
Random Forest

Random Forest

Arbre de décision :



Random Forest



Random Forest

- Pas de sur-apprentissage
- Une fois appris, le modèle est très rapide

Support Vector Machine

QUESTIONS ?