

Apprentissage supervisé

EDOUARD PAUWELS

M2-MAT SID

Support de cours :

www.math.univ-toulouse.fr/~epauwels/LearningM2SID/

- Cours / TP,
- Amenez vos machines en cours.
- Libraries installées et fonctionnelle : python 3, jupyter, numpy, scikit-learn.
- Contrôle continu : un compte rendu de TP à rendre
- Contrôle terminal : devoir sur machine.

- Détecteur de spam
- Risque de crédit
- Prédiction des pics d'ozone
- Aide au diagnostic médical (ex : Breast Cancer)
- Aide au pilotage
- Moteurs de recommandation, publicité
- etc. . .

Qu'est-ce que l'apprentissage ?

Apprentissage (*machine learning*) = discipline visant à la construction de règles d'inférence et de décision pour le traitement automatique des données.

Variantes : machine learning, fouille de données (data-mining).

1 Apprentissage supervisé :

A partir d'un échantillon d'apprentissage $\mathcal{D}_n = \{(x_1, y_1), \dots, (x_n, y_n)\}$, inférer la relation entre x et y .

Synonymes : discrimination, reconnaissance de formes (pattern recognition)

Voc : x_i = caractéristique = feature = variable explicative

2 Apprentissage non supervisé :

A partir d'un échantillon d'apprentissage $\mathcal{D}_n = \{x_1, \dots, x_n\} \subset \mathcal{X}$, inférer des propriétés de \mathcal{X} , par exemple partitionner \mathcal{X} en classes pertinentes (clustering).

Voc : parfois appelé 'Classification' en français (jamais en anglais)

3 Apprentissage séquentiel :

A chaque date n , prendre une décision à l'aide des données passées.

1 Apprentissage supervisé :

A partir d'un échantillon d'apprentissage $\mathcal{D}_n = \{(x_1, y_1), \dots, (x_n, y_n)\}$, inférer la relation entre x et y .

Synonymes : discrimination, reconnaissance de formes (pattern recognition)

Voc : x_i = caractéristique = feature = variable explicative

2 Apprentissage non supervisé :

A partir d'un échantillon d'apprentissage $\mathcal{D}_n = \{x_1, \dots, x_n\} \subset \mathcal{X}$, inférer des propriétés de \mathcal{X} , par exemple partitionner \mathcal{X} en classes pertinentes (clustering).

Voc : parfois appelé 'Classification' en français (jamais en anglais)

3 Apprentissage séquentiel :

A chaque date n , prendre une décision à l'aide des données passées.

1 Apprentissage supervisé :

A partir d'un échantillon d'apprentissage $\mathcal{D}_n = \{(x_1, y_1), \dots, (x_n, y_n)\}$, inférer la relation entre x et y .

Synonymes : discrimination, reconnaissance de formes (pattern recognition)

Voc : x_i = caractéristique = feature = variable explicative

2 Apprentissage non supervisé :

A partir d'un échantillon d'apprentissage $\mathcal{D}_n = \{x_1, \dots, x_n\} \subset \mathcal{X}$, inférer des propriétés de \mathcal{X} , par exemple partitionner \mathcal{X} en classes pertinentes (clustering).

Voc : parfois appelé 'Classification' en français (jamais en anglais)

3 Apprentissage séquentiel :

A chaque date n , prendre une décision à l'aide des données passées.

- Approche structurelle, logique et logique floue.
- **Apprentissage statistique** : modélisation probabiliste des données à des fins *instrumentales*.
- Apprentissage séq. robuste (théorie des jeux, optim. convexe séq.).

Dans tous les cas :

- science relativement *récente*
- à la frontière des *mathématiques* et de l'*informatique* (et intelligence artificielle)
- en *évolution rapide et constante* avec les technologies =
 - ▶ nouveaux moyens (de calcul)
 - ▶ nouveaux problèmes...

- Approche structurelle, logique et logique floue.
- **Apprentissage statistique** : modélisation probabiliste des données à des fins *instrumentales*.
- Apprentissage séq. robuste (théorie des jeux, optim. convexe séq.).

Dans tous les cas :

- science relativement *récente*
- à la frontière des *mathématiques* et de l'*informatique* (et intelligence artificielle)
- en *évolution rapide et constante* avec les technologies =
 - ▶ nouveaux moyens (de calcul)
 - ▶ nouveaux problèmes...

A mettre en place après une étude préliminaire qualitative : allure des distributions (graphiques), présence de données atypiques, corrélations et cohérence, transformations éventuelles des données (normalisation), description multidimensionnelle (PCA), classification (clustering).

Cadre classique (batch) :

Données : *échantillon d'apprentissage* $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathcal{Y}$, constitué d'observations que l'on suppose représentatives et sans lien entre elles.

Objectif : prédire les valeurs de $y \in \mathcal{Y}$ associées à chaque valeur possible de $x \in \mathcal{X}$.

Classification : \mathcal{Y} discret (typiquement, binaire) pour chaque valeur de $x \in \mathcal{X}$, il faut prédire la classe la plus souvent associée.

Régression : \mathcal{Y} continu, voire plus (fonctionnel).

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathcal{Y}$ associant, à chaque entrée possible x une valeur de y prédite.

A mettre en place après une étude préliminaire qualitative : allure des distributions (graphiques), présence de données atypiques, corrélations et cohérence, transformations éventuelles des données (normalisation), description multidimensionnelle (PCA), classification (clustering).

Cadre classique (batch) :

Données : échantillon d'apprentissage $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathcal{Y}$, constitué d'observations que l'on suppose représentatives et sans lien entre elles.

Objectif : prédire les valeurs de $y \in \mathcal{Y}$ associées à chaque valeur possible de $x \in \mathcal{X}$.

Classification : \mathcal{Y} discret (typiquement, binaire) pour chaque valeur de $x \in \mathcal{X}$, il faut prédire la classe la plus souvent associée.

Régression : \mathcal{Y} continu, voire plus (fonctionnel).

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathcal{Y}$ associant, à chaque entrée possible x une valeur de y prédite.

Modèle linéaire gaussien : $y_k = \beta^T x_k + \epsilon_k$, $k = 1, \dots, n$. Hypothèses ? $(\epsilon_k)_{k=1}^n$

- Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

- Seule hypothèse : P existe. Pas caractérisé.
- Pas de "vrai modèle", de "vraies valeurs du paramètre".
On peut caler un modèle linéaire en dehors du cadre gaussien.
Qu'aura on en moins ?.
- Souvent données disponibles avant intervention du statisticien (malheureusement)
- Tous les coups sont permis, seul critère = efficacité prédictive.
- Classification : $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [P_{(X, Y)}(f_n(X) \neq Y)]$.
- Régression : typiquement, $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [E_{(X, Y)} ((Y - f_n(X))^2)]$.

Modèle linéaire gaussien : $y_k = \beta^T x_k + \epsilon_k$, $k = 1, \dots, n$. Hypothèses ? $(\epsilon_k)_{k=1}^n$

- Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

- Seule hypothèse : P existe. Pas caractérisé.
- Pas de "vrai modèle", de "vraies valeurs du paramètre".
On peut caler un modèle linéaire en dehors du cadre gaussien.
Qu'aura on en moins ?.
- Souvent données disponibles avant intervention du statisticien (malheureusement)
- Tous les coups sont permis, seul critère = efficacité prédictive.
- Classification : $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [P_{(X, Y)}(f_n(X) \neq Y)]$.
- Régression : typiquement, $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [E_{(X, Y)} ((Y - f_n(X))^2)]$.

Modèle linéaire gaussien : $y_k = \beta^T x_k + \epsilon_k$, $k = 1, \dots, n$. Hypothèses ? $(\epsilon_k)_{k=1}^n$ i.i.d. Gaussien.

- Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

- Seule hypothèse : P existe. Pas caractérisé.
- Pas de "vrai modèle", de "vraies valeurs du paramètre".
On peut caler un modèle linéaire en dehors du cadre gaussien.
Qu'aura on en moins ?
- Souvent données disponibles avant intervention du statisticien (malheureusement)
- Tous les coups sont permis, seul critère = efficacité prédictive.
- Classification : $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [P_{(X, Y)}(f_n(X) \neq Y)]$.
- Régression : typiquement, $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [E_{(X, Y)} ((Y - f_n(X))^2)]$.

Modèle linéaire gaussien : $y_k = \beta^T x_k + \epsilon_k$, $k = 1, \dots, n$. Hypothèses ? $(\epsilon_k)_{k=1}^n$ i.i.d. Gaussien.

- Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

- Seule hypothèse : P existe. Pas caractérisé.
- Pas de "vrai modèle", de "vraies valeurs du paramètre".
On peut caler un modèle linéaire en dehors du cadre gaussien.
Qu'aura on en moins ?.
- Souvent données disponibles avant intervention du statisticien (malheureusement)
- Tous les coups sont permis, seul critère = efficacité prédictive.
- Classification : $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [P_{(X, Y)}(f_n(X) \neq Y)]$.
- Régression : typiquement, $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [E_{(X, Y)} ((Y - f_n(X))^2)]$.

Modèle linéaire gaussien : $y_k = \beta^T x_k + \epsilon_k$, $k = 1, \dots, n$. Hypothèses ? $(\epsilon_k)_{k=1}^n$ i.i.d. Gaussien.

- Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

- Seule hypothèse : P existe. Pas caractérisé.
- Pas de "vrai modèle", de "vraies valeurs du paramètre".
On peut caler un modèle linéaire en dehors du cadre gaussien.
Qu'aura on en moins ?.
- Souvent données disponibles avant intervention du statisticien (malheureusement)
- Tous les coups sont permis, seul critère = efficacité prédictive.
- Classification : $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [P_{(X, Y)}(f_n(X) \neq Y)]$.
- Régression : typiquement, $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [E_{(X, Y)} ((Y - f_n(X))^2)]$.

Modèle linéaire gaussien : $y_k = \beta^T x_k + \epsilon_k$, $k = 1, \dots, n$. Hypothèses ? $(\epsilon_k)_{k=1}^n$ i.i.d. Gaussien.

- Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

- Seule hypothèse : P existe. Pas caractérisé.
- Pas de "vrai modèle", de "vraies valeurs du paramètre".
On peut caler un modèle linéaire en dehors du cadre gaussien.
Qu'aura on en moins ?
- Souvent données disponibles avant intervention du statisticien (malheureusement)
- Tous les coups sont permis, seul critère = efficacité prédictive.
- Classification : $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [P_{(X, Y)}(f_n(X) \neq Y)]$.
- Régression : typiquement, $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [E_{(X, Y)} ((Y - f_n(X))^2)]$.

Modèle linéaire gaussien : $y_k = \beta^T x_k + \epsilon_k$, $k = 1, \dots, n$. Hypothèses ? $(\epsilon_k)_{k=1}^n$ i.i.d. Gaussien.

- Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

- Seule hypothèse : P existe. Pas caractérisé.
- Pas de "vrai modèle", de "vraies valeurs du paramètre".
On peut caler un modèle linéaire en dehors du cadre gaussien.
Qu'aura on en moins ?
- Souvent données disponibles avant intervention du statisticien (malheureusement)
- Tous les coups sont permis, seul critère = efficacité prédictive.
- Classification : $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [P_{(X, Y)}(f_n(X) \neq Y)]$.
- Régression : typiquement, $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [E_{(X, Y)} ((Y - f_n(X))^2)]$.

Modèle linéaire gaussien : $y_k = \beta^T x_k + \epsilon_k$, $k = 1, \dots, n$. Hypothèses ? $(\epsilon_k)_{k=1}^n$ i.i.d. Gaussien.

- Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

- Seule hypothèse : P existe. Pas caractérisé.
- Pas de "vrai modèle", de "vraies valeurs du paramètre".
On peut caler un modèle linéaire en dehors du cadre gaussien.
Qu'aura on en moins ?.
- Souvent données disponibles avant intervention du statisticien (malheureusement)
- Tous les coups sont permis, seul critère = efficacité prédictive.
- Classification : $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [P_{(X, Y)}(f_n(X) \neq Y)]$.
- Régression : typiquement, $R_n = \mathbb{E}_{(x_1, y_1), \dots, (x_n, y_n)} [E_{(X, Y)} ((Y - f_n(X))^2)]$.

Modèle linéaire gaussien : $y_k = \beta^T x_k + \epsilon_k$, $k = 1, \dots, n$. Hypothèses ? $(\epsilon_k)_{k=1}^n$ i.i.d. Gaussien.

- Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

- Seule hypothèse : P existe. Pas caractérisé.
- Pas de "vrai modèle", de "vraies valeurs du paramètre".
On peut caler un modèle linéaire en dehors du cadre gaussien.
Qu'aura on en moins ?.
- Souvent données disponibles avant intervention du statisticien (malheureusement)
- Tous les coups sont permis, seul critère = efficacité prédictive.
- Classification : $R_n = \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} [P_{(X, Y)}(f_n(X) \neq Y)]$.
- Régression : typiquement, $R_n = \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} [E_{(X, Y)}((Y - f_n(X))^2)]$.

- Théoriquement, quand un modèle est vrai il est optimal de l'utiliser :
 - ▶ Théorème de Gauss-Markov : parmi les estimateurs sans biais, celui des moindres carrés est de variance minimale
 - ▶ MAIS on peut avoir intérêt à sacrifier du biais contre de la variance !

⇒ Même quand il y en a un 'vrai' modèle, on n'a pas forcément intérêt à l'utiliser

- Des approches non-paramétriques peuvent avoir une efficacité proche :
 - ▶ cf Test de Student vs Mann-Whitney
 - ▶ exemple : k-NN versus régression polynomiale
- ... et ils sont beaucoup plus robustes !

- Théoriquement, quand un modèle est vrai il est optimal de l'utiliser :
 - ▶ Théorème de Gauss-Markov : parmi les estimateurs sans biais, celui des moindres carrés est de variance minimale
 - ▶ MAIS on peut avoir intérêt à sacrifier du biais contre de la variance !

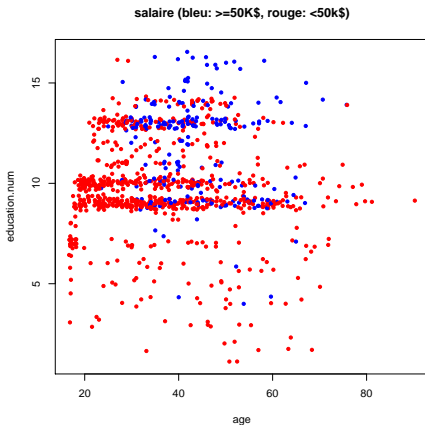
⇒ Même quand il y en a un 'vrai' modèle, on n'a pas forcément intérêt à l'utiliser

- Des approches non-paramétriques peuvent avoir une efficacité proche :
 - ▶ cf Test de Student vs Mann-Whitney
 - ▶ exemple : k-NN versus régression polynomiale
- ... et ils sont beaucoup plus robustes !

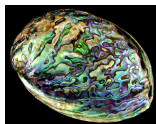
Exemple de problème de classification



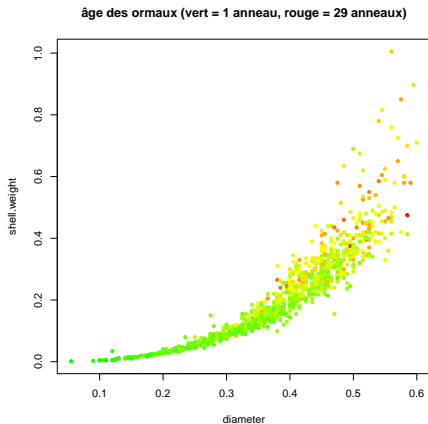
Objectif : prédire qui gagne plus de 50k\$ à partir de données de recensement.



Exemple de problème de régression



Prédire l'âge d'un ormeau (abalone) à partir de sa taille, son poids, etc.





Pattern Classification (2001) - Wiley Interscience, *R. Duda, P. Hart, D. Stork*



The Elements of Statistical Learning (2001) - Springer, *T. Hastie, R. Tibshirani, J. Friedman*
Disponible en ligne : <http://www-stat.stanford.edu/~tibs/ElemStatLearn/>



Data Mining - Technip, *S. Tufféry*



Cours en ligne de Andrew Ng (Stanford) :
<https://www.coursera.org/course/ml>





<http://wikistat.fr/>



Base de données de benchmarking :
<http://archive.ics.uci.edu/ml/>

Référence :

-  <http://cran.r-project.org/>
- The R Project for Statistical Computing
-  <http://scikit-learn.org/stable/>
- Machine learning in Python.
- Avantages : libres, ouverts, bien documentés, complets
- Inconvénients : pas 'presse-bouton', rapidité (MAIS extensions en C possibles !)
- *Aide en ligne* + google indispensables : logiciels vivant !

Alternatives :

- Tous les Data Managers s'y mettent (SAS, Oracle, IBM Dataminer...)
- Quelques outils dédiés faciles à utiliser, par exemple See5/C5.0
<http://www.rulequest.com/see5-info.html>

Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

P inconnu, comment faire ?

- *modèles génératifs* (spécifier P) et manipuler des *données simulées*.
- Illustrer, comprendre ce qu'on peut des concepts statistiques liés à l'apprentissage.
- Avantage : calculer la règle de décision optimale, estimer des taux d'erreurs facilement (jeu de données infinies).

Estimation par Monte-Carlo, loi des grands nombres :

Soit P une mesure de probabilité sur \mathbb{R}^p et $(X_i)_{i \in \mathbb{N}}$ une suite de variables aléatoires indépendantes et identiquement distribuées de loi P . Alors pour tout f fonction continue sur \mathbb{R}^p (plus généralement mesurable), pourvu que l'espérance existe

$$\frac{1}{n} \sum_{i=1}^n f(X_i) \xrightarrow[p.s.]{} \mathbb{E}_{X \sim P}[f(X)] = \int_{x \in \mathbb{R}^p} f(x) dP(x)$$

Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

P inconnu, comment faire ?

- *modèles génératifs* (spécifier P) et manipuler des *données simulées*.
- Illustrer, comprendre ce qu'on peut des concepts statistiques liés à l'apprentissage.
- Avantage : calculer la règle de décision optimale, estimer des taux d'erreurs facilement (jeu de données infinies).

Estimation par Monte-Carlo, loi des grands nombres :

Soit P une mesure de probabilité sur \mathbb{R}^p et $(X_i)_{i \in \mathbb{N}}$ une suite de variables aléatoires indépendantes et identiquement distribuées de loi P . Alors pour tout f fonction continue sur \mathbb{R}^p (plus généralement mesurable), pourvu que l'espérance existe

$$\frac{1}{n} \sum_{i=1}^n f(X_i) \xrightarrow[p.s.]{} \mathbb{E}_{X \sim P}[f(X)] = \int_{x \in \mathbb{R}^p} f(x) dP(x)$$

Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

P inconnu, comment faire ?

- modèles génératifs (spécifier P) et manipuler des *données simulées*.
- Illustrer, comprendre ce qu'on peut des concepts statistiques liés à l'apprentissage.
- Avantage : calculer la règle de décision optimale, estimer des taux d'erreurs facilement (jeu de données infinies).

Estimation par Monte-Carlo, loi des grands nombres :

Soit P une mesure de probabilité sur \mathbb{R}^p et $(X_i)_{i \in \mathbb{N}}$ une suite de variables aléatoires indépendantes et identiquement distribuées de loi P . Alors pour tout f fonction continue sur \mathbb{R}^p (plus généralement mesurable), pourvu que l'espérance existe

$$\frac{1}{n} \sum_{i=1}^n f(X_i) \xrightarrow[p.s.]{} \mathbb{E}_{X \sim P}[f(X)] = \int_{x \in \mathbb{R}^p} f(x) dP(x)$$

Les données $(x_k, y_k)_{k=1}^n$ sont des réalisations de v.a. iid de même loi que

$$(X, Y) \sim P_{(X, Y)}$$

P inconnu, comment faire ?

- *modèles génératifs* (spécifier P) et manipuler des *données simulées*.
- Illustrer, comprendre ce qu'on peut des concepts statistiques liés à l'apprentissage.
- Avantage : calculer la règle de décision optimale, estimer des taux d'erreurs facilement (jeu de données infinies).

Estimation par Monte-Carlo, loi des grands nombres :

Soit P une mesure de probabilité sur \mathbb{R}^p et $(X_i)_{i \in \mathbb{N}}$ une suite de variables aléatoires indépendantes et identiquement distribuées de loi P . Alors pour tout f fonction continue sur \mathbb{R}^p (plus généralement mesurable), pourvu que l'espérance existe

$$\frac{1}{n} \sum_{i=1}^n f(X_i) \xrightarrow[p.s.]{} \mathbb{E}_{X \sim P}[f(X)] = \int_{x \in \mathbb{R}^p} f(x) dP(x)$$

- Plus ou moins adaptée au problème, nature des données, relation entre descripteurs et variable expliquée...
- Apprendre les qualités et les défauts de chaque méthode.
- Apprendre à expérimenter pour identifier la plus pertinentes pour un problème donné.
- Estimer de la performances des méthodes est central (mais pas toujours évident).

1. Qu'est-ce que l'apprentissage supervisé ?
2. Compromis biais-variance
3. Evaluation et selection de modèle
4. Aggrégation de modèles et méthodes d'ensembles

1. Qu'est-ce que l'apprentissage supervisé ?
2. Compromis biais-variance
3. Evaluation et selection de modèle
4. Aggrégation de modèles et méthodes d'ensembles

Données : *échantillon d'apprentissage* $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathcal{Y}$, constitué d'observations que l'on suppose indépendantes et identiquement distribuées selon la loi $P_{\mathcal{X}, \mathcal{Y}}$ sur $\mathcal{X} \times \mathcal{Y}$.

Vocabulaire : X est une variable explicative, Y est une variable à expliquer.

Objectif : prédire les valeurs de $y \in \mathcal{Y}$ associées à chaque valeur possible de $x \in \mathcal{X}$.

Classification : $\mathcal{Y} = \{0, 1\}$.

Régression : $\mathcal{Y} = \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathcal{Y}$ associant, à chaque entrée possible x une valeur de y prédite.

Idéalement, on cherche une règle de décision f qui minimise le risque

$$\begin{aligned} \mathbb{E}_{\mathcal{X}, \mathcal{Y}} \left[(Y - f(X))^2 \right] \\ \mathbb{P}_{\mathcal{X}, \mathcal{Y}} [Y \neq f(X)] \end{aligned}$$

Pourquoi est-ce difficile? Accès à $P_{\mathcal{X}, \mathcal{Y}}$ uniquement par un échantillon.

Données : *échantillon d'apprentissage* $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathcal{Y}$, constitué d'observations que l'on suppose indépendantes et identiquement distribuées selon la loi $P_{\mathcal{X}, \mathcal{Y}}$ sur $\mathcal{X} \times \mathcal{Y}$.

Vocabulaire : X est une variable explicative, Y est une variable à expliquer.

Objectif : prédire les valeurs de $y \in \mathcal{Y}$ associées à chaque valeur possible de $x \in \mathcal{X}$.

Classification : $\mathcal{Y} = \{0, 1\}$.

Régression : $\mathcal{Y} = \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathcal{Y}$ associant, à chaque entrée possible x une valeur de y prédite.

Idéalement, on cherche une règle de décision f qui minimise le risque

$$\mathbb{E}_{\mathcal{X}, \mathcal{Y}} \left[(Y - f(X))^2 \right]$$
$$\mathbb{P}_{\mathcal{X}, \mathcal{Y}} [Y \neq f(X)]$$

Pourquoi est-ce difficile? Accès à $P_{\mathcal{X}, \mathcal{Y}}$ uniquement par un échantillon.

Exercice : Y variable aléatoire réelle avec variance fini. Montrer que $\mathbb{E}[Y] = \min_{y \in \mathbb{R}} \mathbb{E}[(Y - y)^2]$. Que vaut le minimum ?

Théorème :

La meilleure règle possible est la *règle de Bayes* : pour tout $x \in \mathcal{X}$

- en régression : avec $\mathcal{Y} = \mathbb{R}$ et la perte quadratique :

$$f^*(x) = \mathbb{E}[Y|X = x] = \min_{y \in \mathbb{R}} \mathbb{E}[(Y - y)^2|X = x] .$$

- en classification : avec $\mathcal{Y} = \{0, 1\}$ et $\eta(x) = P(Y = 1|X = x)$:

$$f^*(x) = \mathbb{I}\{\eta(x) > 1/2\} = \min_{y \in \{0,1\}} \mathbb{P}[Y \neq y|X = x] .$$

Problème : on ne connaît *jamais* P , donc on ne peut pas la calculer.

Attention : le risque de la règle de Bayes n'est pas nul ! On ne peut pas tout prédire parfaitement.

Exercice : Y variable aléatoire réelle avec variance fini. Montrer que $\mathbb{E}[Y] = \min_{y \in \mathbb{R}} \mathbb{E}[(Y - y)^2]$. Que vaut le minimum ?

Théorème :

La meilleure règle possible est la *règle de Bayes* : pour tout $x \in \mathcal{X}$

- en régression : avec $\mathcal{Y} = \mathbb{R}$ et la perte quadratique :

$$f^*(x) = \mathbb{E}[Y|X = x] = \min_{y \in \mathbb{R}} \mathbb{E}[(Y - y)^2|X = x] .$$

- en classification : avec $\mathcal{Y} = \{0, 1\}$ et $\eta(x) = P(Y = 1|X = x)$:

$$f^*(x) = \mathbb{I}\{\eta(x) > 1/2\} = \min_{y \in \{0,1\}} \mathbb{P}[Y \neq y|X = x] .$$

Problème : on ne connaît *jamaïs* P , donc on ne peut pas la calculer.

Attention : le risque de la règle de Bayes n'est pas nul ! On ne peut pas tout prédire parfaitement.

Exercice : Y variable aléatoire réelle avec variance fini. Montrer que $\mathbb{E}[Y] = \min_{y \in \mathbb{R}} \mathbb{E}[(Y - y)^2]$. Que vaut le minimum ?

Théorème :

La meilleure règle possible est la *règle de Bayes* : pour tout $x \in \mathcal{X}$

- en régression : avec $\mathcal{Y} = \mathbb{R}$ et la perte quadratique :

$$f^*(x) = \mathbb{E}[Y|X = x] = \min_{y \in \mathbb{R}} \mathbb{E}[(Y - y)^2|X = x] .$$

- en classification : avec $\mathcal{Y} = \{0, 1\}$ et $\eta(x) = P(Y = 1|X = x)$:

$$f^*(x) = \mathbb{I}\{\eta(x) > 1/2\} = \min_{y \in \{0,1\}} \mathbb{P}[Y \neq y|X = x] .$$

Problème : on ne connaît *jamais* P , donc on ne peut pas la calculer.

Attention : le risque de la règle de Bayes n'est pas nul ! On ne peut pas tout prédire parfaitement.

Exercice : Y variable aléatoire réelle avec variance fini. Montrer que $\mathbb{E}[Y] = \min_{y \in \mathbb{R}} \mathbb{E}[(Y - y)^2]$. Que vaut le minimum ?

Théorème :

La meilleure règle possible est la *règle de Bayes* : pour tout $x \in \mathcal{X}$

- en régression : avec $\mathcal{Y} = \mathbb{R}$ et la perte quadratique :

$$f^*(x) = \mathbb{E}[Y|X = x] = \min_{y \in \mathbb{R}} \mathbb{E}[(Y - y)^2|X = x] .$$

- en classification : avec $\mathcal{Y} = \{0, 1\}$ et $\eta(x) = P(Y = 1|X = x)$:

$$f^*(x) = \mathbb{I}\{\eta(x) > 1/2\} = \min_{y \in \{0,1\}} \mathbb{P}[Y \neq y|X = x] .$$

Problème : on ne connaît *jamais* P , donc on ne peut pas la calculer.

Attention : le risque de la règle de Bayes n'est pas nul ! On ne peut pas tout prédire parfaitement.

Données : *échantillon d'apprentissage* $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathbb{R}$, observations iid selon la loi $P_{X,Y}$ sur $\mathcal{X} \times \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathbb{R}$.

Formellement un algorithme d'apprentissage est simplement une fonction de la forme

$$h_n : (x, x_1, y_1, \dots, x_n, y_n) \mapsto y \in \mathcal{Y}$$

on note de manière concise $f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n)$.

Vocabulaire, pour un algorithme h_n :

Entrainer : Etant donné $(x_k, y_k)_{1 \leq k \leq n}$, un jeu d'entraînement, construire

$$f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Prédire : Après entraînement, étant donné $x \in \mathcal{X}$, évaluer

$$f_n(x) = h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Données : *échantillon d'apprentissage* $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathbb{R}$, observations iid selon la loi $P_{\mathcal{X}, \mathcal{Y}}$ sur $\mathcal{X} \times \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathbb{R}$.

Formellement un algorithme d'apprentissage est simplement une fonction de la forme

$$h_n : (x, x_1, y_1, \dots, x_n, y_n) \mapsto y \in \mathcal{Y}$$

on note de manière concise $f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n)$.

Vocabulaire, pour un algorithme h_n :

Entraîner : Etant donné $(x_k, y_k)_{1 \leq k \leq n}$, un jeu d'entraînement, construire

$$f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Prédire : Après entraînement, étant donné $x \in \mathcal{X}$, évaluer

$$f_n(x) = h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Données : échantillon d'apprentissage $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathbb{R}$, observations iid selon la loi $P_{X,Y}$ sur $\mathcal{X} \times \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathbb{R}$.

Formellement un algorithme d'apprentissage est simplement une fonction de la forme

$$h_n : (x, x_1, y_1, \dots, x_n, y_n) \mapsto y \in \mathcal{Y}$$

on note de manière concise $f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n)$.

Vocabulaire, pour un algorithme h_n :

Entraîner : Etant donné $(x_k, y_k)_{1 \leq k \leq n}$, un jeu d'entraînement, construire

$$f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Prédire : Après entraînement, étant donné $x \in \mathcal{X}$, évaluer

$$f_n(x) = h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Données : échantillon d'apprentissage $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathbb{R}$, observations iid selon la loi $P_{\mathcal{X}, \mathcal{Y}}$ sur $\mathcal{X} \times \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathbb{R}$.

Formellement un algorithme d'apprentissage est simplement une fonction de la forme

$$h_n : (x, x_1, y_1, \dots, x_n, y_n) \mapsto y \in \mathcal{Y}$$

on note de manière concise $f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n)$.

Vocabulaire, pour un algorithme h_n :

Entraîner : Etant donné $(x_k, y_k)_{1 \leq k \leq n}$, un jeu d'entraînement, construire

$$f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Prédire : Après entraînement, étant donné $x \in \mathcal{X}$, évaluer

$$f_n(x) = h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Données : échantillon d'apprentissage $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathbb{R}$, observations iid selon la loi $P_{X,Y}$ sur $\mathcal{X} \times \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathbb{R}$.

Formellement un algorithme d'apprentissage est simplement une fonction de la forme

$$h_n : (x, x_1, y_1, \dots, x_n, y_n) \mapsto y \in \mathcal{Y}$$

on note de manière concise $f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n)$.

Vocabulaire, pour un algorithme h_n :

Entraîner : Etant donné $(x_k, y_k)_{1 \leq k \leq n}$, un jeu d'entraînement, construire

$$f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Prédire : Après entraînement, étant donné $x \in \mathcal{X}$, évaluer

$$f_n(x) = h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Données : échantillon d'apprentissage $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathbb{R}$, observations iid selon la loi $P_{X,Y}$ sur $\mathcal{X} \times \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathbb{R}$.

Formellement un algorithme d'apprentissage est simplement une fonction de la forme

$$h_n : (x, x_1, y_1, \dots, x_n, y_n) \mapsto y \in \mathcal{Y}$$

on note de manière concise $f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n)$.

Vocabulaire, pour un algorithme h_n :

Entraîner : Etant donné $(x_k, y_k)_{1 \leq k \leq n}$, un jeu d'entraînement, construire

$$f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n).$$

Prédire : Après entraînement, étant donné $x \in \mathcal{X}$, évaluer

$$f_n(x) = h_n(x, x_1, y_1, \dots, x_n, y_n).$$

$\mathcal{X} = \mathbb{R}^d$, échantillon d'entraînement dans $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathbb{R}$.
 h_n l'algorithme des k plus proches voisins.

Que se passe du point de vue calcul numérique.

- A l'entraînement ?
- A la prédiction ?

Les données sont des réalisation d'un processus aléatoires.

On cherche un algorithme d'apprentissage $f_n: \mathcal{X} \mapsto \mathbb{R}$ qui minimise le risque *en prenant en compte l'aléa des données d'entraînement*

$$\begin{aligned} R_n &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \right] \\ &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - h_n(X, X_1, Y_1, \dots, X_n, Y_n))^2 \right) \right] \end{aligned}$$

sans avoir accès à $P_{X, Y}$ autrement que par l'échantillon d'apprentissage.

Les données sont des réalisation d'un processus aléatoires.

On cherche un algorithme d'apprentissage $f_n: \mathcal{X} \mapsto \mathbb{R}$ qui minimise le risque *en prenant en compte l'aléat des données d'entraînement*

$$\begin{aligned} R_n &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \right] \\ &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - h_n(X, X_1, Y_1, \dots, X_n, Y_n))^2 \right) \right] \end{aligned}$$

sans avoir accès à $P_{X, Y}$ autrement que par l'échantillon d'apprentissage.

Les données sont des réalisation d'un processus aléatoires.

On cherche un algorithme d'apprentissage $f_n: \mathcal{X} \mapsto \mathbb{R}$ qui minimise le risque *en prenant en compte l'aléat des données d'entraînement*

$$\begin{aligned} R_n &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \right] \\ &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - h_n(X, X_1, Y_1, \dots, X_n, Y_n))^2 \right) \right] \end{aligned}$$

sans avoir accès à $P_{X, Y}$ autrement que par l'échantillon d'apprentissage.

Pour un $x \in \mathcal{X}$ et un échantillon fixés, on a l'erreur de prédiction suivante

$$\begin{aligned} & \mathbb{E}_Y \left[(Y - f_n(X))^2 \mid X = x \right] \\ &= \mathbb{E}_Y \left[(Y - f^*(X) + f^*(X) - f_n(X))^2 \mid X = x \right] \\ &= \mathbb{E}_Y \left[(Y - f^*(X))^2 \mid X = x \right] + \mathbb{E}_Y \left[(f^*(X) - f_n(X))^2 \mid X = x \right] \\ &\quad + 2\mathbb{E}_Y \left[(Y - f^*(X))(f^*(X) - f_n(X)) \mid X = x \right] \\ &= \mathbb{E}_Y \left[(Y - f^*(X))^2 \mid X = x \right] + (f^*(x) - f_n(x))^2 \\ &= \sigma^2(x) + (f^*(x) - f_n(x))^2 \end{aligned}$$

où $f^*(x) = \mathbb{E}[Y \mid X = x]$ est la décision de Bayes et $\sigma^2(x)$ est l'erreur de Bayes au point x . Prenons l'espérance sur le tirage de l'échantillon (noté \mathbb{E}_n).

$$\begin{aligned} & \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_Y \left[(Y - f_n(X))^2 \mid X = x \right] \right] \\ &= \sigma^2(x) + \mathbb{E}_n \left[(f^*(x) - f_n(x))^2 \right] \\ &= \sigma^2(x) + \text{Var}_n[f^*(x) - f_n(x)] + \mathbb{E}_n \left[f^*(x) - f_n(x) \right]^2 \\ &= \sigma^2(x) + \text{Var}_n[f_n(x)] + \text{Biais}_n \left[f_n(x) \right]^2 \end{aligned}$$

Pour un $x \in \mathcal{X}$ fixé, on a l'erreur de prédiction suivante

$$\begin{aligned} R_n(x) &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[E_Y \left[(Y - f_n(X))^2 \mid X = x \right] \right] \\ &= \sigma^2(x) + \text{Var}_n [f_n(x)] + \text{Biais}_n [f_n(x)]^2 \end{aligned}$$

- Variance : quantifie la dispersion f_n due au caractère aléatoire de l'échantillon
- Biais : quantifie la différence entre f_n et f^* en moyenne (aléat de l'échantillon).
- Risque ne peut pas être plus petit que $\sigma^2(x)$.
- Plus on a de "paramètres" à estimer dans notre modèle plus le biais devient petit, et la variance devient grande.
- Il y a un compromis à trouver, que l'on appelle le compromis "biais-variance".

Message principal : le modèle le plus complexe n'est pas forcément le meilleur.

Pour un $x \in \mathcal{X}$ fixé, on a l'erreur de prédiction suivante

$$\begin{aligned} R_n(x) &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[E_Y \left[(Y - f_n(X))^2 \mid X = x \right] \right] \\ &= \sigma^2(x) + \text{Var}_n [f_n(x)] + \text{Biais}_n [f_n(x)]^2 \end{aligned}$$

- **Variance** : quantifie la dispersion f_n due au caractère aléatoire de l'échantillon
- **Biais** : quantifie la différence entre f_n et f^* en moyenne (aléat de l'échantillon).
- Risque ne peut pas être plus petit que $\sigma^2(x)$.
- Plus on a de "paramètres" à estimer dans notre modèle plus le biais devient petit, et la variance devient grande.
- Il y a un compromis à trouver, que l'on appelle le compromis "biais-variance".

Message principal : le modèle le plus complexe n'est pas forcément le meilleur.

Pour un $x \in \mathcal{X}$ fixé, on a l'erreur de prédiction suivante

$$\begin{aligned} R_n(x) &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[E_Y \left[(Y - f_n(X))^2 \mid X = x \right] \right] \\ &= \sigma^2(x) + \text{Var}_n [f_n(x)] + \text{Biais}_n [f_n(x)]^2 \end{aligned}$$

- Variance : quantifie la dispersion f_n due au caractère aléatoire de l'échantillon
- Biais : quantifie la différence entre f_n et f^* en moyenne (aléat de l'échantillon).
- Risque ne peut pas être plus petit que $\sigma^2(x)$.
- Plus on a de "paramètres" à estimer dans notre modèle plus le biais devient petit, et la variance devient grande.
- Il y a un compromis à trouver, que l'on appelle le compromis "biais-variance".

Message principal : le modèle le plus complexe n'est pas forcément le meilleur.

Pour un $x \in \mathcal{X}$ fixé, on a l'erreur de prédiction suivante

$$\begin{aligned} R_n(x) &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[E_Y \left[(Y - f_n(X))^2 \mid X = x \right] \right] \\ &= \sigma^2(x) + \text{Var}_n [f_n(x)] + \text{Biais}_n [f_n(x)]^2 \end{aligned}$$

- Variance : quantifie la dispersion f_n due au caractère aléatoire de l'échantillon
- Biais : quantifie la différence entre f_n et f^* en moyenne (aléat de l'échantillon).
- Risque ne peut pas être plus petit que $\sigma^2(x)$.
- Plus on a de "paramètres" à estimer dans notre modèle plus le biais devient petit, et la variance devient grande.
- Il y a un compromis à trouver, que l'on appelle le compromis "biais-variance".

Message principal : le modèle le plus complexe n'est pas forcément le meilleur.

Pour un $x \in \mathcal{X}$ fixé, on a l'erreur de prédiction suivante

$$\begin{aligned} R_n(x) &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[E_Y \left[(Y - f_n(X))^2 \mid X = x \right] \right] \\ &= \sigma^2(x) + \text{Var}_n [f_n(x)] + \text{Biais}_n [f_n(x)]^2 \end{aligned}$$

- Variance : quantifie la dispersion f_n due au caractère aléatoire de l'échantillon
- Biais : quantifie la différence entre f_n et f^* en moyenne (aléat de l'échantillon).
- Risque ne peut pas être plus petit que $\sigma^2(x)$.
- Plus on a de "paramètres" à estimer dans notre modèle plus le biais devient petit, et la variance devient grande.
- Il y a un compromis à trouver, que l'on appelle le compromis "biais-variance".

Message principal : le modèle le plus complexe n'est pas forcément le meilleur.

Pour un $x \in \mathcal{X}$ fixé, on a l'erreur de prédiction suivante

$$\begin{aligned} R_n(x) &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[E_Y \left[(Y - f_n(X))^2 \mid X = x \right] \right] \\ &= \sigma^2(x) + \text{Var}_n [f_n(x)] + \text{Biais}_n [f_n(x)]^2 \end{aligned}$$

- Variance : quantifie la dispersion f_n due au caractère aléatoire de l'échantillon
- Biais : quantifie la différence entre f_n et f^* en moyenne (aléat de l'échantillon).
- Risque ne peut pas être plus petit que $\sigma^2(x)$.
- Plus on a de "paramètres" à estimer dans notre modèle plus le biais devient petit, et la variance devient grande.
- Il y a un compromis à trouver, que l'on appelle le compromis "biais-variance".

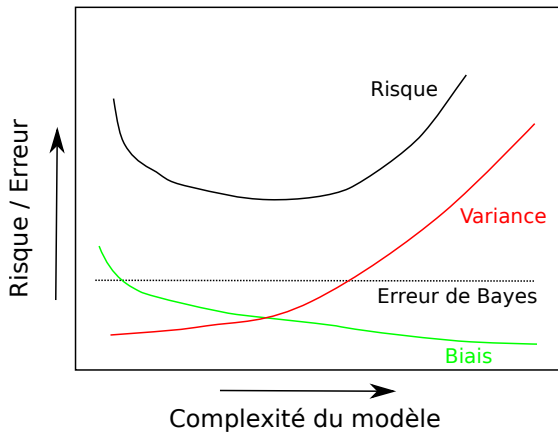
Message principal : le modèle le plus complexe n'est pas forcément le meilleur.

Pour un $x \in \mathcal{X}$ fixé, on a l'erreur de prédiction suivante

$$\begin{aligned} R_n(x) &= \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[E_Y \left[(Y - f_n(X))^2 \mid X = x \right] \right] \\ &= \sigma^2(x) + \text{Var}_n [f_n(x)] + \text{Biais}_n [f_n(x)]^2 \end{aligned}$$

- Variance : quantifie la dispersion f_n due au caractère aléatoire de l'échantillon
- Biais : quantifie la différence entre f_n et f^* en moyenne (aléat de l'échantillon).
- Risque ne peut pas être plus petit que $\sigma^2(x)$.
- Plus on a de "paramètres" à estimer dans notre modèle plus le biais devient petit, et la variance devient grande.
- Il y a un compromis à trouver, que l'on appelle le compromis "biais-variance".

Message principal : le modèle le plus complexe n'est pas forcément le meilleur.



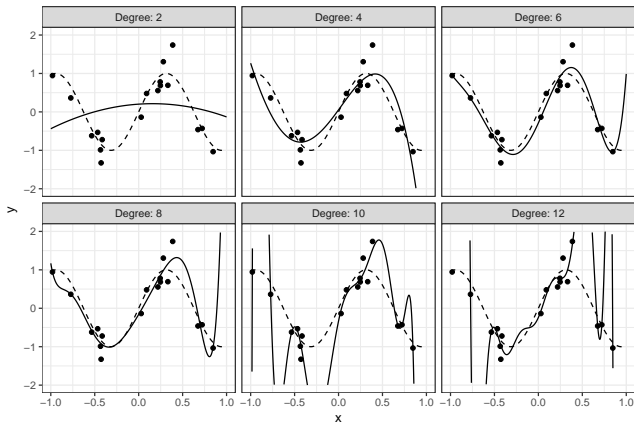
Attention : cette interprétation est parfois discutée typiquement en *deep learning*.

Compromis biais variance : régression polynomiale

Etant donné un échantillon $(x_k, y_k)_{1 \leq k \leq n}$, cherchons une règle de décision polynomiale de degré d .

$$\min_{P \in \mathbb{R}_d[x]} \sum_{i=1}^n (P(x_i) - y_i)^2$$

Comment fait on cela ?

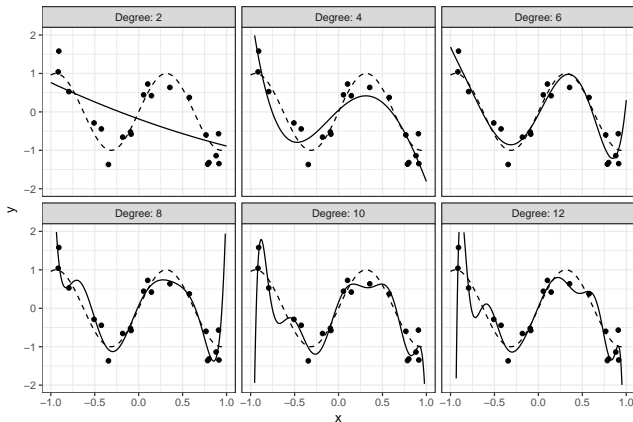


Compromis biais variance : régression polynomiale

Etant donné un échantillon $(x_k, y_k)_{1 \leq k \leq n}$, cherchons une règle de décision polynomiale de degré d .

$$\min_{P \in \mathbb{R}_d[x]} \sum_{i=1}^n (P(x_i) - y_i)^2$$

Comment fait on cela ?

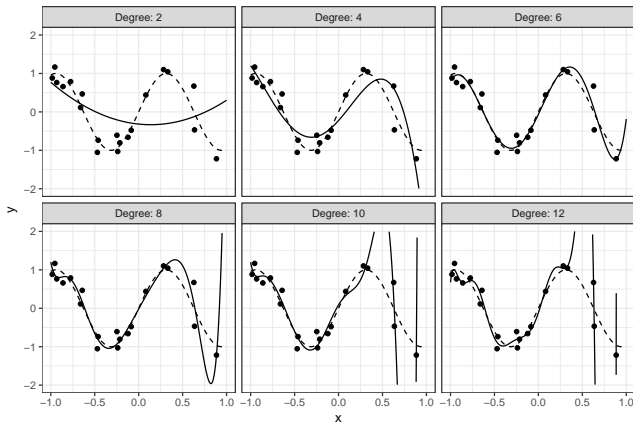


Compromis biais variance : régression polynomiale

Etant donné un échantillon $(x_k, y_k)_{1 \leq k \leq n}$, cherchons une règle de décision polynomiale de degré d .

$$\min_{P \in \mathbb{R}_d[x]} \sum_{i=1}^n (P(x_i) - y_i)^2$$

Comment fait on cela ?

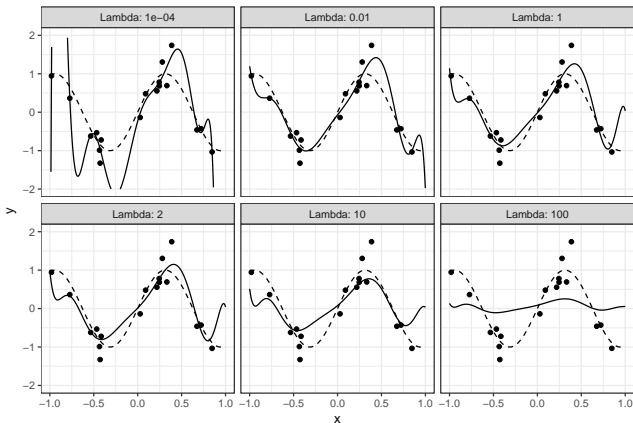


Compromis biais variance : régularisation

Etant donné un échantillon $(x_k, y_k)_{1 \leq k \leq n}$, cherchons une règle de décision polynomiale de degré 10.

$$\min_{P \in \mathbb{R}_d[x]} \sum_{i=1}^n (P(x_i) - y_i)^2 + \lambda \|P\|^2$$

où $\|P\|$ est la somme des carrés des coefficients de P . Comment fait on cela ?

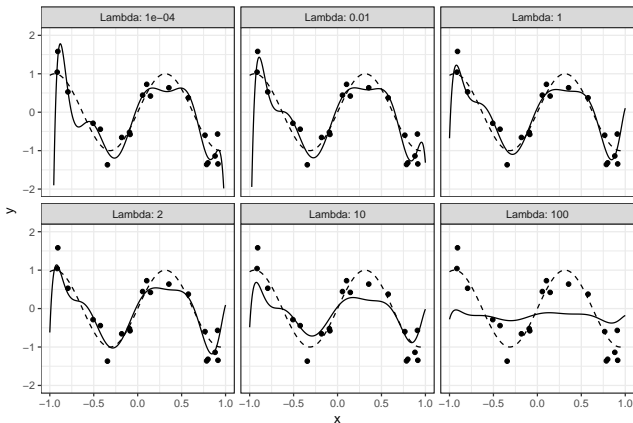


Compromis biais variance : régularisation

Etant donné un échantillon $(x_k, y_k)_{1 \leq k \leq n}$, cherchons une règle de décision polynomiale de degré 10.

$$\min_{P \in \mathbb{R}_d[x]} \sum_{i=1}^n (P(x_i) - y_i)^2 + \lambda \|P\|^2$$

où $\|P\|$ est la somme des carrés des coefficients de P . Comment fait on cela ?

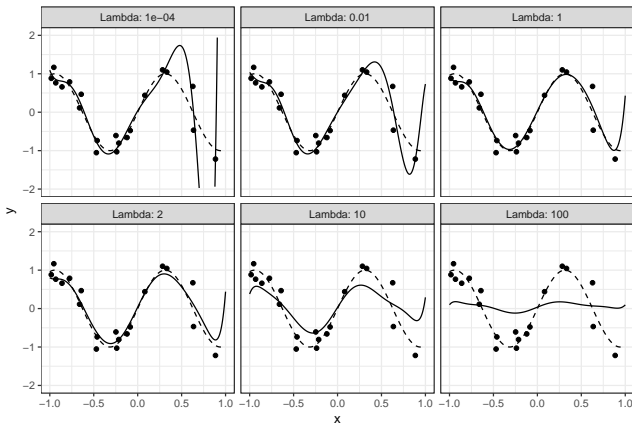


Compromis biais variance : régularisation

Etant donné un échantillon $(x_k, y_k)_{1 \leq k \leq n}$, cherchons une règle de décision polynomiale de degré 10.

$$\min_{P \in \mathbb{R}_d[x]} \sum_{i=1}^n (P(x_i) - y_i)^2 + \lambda \|P\|^2$$

où $\|P\|$ est la somme des carrés des coefficients de P . Comment fait on cela ?

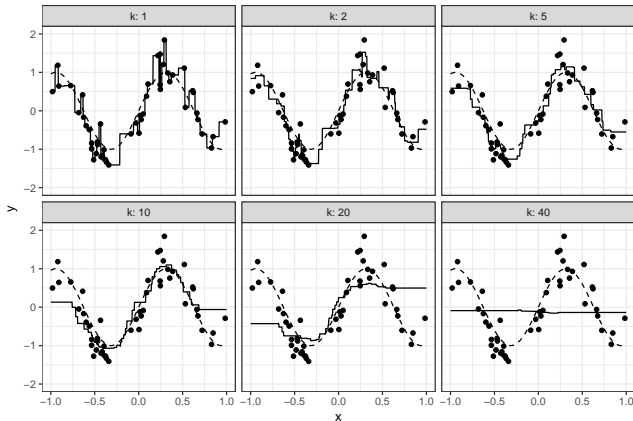


Compromis biais variance : kNN

Echantillon : $S_n = (x_i, y_i)_{1 \leq i \leq n}$ et $k \in \mathbb{N}$ fixés, l'estimateur des k -plus proche voisins en x :

$$f_n: x \mapsto \frac{1}{k} \sum_{i=1}^k y_{l_i},$$

où l_1, \dots, l_k sont les indices des k plus proches voisins de x dans S_n .



$$Y = f^*(X) + \epsilon$$

X et ϵ indépendantes.

$$\text{Var}[\epsilon] = \sigma^2, \mathbb{E}[\epsilon] = 0.$$

Que valent $E[Y|X = x]$ et $\sigma(x)$?

Echantillon : $S_n = (x_i, y_i)_{1 \leq i \leq n}$ et $k \in \mathbb{N}$ fixés, la prédiction en x fixé est :

$$f_n : x \mapsto \frac{1}{k} \sum_{i=1}^k y_{l_i},$$

où l_1, \dots, l_k sont les indices des k plus proches voisins de x dans S_n .

- Que vaut la variance $\text{Var}_n(f_n(x))$?
- Que vaut le biais $\mathbb{E}_n [f_n(x) - f^*(x)]^2$?

$$R_n(x) = \sigma^2 + \left[f^*(x) - \frac{1}{k} \sum_{i=1}^k f^*(x_{l_i}) \right]^2 + \frac{\sigma^2}{k},$$

Comment se comporte le risque en fonction de k ?

$$Y = f^*(X) + \epsilon$$

X et ϵ indépendantes.

$$\text{Var}[\epsilon] = \sigma^2, \mathbb{E}[\epsilon] = 0.$$

Que valent $E[Y|X = x]$ et $\sigma(x)$?

Echantillon : $S_n = (x_i, y_i)_{1 \leq i \leq n}$ et $k \in \mathbb{N}$ fixés, la prédiction en x fixé est :

$$f_n : x \mapsto \frac{1}{k} \sum_{i=1}^k y_{l_i},$$

où l_1, \dots, l_k sont les indices des k plus proches voisins de x dans S_n .

- Que vaut la variance $\text{Var}_n(f_n(x))$?
- Que vaut le biais $\mathbb{E}_n [f_n(x) - f^*(x)]^2$?

$$R_n(x) = \sigma^2 + \left[f^*(x) - \frac{1}{k} \sum_{i=1}^k f^*(x_{l_i}) \right]^2 + \frac{\sigma^2}{k},$$

Comment se comporte le risque en fonction de k ?

$$Y = f^*(X) + \epsilon$$

X et ϵ indépendantes.

$$\text{Var}[\epsilon] = \sigma^2, \mathbb{E}[\epsilon] = 0.$$

Que valent $E[Y|X = x]$ et $\sigma(x)$?

Echantillon : $S_n = (x_i, y_i)_{1 \leq i \leq n}$ et $k \in \mathbb{N}$ fixés, la prédiction en x fixé est :

$$f_n : x \mapsto \frac{1}{k} \sum_{i=1}^k y_{l_i},$$

où l_1, \dots, l_k sont les indices des k plus proches voisins de x dans S_n .

- Que vaut la variance $\text{Var}_n(f_n(x))$?
- Que vaut le biais $\mathbb{E}_n [f_n(x) - f^*(x)]^2$?

$$R_n(x) = \sigma^2 + \left[f^*(x) - \frac{1}{k} \sum_{i=1}^k f^*(x_{l_i}) \right]^2 + \frac{\sigma^2}{k},$$

Comment se comporte le risque en fonction de k ?

$$Y = f^*(X) + \epsilon$$

X et ϵ indépendantes.

$$\text{Var}[\epsilon] = \sigma^2, \mathbb{E}[\epsilon] = 0.$$

Que valent $E[Y|X = x]$ et $\sigma(x)$?

Echantillon : $S_n = (x_i, y_i)_{1 \leq i \leq n}$ et $k \in \mathbb{N}$ fixés, la prédiction en x fixé est :

$$f_n : x \mapsto \frac{1}{k} \sum_{i=1}^k y_{l_i},$$

où l_1, \dots, l_k sont les indices des k plus proches voisins de x dans S_n .

- Que vaut la variance $\text{Var}_n(f_n(x))$?
- Que vaut le biais $\mathbb{E}_n [f_n(x) - f^*(x)]^2$?

$$R_n(x) = \sigma^2 + \left[f^*(x) - \frac{1}{k} \sum_{i=1}^k f^*(x_{l_i}) \right]^2 + \frac{\sigma^2}{k},$$

Comment se comporte le risque en fonction de k ?

$$Y = f^*(X) + \epsilon$$

X et ϵ indépendantes.

$$\text{Var}[\epsilon] = \sigma^2, \mathbb{E}[\epsilon] = 0.$$

Que valent $E[Y|X = x]$ et $\sigma(x)$?

Echantillon : $S_n = (x_i, y_i)_{1 \leq i \leq n}$ et $k \in \mathbb{N}$ fixés, la prédiction en x fixé est :

$$f_n : x \mapsto \frac{1}{k} \sum_{i=1}^k y_{l_i},$$

où l_1, \dots, l_k sont les indices des k plus proches voisins de x dans S_n .

- Que vaut la variance $\text{Var}_n(f_n(x))$?
- Que vaut le biais $\mathbb{E}_n [f_n(x) - f^*(x)]^2$?

$$R_n(x) = \sigma^2 + \left[f^*(x) - \frac{1}{k} \sum_{i=1}^k f^*(x_{l_i}) \right]^2 + \frac{\sigma^2}{k},$$

Comment se comporte le risque en fonction de k ?

Travaux pratiques avec scikit-learn.

www.math.univ-toulouse.fr/~epauwels/LearningM2SID/

Formellement un algorithme d'apprentissage est simplement une fonction de la forme

$$h_n : (x, x_1, y_1, \dots, x_n, y_n) \mapsto y \in \mathcal{Y}$$

on note de manière concise $f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n)$.

Tutoriel python :

- Modèle `sklearn` = fonction h_n .
- `fit` construit f_n : fixe l'échantillon d'entraînement $(x_1, y_1, \dots, x_n, y_n)$.
- `predict` évalue f_n : argument à choisir selon la quantité à évaluer.

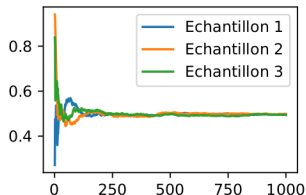
Estimation par Monte-Carlo, loi des grands nombres :

Soit P une mesure de probabilité sur \mathbb{R}^p et $(X_i)_{i \in \mathbb{N}}$ une suite de variables aléatoires indépendantes et identiquement distribuées de loi P . Alors pour tout f fonction continue sur \mathbb{R}^p (plus généralement mesurable), pourvu que l'espérance existe

$$\frac{1}{n} \sum_{i=1}^n f(X_i) \xrightarrow[n \rightarrow \infty]{p.s.} \mathbb{E}_{X \sim P}[f(X)] = \int_{x \in \mathbb{R}^p} f(x) dP(x)$$

Exemple : Espérance d'une loi uniforme sur $[0, 1]$

```
from numpy.random import *
from numpy import *
X = rand(n)
avgs = cumsum(X) / arange(1,n+1)
plot(arange(1,n+1),avgs, label = "Echantillon 1")
```



1. Qu'est-ce que l'apprentissage supervisé ?
2. Compromis biais-variance
3. Evaluation et selection de modèle
4. Aggrégation de modèles et méthodes d'ensembles

Données : *échantillon d'apprentissage* $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathbb{R}$, observations iid selon la loi $P_{X,Y}$ sur $\mathcal{X} \times \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathbb{R}$.

Formellement un algorithme d'apprentissage est simplement une fonction de la forme

$$h_n : (x, x_1, y_1, \dots, x_n, y_n) \mapsto y \in \mathcal{Y}$$

on note de manière concise $f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n)$.

On cherche une règle de décision f_n qui minimise le risque

$$R_n = \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \right].$$

Données : *échantillon d'apprentissage* $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathbb{R}$, observations iid selon la loi $P_{X,Y}$ sur $\mathcal{X} \times \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathbb{R}$.

Formellement un algorithme d'apprentissage est simplement une fonction de la forme

$$h_n : (x, x_1, y_1, \dots, x_n, y_n) \mapsto y \in \mathcal{Y}$$

on note de manière concise $f_n : x \mapsto h_n(x, x_1, y_1, \dots, x_n, y_n)$.

On cherche une règle de décision f_n qui minimise le risque

$$R_n = \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \right].$$

Comment choisir h_n ?

C'est à dire : le modèle, l'architecture, les hyper-paramètres,...

On cherche une règle de décision f qui minimise le risque

$$R_n = \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[E_{(X, Y)} \left((Y - f_n(X))^2 \right) \right].$$

On cherche une règle de décision f qui minimise le risque

$$R_n = \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \right] .$$

Performance de prédiction : On cherche à prédire correctement sur de nouvelles données, inconnues, mais issues de la même distribution que l'échantillon. C'est l'objet de

$$\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) = \mathbb{E}_{(X, Y)} \left((Y - h_n(X, X_1, Y_1, \dots, X_n, Y_n))^2 \right) .$$

On cherche une règle de décision f qui minimise le risque

$$R_n = \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \right] .$$

Performance de prédiction : On cherche à prédire correctement sur de nouvelles données, inconnues, mais issues de la même distribution que l'échantillon. C'est l'objet de

$$\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) = \mathbb{E}_{(X, Y)} \left((Y - h_n(X, X_1, Y_1, \dots, X_n, Y_n))^2 \right) .$$

Performance d'estimation : Dépend de l'échantillon : une réalisation de variables aléatoires iid. Prédiction performante en moyenne sur l'ensemble des tirages possibles de l'échantillon :

$$\mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\dots \right]$$

On cherche une règle de décision f qui minimise le risque

$$R_n = \mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \right] .$$

Performance de prédiction : On cherche à prédire correctement sur de nouvelles données, inconnues, mais issues de la même distribution que l'échantillon. C'est l'objet de

$$\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) = \mathbb{E}_{(X, Y)} \left((Y - h_n(X, X_1, Y_1, \dots, X_n, Y_n))^2 \right) .$$

Performance d'estimation : Dépend de l'échantillon : une réalisation de variables aléatoires iid. Prédiction performante en moyenne sur l'ensemble des tirages possibles de l'échantillon :

$$\mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\dots \right]$$

Problème : On ne connaît pas P , on a juste accès à $(x_k, y_k)_{1 \leq k \leq n}$ une réalisation de l'échantillon. Solutions :

- Jeu de données test
- Ré-échantillonnage

Pour $(x_k, y_k)_{1 \leq k \leq n}$ fixé :

$$\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) = \mathbb{E}_{(X, Y)} \left((Y - h_n(X, x_1, y_1, \dots, x_n, y_n))^2 \right) .$$

Pour $(x_k, y_k)_{1 \leq k \leq n}$ fixé :

$$\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) = \mathbb{E}_{(X, Y)} \left((Y - h_n(X, x_1, y_1, \dots, x_n, y_n))^2 \right) .$$

Loi forte des grands nombres : Soit $G(\cdot, \cdot)$ une fonction mesurable, et $(\tilde{x}_k, \tilde{y}_k)_{1 \leq k \leq \tilde{n}}$ un échantillon indépendant tiré selon la loi de (X, Y) . Alors, presque sûrement :

$$\frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} G(\tilde{x}_i, \tilde{y}_i) \xrightarrow{\tilde{n} \rightarrow \infty} \mathbb{E}_{X, Y} [G(X, Y)]$$

Pour $(x_k, y_k)_{1 \leq k \leq n}$ fixé :

$$\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) = \mathbb{E}_{(X, Y)} \left((Y - h_n(X, x_1, y_1, \dots, x_n, y_n))^2 \right) .$$

Loi forte des grands nombres : Soit $G(\cdot, \cdot)$ une fonction mesurable, et $(\tilde{x}_k, \tilde{y}_k)_{1 \leq k \leq \tilde{n}}$ un échantillon indépendant tiré selon la loi de (X, Y) . Alors, presque sûrement :

$$\frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} G(\tilde{x}_i, \tilde{y}_i) \xrightarrow{\tilde{n} \rightarrow \infty} \mathbb{E}_{X, Y} [G(X, Y)]$$

Approximation sur un échantillon fini :

$$\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \simeq \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} (\tilde{y}_i - f_n(\tilde{x}_i))^2 = \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} (\tilde{y}_i - h_n(\tilde{x}_i, x_1, y_1, \dots, x_n, y_n))^2$$

Pour $(x_k, y_k)_{1 \leq k \leq n}$ fixé :

$$\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) = \mathbb{E}_{(X, Y)} \left((Y - h_n(X, x_1, y_1, \dots, x_n, y_n))^2 \right) .$$

Loi forte des grands nombres : Soit $G(\cdot, \cdot)$ une fonction mesurable, et $(\tilde{x}_k, \tilde{y}_k)_{1 \leq k \leq \tilde{n}}$ un échantillon indépendant tiré selon la loi de (X, Y) . Alors, presque sûrement :

$$\frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} G(\tilde{x}_i, \tilde{y}_i) \xrightarrow{\tilde{n} \rightarrow \infty} \mathbb{E}_{X, Y} [G(X, Y)]$$

Approximation sur un échantillon fini :

$$\mathbb{E}_{(X, Y)} \left((Y - f_n(X))^2 \right) \simeq \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} (\tilde{y}_i - f_n(\tilde{x}_i))^2 = \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} (\tilde{y}_i - h_n(\tilde{x}_i, x_1, y_1, \dots, x_n, y_n))^2$$

Conditions/remarques :

- G indépendant de n : impossible de réutiliser $(x_k, y_k)_{1 \leq k \leq n}$ dans $(\tilde{x}_k, \tilde{y}_k)_{1 \leq k \leq \tilde{n}}$.
- Plus \tilde{n} est grand, meilleure est l'approximation.

C'est la raison pour laquelle on garde un échantillon de test.

Comment évaluer

$$\mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} [\dots] ?$$

Lois des grands nombres ? On a qu'un seul échantillon et il en faudrait idéalement beaucoup.

Comment évaluer

$$\mathbb{E}_{(X_1, Y_1), \dots, (X_n, Y_n)} \left[\dots \right] ?$$

Lois des grands nombres ? On a qu'un seul échantillon et il en faudrait idéalement beaucoup.



Bootstrap : Sous-échantillonner, ré-échantillonner, pour “simuler” le tirage d'un nouvel échantillon.

Etant donné $S = (x_k)_{1 \leq k \leq n}$, réels, notons $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$. Quelle est la variance de \bar{x} ?

$$\text{Var}_{X_1, \dots, X_n} \left[\frac{1}{n} \sum_{i=1}^n X_i \right]$$

Etant donné $S = (x_k)_{1 \leq k \leq n}$, réels, notons $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$. Quelle est la variance de \bar{x} ?

$$\text{Var}_{X_1, \dots, X_n} \left[\frac{1}{n} \sum_{i=1}^n X_i \right]$$

Réponse : Si X_1, \dots, X_n sont iid, c'est $\frac{1}{n} \text{Var}(X)$, il suffit d'estimer $\text{Var}(X)$.

Etant donné $S = (x_k)_{1 \leq k \leq n}$, réels, notons $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$. Quelle est la variance de \bar{x} ?

$$\text{Var}_{X_1, \dots, X_n} \left[\frac{1}{n} \sum_{i=1}^n X_i \right]$$

Réponse : Si X_1, \dots, X_n sont iid, c'est $\frac{1}{n} \text{Var}(X)$, il suffit d'estimer $\text{Var}(X)$.

Une technique algorithmique : Pour $j = 1, \dots, K$,

- notons S_j un échantillon de taille n tiré indépendamment et avec remise de S .
- \bar{x}_j la moyenne empirique de S_j .

On traite les S_j comme de nouveaux échantillons et $(\bar{x}_j)_{j=1}^K$ comme différentes réalisations de \bar{x} .

$$\text{Var}_{X_1, \dots, X_n} \left[\frac{1}{n} \sum_{i=1}^n X_i \right] \simeq \text{Var}(\bar{x}_1, \dots, \bar{x}_K)$$

Etant donné $S = (x_k)_{1 \leq k \leq n}$, réels, notons $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$. Quelle est la variance de \bar{x} ?

$$\text{Var}_{X_1, \dots, X_n} \left[\frac{1}{n} \sum_{i=1}^n X_i \right]$$

Réponse : Si X_1, \dots, X_n sont iid, c'est $\frac{1}{n} \text{Var}(X)$, il suffit d'estimer $\text{Var}(X)$.

Une technique algorithmique : Pour $j = 1, \dots, K$,

- notons S_j un échantillon de taille n tiré indépendamment et avec remise de S .
- \bar{x}_j la moyenne empirique de S_j .

On traite les S_j comme de nouveaux échantillons et $(\bar{x}_j)_{j=1}^K$ comme différentes réalisations de \bar{x} .

$$\text{Var}_{X_1, \dots, X_n} \left[\frac{1}{n} \sum_{i=1}^n X_i \right] \simeq \text{Var}(\bar{x}_1, \dots, \bar{x}_K)$$

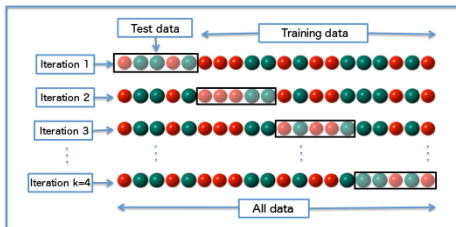
Plus généralement, pour calculer $\mathbb{E}_{X_1, \dots, X_n} [G(X_1, \dots, X_n)]$, on utilise l'approximations

$$\mathbb{E}_{X_1, \dots, X_n} [G(X_1, \dots, X_n)] \simeq \frac{1}{K} \sum_{j=1}^K G(x_{j1}, \dots, x_{jn})$$

où x_{ji} désigne le i ème élément de S_j , $i = 1, \dots, n$, $j = 1, \dots, K$.

Comment évaluer/sélectionner un modèle (par exemple k dans les k plus proches voisins) ?

Combiner ré-échantillonnage et estimation d'erreur de test



Remarque : Si la validation croisée est utilisée pour sélectionner un modèle (régler un hyper paramètre), utiliser un échantillon de test pour évaluer la performance de prédiction du modèle choisi.

1. Qu'est-ce que l'apprentissage supervisé ?
2. Compromis biais-variance
3. Evaluation et selection de modèle
4. Aggrégation de modèles et méthodes d'ensembles

Données : *échantillon d'apprentissage* $(x_k, y_k)_{1 \leq k \leq n}$ dans $\mathcal{X} \times \mathcal{Y}$, constitué d'observations que l'on suppose indépendantes et identiquement distribuées selon la loi $P_{\mathcal{X}, \mathcal{Y}}$ sur $\mathcal{X} \times \mathcal{Y}$.

Vocabulaire : X est une variable explicative, Y est une variable à expliquer.

Objectif : prédire les valeurs de $y \in \mathcal{Y}$ associées à chaque valeur possible de $x \in \mathcal{X}$.

Classification : $\mathcal{Y} = \{0, 1\}$.

Régression : $\mathcal{Y} = \mathbb{R}$.

Règle de décision : à partir de l'échantillon d'apprentissage, construire $f_n : \mathcal{X} \rightarrow \mathcal{Y}$ associant, à chaque entrée possible x une valeur de y prédite.

Idéalement, on cherche une règle de décision f qui minimise le risque

$$\begin{aligned} \mathbb{E}_{\mathcal{X}, \mathcal{Y}} \left[(Y - f(X))^2 \right] \\ \mathbb{P}_{\mathcal{X}, \mathcal{Y}} [Y \neq f(X)] \end{aligned}$$

sans avoir accès à $P_{\mathcal{X}, \mathcal{Y}}$ autrement que par l'échantillon d'apprentissage.

- Construction de différentes règles de décisions :

$$f_{n1}, f_{n2}, \dots, f_{nK}$$

Souvent issues d'un même classifieur de base.

- Production d'un classifieur agrégé :

$$F_n(f_{n1}, f_{n2}, \dots, f_{nK}) : \mathcal{X} \rightarrow \mathcal{Y}$$

- Construction de différentes règles de décisions :

$$f_{n1}, f_{n2}, \dots, f_{nK}$$

Souvent issues d'un même classifieur de base.

- Production d'un classifieur agrégé :

$$F_n(f_{n1}, f_{n2}, \dots, f_{nK}) : \mathcal{X} \rightarrow \mathcal{Y}$$

Pourquoi ?

- Faciliter l'apprentissage (réduire le terme de variance).
- Augmenter la performance de prédiction (réduire le terme de biais).

- Construction de différentes règles de décisions :

$$f_{n1}, f_{n2}, \dots, f_{nK}$$

Souvent issues d'un même classifieur de base.

- Production d'un classifieur agrégé :

$$F_n(f_{n1}, f_{n2}, \dots, f_{nK}) : \mathcal{X} \rightarrow \mathcal{Y}$$

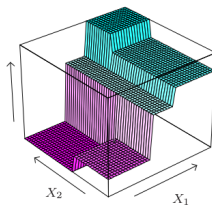
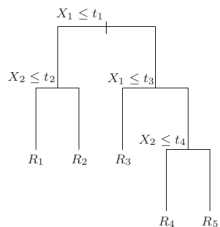
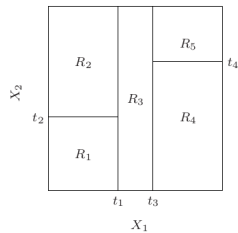
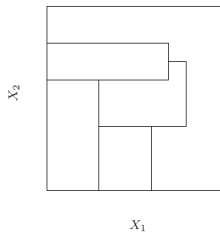
Pourquoi ?

- Faciliter l'apprentissage (réduire le terme de variance).
- Augmenter la performance de prédiction (réduire le terme de biais).

Plan :

- CART trees, bagging, random forests, boosting, gradient boosting.
- Source : Elements of Statistical Learning.

CART : classification and regression tree :



- Un arbre induit une partition de l'espace
- Dans chaque sous ensemble de la partition, on prédit la valeur moyenne observée.
- L'arbre se construit par récurrence.
- On choisit de manière gloutonne, la feuille, la variable et le seuil qui induisent la meilleure attache aux données (residual sum of squares en régression, Gini index en classification).

Contrôle de la complexité du modèle : profondeur maximale.

- Un arbre induit une partition de l'espace
- Dans chaque sous ensemble de la partition, on prédit la valeur moyenne observée.
- L'arbre se construit par récurrence.
- On choisit de manière gloutonne, la feuille, la variable et le seuil qui induisent la meilleure attache aux données (residual sum of squares en régression, Gini index en classification).

Contrôle de la complexité du modèle : profondeur maximale.

Dans tout ce qui suit on utilise un prédicteur de base que l'on note h_n :

$$h_n(\cdot, S) : \mathbb{R} \rightarrow \mathbb{R}$$

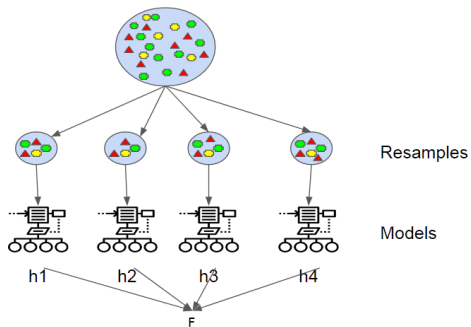
où $S = \{(x_i, y_i)\}_{i=1}^n$ est un échantillon d'apprentissage. On illustrera en TP tous nos modèles d'agrégation avec des arbres de régression.

Bagging est l'acronyme de "bootstrap aggregation".

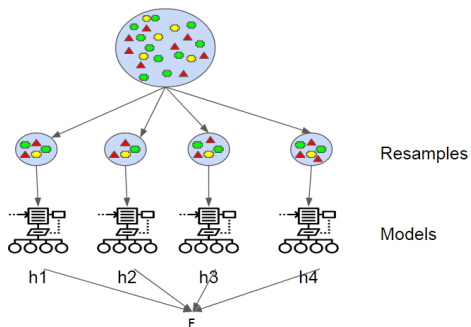
- Etant donné S , générer K nouveaux échantillons, S_1, \dots, S_K de taille n par la méthode du bootstrap.
- Pour de la régression, agréger avec une moyenne

$$F_n(\cdot) = \frac{1}{K} \sum_{k=1}^K h_n(\cdot, S_k)$$

Pour de la classification, on agrège avec un vote majoritaire.



Source hackernoon.com



Source hackernoon.com

Effet du bagging :

- La moyenne réduit le terme de variance.
 - ▶ On stabilise des méthodes et on évite le sur-apprentissage.
- Le fait de moyenner ne permet pas d'améliorer le biais
 - ▶ Si h_n a un biais fort, alors F_n aussi.

Random Forests = bagging + arbres + double bootstrap.

Pour chaque arbre, on sous échantillonne à la fois les individus et les variables aléatoirement.

“Feature bagging” : réduire la corrélation entre les arbres, éviter que les mêmes variables soient sélectionnées tout le temps dans le processus de construction des arbres.

L'idée du boosting est de renforcer un prédicteur faible (un peu mieux que random). Pour la régression :

- On se concentre sur les exemples mal prédits en repondérant les données

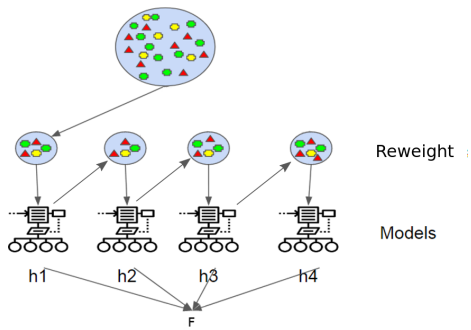
$$\frac{1}{n} \sum_{i=1}^n x_i \rightarrow \sum_{i=1}^n w_i x_i$$

- On agrège avec une médiane pondérée.
- Nouvelle notation :

$$h_n(\cdot, S, w)$$

pour expliciter le fait qu'on a un échantillon S avec des poids w , positifs, sommant à 1.

Boosting



Source hackernoon.com

Adaboost pour de la régression : $S = \{(x_i, y_i)\}_{i=1}^n$ pondérés avec des poids $w_i = 1/n$,
 $i = 1, \dots, n$ itération, pour $k = 1, 2, \dots, K$

- $f_k(\cdot) = h_n(\cdot, S, w)$
- $D = \max_{i=1, \dots, n} (f_k(x_i) - y_i)^2$
- $L_i = (f_k(x_i) - y_i)^2 / D^2$, $i = 1, \dots, n$
- $\bar{L}_k = \sum_{i=1}^n w_i L_i$
- $\beta_k = \frac{\bar{L}_k}{1 - \bar{L}_k}$.
- $w_i \leftarrow w_i \times \beta_k^{1 - L_i}$, $i = 1, \dots, n$.
- $w_i \leftarrow \frac{w_i}{\sum_{j=1}^n w_j}$, $i = 1, \dots, n$.

Retourne $F: x \mapsto \text{median}_{(\beta)}(f_1(x), \dots, f_K(x))$, une médiane pondérée.

Adaboost pour de la régression : $S = \{(x_i, y_i)\}_{i=1}^n$ pondérés avec des poids $w_i = 1/n$,
 $i = 1, \dots, n$ itération, pour $k = 1, 2, \dots, K$

- $f_k(\cdot) = h_n(\cdot, S, w)$
- $D = \max_{i=1, \dots, n} (f_k(x_i) - y_i)^2$
- $L_i = (f_k(x_i) - y_i)^2 / D^2, i = 1, \dots, n$
- $\bar{L}_k = \sum_{i=1}^n w_i L_i$
- $\beta_k = \frac{\bar{L}_k}{1 - \bar{L}_k}$.
- $w_i \leftarrow w_i \times \beta_k^{1 - L_i}, i = 1, \dots, n.$
- $w_i \leftarrow \frac{w_i}{\sum_{j=1}^n w_j}, i = 1, \dots, n.$

Retourne $F: x \mapsto \text{median}_{(\beta)}(f_1(x), \dots, f_K(x))$, une médiane pondérée.

Adaboost pour de la régression : $S = \{(x_i, y_i)\}_{i=1}^n$ pondérés avec des poids $w_i = 1/n$, $i = 1, \dots, n$ itération, pour $k = 1, 2, \dots, K$

- $f_k(\cdot) = h_n(\cdot, S, w)$
- $D = \max_{i=1, \dots, n} (f_k(x_i) - y_i)^2$
- $L_i = (f_k(x_i) - y_i)^2 / D^2, i = 1, \dots, n$
- $\bar{L}_k = \sum_{i=1}^n w_i L_i$
- $\beta_k = \frac{\bar{L}_k}{1 - \bar{L}_k}$.
- $w_i \leftarrow w_i \times \beta_k^{1 - L_i}, i = 1, \dots, n.$
- $w_i \leftarrow \frac{w_i}{\sum_{j=1}^n w_j}, i = 1, \dots, n.$

Retourne $F: x \mapsto \text{median}_{(\beta)}(f_1(x), \dots, f_K(x))$, une médiane pondérée.

Qualitativement :

- Si beaucoup de L_i sont proches de 1, $\beta_k \geq 1$ et les poids s'uniformisent.
- Si beaucoup de L_i sont proches de 0, $\beta_k \leq 1$ et on renforce le poids des L_i grands.

Adaboost pour de la régression : $S = \{(x_i, y_i)\}_{i=1}^n$ pondérés avec des poids $w_i = 1/n$, $i = 1, \dots, n$ itération, pour $k = 1, 2, \dots, K$

- $f_k(\cdot) = h_n(\cdot, S, w)$
- $D = \max_{i=1, \dots, n} (f_k(x_i) - y_i)^2$
- $L_i = (f_k(x_i) - y_i)^2 / D^2, i = 1, \dots, n$
- $\bar{L}_k = \sum_{i=1}^n w_i L_i$
- $\beta_k = \frac{\bar{L}_k}{1 - \bar{L}_k}$.
- $w_i \leftarrow w_i \times \beta_k^{1 - L_i}, i = 1, \dots, n.$
- $w_i \leftarrow \frac{w_i}{\sum_{j=1}^n w_j}, i = 1, \dots, n.$

Retourne $F: x \mapsto \text{median}_{(\beta)}(f_1(x), \dots, f_K(x))$, une médiane pondérée.

Qualitativement :

- Si beaucoup de L_i sont proches de 1, $\beta_k \geq 1$ et les poids s'uniformisent.
- Si beaucoup de L_i sont proches de 0, $\beta_k \leq 1$ et on renforce le poids des L_i grands.

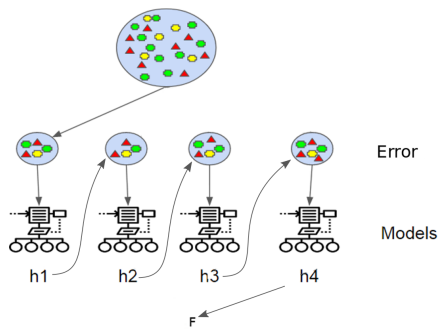
Effet du boosting :

- On améliore le biais d'un régresseur faible
- Pas d'effet positifs sur la variance.

On renforce un prédicteur faible, en essayant de corriger les erreurs d'entraînement,

- On cherche à prédire l'erreur d'entraînement du modèle précédent
- On agrège avec une somme pondérée.

Gradient boosting



Source hackernoon.com

Gradient boosting pour la régression : $S_0 = \{(x_i, y_i)\}_{i=1}^n$, et $f_0 = h_n(\cdot, S_0)$ pour $k = 1, 2, \dots, K$

- Nouveau jeu d'entraînement : $S_k = ((x_i, y_i - f_{k-1}(x_i)))_{i=1}^n$.
- $f_k(\cdot) = f_{k-1}(\cdot) + \gamma h_n(\cdot, S_k)$ pour un γ bien choisi.

Retourne $F : x \mapsto f_K(x)$.

Gradient boosting pour la régression : $S_0 = \{(x_i, y_i)\}_{i=1}^n$, et $f_0 = h_n(\cdot, S_0)$ pour $k = 1, 2, \dots, K$

- Nouveau jeu d'entraînement : $S_k = ((x_i, y_i - f_{k-1}(x_i)))_{i=1}^n$.
- $f_k(\cdot) = f_{k-1}(\cdot) + \gamma h_n(\cdot, S_k)$ pour un γ bien choisi.

Retourne $F: x \mapsto f_K(x)$.

Effet du gradient boosting :

- On améliore le biais d'un régresseur faible
- Pas d'effet positifs sur la variance.
- Souvent performant en pratique.

