Introduction à l'apprentissage

Cristina Butucea

Septembre 2023

Références

- An Introduction to Statistical Learning, with Applications in R;
  James, Witten, Hastie, Tibshirani, *Springer Texts in Statistics*

- The Elements of Statistical Learning, Data Mining, Inference and Prediction;
  Hastie, Tibshirani, Friedman, *Springer Series in Statistics*;

- Pattern recognition and machine learning;
  Bishop; *Springer Science*

- Régression avec R
  Cornillon, Matzner-Løber;

- A Probabilistic Theory of Pattern Recognition;
  Devroye, Lugosi, *Springer Verlag*;

Introduction à l'apprentissage

Historiquement,

- -'40-'50 formalisme mathématique basé sur la logique et le calcul symbolique
- -'60-'70 IA pour certains comportements "appris" à partir des données - sans analyse statistique
- -'80 réseaux de néurones artificiels (perceptron de Rosenblatt '57)
- depuis '90 apprentissage statistique: formalisme mathématique basé sur proba-stat, théorie de l'information et optimisation

L'évolution suit l'augmentation de la puissance de calcul!

	Démarche statistique	vs. Apprentissage
Données	issues d'un modèle	dégagent un modèle
Modèle	à estimer	à reproduire / prédire
Evaluation risque	comparaison au modèle théorique	qualité de prédiction

On distingue:

- Apprentissage supervisé
  * Les données $X_i \in \chi$ ont des 'labels' $Y_i \in \mathcal{Y}$
  * Pour un nouveau X, prédire $\tilde{Y}$
  * Exemples: publicité, reconnaissance vocale ou manuscrite

On distingue:

- Apprentissage supervisé
  - Les données $X_i \in \chi$ ont des 'labels' $Y_i \in \mathcal{Y}$
  - Pour un nouveau X, prédire $\tilde{Y}$
  - Exemples: publicité, reconnaissance vocale ou manuscrite
- Apprentissage non-supervisé
  - Les données X_i dans χ n'ont pas de 'labels'
  - Trouver une ou des structures
  - Exemples: séparer en groupes, trouver un espace approximant de plus petite dimension

On distingue:

- Apprentissage supervisé
  - Les données $X_i \in \chi$ ont des 'labels' $Y_i \in \mathcal{Y}$
  - Pour un nouveau X, prédire $\tilde{Y}$
  - Exemples: publicité, reconnaissance vocale ou manuscrite
- Apprentissage non-supervisé
  - Les données X_i dans χ n'ont pas de 'labels'
  - Trouver une ou des structures
  - Exemples: séparer en groupes, trouver un espace approximant de plus petite dimension
- Apprentissage semi-supervisé

On distingue:

- Apprentissage supervisé
  - Les données $X_i \in \chi$ ont des 'labels' $Y_i \in \mathcal{Y}$
  - Pour un nouveau X, prédire $\tilde{Y}$
  - Exemples: publicité, reconnaissance vocale ou manuscrite
- Apprentissage non-supervisé
  - Les données X_i dans χ n'ont pas de 'labels'
  - Trouver une ou des structures
  - Exemples: séparer en groupes, trouver un espace approximant de plus petite dimension
- Apprentissage semi-supervisé
- Apprentissage actif (Reinforcement Learning)

Apprentissage supervisé

But: prédire une donnée de sortie $\tilde{Y}$,
à partir d'une donnée (éventuellement multivariée) d'entrée X.

Apprentissage supervisé

But: prédire une donnée de sortie $\tilde{Y}$,
à partir d'une donnée (éventuellement multivariée) d'entrée X.

Cette prédiction sera basée sur les données disponibles:
$(X_1, Y_1), ..., (X_n, Y_n)$.

Apprentissage supervisé

But: prédire une donnée de sortie $\tilde{Y}$,
à partir d'une donnée (éventuellement multivariée) d'entrée X.

Cette prédiction sera basée sur les données disponibles:
$(X_1, Y_1), ..., (X_n, Y_n)$.

Exemple

Saisie automatique d'écriture manuscrite - les chiffres.
Ici: X est un vecteur de taille p avec des valeurs comprises entre 0 et 1 (pour chaque pixel), X appartient à $[0,1]^p$. La réponse Y appartient à $\{0, 1, ..., 9\}$.

Autres exemples: reconnaissance vocale et d'images, systèmes de recommandation, scoring bancaire, voitures automatiques, soins de santé.

Prédicteur

Hypothèse: On observe $\mathcal{D} = \mathcal{D}_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ i.i.d., de loi P. On suppose que X_i sont dans $\mathbb{R}^p$ et Y_i dans un ensemble $\mathcal{Y}$.

Prédicteur

Hypothèse: On observe $\mathcal{D} = \mathcal{D}_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ i.i.d., de loi P. On suppose que X_i sont dans $\mathbb{R}^p$ et Y_i dans un ensemble $\mathcal{Y}$.

On appelle

X_i descripteurs, covariables, régresseurs, "features", design,

Y_i réponses, variables dépendantes (des X!), étiquettes ou "labels" en classification.

Prédicteur

Hypothèse: On observe $\mathcal{D} = \mathcal{D}_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ i.i.d., de loi P. On suppose que X_i sont dans $\mathbb{R}^p$ et Y_i dans un ensemble $\mathcal{Y}$.

On appelle

X_i descripteurs, covariables, régresseurs, "features", design,

Y_i réponses, variables dépendantes (des X!), étiquettes ou "labels" en classification.

On veut prédire $\tilde{Y}$ d'une nouvelle donnée X, en utilisant $\mathcal{D}_n$.

On appelle
Prédicteur $f : \mathbb{R}^p \to \mathcal{Y}$ et
Prédiction $\tilde{Y} = f(X)$ la réponse d'un nouvel individu caractérisé par X.

On appelle
Prédicteur $f : \mathbb{R}^p \to \mathcal{Y}$ et
Prédiction $\tilde{Y} = f(X)$ la réponse d'un nouvel individu caractérisé par X.

Exemple

$\mathcal{Y} = \{0, 1\}$ *dans un problème de classification binaire;*
Le prédicteur s'appelle aussi classifieur.

Exemple

$\mathcal{Y} = \mathbb{R}$ *dans un problème de régression ou de scoring.*

On appelle
Prédicteur $f : \mathbb{R}^p \to \mathcal{Y}$ et
Prédiction $\tilde{Y} = f(X)$ la réponse d'un nouvel individu caractérisé par X.

Exemple

$\mathcal{Y} = \{0, 1\}$ *dans un problème de classification binaire;*
Le prédicteur s'appelle aussi classifieur.

Exemple

$\mathcal{Y} = \mathbb{R}$ *dans un problème de régression ou de scoring.*

Pour comparer différents prédicteurs on cherche à évaluer le **risque de prédiction.**

Fonction de perte

On choisit une fonction de perte ou coût de la prédiction

$$\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+,$$

qui s'applique à l'observation Y et à sa prédiction $\tilde{Y} = f(X)$. Exemples

Fonction de perte

On choisit une fonction de perte ou coût de la prédiction

$$\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+,$$

qui s'applique à l'observation Y et à sa prédiction $\tilde{Y} = f(X)$. Exemples

Y	catégorielle	$\ell(Y, f(X))$
	perte 0-1	$I(Y \neq f(X))$
	exponen. ou AdaBoost	$\exp(-Y \cdot f(X))$;
	LogitBoost	$\log(1 + \exp(-2Y \cdot f(X)))$;
Y	continue	
	$\mathbb{L}_2$ ou quadratique	$\frac{1}{2}(Y - f(X))^2$;
	$\mathbb{L}_1$	$\lvert Y - f(X)\rvert$;
	Huber	$\begin{cases} \frac{1}{2} \cdot (Y - f(X))^2, & \text{si } \lvert Y - f(X)\rvert \leq \delta \\ \delta \cdot \lvert Y - f(X)\rvert - \frac{1}{2} \cdot \delta^2, & \text{sinon.} \end{cases}$

Risque de prédiction

Risque du prédicteur f:

$$R(f) = E_{(X,Y)\sim P}\left[\ell(Y, f(X))\right].$$

Risque de prédiction

Risque du prédicteur f:

$$R(f) = E_{(X,Y)\sim P}\left[\ell(Y, f(X))\right].$$

Ce risque ne peut être évalué, car la loi P est inconnue.

Risque de prédiction

Risque du prédicteur f:

$$R(f) = E_{(X,Y)\sim P}\left[\ell(Y, f(X))\right].$$

Ce risque ne peut être évalué, car la loi P est inconnue.
Typiquement, on sépare l'échantillon en deux parties:

- un échantillon 'train' $\mathcal{D}_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ pour apprendre f
- un échantillon 'test' $\mathcal{T}_m = \{(X'_1, Y'_1), ..., (X'_m, Y'_m)\}$ pour évaluer/estimer son risque par

$$\hat{R}(f) = \frac{1}{m}\sum_{i=1}^{m} \ell(Y'_i, f(X'_i)).$$

Prédicteur optimal

Un prédicteur f^* est dit **optimal** si son risque est minimal:

$$f^* \in \arg\min_{f \in \mathcal{F}} R(f).$$

Prédicteur optimal

Un prédicteur f^* est dit **optimal** si son risque est minimal:

$$f^* \in \arg\min_{f \in \mathcal{F}} R(f).$$

La fonction f^* est également appelée **fonction cible** ou **oracle**. On note

$$R^* = R(f^*).$$

Des formes explicites existent dans certains cas particuliers, cf. aux exemples suivants.

Exemples

1. Pour la perte quadratique, le risque s'écrit

$$\begin{aligned} R(f) &= E\left[(Y - f(X))^2\right] \\ &= E\left[E[(Y - E(Y|X))^2]|X\right] + E\left[(E(Y|X) - f(X))^2\right] \end{aligned}$$

Alors $f^*(X) = E(Y|X)$ dit **régresseur de Bayes** minimise ce risque et

$$R^* = R(f^*) = E\left[(Y - E(Y|X))^2\right].$$

Exemples

1. Pour la perte quadratique, le risque s'écrit

$$\begin{aligned} R(f) &= E\left[(Y - f(X))^2\right] \\ &= E\left[E[(Y - E(Y|X))^2]|X\right] + E\left[(E(Y|X) - f(X))^2\right] \end{aligned}$$

Alors $f^*(X) = E(Y|X)$ dit **régresseur de Bayes** minimise ce risque et

$$R^* = R(f^*) = E\left[(Y - E(Y|X))^2\right].$$

2. Pour la perte $\mathbb{L}_1$, f^* est la médiane a posteriori (MAP), c-à-d la médiane de la loi de Y/X.

3. Pour la classification binaire et la perte $\ell(Y, f(X)) = I(Y \neq f(X))$, le risque $R(f) = P(Y \neq f(X))$:

$$f^* = I\left(E(Y|X) \geq \frac{1}{2}\right)$$

est le **classifieur de Bayes**.

3. Pour la classification binaire et la perte $\ell(Y, f(X)) = I(Y \neq f(X))$, le risque $R(f) = P(Y \neq f(X))$:

$$f^* = I\left(E(Y|X) \geq \frac{1}{2}\right)$$

est le **classifieur de Bayes**.

Remarque: Les estimateurs et classifieurs Bayésiens ont le plus petit risque:

$$R(f^*) \leq R(f), \quad \text{pour tout autre } f,$$

mais ne sont **pas calculables!**
Ils dépendent de $\eta(X) := E(Y|X)$, c-à-d de la loi P inconnue!

Méthodes plug-in

On note $\eta(X) = E(Y|X)$ et on rappelle que :

1. pour la régression et la perte quadratique, le régresseur de Bayes

$$f^*(x) = \eta(X)$$

2. pour la clasiffication binaire et perte 0-1, le classifieur de Bayes

$$f^*(X) = I\left(\eta(X) \geq \frac{1}{2}\right).$$

Méthodes plug-in

On note $\eta(X) = E(Y|X)$ et on rappelle que :

1. pour la régression et la perte quadratique, le régresseur de Bayes

$$f^*(x) = \eta(X)$$

2. pour la clasiffication binaire et perte 0-1, le classifieur de Bayes

$$f^*(X) = I\left(\eta(X) \geq \frac{1}{2}\right).$$

Méthodes de plug-in: -Estimer $\eta(X)$ par $\hat{\eta}(X)$ en utilisant $\mathcal{D}_n$
- 'Plug-in' la formule, i.e. pour la régression

$$\hat{f}(X) = \hat{\eta}(X)$$

pour la classification

$$\hat{f}(X) = I\left(\hat{\eta}(X) \geq \frac{1}{2}\right).$$

kNN

On dispose de l'échantillon $\mathcal{D}_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ i.i.d., de loi P.

kNN

On dispose de l'échantillon $\mathcal{D}_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ i.i.d., de loi P.

Une distance d est définie entre les points de χ. Par exemple, la distance euclidienne:

$$d(X_i, X_j) = \|X_i - X_j\|.$$

On appelle voisins de X, les points $X_1, ..., X_n$ rangés du plus proche au plus éloignés de X.

kNN

On dispose de l'échantillon $\mathcal{D}_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ i.i.d., de loi P.

Une distance d est définie entre les points de χ. Par exemple, la distance euclidienne:

$$d(X_i, X_j) = \|X_i - X_j\|.$$

On appelle voisins de X, les points $X_1, ..., X_n$ rangés du plus proche au plus éloignés de X.
1NN: le plus proche voisin de X est le point X_k tel que

$$d(X, X_k) = \min_{j=1,...,n} d(X, X_j).$$

kNN

On dispose de l'échantillon $\mathcal{D}_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ i.i.d., de loi P.

Une distance d est définie entre les points de χ. Par exemple, la distance euclidienne:

$$d(X_i, X_j) = \|X_i - X_j\|.$$

On appelle voisins de X, les points $X_1, ..., X_n$ rangés du plus proche au plus éloignés de X.
1NN: le plus proche voisin de X est le point X_k tel que

$$d(X, X_k) = \min_{j=1,...,n} d(X, X_j).$$

kNN: les $k \geq 1$ plus proches voisins de X,

$$X_{i_1}, ..., X_{i_k}$$

ayant les plus petites distances à X.

On appelle **classifieur kNN:**
pour la régression

$$\hat{\eta}^{k\mathrm{NN}}(X) = \frac{1}{k}\left(Y_{i_1} + \ldots Y_{i_k}\right);$$

pour la classification

$$\hat{f}^{k\mathrm{NN}}(X) = I\left(\hat{\eta}^{k\mathrm{NN}}(X) \geq \frac{1}{2}\right).$$

C'est un vote à la majorité.
Il ne dépend des X_i que via le choix des k plus proches voisins, qui votent.

Regressogramme

On dispose de l'échantillon $\mathcal{D}_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ i.i.d., de loi P.

Étant donnée une partition de l'espace χ contenant tous les X_i:

$$C_1, ..., C_k, \quad \text{on note } |C_k| \text{ l'effectif de la } k\text{ème classe}$$

et

$$\hat{\mu}_k = \frac{1}{|C_k|} \sum_{i:X_i \in C_k} Y_i,$$

la moyenne des réponses Y_i dans la classe C_k.

Alors, le régressogramme est constant, égal à $\hat{\mu}_k$ sur chaque classe C_k:

$$\hat{f}^R(X) = \hat{\mu}_k \quad \text{si } X \in C_k.$$

Alors, le régressogramme est constant, égal à $\hat{\mu}_k$ sur chaque classe C_k:

$$\hat{f}^R(X) = \hat{\mu}_k \quad \text{si } X \in C_k.$$

Le classifieur plug-in du régressogramme est

$$\hat{f}^R(X) = I(\hat{\mu}_k \geq \frac{1}{2}) \quad \text{si } X \in C_k.$$

Nadaraya-Watson

On choisit un noyau (*kernel*) K et une fenêtre (*bandwidth*) $h > 0$ et on estime la régression au point X par

$$\hat{\eta}^{NW}(X) = \frac{\sum_{i=1}^{n} Y_i K\left(\frac{X - X_i}{h}\right)}{\sum_{i=1}^{n} K\left(\frac{X - X_i}{h}\right)},$$

si dénominateur non nul!
Classifieur

$$f^{NW}(X) = I\left(\hat{\eta}^{NW}(X) \geq \frac{1}{2}\right).$$

Ici, pas de binning, mais fenêtre glissante!

Règle de prédiction

Si le prédicteur f est estimé à partir de $\mathcal{D}_n$, on appelle **règle de prédiction**

$$\begin{aligned} \hat{f} : \mathbb{R}^p \quad \times \quad & \left(\bigcup_{n \geq 1} (\mathbb{R}^p \times \mathcal{Y}) \right) \to \mathcal{Y} \\ (X \quad , \quad & \mathcal{D}_n) \to \hat{f}(X) = \hat{f}(X, \mathcal{D}_n). \end{aligned}$$

Le risque de $\hat{f}$ est:

$$R(\hat{f}) = E\left[\ell(Y, \hat{f}(X))|\mathcal{D}_n\right].$$

On appelle $\mathcal{D}_n$ - **l'échantillon d'apprentissage** (train sample).

Une règle de prédiction est dite
-*universellement consistante*, si $E_P[R(\hat{f})] \to R^*$, pour toute loi P.
-*universellement uniformément consistante*, si

$$\sup_P \left\{ E_P[R(\hat{f})] - R^* \right\} \to 0, \quad n \to \infty.$$

On appelle *excès de risque* (excess risk) d'une régle de prédiction $\hat{f}$:

$$R(\hat{f}) - R(f^*).$$

Nous pouvons établir (dans beaucoup de cas) des *vitesses de convergences* de l'excès de risque, c'est-à-dire trouver φ_n qui décroit vers 0 telle que:

$$0 \leq R(\hat{f}) - R(f^*) \leq \varphi_n, \quad \text{avec grande probabilité.}$$

On peut s'intéresser à l'*optimalité* de ces vitesses.

Le risque $R(\hat{f})$ ne peut être calculé explicitement, mais approximé à partir des données d'un autre échantillon

$$\mathcal{T}_m = \{(X'_1, Y'_1), ..., (X'_m, Y'_m)\}$$

dit **échantillon de test** (test sample).

On calcule un estimateur du risque:

$$\hat{R}(\hat{f}) = \frac{1}{m} \sum_{i=1}^{m} \ell(\hat{f}(X'_i), Y'_i).$$

Ici $\mathcal{Y} = \{0, 1\}$ et $\ell(Y, f(X)) = I(Y \neq f(X))$, donc

$$R(f) = P(Y \neq f(X)).$$

On note $\eta(X) = E(Y|X)$ la fonction de régression de Y sachant X.

Classifieur de Bayes

$$f^*(X) = I(\eta(X) \geq \frac{1}{2}).$$

Théorème

Pour toute autre règle f, on a $R(f) \geq R(f^*)$.

On peut aussi écrire

$$\begin{aligned} R(f^*) &= E\left[I(f^*(X) = 1)(1 - \eta(X)) + I(f^*(X) = 0)\eta(X)\right] \\ &= E\left[\min\{\eta(X), 1 - \eta(X)\}\right] \\ &= \frac{1}{2} - E\left[\left|\eta(X) - \frac{1}{2}\right|\right]. \end{aligned}$$

Conséquences: Pour toute loi P, $R(f^*) \leq \frac{1}{2}$ et $R(f^*) = \frac{1}{2}$ seulement quand

$$\eta(X) = \frac{1}{2}, \quad \text{avec probabilité 1.}$$

Preuve du théorème: Pour tout classifieur f,

$$R(f) = \frac{1}{2} - E\left[(\eta(X) - \frac{1}{2})(I(f(X) = 1) - I(f(X) = 0))\right].$$

Ainsi,

$$R(f) - R(f^*) = E\left[2A \cdot (\eta(X) - \frac{1}{2})\right],$$

avec $A = I(f^*(X) = 1) - I(f(X) = 1) = I(f(X) = 0) - I(f^*(X) = 0)$.
Soit $A = 0$, soit $A = sgn(\eta(X) - \frac{1}{2})$, donc

$$R(f) - R(f^*) = E[|2\eta(X) - 1|] \geq 0, \quad \text{pour tout } f.$$

On appelle un classifieur de type plug-in si il est basé sur un estimateur $\hat{\eta}$ de η:

$$\hat{f}(X) = I(\hat{\eta}(X) \geq \frac{1}{2}).$$

Théorème

Pour tout autre classifieur de type plug-in, c-à-d basé sur $\tilde{\eta}$: $\tilde{f}(X) = I(\tilde{\eta}(X) \geq \frac{1}{2})$,

$$R(\tilde{f}) - R(f^*) \leq 2E\left[|\tilde{\eta}(X) - \eta(X)|\right].$$

Preuve du théorème: Comme précédemment,

$$R(\tilde{f}) - R(f^*) = E[A \cdot (2\eta(X) - 1)] = E[|A| \cdot |2\eta(X) - 1|].$$

Soit $A = 0$ et l'inégalité est triviale, soit $A \neq 0$ et on voit que $|A| = I(\tilde{f} \neq f^*)$. On peut vérifier que sous cet évènement:

$$|2\eta(X) - 1| \leq 2|\eta(X) - \tilde{\eta}(X)|.$$

Méthodes de type plug-in pour des variables explicatives quantitatives.
Cas binaire: π_0, π_1 sont $P(Y = 0)$, $P(Y = 1)$, respectivement,
Les densités f_0 et f_1 de chaque groupe sont supposées gaussiennes,
$N(\mu_0, \Sigma_0)$ et $N(\mu_1, \Sigma_1)$!

Méthodes de type plug-in pour des variables explicatives quantitatives.
Cas binaire: π_0, π_1 sont $P(Y=0)$, $P(Y=1)$, respectivement,
Les densités f_0 et f_1 de chaque groupe sont supposées gaussiennes, $N(\mu_0, \Sigma_0)$ et $N(\mu_1, \Sigma_1)$!

Le classifieur Bayésien: $f^*(X) = \arg\max_{j=0,1} \{\pi_j \cdot f_j(X)\}$.

Méthodes de type plug-in pour des variables explicatives quantitatives.
Cas binaire: π_0, π_1 sont $P(Y=0)$, $P(Y=1)$, respectivement,
Les densités f_0 et f_1 de chaque groupe sont supposées gaussiennes,
$N(\mu_0, \Sigma_0)$ et $N(\mu_1, \Sigma_1)$!

Le classifieur Bayésien: $f^*(X) = \arg\max_{j=0,1} \{\pi_j \cdot f_j(X)\}$.

Dans le cas gaussien de même variance - règle linéaire (en X) donc LDA,
sinon, si les variances sont différentes - règle quadratique (en X) - QDA.

Méthodes de type plug-in pour des variables explicatives quantitatives.
Cas binaire: π_0, π_1 sont $P(Y=0)$, $P(Y=1)$, respectivement,
Les densités f_0 et f_1 de chaque groupe sont supposées gaussiennes,
$N(\mu_0, \Sigma_0)$ et $N(\mu_1, \Sigma_1)$!

Le classifieur Bayésien: $f^*(X) = \arg\max_{j=0,1} \{\pi_j \cdot f_j(X)\}$.

Dans le cas gaussien de même variance - règle linéaire (en X) donc LDA, sinon, si les variances sont différentes - règle quadratique (en X) - QDA.

Pour construire le classifieur, on estime les paramètres inconnus par maximum de vraisemblance en utilisant $\mathcal{D}_n$, puis

$$\hat{f}(X) = \arg\max_{j=0,1} \hat{\delta}_j(X), \quad \text{où:}$$

$\hat{\delta}_j(X)$ est une fonction de $\hat{\mu}_j$ et de $\hat{\Sigma}_j$.

LDA et QDA

1. **LDA** : frontière linéaire de séparation $\{x : \delta_1(x) = \delta_0(x)\}$, où

$$\delta_j(X) = X^\top \Sigma^{-1} \mu_j - \frac{1}{2} \mu_j^\top \Sigma^{-1} \mu_j + \log \pi_j$$

Ici, $\Sigma_0 = \Sigma_1 = \Sigma$. Pour obtenir $\hat{\delta}_j$ on remplace par les Estimateurs qui Max la Vraisemblance: $\hat{\pi}_0$, $\hat{\pi}_1$, $\hat{\mu}_0$, $\hat{\mu}_1$,

$$\hat{\Sigma} = \frac{1}{n-2} \sum_{j=0}^{1} \sum_{i:Y_i=j} (X_i - \hat{\mu}_j)(X_i - \hat{\mu}_j)^\top$$

2. **QDA** : frontière quadratique $\{x : \delta_1(x) = \delta_0(x)\}$, où

$$\delta_j(X) = -\frac{1}{2}(X - \mu_j)^\top \Sigma^{-1}(X - \mu_j) - \frac{1}{2} \log \det \Sigma_j + \log \pi_j$$

et on remplace ici par

$$\hat{\Sigma}_j = \frac{1}{N_j - 1} \sum_{i:Y_i=j} (X_i - \hat{\mu}_j)(X_i - \hat{\mu}_j)^\top.$$

Estimation du risque de classification

On suppose maintenant que l'on dispose d'un autre échantillon $\mathcal{T}_n = \{(X'_1, Y'_1), ..., (X'_m, Y'_m)\}$ dit **échantillon test ou de validation**.

Estimation du risque de classification

On suppose maintenant que l'on dispose d'un autre échantillon $\mathcal{T}_n = \{(X_1', Y_1'), ..., (X_m', Y_m')\}$ dit **échantillon test ou de validation**.

Étant donné $\mathcal{D}_n$, pour chaque prédicteur $\hat{f}$, on estime son risque $R(\hat{f})$ par

$$\hat{R}(\hat{f}) = \frac{1}{m} \sum_{i=1}^{m} \ell(Y_i', \hat{f}(X_i')),$$

et ici $\ell(Y_i', \hat{f}(X_i')) = I(Y_i' \neq \hat{f}(X_i'))$.

Estimation du risque de classification

On suppose maintenant que l'on dispose d'un autre échantillon $\mathcal{T}_n = \{(X_1', Y_1'), ..., (X_m', Y_m')\}$ dit **échantillon test ou de validation**.

Étant donné $\mathcal{D}_n$, pour chaque prédicteur $\hat{f}$, on estime son risque $R(\hat{f})$ par

$$\hat{R}(\hat{f}) = \frac{1}{m} \sum_{i=1}^{m} \ell(Y_i', \hat{f}(X_i')),$$

et ici $\ell(Y_i', \hat{f}(X_i')) = I(Y_i' \neq \hat{f}(X_i'))$.

Sachant $\mathcal{D}_n$, $\hat{f}$ est une fonction fixe, donc

$$Z_i = I(Y_i' \neq \hat{f}(X_i'))$$

suit une loi de Bernoulli de probabilité $P(Y \neq \hat{f}(X))$.

Théorème (L'inégalité de Hoeffding)

Si $Z_1, ..., Z_m$ *indépendantes et bornées:* $Z_i \in [a_i, b_i]$ *alors*

$$P\left(|\frac{1}{m}\sum_{i=1}^{m}(Z_i - E(Z_i))| \geq t\right) \leq 2\exp\left(-\frac{2(mt)^2}{\sum_{i=1}^{m}(b_i - a_i)^2}\right).$$

Théorème (L'inégalité de Hoeffding)

Si $Z_1, ..., Z_m$ indépendantes et bornées: $Z_i \in [a_i, b_i]$ alors

$$P\left(|\frac{1}{m}\sum_{i=1}^{m}(Z_i - E(Z_i))| \geq t\right) \leq 2\exp\left(-\frac{2(mt)^2}{\sum_{i=1}^{m}(b_i - a_i)^2}\right).$$

On obtient, pour tout $t > 0$,

$$P\left(|\hat{R}(\hat{f}) - P(\hat{f}(X) \neq Y)| \geq t|\mathcal{D}_n\right) \leq 2\exp(-2mt^2),$$

autrement dit

$$|\hat{R}(\hat{f}) - P(\hat{f}(X) \neq Y)| \leq \sqrt{\frac{\log(2/\delta)}{2m}},$$

avec probabilité plus grande que $1 - \delta$.

Si on dispose de plusieurs classifieurs $\hat{f}_1, ..., \hat{f}_K$ (un dictionnaire), alors

$$\begin{aligned} & P\left(\max_{j=1,...,K} |\hat{R}(\hat{f}_j) - P(\hat{f}_j(X) \neq Y)| \geq t | \mathcal{D}_n\right) \\ \leq & \sum_{j=1}^{K} P\left(|\hat{R}(\hat{f}_j) - P(\hat{f}_j(X) \neq Y)| \geq t | \mathcal{D}_n\right) \\ \leq & 2K \exp(-2mt^2). \end{aligned}$$

Ainsi, on obtient

$$\max_{j=1,...,K} |\hat{R}(\hat{f}_j) - P(\hat{f}_j(X) \neq Y)| \leq \sqrt{\frac{\log(2K/\delta)}{2m}}$$

avec probabilité supérieure à $1 - \delta$.
Également,

$$E\left(\max_{j=1,...,K} |\hat{R}(\hat{f}_j) - P(\hat{f}_j(X) \neq Y)|\right) \leq \sqrt{\frac{\log(2K)}{2m}}.$$

Minimisation du risque empirique

Pour choisir le meilleur classifieur du dictionnaire, on choisit

$$\hat{f}_{erm} = \arg\min\{\hat{R}(f) : f \text{ parmi les } \hat{f}_1, ..., \hat{f}_K\}.$$

Alors

$$R(\hat{f}_{erm}) \leq \min_{k=1,...,K} R(\hat{f}_k) + 2\sqrt{\frac{\log(2K/\delta)}{2m}},$$

avec probabilité supérieure à $1-\delta$, et

$$E(\hat{f}_{erm}) \leq \min_{k=1,...,K} R(\hat{f}_k) + 2\sqrt{\frac{\log(2K)}{2m}}.$$

Séparation d'un échantillon en $\mathcal{D}_n$ (train) et $\mathcal{T}_n$ (test) par:
-1. Hold Out, Leave-One-Out
-2. V-fold Cross-Validation

CART

Un arbre de décision est une suite de partitions de plus en plus fines de l'ensemble de tous les individus observés.
À chaque étape, on choisit une variable X^j (coordonnée des $X_i = (X_i^1, ..., X_i^p)$) et un seuil s, puis on sépare le groupe considéré en

$$\text{groupe Gauche: si } X_i^j \leq s, \text{ groupe Droite: si } X_i^j > s.$$

Le choix de X^j et de s se fait par un algorithme 'glouton' (*greedy*) qui parcourt toutes les variables et tous les seuils de manière à ce que la *deviance* de la nouvelle partition soit minimale.

Pour la **régression**, on évalue la somme des carrés

$$SC = \sum_{k=1}^{K} \sum_{i:X_i \in C_k} (Y_i - \hat{\mu}_k)^2.$$

Si une classe C a été séparée en *Gauche* et *Droite* suivant la variable j et le seuil s la somme des carrés de C est remplacée par

$$\sum_{i:X_i \in C_{j,s,Gauche}} (Y_i - \hat{\mu}_{Gauche})^2 + \sum_{i:X_i \in C_{j,s,Droite}} (Y_i - \hat{\mu}_{Droite})^2,$$

où $\hat{\mu}_{Gauche}$ et $\hat{\mu}_{Droite}$ sont les moyennes des valeurs Y_i dans chaque qroupe. On minimise les SC ainsi obtenues suivant tous les j et s pour choisir la partition suivante.

Pour la **classification**, on calcule la proportion de chaque modalité dans la classe C_k et sa "deviance":

$$\hat{p}_k(1) = \frac{1}{|C_k|} \sum_{i:X_i \in C_k} I(Y_i = 1), \quad \hat{p}_k(0) = 1 - \hat{p}_k(1),$$

$$D(C_k) = -2 \cdot |C_k| \cdot (\hat{p}_k(1) \log(\hat{p}_k(1)) + \hat{p}_k(0) \log(\hat{p}_k(0))).$$

Si une classe C_k est divisée en $C_{j,s,Gauche}$ et $C_{j,s,Droite}$, on remplace

$$D(C_k) \quad \text{par } D(C_{j,s,Gauche}) + \cdot D(C_{j,s,Droite}),$$

puis, on minimise en j et s l'entropie totale.
Remarque: pour une classification parfaite, par exemple $\hat{p}_k(1) = 1$, on a $D(C_k) = 0$ (cas idéal).

Régression linéaire

$$Y = X \cdot \beta + \xi, \quad Y, \xi \text{ appartiennent à } \mathbb{R}^n,$$

X appartient à $\mathbb{R}^{n \times p}$, β dans $\mathbb{R}^p$. On suppose que les ξ_i i.i.d., $E(\xi_i) = 0$ et $Var(\xi_i) = \sigma^2$. On suppose que les colonnes sont standardisées $\frac{1}{n}\|X^j\|_2^2 = 1$ (c-à-d de variance 1) et (quasi-)orthogonales.
L'estimateur de moindres carrés

$$\hat{\beta}^{MC} = \arg \min_{\beta \in \mathbb{R}^p} \|Y - X \cdot \beta\|_2^2,$$

existe et est unique si $X^\top \cdot X$ est une matrice inversible.
Dans ce cas: $\hat{\beta}^{MC} = (X^\top X)^{-1} X^\top Y$.

Remarques: - si $n \geq p$ et si $rang(X^\top \cdot X) = p$, alors $X^\top X$ est inversible.
-par contre, si $n < p$ alors $rang(X^\top X) = n < p$, donc $X^\top X$ ne peut être inversible.

Ridge regression

ou régularisation de Tikhonov, consiste à pénaliser par $\|\beta\|_2^2$:

$$\hat{\beta}^R = \arg\min_{\beta \in \mathbb{R}^p} \frac{1}{n}\|Y - X \cdot \beta\|_2^2 + \lambda\|\beta\|_2^2$$

On obtient

$$\hat{\beta}^R = (\frac{X^\top X}{n} + \lambda I_p)^{-1}\frac{1}{n}X^\top Y.$$

Biais: $E(\hat{\beta}^R) - \beta = -\lambda(\frac{X^\top X}{n} + \lambda I_p)^{-1}\beta$

Variance: $Var(\hat{\beta}^R) = \frac{\sigma^2}{n}(\frac{X^\top X}{n} + \lambda I_p)^{-1} \cdot \frac{X^\top X}{n} \cdot (\frac{X^\top X}{n} + \lambda I_p)^{-1}$.

Remarque $Var(\hat{\beta}^R) \ll Var(\hat{\beta}^{MC})$.

Lasso

Least Absolute Shrinkage and Selection Operator

$$\hat{\beta}^L = \arg\min_{\beta \in \mathbb{R}^p} \frac{1}{2n} \| Y - X \cdot \beta \|_2^2 + \lambda \|\beta\|_1$$

Dans le cas simple y, t nombres réels et $\lambda > 0$ alors

$$\arg\min_t \frac{1}{2}(y-t)^2 + \lambda \cdot |t| \text{ admet la solution}$$

où

$$\begin{aligned} t &= signe(y) \cdot (|y| - \lambda)_+ \\ &= y \left(1 - \frac{\lambda}{|y|}\right)_+ \end{aligned}$$

Algorithme de descente de coordonnées

On résoud la minimisation Lasso pour chaque coordonnée j de 1 à p:

$$\begin{aligned}
\hat{\beta}_j &= \arg\min_{\beta_j} \frac{1}{2n}\| Y - \sum_{k:k\neq j} \beta_k X^k - \beta_j X^j \|_2^2 + \lambda(|\beta_j| + \sum_{k:k\neq j} |\beta_k|) \\
&= \arg\min_{\beta_j} \frac{1}{2n}\beta_j^2 \|X^j\|_2^2 - \frac{1}{n}\langle Y - \sum_{k:k\neq j} \beta_k X^k, \beta_j X^j \rangle + \lambda|\beta_j| \\
&= \arg\min_{\beta_j} \frac{1}{2n}\beta_j^2 - \frac{1}{n\|X^j\|_2^2}\langle Y - \sum_{k:k\neq j} \beta_k X^k, \beta_j X^j \rangle + \frac{\lambda}{\|X^j\|_2^2}|\beta_j| \\
&= \arg\min_{\beta_j} \frac{1}{2n}\left(\beta_j - \frac{1}{\|X^j\|_2^2}\langle Y - \sum_{k:k\neq j} \beta_k X^k, X^j \rangle \right)^2 + \frac{\lambda}{\|X^j\|_2^2}|\beta_j|
\end{aligned}$$

Alors, on obtient

$$\hat{\beta}_j = \frac{A}{\|X^j\|^2}\left(1-\frac{n\lambda}{|A|}\right)_+, \text{ où } A = \left(Y - \sum_{k:k\neq j}\beta_k X^k\right)^\top X^j.$$

Par conséquent, la procédure initialise $\hat{\beta}^{(0)} = b$, puis recalcule toutes les coordonnées $\hat{\beta}^{(1)}$ et réitère jusqu'à convergence.

Elastic net

$$\hat{\beta}^{EN} = \arg\min_{\beta\in\mathbb{R}^p} \frac{1}{2n}\|Y - X\cdot\beta\|_2^2 + \lambda\left(\gamma\|\beta\|_1 + (1-\gamma)\frac{1}{2}\|\beta\|_2^2\right).$$

Théorème : Si les erreurs ξ_i sont indépendantes et de même loi gaussienne $N(0, \sigma^2)$, et si X a les colonnes standardisées (de variance 1: $\frac{1}{n}\|X^j\|_2^2 = 1$) et presque orthogonales, alors pour tout $\delta > 0$, l'estimateur Lasso avec

$$\lambda = 2\sigma\sqrt{\frac{2}{n}\log\frac{2p}{\delta}}$$

on a, avec probabilité supérieure à $1 - \delta$,

$$\frac{1}{2n}\|X(\hat{\beta} - \beta)\|_2^2 \leq 32\frac{\sigma^2 s}{n}\log\frac{2p}{\delta}$$

et

$$\|\hat{\beta} - \beta\|_1 \leq 16\sigma \cdot s\sqrt{\frac{2}{n}\log\frac{2p}{\delta}}$$

où s est le nombre de coordonnées non-nulles de β.

Remarque: Si les colonnes de X sont orthogonales, on a $\frac{1}{n}X^\top X = I_n$, donc

$$\frac{1}{2n}\|X(\hat{\beta} - \beta)\|_2^2 = \|\hat{\beta} - \beta\|_2^2.$$

Factorisation des matrices

En classification non supervisée, on observe seulement X de taille $n \times p$ et pas les labels Y. Pour la matrice X, on note les colonnes X^j, $j = 1, ..., p$, et les lignes $X_i^\top$, i de 1 à n:

$$X = (\dots X^j \dots) = \begin{pmatrix} X_1^\top \\ \vdots \\ X_n^\top \end{pmatrix}.$$

On suppose que chaque entrée représente un coefficient observée avec un bruit:
$X = M + E$, où M et E sont des matrices de taille $n \times p$.

On dit que M est factorisée si on peut l'écrire comme produit

$$M = A \cdot B, \quad A \text{ de taille } n \times K \text{ et } B \text{ de taille } K \times p.$$

Exemples: 1. systèmes de recommandation: n clients d'un magasin, p articles vendus, et M_i^j le score qui mesure l'avis d'un client sur un article.
On peut regrouper les articles par catégories (vêtements, meubles, etc. pour vente par correspondance) puis chercher
-A_i^k les avis de l'utilisateur i sur la catégorie k,
-B_k^j les notes des clients de k sur l'article j.
Ainsi,

$$M_{ij} = \sum_{k=1}^{K} A_i^j B_k^j.$$

2. finance ou économie: variables latentes (cachés).
3. Netflix, MovieLens (1682 films, 943 utilisateurs)
Remarque: D'autant plus intéressant que K est petit (plus petit que n et p). Dans ce cas le rang de M est au plus K.

En ayant observé X, on souhaite trouver la matrice de type $M = A \cdot B$ la plus proche de X:

$$\min_{A:n\times K, B:K\times p} \sum_{i=1}^{n}\sum_{j=1}^{p}(X_{ij} - [A\cdot B]_{ij})^2.$$

On note $\|X - M\|_F^2 = \sum_{i=1}^{n}\sum_{j=1}^{p}(X_{ij} - M_{ij})^2$

Rappel: SVD de X. Si $X^\top \cdot X$ admet les valeurs propres $\sigma_1^2 \geq ... \geq \sigma_{n\wedge p}^2$, alors on peut décomposer

$$X = U^\top \cdot diag[\sigma_1, ..., \sigma_{n\wedge p}] \cdot V,$$

où les lignes de U sont orthonormées et les lignes de V aussi.
On a U de dimension $n \times n$, V de dimension $p \times p$ et $diag[\sigma_1, ..., \sigma_{n\wedge p}]$ de dimension $n \times p$.

Projection sur les matrices de rang petit

On veut approcher X par la meilleure matrice de rang petit, au sens

$$\hat{M}_K = \arg \min_{M:rang(M)=K} \|X - M\|^2.$$

Si X admet la SVD (décomposition en valeurs singulières)

$$X = U^\top \cdot diag[\sigma_1, ..., \sigma_{n\wedge p}] \cdot V,$$

alors $\hat{M}_K = U_K^\top \cdot diag[\sigma_1, ..., \sigma_K] \cdot V_K$, où U_K, V_K contiennent les K premières lignes de U et V.

Ce problème est analogue à la sélection de modèle (pénalité ℓ_0) mais on peut l'implémenter facilement grâce à la relation d'ordre des valeurs singulières. Le choix de K se fait par validation croisée.

Au final $\hat{A} = U_K^\top \cdot diag[\sigma_1, ..., \sigma_K]$ et $\hat{B} = V_K$.

Soft SVD

Soft Thresholding (seuillage doux) des valeurs singulières est une version régularisée de l'approximation par la meilleure matrice de rang petit.
On veut résoudre le problème

$$\hat{M}_\lambda = \arg\min_M \frac{1}{2}\|X - M\|_F^2 + \lambda\|M\|_*, \quad \lambda > 0,$$

et $\|M\|_*$ et la somme des valeurs singulières de M.

Une solution explicite existe également: elle consiste en un seuillage doux des valeurs singulières de X.
Soit $D_\lambda = diag[(\sigma_1 - \lambda)_+, ..., (\sigma_{n\wedge p} - \lambda)_+]$ de taille $n \times p$, alors

$$\hat{M}_\lambda = U^\top \cdot D_\lambda \cdot V.$$

Choix de λ par validation croisée.

Cas particuliers

1. Complétion de matrices: si on dispose seulement des entrées $X_{i,j}$, $(i,j) \in \Omega$, le seuillage s'écrit

$$\hat{M}_{\lambda,\Omega} = \arg\min_{M} \frac{1}{2} \sum_{(i,j)\in\Omega} (X_{i,j} - M_{i,j})^2 + \lambda \|M\|_*, \quad \lambda > 0,$$

où Ω est l'ensemble d'entrées (i,j) où l'on observe X_{ij} et $\bar{\Omega}$ est l'ensemble où X_{ij} est manquant.

Si M donnée, on note $X(M)$ la matrice où l'on remplace les données manquantes de X par les entrées de M.
On note la solution de Soft SVD $\hat{M}_\lambda$ par $ST_\lambda(X)$.

L'algorithme softImpute calcule de manière itérative cette solution:

-on pose $M = 0$ et on calcule $X(M)$.

-choix d'une grille des valeurs $\lambda_1 > \lambda_2 > ... > \lambda_N > 0$ avec $\lambda_1 < \sigma_1(X(M))$;

-pour chaque $j = 1, ..., N$: $M^{nouveau} = ST_{\lambda_j}(X(M))$ puis on remplace M par $M^{nouveau}$ et réitère jusqu'à stabilisation de $\|M - M^{nouveau}\|$.

-retourner $\hat{M}_j$.

Pour chaque j, $\hat{M}_j$ est proche de la solution du problème de point fixe

$$M = ST_{\lambda_j}(X(M)),$$

donc de la solution du problème initial. Choix de λ_j par validation croisée.

2. ACP: on recherche un sous-espace vectoriel V de petite dimension

$$\min_V \|X - P_v X\|_F^2, \quad P_V \text{ est la matrice de projection sur } V.$$

3. K-means: On recherche K clusters de centres $C_1, ..., C_K$ de même dimension que les X_i et une matrice L de taille $n \times K$ ayant des 0 et une seule valeur de 1 sur chaque ligne (attribue chaque individu à un seul centre C_k):

$$\min_{L,C} \|X - L \cdot C\|_F^2.$$