

Coralie Joucla

Machine learning :
principes de bases et enjeux médicaux

Café Neuro

Jeudi 17 mars 2022



Intelligence artificielle

Définitions

L'ordinateur et la machine

- ▶ Un ordinateur est une machine :
 - ▶ Munie d'une mémoire dans laquelle on insère des **données**,
 - ▶ Et d'un **algorithme**
 - ▶ Codé dans un langage propre
 - ▶ Qui définit le traitement à effectuer sur ces données.

L'ordinateur et la machine

- ▶ Un ordinateur est une machine :
 - ▶ Munie d'une mémoire dans laquelle on insère des **données**,
 - ▶ Et d'un **algorithme**
 - ▶ Codé dans un langage propre
 - ▶ Qui définit le traitement à effectuer sur ces données.
- ▶ La machine :
 - ▶ **Ne peut pas** décider d'elle-même d'exécuter telle ou telle consigne
 - ▶ Ne peut qu'exécuter les étapes déterminées à l'avance **par l'auteur humain** de l'algorithme (Sabouret 2019).

Intelligence artificielle

- ▶ D'après Marvin Minsky, l'IA est :
 - ▶ Discipline informatique de construction de programmes
 - ▶ Réalisation de tâches qui sont (pour l'instant) mieux accomplies par des humains
 - ▶ Demandent des processus mentaux de haut niveau
 - ▶ Apprentissage perceptuel, organisation de la mémoire, raisonnement critique...

Intelligence artificielle

- ▶ D'après Marvin Minsky, l'IA est :
 - ▶ Discipline informatique de construction de programmes
 - ▶ Réalisation de tâches qui sont (pour l'instant) mieux accomplies par des humains
 - ▶ Demandent des processus mentaux de haut niveau
 - ▶ Apprentissage perceptuel, organisation de la mémoire, raisonnement critique...
- ▶ D'après Sabouret, c'est l'écriture d'algorithmes plus sophistiqués, qui :
 - ▶ Calculent automatiquement les bonnes réponses à partir des données fournies,
 - ▶ Au lieu d'écrire soi-même tout le détail des instructions à la main.
 - ▶ Le traitement des données est décrit dans un programme qui est lui-même une donnée fournie à la machine.

Intelligence artificielle

- ▶ Le traitement des données est décrit dans un programme qui est lui-même une donnée fournie à la machine.
 - ▶ Le résultat fourni n'est donc pas créé de toutes pièces par l'IA, mais dépend des instructions données par l'informaticien.

Intelligence artificielle

- ▶ Le traitement des données est décrit dans un programme qui est lui-même une donnée fournie à la machine.
 - ▶ Le résultat fourni n'est donc pas créé de toutes pièces par l'IA, mais dépend des instructions données par l'informaticien.
 - ▶ Un ordinateur ne peut écrire lui-même un programme pour résoudre un problème pour lequel il n'a pas été conçu car il y a :
 - ▶ Trop de données différentes,
 - ▶ Trop de règles requises pour que ce programme fonctionne correctement.

Intelligence artificielle

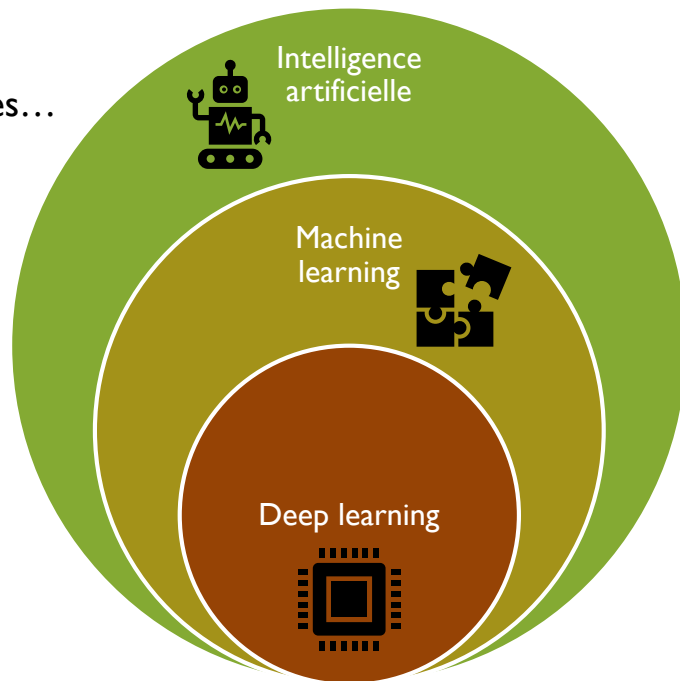
- ▶ Le traitement des données est décrit dans un programme qui est lui-même une donnée fournie à la machine.
 - ▶ Le résultat fourni n'est donc pas créé de toutes pièces par l'IA, mais dépend des instructions données par l'informaticien.
 - ▶ Un ordinateur ne peut écrire lui-même un programme pour résoudre un problème pour lequel il n'a pas été conçu car il y a :
 - ▶ Trop de données différentes,
 - ▶ Trop de règles requises pour que ce programme fonctionne correctement.
- ▶ Le programme d'IA qui résoudrait tous les problèmes n'existe pas car, justement, chaque problème nécessite un programme spécifique et des données adaptées

Qu'est-ce qui rend une machine « intelligente » ?

Imite comportement humain

Narrow IA très utilisée

Assistant virtuel, voitures autonomes...

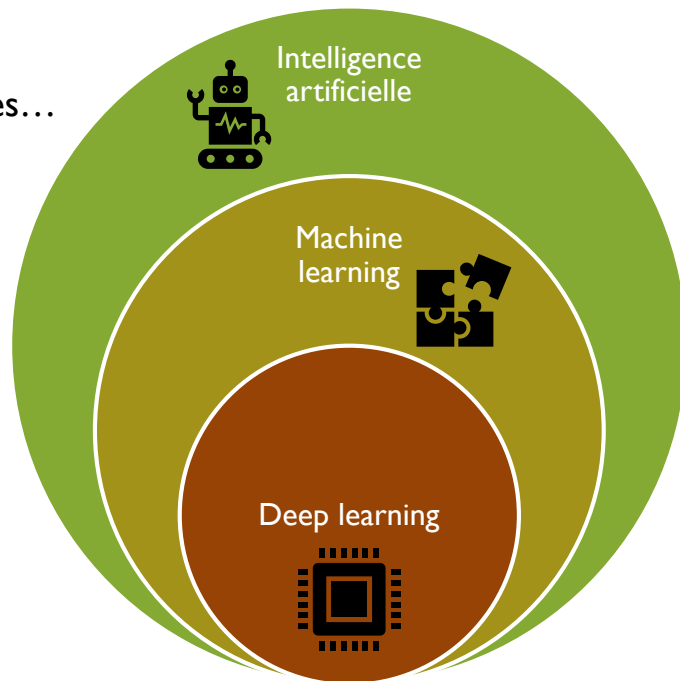


Qu'est-ce qui rend une machine « intelligente » ?

Imite comportement humain

Narrow IA très utilisée

Assistant virtuel, voitures autonomes...



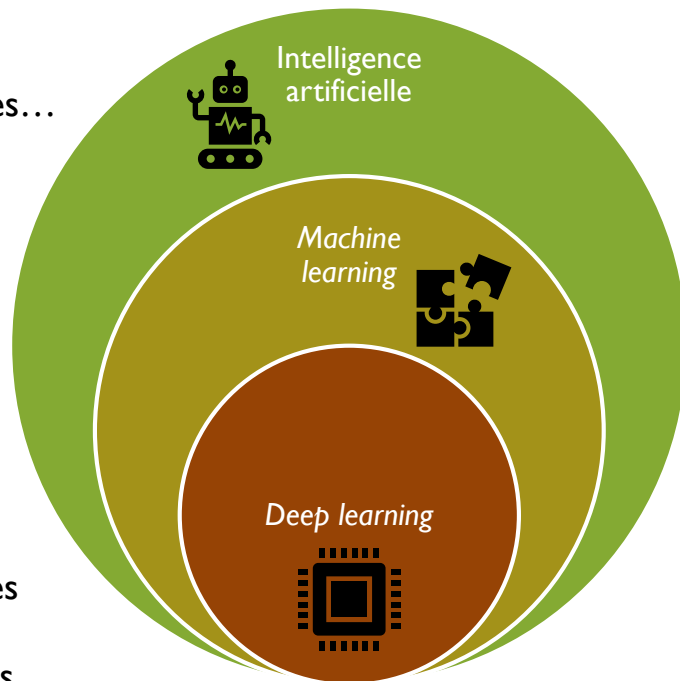
Apprend d'exemples sans avoir à être explicitement programmé

Qu'est-ce qui rend une machine « intelligente » ?

Imite comportement humain

Narrow IA très utilisée

Assistant virtuel, voitures autonomes...



Apprend d'exemples sans avoir à être explicitement programmé

Type de *machine learning* utilisant des réseaux de neurones profonds, ne requiert pas d'extraction de features



Machine learning

Etape I : définition du problème à résoudre

Différents types de *machine learning*

CLASSICAL MACHINE LEARNING

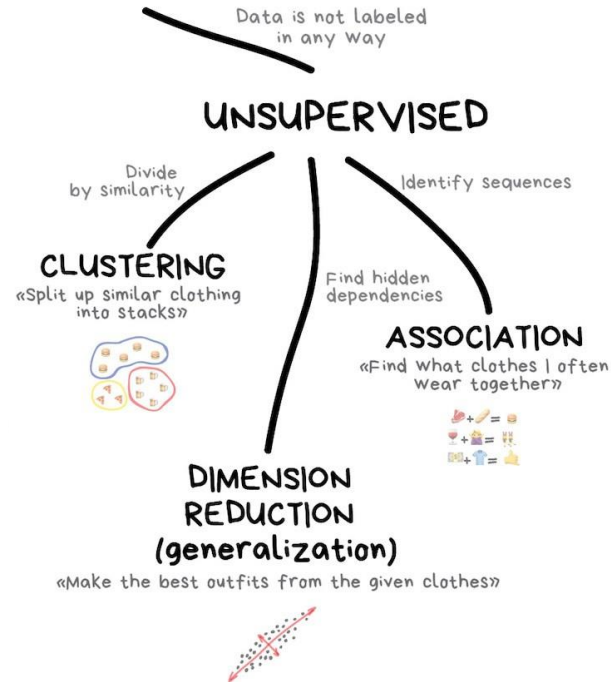
Data is not labeled
in any way

UNSUPERVISED

- ▶ Découvrir des modèles cachés / structures intrinsèques dans les données
- ▶ Tirer des conclusions à partir d'ensembles de données constitués de données d'entrée sans réponses étiquetées

Différents types de *machine learning*

CLASSICAL MACHINE LEARNING



Différents types de *machine learning*

CLASSICAL MACHINE LEARNING

Data is pre-categorized
or numerical

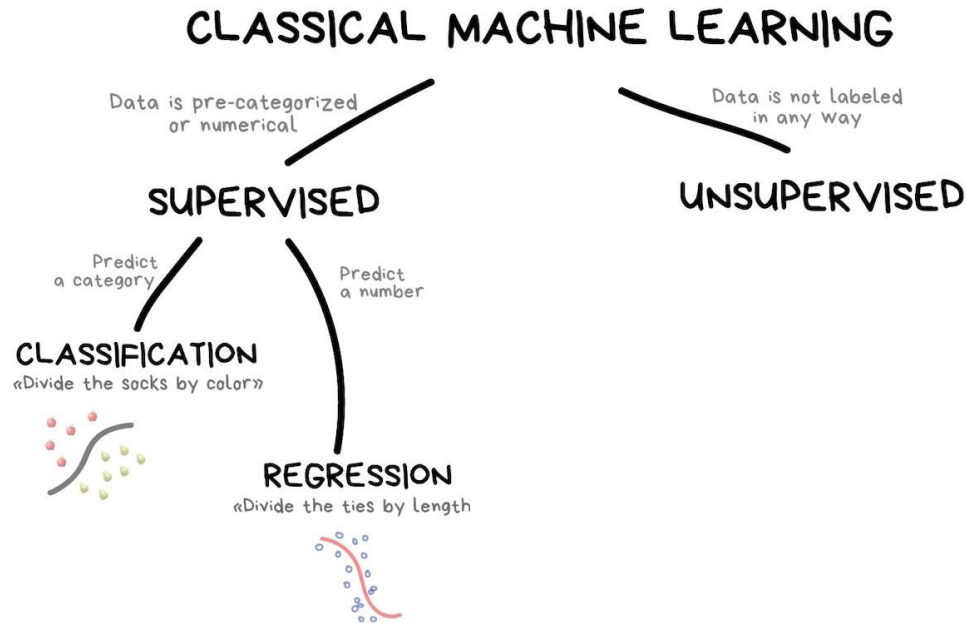
SUPERVISED

Data is not labeled
in any way

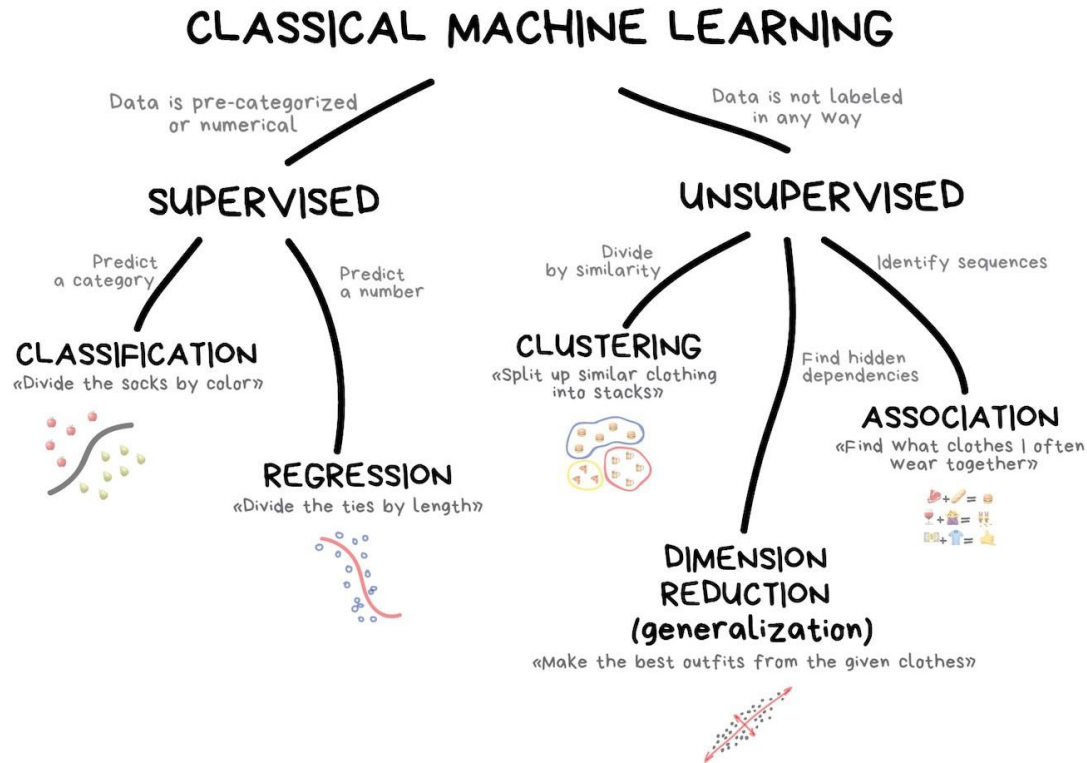
UNSUPERVISED

- ▶ Partir d'un ensemble connu de données d'entrée et de réponses connues à ces données (étiquettes)
- ▶ Construire un modèle pour prédire la réponse à de nouvelles données

Différents types de *machine learning*



Différents types de *machine learning*





Machine learning

Etape 2 : acquisition des données
d'apprentissage et de test

Le *machine learning* : principes

- ▶ Le *machine learning* est utile lorsque :
 - ▶ Tâche complexe ou grande quantité de données, mais pas de modèle prédéterminé

Le *machine learning* : principes

- ▶ Le *machine learning* est utile lorsque :
 - ▶ Tâche complexe ou grande quantité de données, mais pas de modèle prédéterminé
 - ▶ L'ordinateur apprend de son expérience pour trouver des patterns dans les données

Le *machine learning* : principes

- ▶ Le *machine learning* est utile lorsque :
 - ▶ Tâche complexe ou grande quantité de données, mais pas de modèle prédéterminé
 - ▶ L'ordinateur apprend de son expérience pour trouver des patterns dans les données

- ▶ Classification de données : classe les données fournies en catégories pré-déterminées

Le *machine learning* : principes

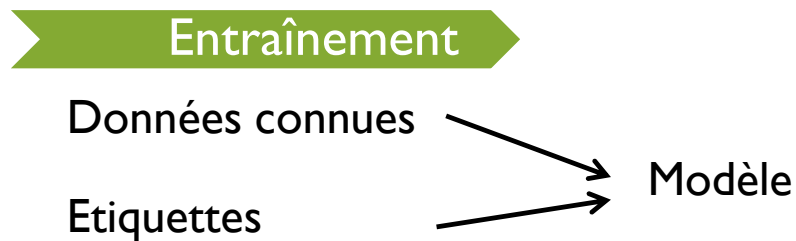
- ▶ Le *machine learning* est utile lorsque :
 - ▶ Tâche complexe ou grande quantité de données, mais pas de modèle prédéterminé
 - ▶ L'ordinateur apprend de son expérience pour trouver des patterns dans les données
- ▶ Classification de données : classe les données fournies en catégories pré-déterminées

Entraînement



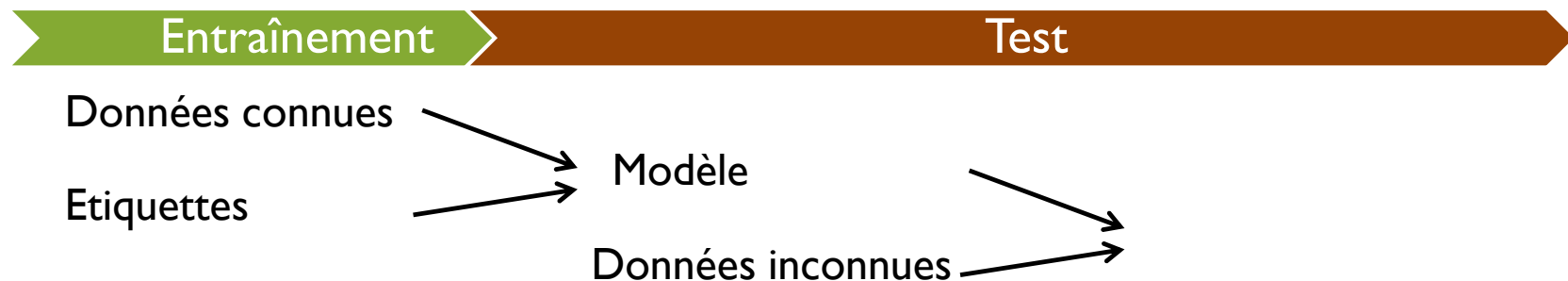
Le *machine learning* : principes

- ▶ Le *machine learning* est utile lorsque :
 - ▶ Tâche complexe ou grande quantité de données, mais pas de modèle prédéterminé
 - ▶ L'ordinateur apprend de son expérience pour trouver des patterns dans les données
- ▶ Classification de données : classe les données fournies en catégories pré-déterminées



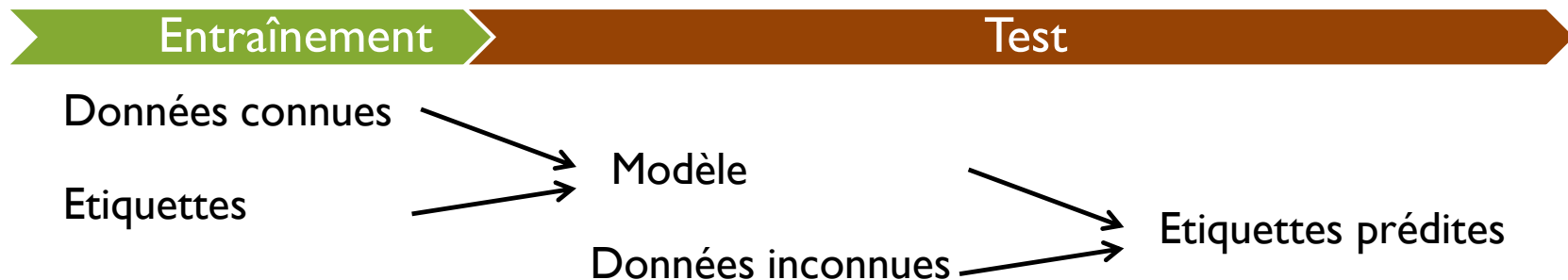
Le *machine learning* : principes

- ▶ Le *machine learning* est utile lorsque :
 - ▶ Tâche complexe ou grande quantité de données, mais pas de modèle prédéterminé
 - ▶ L'ordinateur apprend de son expérience pour trouver des patterns dans les données
- ▶ Classification de données : classe les données fournies en catégories pré-déterminées



Le *machine learning* : principes

- ▶ Le *machine learning* est utile lorsque :
 - ▶ Tâche complexe ou grande quantité de données, mais pas de modèle prédéterminé
 - ▶ L'ordinateur apprend de son expérience pour trouver des patterns dans les données
- ▶ Classification de données : classe les données fournies en catégories pré-déterminées



Nécessité d'une grande quantité de données

- ▶ Capacités du modèle dépendent de la taille de l'échantillon d'entraînement
 - ▶ nombre *features* augmente => nombre essais nécessaire augmente

Nécessité d'une grande quantité de données

- ▶ Capacités du modèle dépendent de la taille de l'échantillon d'entraînement
 - ▶ nombre *features* augmente => nombre essais nécessaire augmente
- ▶ Classification :
 - ▶ Nécessité de 2^d essais

Nécessité d'une grande quantité de données

- ▶ Capacités du modèle dépendent de la taille de l'échantillon d'entraînement
 - ▶ nombre *features* augmente => nombre essais nécessaire augmente
- ▶ Classification :
 - ▶ Nécessité de 2^d essais
- ▶ Difficulté :
 - ▶ Données complexes => nécessité grand nombre de *features*
 - ▶ Grand nombre de *features* => nécessité grand nombre de données
 - ▶ => difficile à obtenir en médecine (fatigue du sujet, contraintes techniques etc...)

Similarités de classes

- ▶ Pour discriminer au mieux les essais :
- ▶ Homogénéité intra-classe :
 - ▶ Essais classe A très semblables entre eux
 - ▶ Essais classe B très semblables entre eux

Similarités de classes

- ▶ Pour discriminer au mieux les essais :
- ▶ Homogénéité intra-classe :
 - ▶ Essais classe A très semblables entre eux
 - ▶ Essais classe B très semblables entre eux
- ▶ Hétérogénéité inter-classes :
 - ▶ Essais A et B très différents

Similarités de classes

- ▶ Pour discriminer au mieux les essais :
- ▶ Homogénéité intra-classe :
 - ▶ Essais classe A très semblables entre eux
 - ▶ Essais classe B très semblables entre eux
- ▶ Hétérogénéité inter-classes :
 - ▶ Essais A et B très différents
- ▶ Sinon => augmentation probabilité essais mal classés

Equilibre de classes

- ▶ Avec deux catégories, probabilité de classer correctement au hasard = 50%



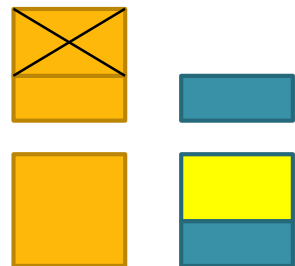
Equilibre de classes

- ▶ Avec deux catégories, probabilité de classer correctement au hasard = 50%
- ▶ Si quantité données classe A \ll classe B
 - ▶ Classificateur devient plus sensible à détection classe majoritaire
 - ▶ Exemple : sur 1000 personnes à l'aéroport, 2 sont des terroristes
 - ▶ Il suffit au classificateur d'étiqueter tout le monde comme « non terroriste » pour obtenir 99,8% de bonnes réponses !



Equilibre de classes

- ▶ Avec deux catégories, probabilité de classer correctement au hasard = 50%
- ▶ Si quantité données classe A \ll classe B
 - ▶ Classificateur devient plus sensible à détection classe majoritaire
 - ▶ Exemple : sur 1000 personnes à l'aéroport, 2 sont des terroristes
 - ▶ Il suffit au classificateur d'étiqueter tout le monde comme « non terroriste » pour obtenir 99,8% de bonnes réponses !
- ▶ 3 possibilités :
 - ▶ Ignorer le problème : acceptable si écart faible
 - ▶ Sous-échantillonner : éliminer des essais de la classe majoritaire
 - ▶ Sur-échantillonner : création d'essais synthétiques de la classe minoritaire





Machine learning

Etape 3 : nettoyage des données
et extraction de *features*

Nettoyage des données

- ▶ Les données recueillies sont souvent :
 - ▶ Bruitées : artéfacts en EEG, bruit blanc dans le signal sonore, etc...
 - ▶ Incomplètes : données manquantes, perdus de vue dans les BDD patients, etc...

Nettoyage des données

- ▶ Les données recueillies sont souvent :
 - ▶ Bruitées : artéfacts en EEG, bruit blanc dans le signal sonore, etc...
 - ▶ Incomplètes : données manquantes, perdus de vue dans les BDD patients, etc...
- ▶ Nécessité absolue de nettoyer la base de données au préalable
 - ▶ Voir méthodes recommandées en fonction du type de données

Extraction de *features*

- ▶ But : sélectionner infos :
 - ▶ Pertinentes
 - ▶ Donnent un caractère distinctif aux classes

Extraction de *features*

- ▶ But : sélectionner infos :
 - ▶ Pertinentes
 - ▶ Donnent un caractère distinctif aux classes
- ▶ Le choix de ces caractéristiques destinées à représenter le signal est **complexe** et **crucial** :
 - ▶ Des *features* non pertinentes ajoutent du bruit et peuvent fausser la classification
 - ▶ Si des *features* pertinentes sont absentes, il ne reste que du bruit à classer

Extraction de *features*

- ▶ But : sélectionner infos :
 - ▶ Pertinentes
 - ▶ Donnent un caractère distinctif aux classes
- ▶ Le choix de ces caractéristiques destinées à représenter le signal est **complexe** et **crucial** :
 - ▶ Des *features* non pertinentes ajoutent du bruit et peuvent fausser la classification
 - ▶ Si des *features* pertinentes sont absentes, il ne reste que du bruit à classer
 - ▶ Influe directement sur la performance de l'algorithme : « *garbage in, garbage out* »
 - ▶ Dépend du type d'info recherché, du protocole d'acquisition des données etc...

Extraction de *features*

- ▶ Les *features* sont des éléments extraits des données permettant de distinguer les 2 catégories souhaitées

Extraction de *features*

- ▶ Les *features* sont des éléments extraits des données permettant de distinguer les 2 catégories souhaitées



Extraction de *features*

- ▶ Les *features* sont des éléments extraits des données permettant de distinguer les 2 catégories souhaitées
- ▶ Certaines sont pertinentes, d'autres moins

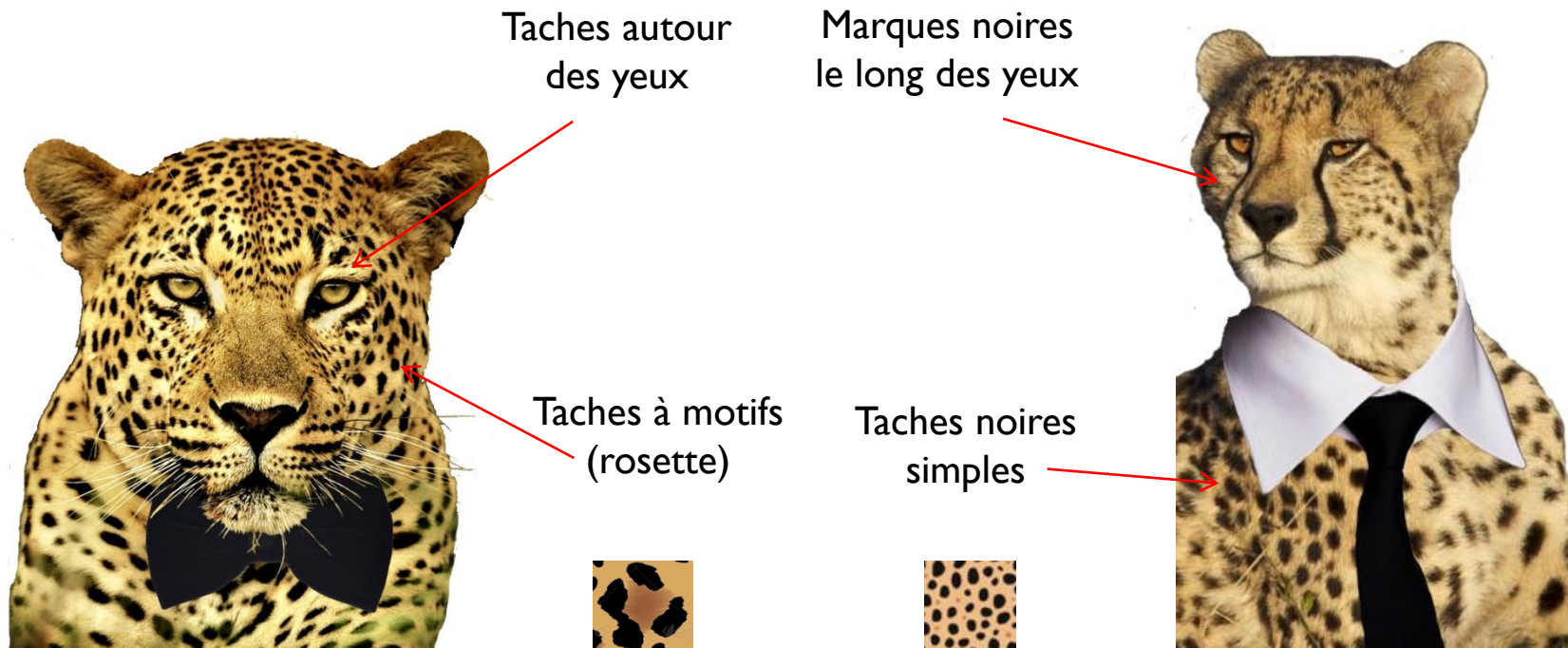


- ▶ Couleur du pelage ?
- ▶ Couleur des yeux ?
- ▶ Taille des oreilles ?
- ▶ Forme du museau ?
- ▶ ...



Extraction de *features*

- ▶ Les *features* sont des éléments extraits des données permettant de distinguer les 2 catégories souhaitées



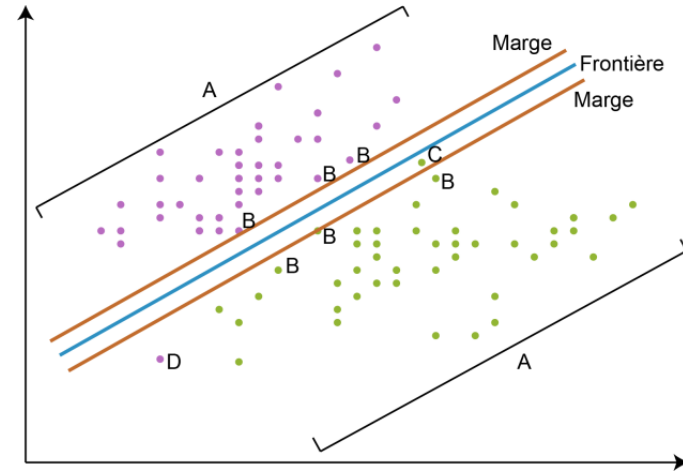


Machine learning

Etape 4 : évaluation de la performance
et ajustement du modèle

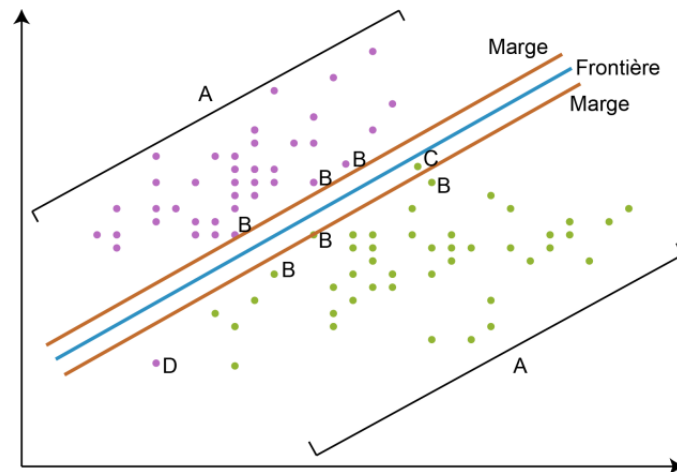
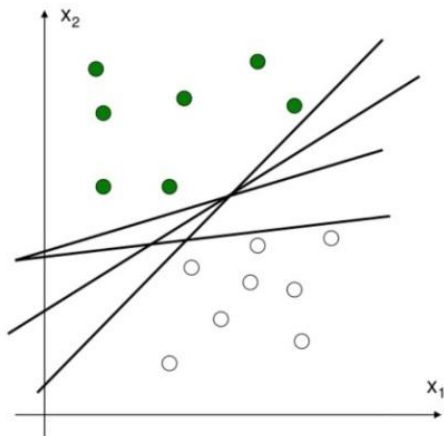
Fonctionnement général du SVM

- ▶ Algorithme d'apprentissage pour la discrimination dichotomique
- ▶ Principe : trouver la frontière de décision séparant tous les essais d'entraînement



Fonctionnement général du SVM

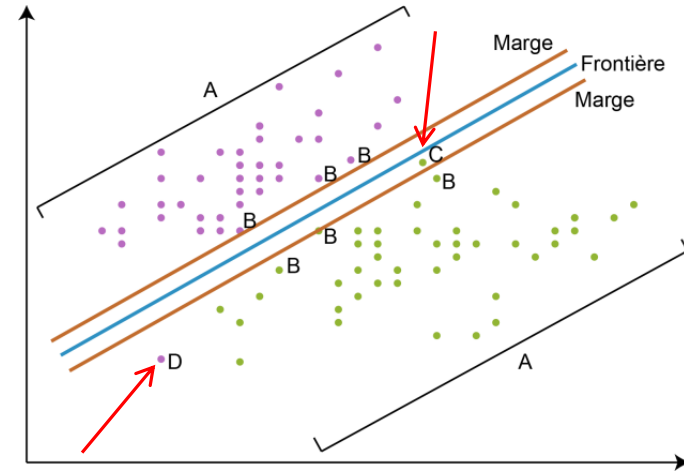
- ▶ Algorithme d'apprentissage pour la discrimination dichotomique
- ▶ Principe : trouver la frontière de décision séparant tous les essais d'entraînement



- ▶ Quelle est la meilleure séparation ?
 - ▶ Marge = distance entre la frontière et les essais les plus proches (vecteurs supports)
 - ▶ La frontière de décision optimale est celle qui maximise cette marge
=> capacité de généralisation la plus grande possible

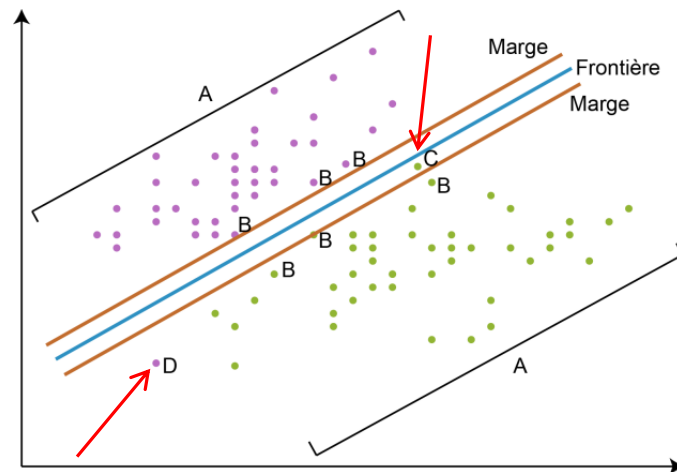
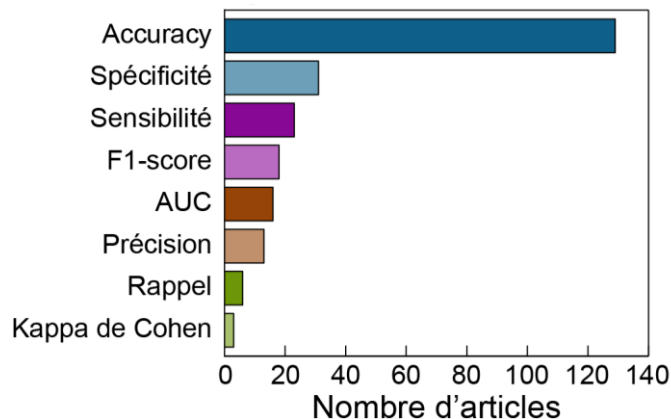
Mesures de performance

- ▶ Une frontière est rarement parfaite :
 - ▶ Des essais peuvent être mal classés,
 - ▶ Il est donc nécessaire de quantifier la performance du modèle établi



Mesures de performance

- ▶ Une frontière est rarement parfaite :
 - ▶ Des essais peuvent être mal classés,
 - ▶ Il est donc nécessaire de quantifier la performance du modèle établi



- ▶ La métrique la plus utilisée est l'*accuracy* :
 - ▶ Pourcentage d'essais correctement classés, par rapport au nombre total d'essais utilisés pour le test du modèle

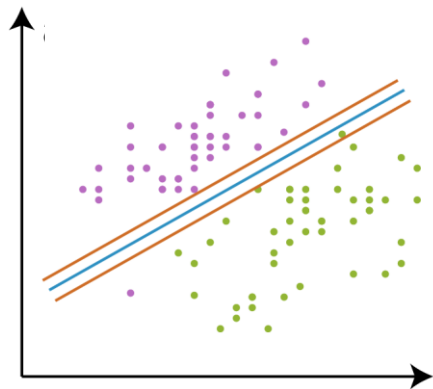
Séparation des données

- ▶ Les données ne sont pas toujours séparables par une ligne
- ▶ Il est donc intéressant de recourir à des fonctions *kernel* pour obtenir d'autres formes de frontières

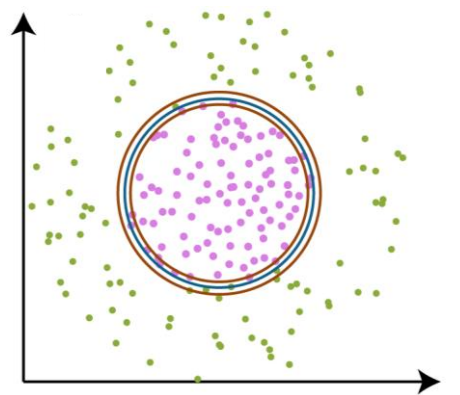
Séparation des données

- ▶ Les données ne sont pas toujours séparables par une ligne
- ▶ Il est donc intéressant de recourir à des fonctions *kernel* pour obtenir d'autres formes de frontières

Kernel linéaire

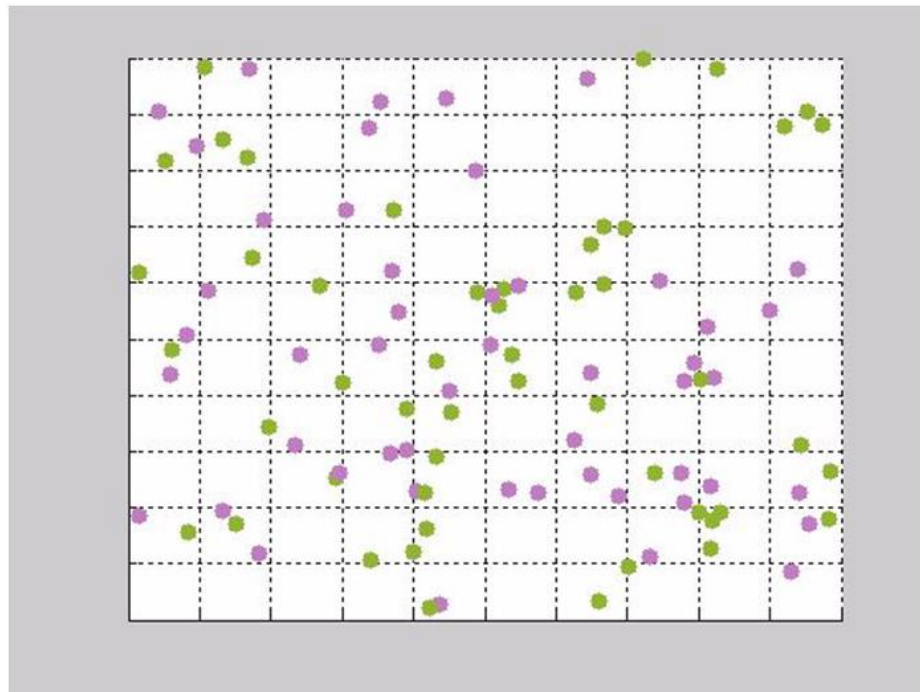


Kernel RBF



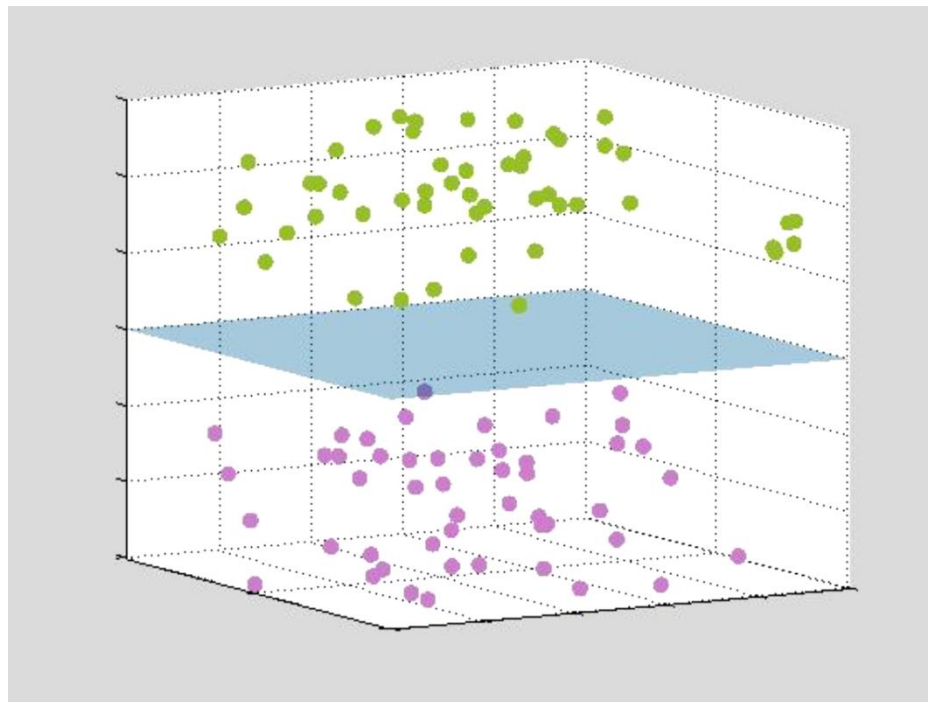
Kernels : projection 2D->3D

- ▶ Chaque essai est représenté par des *features*, qui forment des coordonnées dans un espace à n dimensions
 - ▶ Exemple : 2 *features* = 2 dimensions, observables sur un plan.



Kernels : projection 2D->3D

- ▶ Chaque essai est représenté par des *features*, qui forment des coordonnées dans un espace à n dimensions
 - ▶ Exemple : 2 *features* = 2 dimensions, observables sur un plan.
- ▶ S'il n'est pas possible de séparer les données sur ces n dimensions, le *kernel* va être capable de les projeter sur $n+1$ dimensions
 - ▶ Exemple : projection de 2D à 3D
- ▶ Il peut être possible de séparer les données dans ce nouvel espace
- ▶ **Exemple : kernel linéaire**

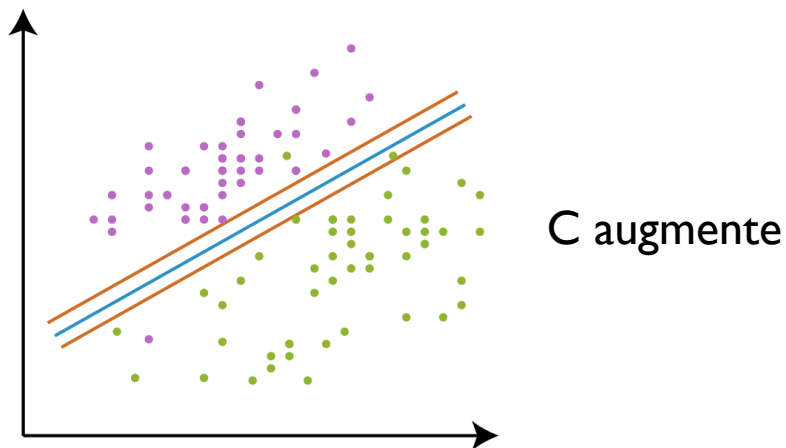
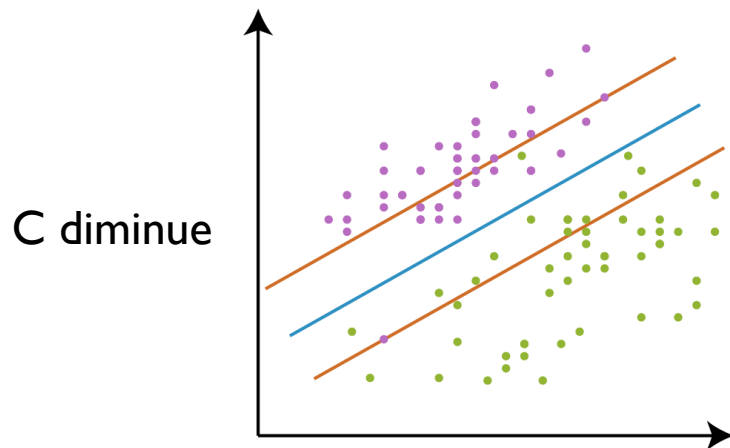


Des hyperparamètres régulent les *kernels*

- ▶ Hyperparamètre C : permet de faire un compromis entre une classification parfaite des essais d'entraînement et la simplicité du modèle créé.
 - ▶ Présent dans les *kernels* linéaire et RBF

Des hyperparamètres régulent les *kernels*

- ▶ Hyperparamètre C : permet de faire un compromis entre une classification parfaite des essais d'entraînement et la simplicité du modèle créé.
 - ▶ Présent dans les *kernels* linéaire et RBF
 - ▶ C diminue \Rightarrow marge plus permissive \Rightarrow classification à l'entraînement moins bonne
 - ▶ Mais frontière plus souple \Rightarrow meilleure capacité de généralisation sur les essais inconnus

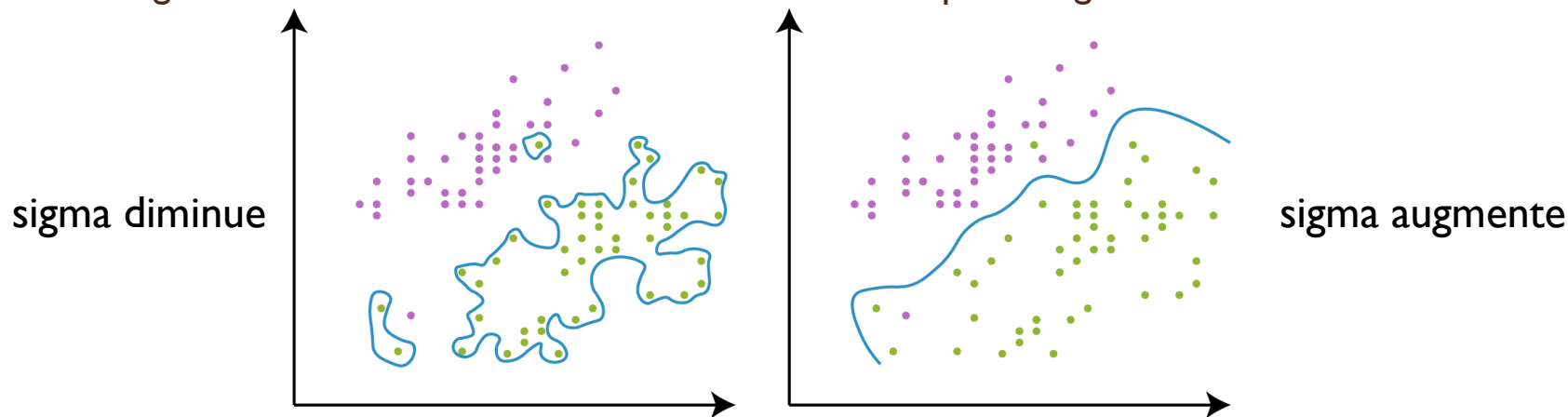


Des hyperparamètres régulent les *kernels*

- ▶ Hyperparamètre C : permet de faire un compromis entre une classification parfaite des essais d'entraînement et la simplicité du modèle créé.
- ▶ Hyperparamètre σ : rayon de la zone d'influence de la frontière
 - ▶ Présent dans le *kernel* RBF seulement

Des hyperparamètres régulent les *kernels*

- ▶ Hyperparamètre C : permet de faire un compromis entre une classification parfaite des essais d'entraînement et la simplicité du modèle créé.
- ▶ Hyperparamètre sigma : rayon de la zone d'influence de la frontière
 - ▶ Présent dans le *kernel* RBF seulement
 - ▶ sigma diminue => moindre influence sur les essais plus éloignés de la frontière

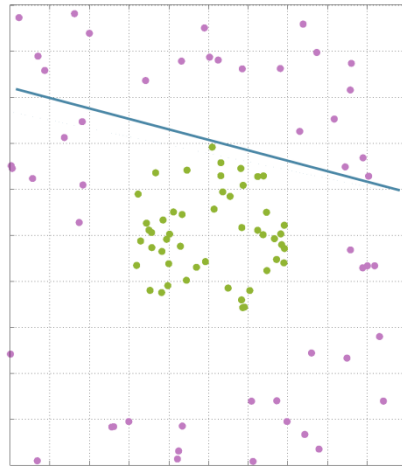
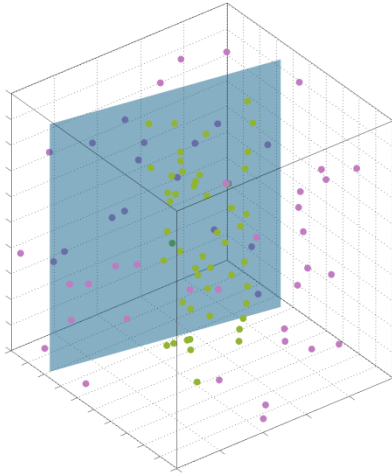


Des hyperparamètres régulent les *kernels*

- ▶ Hyperparamètre C : permet de faire un compromis entre une classification parfaite des essais d'entraînement et la simplicité du modèle créé.
- ▶ Hyperparamètre sigma : rayon de la zone d'influence de la frontière
- ▶ Le réglage de ces hyperparamètres est **crucial** car ils influencent fortement la performance du modèle

Exemple de l'influence des hyperparamètres

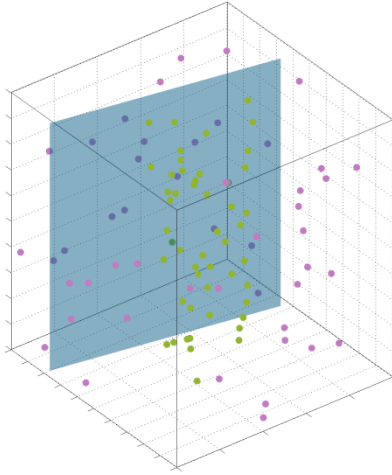
Kernel RBF



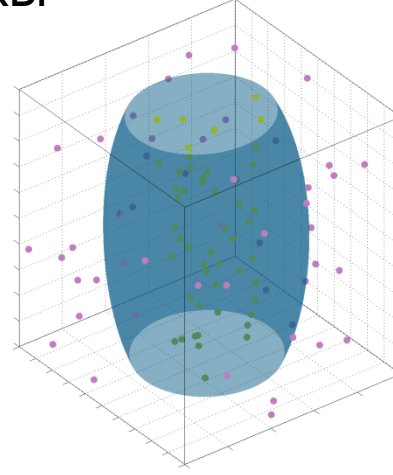
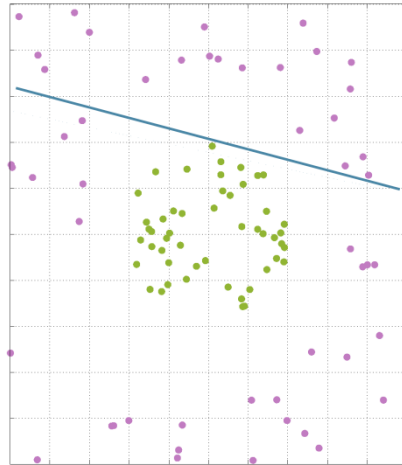
$C = 1, \text{sigma} = 1$

Exemple de l'influence des hyperparamètres

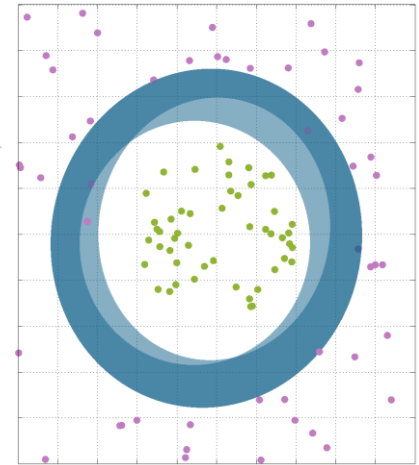
Kernel RBF



$C = 1, \sigma = 1$



$C = 1, \sigma = 10$



Validation de la performance

- ▶ Nécessité de mesurer la performance de généralisation = perf sur données inconnues
 - ▶ Division des données : ensemble d'entraînement + ensemble de test

Validation de la performance

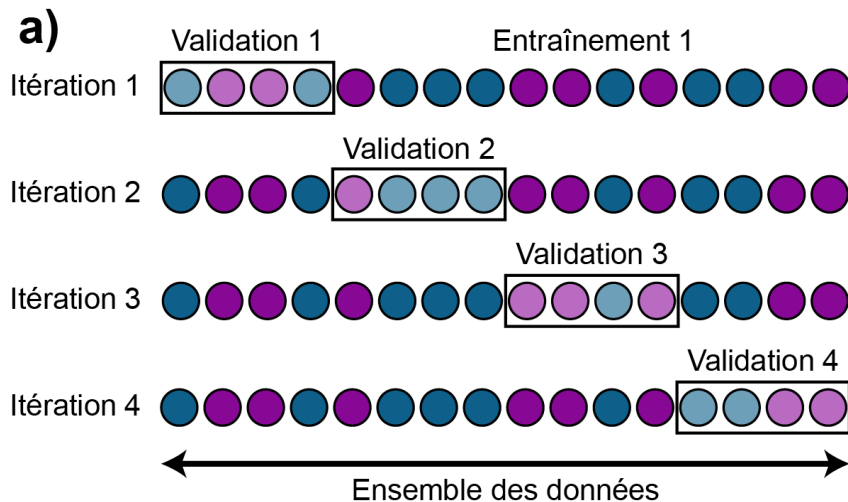
- ▶ Nécessité de mesurer la performance de généralisation = perf sur données inconnues
 - ▶ Division des données : ensemble d'entraînement + ensemble de test
- ▶ Optimisation des hyperparamètres => division des données en 3 sets :
 - ▶ Ensemble d'entraînement + ensemble de validation pour optimiser les hyperparamètres
 - ▶ Ensemble de test pour évaluer la performance sur les paramètres obtenus

Validation de la performance

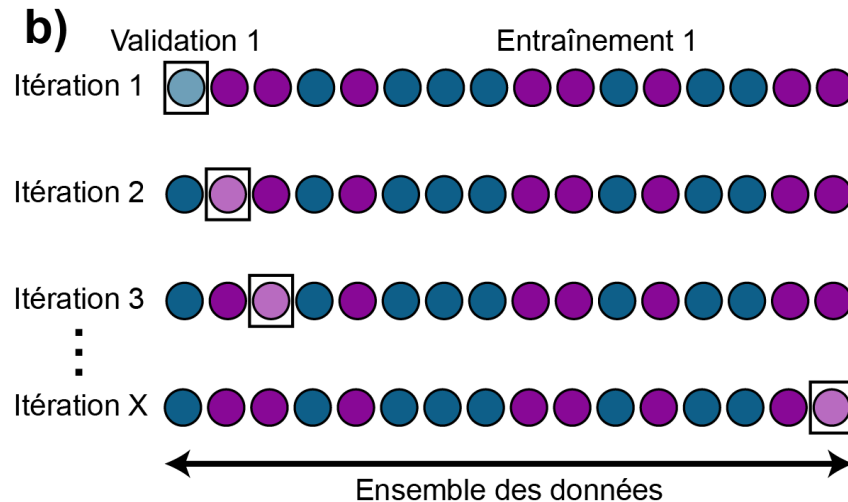
- ▶ Nécessité de mesurer la performance de généralisation = perf sur données inconnues
 - ▶ Division des données : ensemble d'entraînement + ensemble de test
- ▶ Optimisation des hyperparamètres => division des données en 3 sets :
 - ▶ Ensemble d'entraînement + ensemble de validation pour optimiser les hyperparamètres
 - ▶ Ensemble de test pour évaluer la performance sur les paramètres obtenus
- ▶ Procédure de validation croisée (cross-validation)
 - ▶ Problème : faible quantité de données => division nuit à performance
 - ▶ Utilisation des partitions comme ensemble d'entraînement ou de test successivement

Validation croisée

► K-fold

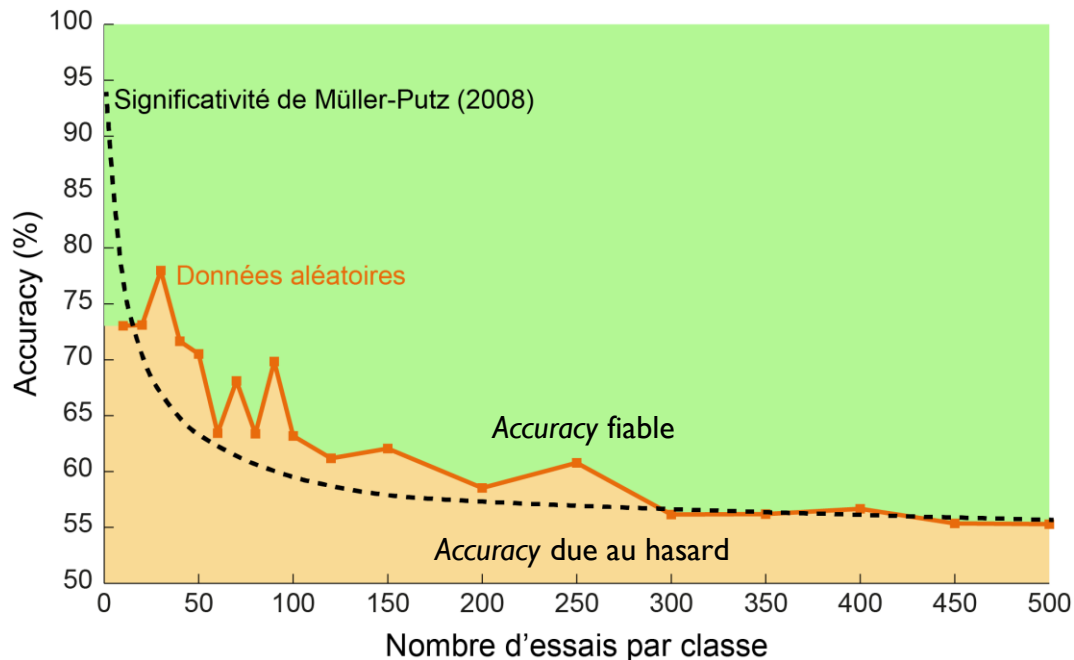


► Leave-one-out



Classification de données aléatoires

- ▶ Test du pipeline sur des données générées aléatoirement
- ▶ Comparaison des performances après optimisation en fonction du nombre d'essais par classe



Que faire des résultats ?

Interprétation des chiffres obtenus

Interprétation et remise en contexte

- ▶ De manière générale, le SVM est une méthode statistique qui donne un score
 - ▶ Il faut l'interpréter et le remettre en contexte car il ne suffit pas
 - ▶ Or cet effort n'est pas toujours fait dans les études publiées

Interprétation et remise en contexte

- ▶ De manière générale, le SVM est une méthode statistique qui donne un score
 - ▶ Il faut l'interpréter et le remettre en contexte car il ne suffit pas
 - ▶ Or cet effort n'est pas toujours fait dans les études publiées
- ▶ Il est nécessaire ici de fixer un positionnement de travail :
 - ▶ Soit donner la priorité à la performance
 - ▶ Soit donner la priorité à la robustesse et à la fiabilité

Interprétation et remise en contexte

- ▶ De manière générale, le SVM est une méthode statistique qui donne un score
 - ▶ Il faut l'interpréter et le remettre en contexte car il ne suffit pas
 - ▶ Or cet effort n'est pas toujours fait dans les études publiées
- ▶ Il est nécessaire ici de fixer un positionnement de travail :
 - ▶ Soit donner la priorité à la performance
 - ▶ Soit donner la priorité à la robustesse et à la fiabilité
- ▶ Ceci pose notamment la question de la reproductibilité de ces études :
 - ▶ Manque de recul et de formation sur les algorithmes de classification
 - ▶ Manque de reproductibilité des études dû au manque de précision



IA et médecine

Applications, implications

Applications de l'IA en médecine

- ▶ Détection et le diagnostic des maladies
 - ▶ Monitoring permanent des patients
 - ▶ Modèle d'IA prédictif pour bébés prématurés => 75% détection septicémies graves

Applications de l'IA en médecine

- ▶ **Détection et le diagnostic des maladies**
 - ▶ Monitoring permanent des patients
 - ▶ Modèle d'IA prédictif pour bébés prématurés => 75% détection septicémies graves
- ▶ **Traitement personnalisé des maladies**
 - ▶ Recommandations personnalisées en temps réel aux patients, 24 heures sur 24
 - ▶ Réponses aux questions selon antécédents médicaux, préférences et besoins personnels du patient

Applications de l'IA en médecine

- ▶ **Détection et le diagnostic des maladies**
 - ▶ Monitoring permanent des patients
 - ▶ Modèle d'IA prédictif pour bébés prématurés => 75% détection septicémies graves
- ▶ **Traitement personnalisé des maladies**
 - ▶ Recommandations personnalisées en temps réel aux patients, 24 heures sur 24
 - ▶ Réponses aux questions selon antécédents médicaux, préférences et besoins personnels du patient
- ▶ **Imagerie médicale**
 - ▶ Aussi efficace que les radiologues humains pour détecter les signes du cancer du sein
 - ▶ Rend plus gérable le nombre stupéfiant d'images médicales que les cliniciens doivent analyser

Applications de l'IA en médecine

- ▶ **Détection et le diagnostic des maladies**
 - ▶ Monitoring permanent des patients
 - ▶ Modèle d'IA prédictif pour bébés prématurés => 75% détection septicémies graves
- ▶ **Traitement personnalisé des maladies**
 - ▶ Recommandations personnalisées en temps réel aux patients, 24 heures sur 24
 - ▶ Réponses aux questions selon antécédents médicaux, préférences et besoins personnels du patient
- ▶ **Imagerie médicale**
 - ▶ Aussi efficace que les radiologues humains pour détecter les signes du cancer du sein
 - ▶ Rend plus gérable le nombre stupéfiant d'images médicales que les cliniciens doivent analyser
- ▶ **Essais cliniques**
 - ▶ Attribution et recherche des codes patients accélérée
 - ▶ Mise à jour des données pertinentes

Limitations et régulation de l'IA

- ▶ Pas de directives universelles claires à l'heure actuelle

Limitations et régulation de l'IA

- ▶ Pas de directives universelles claires à l'heure actuelle
- ▶ Personnes qui créent les algorithmes à utiliser en clinique ne sont pas toujours les médecins qui traitent les patients
 - ▶ Les informaticiens devraient en savoir plus sur la médecine
 - ▶ Les cliniciens doivent apprendre pour quelles tâches un algorithme spécifique est ou n'est pas adapté

Limitations et régulation de l'IA

- ▶ Pas de directives universelles claires à l'heure actuelle
- ▶ Personnes qui créent les algorithmes à utiliser en clinique ne sont pas toujours les médecins qui traitent les patients
 - ▶ Les informaticiens devraient en savoir plus sur la médecine
 - ▶ Les cliniciens doivent apprendre pour quelles tâches un algorithme spécifique est ou n'est pas adapté
- ▶ Essais cliniques
 - ▶ Exigence de transparence extrême autour des méthodes scientifiques utilisées
 - ▶ Algorithmes souvent « boîtes noires »
 - ▶ Impossibilité de « débiller » cette boîte noire: impact sur acceptabilité de l'étude ?

Acceptabilité par les patients

- ▶ Si algorithme non transparent pour les informaticiens et pour les cliniciens
 - ▶ Doute du patient

Acceptabilité par les patients

- ▶ Si algorithme non transparent pour les informaticiens et pour les cliniciens
 - ▶ Doute du patient

- ▶ A choisir, préfère-t-on être mal diagnostiqué par un humain ou par un algorithme ?

Acceptabilité par les patients

- ▶ Si algorithme non transparent pour les informaticiens et pour les cliniciens
 - ▶ Doute du patient
- ▶ A choisir, préfère-t-on être mal diagnostiqué par un humain ou par un algorithme ?
- ▶ Quid si l'algorithme est généralement plus performant que l'humain ?

Acceptabilité par les patients

- ▶ Si algorithme non transparent pour les informaticiens et pour les cliniciens
 - ▶ Doute du patient
- ▶ A choisir, préfère-t-on être mal diagnostiqué par un humain ou par un algorithme ?
- ▶ Quid si l'algorithme est généralement plus performant que l'humain ?
- ▶ Sacrifice de la relation avec le médecin ?

Le mythe du remplacement du médecin

- ▶ Les médecins seront-ils remplacés par l'IA ?
 - ▶ Non, mais...
 - ▶ Ceux qui comprennent l'IA seront avantagés
 - ▶ => déjà une exigence courante chez les radiologues aux USA

Le mythe du remplacement du médecin

- ▶ Les médecins seront-ils remplacés par l'IA ?
 - ▶ Non, mais...
 - ▶ Ceux qui comprennent l'IA seront avantagés
 - ▶ => déjà une exigence courante chez les radiologues aux USA
- ▶ Vinod Khosla (Silicon Valley) : 80% des médecins seront remplacés
 - ▶ En réalité : transpositeurs médicaux, techniciens et secrétaires médicales sont susceptibles d'être informatisés
 - ▶ Mais seulement 0,42% de chances que médecins et chirurgiens soient remplacés

Place du *machine learning* en médecine

- ▶ Le *machine learning* a déjà fait ses preuves pour :
 - ▶ Facilitation et accélération tâches répétitives
 - ▶ Pallie variations intra et inter individuelles de traitement par les experts

Place du *machine learning* en médecine

- ▶ Le *machine learning* a déjà fait ses preuves pour :
 - ▶ Facilitation et accélération tâches répétitives
 - ▶ Pallie variations intra et inter individuelles de traitement par les experts

- ▶ Sans toutefois remplacer l'humain
 - ▶ Diagnostic nécessite une capacité de résolution de problème créative
 - ▶ Même exercice mais avec outils plus perfectionnés
 - ▶ Interprétation des données restera toujours un territoire humain

Optimisme du corps médical

- ▶ Sarwar 2019 : enquête auprès de 487 médecins dans 54 pays
 - ▶ 42,4% médecins : IA va créer de nouveaux postes et augmenter l'emploi
 - ▶ Accroissement efficacité diagnostic
 - ▶ 48,3% : diagnostic doit être effectué principalement par médecin
 - ▶ 25,3% : tâche peut être partagée à égalité IA/humain
- ▶ Consensus : transformation médecine
 - ▶ IA = outil supplémentaire dans le processus de diagnostic
 - ▶ Dégage du temps pour le médecin : renforcement relation soignant-soigné



Merci de votre attention !