Apprentissage supervisé
La science
pour « rendre une machine »
« dispose d'une forme »
d'intelligence

Mourad NACHAOUI

FST de Béni-Mellal

Sommaire

Apprentissage supervisé

Formalisation :

Soit $E_n := \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ un ensemble de données d'entraînement. Les X_i sont des variables d'entrées à valeur dans un ensemble $\mathcal{X}$. De même les Y_i sont des variables sortie à valeur dans un ensemble $\mathcal{Y}$. On appelle les X_i descripteurs, covariables, régresseurs, (en anglais features) et les Y_i étiquettes (en anglais labels).
Nous dénoterons par $\mathcal{X}$ un objet quelconque, x son vecteur représentant dans $\mathcal{X}$ et y la valeur cible associée à x.

Etant donné un ensemble d'entraînement E, on cherche à déterminer $f : \mathcal{X} \mapsto \mathcal{Y}$ une fonction modélisant la relation entre les $\mathcal{X}$ décrits dans l'espace de représentation $\mathcal{X}$ et la variable cible $\mathcal{Y}$:

$$f(X) = Y$$

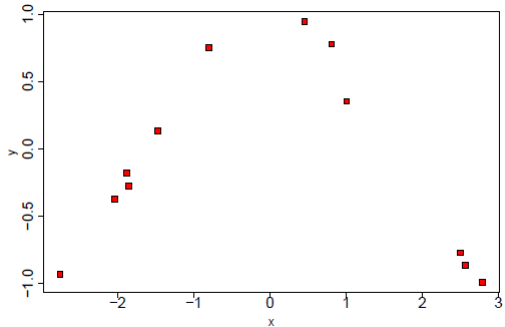
En revanche, ne connaissant pas la vraie nature de la relation entre $\mathcal{X}$ et $\mathcal{Y}$ et les données observées étant soit bruitées, soit incomplètes ; il n'est pas raisonnable de supposer une relation déterministe. Aussi, il est davantage raisonnable de poser le problème en les termes suivants :

$$f(X) = Y + \varepsilon$$

où ε est l'erreur ou le résidu. Autrement dit, il s'agit d'approximer f en commettant le moins d'erreurs possibles sur E tout en faisant de bonnes prédictions pour des valeurs de $\mathcal{X}$ non encore observées.

Exemple de problème de régression

L'objectif est de déterminer une fonction f qui étant donné un nouveau $x \in \mathbb{R}$ prédise correctement $y \in \mathbb{R}$



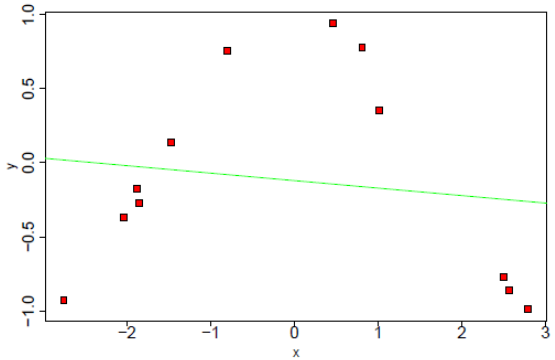
Régression linéaire simple

- Nous observons 12 couples de données avec en abscisse la variable X et en ordonnées la variable cible Y dont les éléments sont des réels.
- L'objectif est d'estimer une fonction $Y = f(X) + \epsilon$ qui représente la relation entre Y et X afin de prédire la valeur $\hat{y} = \hat{f}(\mathbf{x})$ pour une valeur de $\mathbf{x}$ quelconque.
- Pour un problème de régression on parlera également de **prédicteur** pour la fonction $\hat{f}$.
- En statistique une méthode très classique est donnée par les **Moindres Carrés Ordinaires (MCO)** que l'on notera par $scr(f)$ (somme des carrés des résidus ou "Residual Sum of Squares") :

$$\begin{aligned} scr(f) &= \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i))^2 \\ &= \sum_{i=1}^{n} (\epsilon_i)^2 \end{aligned}$$

- La régression linéaire simple consiste à prendre pour hypothèse que la relation f est un polynôme de degré 1 de X : $f(X) = a + bX$
- Ce qui nous donne :
$$scr(f) = scr(a, b) = \sum_{i=1}^{n} (y_i - (a + bx_i))^2$$
- $\mathbb{P} = \{a, b\}$ est l'ensemble des paramètres du modèle et on cherche les estimations $\hat{a}$ et $\hat{b}$ qui minimisent scr.
- Il faut déterminer les points critiques (ou stationnaires), solutions des équations normales (dérivées premières nulles). On obtient une solution analytique :
$$\hat{a} = \overline{\mathbf{y}} - \hat{b}\overline{\mathbf{x}} \text{ et } \hat{b} = \frac{\sum_{i=1}^{n} (x_i - \overline{\mathbf{x}})(y_i - \overline{\mathbf{y}})}{\sum_{i=1}^{n} (x_i - \overline{\mathbf{x}})^2}$$
où $\overline{\mathbf{y}} = \frac{1}{n} \sum_{i=1}^{n} y_i$ est la moyenne empirique de Y.
- Le modèle de prédiction est alors donné par :
$$\hat{f}(\mathbf{x}) = \hat{a} + \hat{b}\mathbf{x}$$

- Régression linéaire simple



Régression linéaire multiple (polynôme de degré > 1)

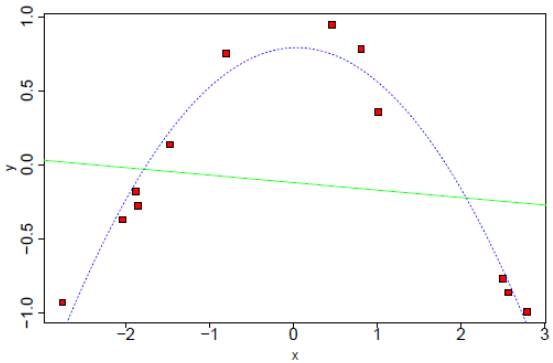
- La régression linéaire simple fait l'hypothèse que la fonction f est un polynôme de degré 1 et clairement ceci n'est pas une hypothèse raisonnable pour l'exemple traité.
- Autre type d'hypothèse : f est un polynôme de degré 2 de X : $f(X) = a + bX + cX^2$
- Dans ce cas $\mathbb{P} = \{a, b, c\}$ et on cherche à minimiser :

$$scr(f) = scr(a, b, c) = \sum_{i=1}^{n} (y_i - (a + bx_i + cx_i^2))^2$$

- Remarque : on parle de modèle linéaire car f est une fonction linéaire des paramètres $\mathbb{P}$! Les variables peuvent être tout type de fonction des variables initiales.

Régression linéaire multiple

- Régression linéaire multiple utilisant des fonctions de base polynômiales (jusqu'au degré 2).



Régression non paramètrique

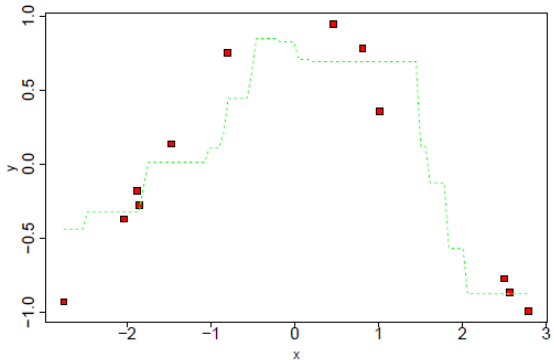
- La régression linéaire est un **modèle paramétrique** : on choisit une famille de fonctions avec un nombre fini de paramètres ($\mathbb{P}$) et le problème revient alors à estimer les paramètres qui minimisent $scr(\mathbb{P})$.
- Il existe des **modèles non paramétriques** de régression. Dans ce cas une hypothèse courante est basée sur les "**plus proches voisins**" : "deux objets similaires doivent avoir deux valeurs cibles similaires".
- La méthode la plus simple dans ce cas consiste à moyenner les y_i des $\mathbf{x}_i$ proches de $\mathbf{x}$. Formellement il s'agit d'étendre la méthode des moyennes mobiles et on obtient l'estimateur suivant :

$$\hat{f}(\mathbf{x}) = \frac{\sum_{i=1}^{n} K_\lambda(\mathbf{x}, \mathbf{x}_i) y_i}{\sum_{i=1}^{n} K_\lambda(\mathbf{x}, \mathbf{x}_i)}$$

où, pour notre exemple ci-dessous, $K_\lambda(\mathbf{x}, \mathbf{x}_i)$ vaut 1 si $\|\mathbf{x} - \mathbf{x}_i\| < \lambda$ et 0 sinon (boule centrée en $\mathbf{x}$ et de rayon λ).

Régression non paramètrique (suite)

- Avec $\lambda = 1$



Régression non paramètrique (suite)

- On peut réécrire $\hat{f}$ de la façon équivalente suivante :

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{n} \left(\frac{K_\lambda(\mathbf{x}, \mathbf{x}_i)}{\sum_{i=1}^{n} K_\lambda(\mathbf{x}, \mathbf{x}_i)} \right) y_i$$

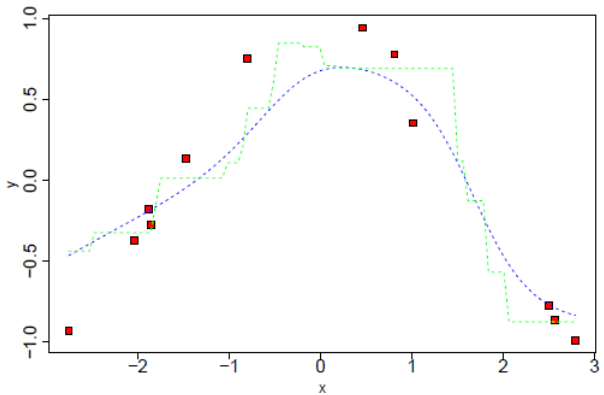
  On donne un poids uniforme non nul pour tout y_i dont le $\mathbf{x}_i$ appartient au voisinage de $\mathbf{x}$.
- Les **estimateurs à noyau** généralisent la méthode précédente en donnant des poids différents aux plus proches voisins $\mathbf{x}_i$ de $\mathbf{x}$ selon la distance entre $\mathbf{x}_i$ et $\mathbf{x}$. La fonction K_λ est de manière générale appelée fonction noyau (ou noyau de Parzen).
- Exemple du noyau gaussien :

$$K_\lambda(\mathbf{x}, \mathbf{x}_i) = \frac{1}{\lambda\sqrt{2\pi}} \exp\left(-\frac{1}{2} \left(\frac{\mathbf{x} - \mathbf{x}_i}{\lambda} \right)^2 \right)$$

- Pour toute fonction noyau, λ est un paramètre important qui permet de préciser la notion de voisinage autour de $\mathbf{x}$.

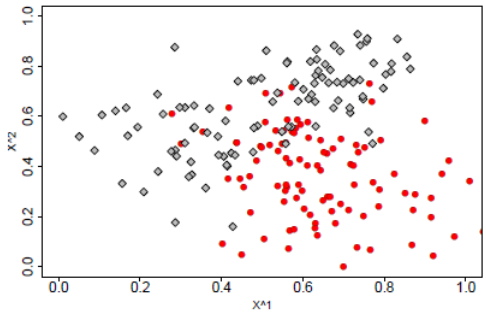
Régression non paramètrique (suite)

- Avec noyau gaussien et $\lambda = 1$.



Exemple de problème de catégorisation

- L'objectif est de déterminer une fonction $\hat{f}$ qui étant donné un nouveau $\mathbf{x} \in \mathbb{R}^2$ prédit correctement sa classe $y \in \{C_1, C_2\}$



Régression linéaire multiple avec variables artificielles

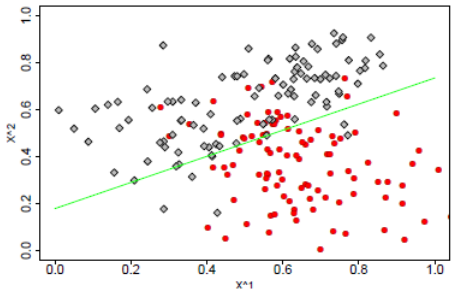
- On attribue des valeurs numériques à chacune des deux classes comme par exemple $C_1 \leftrightarrow 1$ et $C_2 \leftrightarrow -1$.
- On remplace Y variable discrète par Z une variable numérique remplie de -1 et 1.
- On traite le problème comme une régression linéaire multiple : $Z = g(X)$.
- Pour un nouveau $\mathbf{x}$ on applique la règle de décision suivante :

$$\hat{f}(\mathbf{x}) = \begin{cases} C_1 & \text{si } \hat{g}(\mathbf{x}) \geq 0 \\ C_2 & \text{si } \hat{g}(\mathbf{x}) < 0 \end{cases}$$

- La ligne de niveau $\{\mathbf{x} \in \mathbb{R}^2 : \hat{g}(\mathbf{x}) = 0\}$ est la **frontière de décision**.
- Pour un problème de catégorisation on parlera également de **classifieur** pour la fonction $\hat{f}$.

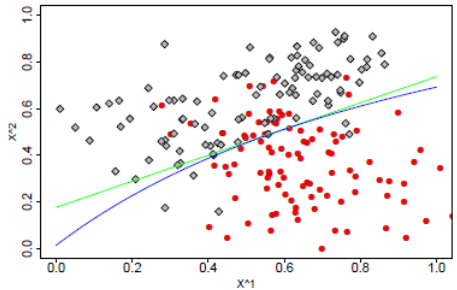
Régression linéaire multiple (suite)

- Hypothèse : $Z = g(X) = a + bX^1 + cX^2$ (polynôme de degré 1 des $\{X^j\}_{j=1}^2$)
- En vert on a tracé la frontière de décision.



Régression linéaire multiple (suite)

- Hypothèse : $Z = g(X) = a + bX^1 + cX^2 + dX^1X^2$ (polynôme de degré 2 des $\{X^j\}_{j=1}^2$).
- En bleu on a tracé la frontière de décision.



Etant donné un espace d'hypothèses $\mathbb{H}$, pour déterminer la fonction de prédiction (une instance de $\mathbb{H}$), il faut estimer les paramètres $\mathbb{P}$ qui optimisent un critère de performance sur les données E. Pour le problème de régression nous avons déjà évoqué les Moindres Carrés Ordinaires :

$$\mathcal{J}(f) = \sum_{i=1}^{n} (y_i - f(x_i))^2.$$

Lorsque les données n'ont pas toutes une importance uniforme, une façon de généraliser le $\mathcal{J}$ est l'utilisation de concepts issus de la décision statistique.

But d'apprentissage automatique

Prédire une donnée de sortie y à partir d'une donnée d'entrée x, ou bien plus généralement produire la meilleure action a à partir d'une donnée d'entrée x et en vue d'une donnée y qui n'est pas connue au moment où la décision est prise.

Difficulté

Y n'est pas une fonction déterministe de X.

- Il peut y avoir du bruit e.g. $Y = f(X) + \varepsilon$.
- Plus généralement, $Y = f(X, Z)$ où Z n'est pas observé. On peut difficilement faire bien systématiquement.

Approches possibles

- Essayer de faire bien dans le pire cas $\rightarrow$ approches théorie de jeux, stratégie minimax au coup par coup.
- Essayer de faire bien en moyenne. $\rightarrow$ objectif de l'apprentissage

Idée

Modéliser X et Y comme des variables aléatoires. $\rightarrow$ La meilleure décision "en moyenne" peut être prise à partir de $P(Y = |X = x)$. Cependant

- On ne la connait pas. Faire un modèle simple n'est pas possible.
- X et éventuellement Y sont des objets de grande dimension $\rightarrow$ le problème de déterminer $P(Y = |X = x)$ est a priori beaucoup plus difficile que le problème initial.

Information disponible

des observations de X : $(x_1, \ldots, x_n)$, ou des observations de (X, Y) : $(x_1, y_1), \ldots, (x_n, y_n)$.

Idée

Apprendre ! Utiliser une stratégie qui marche pour un ensemble d'observations existante et qui puisse se généraliser aux autres observations

Formalisme

- Les (x_i, y_i) sont des réalisations des variables aléatoires X_i, Y_i réelles. Le but est de minimiser l'espérance d'une mesure de « performance » par rapport à la distribution des données d'essai.
- Hypothèses classiques (jamais rencontrées en pratique ...): les variables aléatoires (x_i, y_i) sont indépendants et distribué de manière identique avec la même distribution que la distribution testing. Dans ce cours, nous allons ignorer le décalage potentiel entre les distributions d'échantillon d'entrainement et de test (bien que ce soit un sujet de recherche).
- Un algorithme d'apprentissage automatique $\mathcal{A}$ est alors une fonction qui part d'un jeu de données, c'est-à-dire un élément de $(X \times Y)^n$, à une fonction de X à Y.

$$\begin{aligned} \mathcal{A} : \bigcup_{n \in \mathbb{N}} (\mathcal{X} \times \mathcal{Y})^n &\to \mathbb{H} \\ E_n &\mapsto \hat{f}_n \end{aligned}$$

La fonction estimée $\hat{f}^n$ est construite en vue d'être utilisée pour prédire Y à partir d'un nouveau X où (X, Y) est une paire de données de test, c'est-à-dire pas nécessairement observée dans les données d'entraînement. Distinguer phase d'apprentissage et phase de test.

Évaluation pratique des performances

En pratique, nous n'avons pas accès à la distribution de test, mais à des échantillons de celle-ci. Dans la plupart des cas, compte tenu des données fournies au machine learning, il est divisé en trois parties:

- l'ensemble de formation, sur lequel seront estimés les modèles d'apprentissage,
- l'ensemble de validation, pour estimer les hyperparamètres (toutes les techniques d'apprentissage en ont),
- l'ensemble de test, pour évaluer les performances du modèle final (formellement, l'ensemble de test ne peut être utilisé une fois!)

entrainement	validation	test

cadre probabiliste

Soit (X, Y) les variables aléatoire formant les paires de données de test possible, distribuées selon une loi P ; en notation : $(X, Y) \backsim P_{X,Y}$.
Soit $\mathcal{A}$ un ensemble d'actions, de décisions ou de prédictions possibles.

fonction de perte

Soit : $\ell : \mathcal{A} \times \mathcal{Y} \to \mathbb{R}$ une fonction de perte, ou fonction de coût. La fonction de perte spécifie le prix à payer $\ell(a, y)$ pour avoir pris la décision a quand la variable de sortie prend la valeur y.

Risque

Soit une fonction de prédiction $f : \mathcal{X} \to \mathcal{A}$ et $\mathbb{H}$ un sous-ensemble de fonctions de $\mathcal{X}$ vers $\mathcal{A}$. Pour un problème de décision défini par $(X, Y) \backsim P_{X,Y}$, $\mathcal{A}$ et ℓ , on définit le risque $R(f)$ (au sens de Vapnik) pour une fonction de prédiction f:

$$R(f) := \mathcal{E}[\ell(f(X), Y)] = \int_{\mathcal{X}\times\mathcal{Y}} \ell(f(x), y) dP(x, y)$$

où dP est une mesure de probabilité sur $\mathcal{X} \times \mathcal{Y}$.

prédicteur

un prédicteur optimal est un prédicteur pour lequel le risque est minimal. S'il existe un prédicteur atteignant l'infimum du risque, ce prédicteur f^* ? est appelé fonction cible et on a

$$f^* := arg \min_{f \in \mathbb{H}} R(f)$$

prédicteur de Bayes

Notons par $R(a|X) = \mathcal{E}[\ell(a, Y)|X]$ le risque conditionnel. Le prédicteur de Bayes comme le prédicteur qui minimise le risque conditionnel au sens où :

$$f^*(X) := arg \min_{a \in \mathcal{A}} R(a|X).$$

Si le prédicteur de Bayes est dans $\mathbb{H}$, il est alors identique à la fonction cible.

excès de risque

On appelle excès de risque la différence entre le risque du prédicteur considéré et le risque de la fonction cible. C'est la quantité

$$R(\hat{f}_n) - R(f^*) = \mathbb{E}[\ell(\hat{f}_n(X), Y)|E_n] - \inf_{f \in \mathbb{H}} \mathbb{E}[\ell(f(X), Y)|Y]$$

Exemples

exemple 1:(régression au sens des moindres carrés : perte quadratique)

Cas: $\mathcal{A} = \mathcal{Y} = \mathbb{R}$

- fonction de perte: $\ell(a, y) = \frac{1}{2}(a - y)^2$.
- risque : $R(f) = \frac{1}{2}\mathbb{E}[(f(X) - Y)^2]$.
- fonction cible : $f^*(X) = \mathbb{E}[Y|X)]$

exemple 2:(classification à K-classes : perte 0-1)

Cas: $\mathcal{A} = \mathcal{Y} = \{0, \ldots, n\}$.

- fonction de perte: $\ell(a, y) = 1_{a \neq y}$.
- risque : $R(f) = P(f(X) \neq Y)$.
- fonction cible : $f^*(X) = arg\max_k P(Y = k|X))$.

Idée

estimer le risque grâce à l'ensemble d'apprentissage disponible, i.e remplacer la distribution de probabilité $P_{X,Y}$ par la distribution empirique $P_n = \frac{1}{n}\sum_{i=1}^{n} \delta(x_i, y_i)$[1]

Risque empirique

On définit donc le risque empirique

$$\hat{R}_n(f) := \mathbb{E}_n[\ell(f(X), Y)] = \frac{1}{n}\sum_{i=1}^{n} \ell(f(x_i), y_i).$$

Le principe de minimisation du risque empirique

$$\mathcal{S} \subset \mathbb{H}, \quad \min_{f \in \mathcal{S}} \hat{R}_n(f) = \min_{f \in \mathbb{H}} \underbrace{\frac{1}{n}\sum_{i=1}^{n} \ell(f(x_i), y_i)}_{\text{erreur d'entraînement}}$$

[1] $P_n \to P_{X,Y}$ as $n \to \infty$ dans un sens qui peut être formalisé – au sens faible par la loi des grands nombres, et au sens fort grâce au théorème de Glivenko-Cantelli, également appelé "théorème fondamental de la statistique".

Exemple 3. (La régression linéaire)

On considère le cas $\mathcal{X} = \mathbb{R}^p, \mathcal{Y} = \mathbb{R}$ et ℓ est la perte quadratique. On se restreint à des fonctions linéaires de la forme $f_w : x \mapsto w^T x$. L'espace d'hypothèse est donc $\mathcal{S} = \{f_w | w \in \mathbb{R}^p\}$. On a alors :

$$\hat{R}_n(f) = \frac{1}{2n} \sum_{i=1}^{n} (w^T x_i - y_i)^2 = \frac{1}{2n} \|Xw - Y\|_2^2$$

avec $y^T = (y_1, \ldots, y_n) \in \mathbb{R}^n$ le vecteur de sorties, $X \in \mathbb{R}^{n*p}$ la matrice de design. Le problème $\min_{w \in \mathbb{R}^p} \hat{R}_n(f_w)$ est résolu par les équations normales

$$X^T X w - X^T y = 0.$$

Problème : $X^T X$ n'est pas inversible quand $p > n$, le prédicteur n'est pas unique.
Si $X^T X$ est inversible, alors

$$\hat{f}_{\mathcal{S}}(x') = x'^T (X^T X)^{-1} X^T y.$$

(Consistance par rapport à une loi P)

Pour des données d'entraînement et de test i.i.d de loi P, on dit que l'algorithme d'apprentissage est consistant si le prédicteur qu'il définit satisfait

$$\lim_{n\to\infty} \mathbb{E}[R(\hat{f}_n)] - R(f^*) = 0.$$

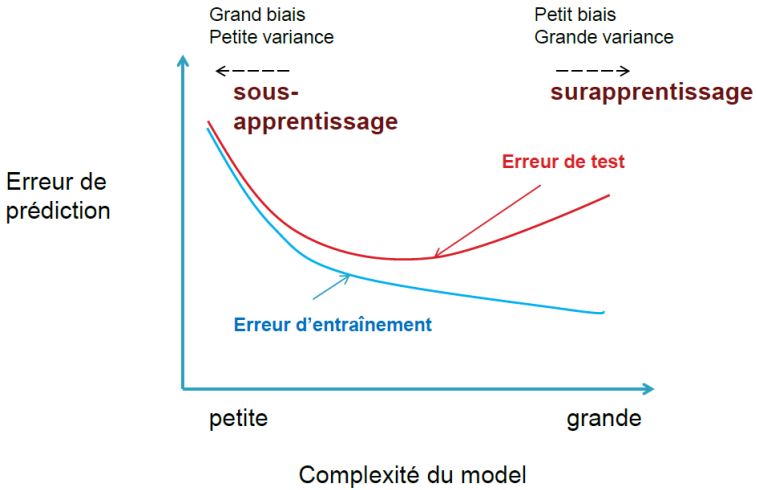
Consistance universelle

On dit qu'un algorithme d'apprentissage est universellement consistant s'il est consistant pour toute loi P.

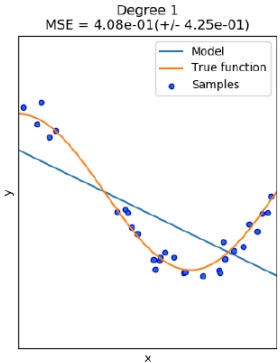
Phénomène de surapprentissage

- Le choix d'une méthode revient à choisir un espace d'hypothèses, une fonction de perte et une technique d'inférence.
- Ce qu'on attend d'une bonne méthode n'est pas tant sa capacité à reproduire à l'identique le résultat des données d'entraînement mais de produire les résultats corrects sur des données de test c-à-d non observées : c'est le principe de généralisation.
- Dans cette perspective il faut une bonne adéquation entre la complexité de la classe d'hypothèse choisie $\mathbb{H}$ et la véritable relation entre X et Y . Si la complexité de $\mathbb{H}$ n'est pas assez suffisante on parle de sous-apprentissage.
- Quand au contraire, la complexité de $\mathbb{H}$ est trop grande, il arrive que l'erreur sur E soit proche de zéro alors que l'erreur sur les données de test est grande. Dans ce cas on parle de sur-apprentissage.

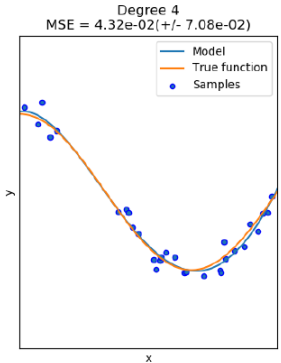
Phénomène de surapprentissage



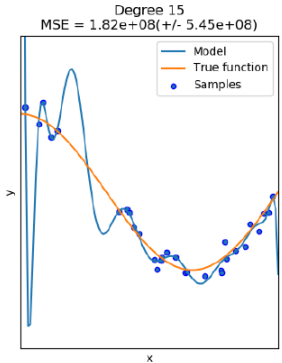
Régression par polynomes de degré d : comment choisir d ? $\rightarrow$ minimisation de $\hat{R}_n$ sur $\mathcal{S}_d = \{x \mapsto \sum_{i=1}^{d} a_i x_i^d\}$. Remarque : $\mathcal{S}_1 \subset \mathcal{S}_4 \subset \mathcal{S}_{15} \subset \mathbb{H}$



$\mathcal{S}_1$



$\mathcal{S}_4$



$\mathcal{S}_{15}$

sous-apprentissage (under-tting)
biais élévée, variance faible

sur-apprentissage (over-tting)
biais faible variance élevée

Dans l'exemple de régression, la complexité d'un modèle ou d'une classe d'hypothèses $\mathbb{H}$ est l'ordre du polynôme.

- Si l'ordre est trop petit, il y a sous-apprentissage et s'il est trop grand il y a sur-apprentissage.
- Pour le polynôme de degré 1 :
  - la complexité des données et celle du modèle ne coïncident pas,
  - l'erreur mesurée sur les données E est très grande,
  - mais la fonction de prédiction étant une droite la variance du modèle est faible (si on change E la "droite changera peu").
- Pour le polynôme de degré 15 :
  - la complexité des données et celle du modèle ne coïncident pas,
  - l'erreur mesurée sur les données E est très faible,
  - mais la fonction de prédiction est instable et donc la variance du modèle est très grande (si on change E la "courbe changera beaucoup").
- Pour le polynôme de degré 4 : compromis entre le bais et la variance.

Sélection de modèle

- Rappel : base de test et base d'apprentissage sont supposées représentatives des données et indépendantes
  $\rightarrow$ l'erreur sur la base de test est une estimation du risque moyen de prédiction
  $\rightarrow$ on choisit le modèle $\mathbb{H}$ minimisant l'erreur sur la base de test
- On remarque :
  - erreur d'apprentissage $\ll$ erreur de test : sur-apprentissage
  - erreur d'apprentissage $\simeq$ erreur de test, et erreurs "grandes" : sous-apprentissage
  - erreur d'apprentissage $\simeq$ erreur de test, et erreurs "petites" : OK

Décomposition du risque

Soit un risque $R(\cdot)$ défini par la donnée de v.a. X, Y et par une fonction de perte ℓ. On se donne une règle d'apprentissage dont le codomaine est l'espace d'hypothèse $\mathcal{S} \subset \mathbb{H}$ on définit :

- la fonction cible $f^* = arg\min_{f \in \mathbb{H}} R(f)$
- la meilleur approximation de la fonction cible dans $\mathcal{S}$: $f_{\mathcal{S}}^* := arg\min_{f \in \mathcal{S}} R(f)$
- le prédicteur obtenu par la règle d'apprentissage à partir de données $\hat{f}_{\mathcal{S}}$.

On a la décomposition :

$$\underbrace{R(\hat{f}_{\mathcal{S}}) - R(f^*)}_{\text{excès de risque}} = \underbrace{R(\hat{f}_{\mathcal{S}}) - R(f_{\mathcal{S}}^*)}_{\text{erreur d'estimation}} + \underbrace{R(f_{\mathcal{S}}^*) - R(f^*)}_{\text{erreur d'approximation}}$$

Compromis Biais-Variance

Cette décomposition est plus adaptée aux règles d'apprentissage autre que la minimisation du risque empirique. On a :

$$\mathbb{E}[R(\hat{f}] = \mathbb{E}[(\hat{f}(X) - Y)^2] = \underbrace{\mathbb{E}[(\hat{f}(X) - \mathbb{E}[\hat{f}(X)|X])^2]}_{\text{variance de } \hat{f}} + \underbrace{\mathbb{E}[(\mathbb{E}[\hat{f}(X)|X] - \mathbb{E}[Y|X])^2]}_{\text{bais de } \hat{f}} + \underbrace{\mathbb{E}[(Y - \mathbb{E}[Y|X])^2]}_{\text{variance du bruit}}$$

Pour bien interpréter cette expression : par exemple, $\mathbb{E}[\hat{f}(X)|X]$ est l'espérance de $\hat{f}(X)$ par rapport aux variations de données d'entraînement E_n, pour un X (test) fixé (qui est indépendant de E_n). Aussi, les trois termes peuvent être vus comme intégrer les "variances" et "biais" par rapport à la loi sur les X.

Contrôle de la complexité

Un problème mal posé

On dit qu'un problème est bien posé au sens de Hadamard si

- Il admet une solution
- Cette solution est unique
- La solution dépend de façon continue des paramètres du problème dans une topologie bien choisie.

Problème de l'apprentissage comme un problème essentiellement mal posé car sous-contraint et disposant d'information par essence incomplète

Espace d'hypothèse et régularisation

- Contrôle explicite de la complexité : degré du polynôme, choix de la largeur de bande pour les méthodes de lissage, choix des variables, etc $\rightarrow$ problème de choix de l'espace d'hypothèse $\mathcal{S}$.
- Contrôle implicite de la complexité pour les méthodes par minimisation du risque empirique : régularisation de Tikhonov.

Le principe de la régularisation est de pénaliser la valeur d'une norme $f \mapsto \|f\|$ qui "contrôle la complexité" de la fonction f

$$\min_{f \in \mathcal{S}} \hat{R}_n(f) + \lambda \|f\|$$

exemple : norme hilbertienne, norme ℓ_q, norme de Sobolev, Total Variation TV...

Espace d'hypothèse et régularisation

La régularisation induit un compromis entre la minimisation du risque empirique et le choix d'une fonction trop complexe. Elle a l'avantage que la complexité de la fonction ne doit pas être connue à l'avance. Le compromis est contrôlé par λ le paramètre de régularisation ou hyperparamètre. Il fait néanmoins choisir $\lambda \rightarrow$ problème analogue au problème de sélection de modèle. Nous allons couvrir cela plus en détails sur le cours de la sélection de modèle.

Régression ridge

Forme de régularisation la plus classique en statistiques

$$\min_{w \in \mathbb{R}^p} \frac{1}{2n} \|y - Xw\|_2^2 + \frac{\lambda}{2} \|w\|_2^2.$$

Grâce à la régularisation, le problème est devenu fortement convexe (nous allons revoir ce concept dans le cours d'analyse convexe). Donc la solution est unique :

$$w^{(ridge)} = (X^T X + n\lambda I)^{-1} X^T y$$

Effet de lissage du spectre de la matrice de design. Notion de shrinkage. La régularisation a pour effet de transformer le problème en un problème bien posé au sens de Hadamard. Le paramètre de régularisation contrôle le conditionnement de la matrice.