Introduction à l'apprentissage statistique

Notion de famille de modèles et phénomène de surapprentissage

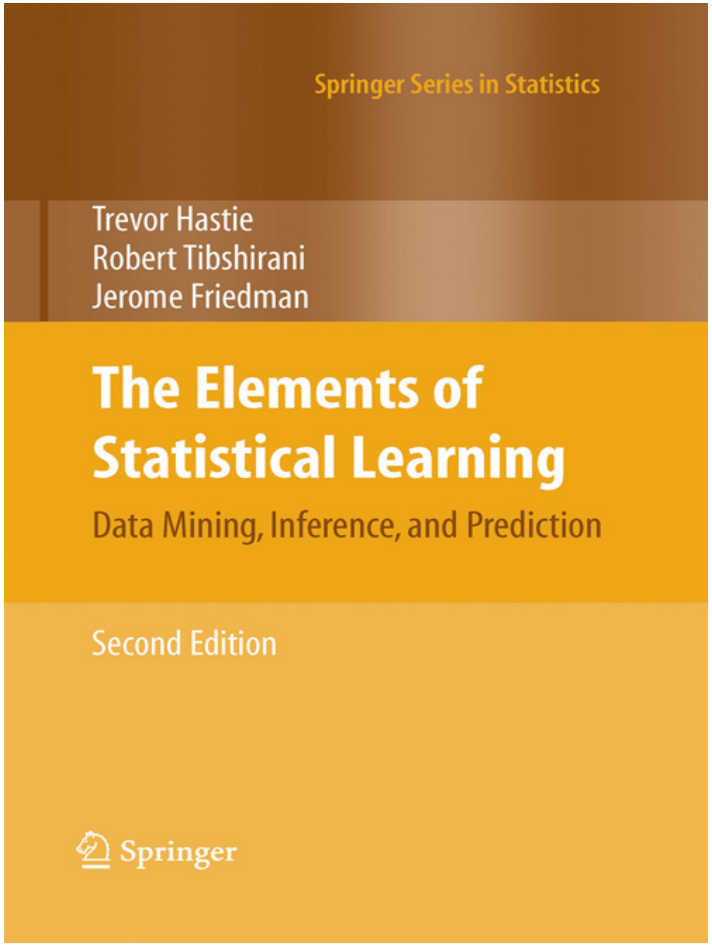
masedki.github.io

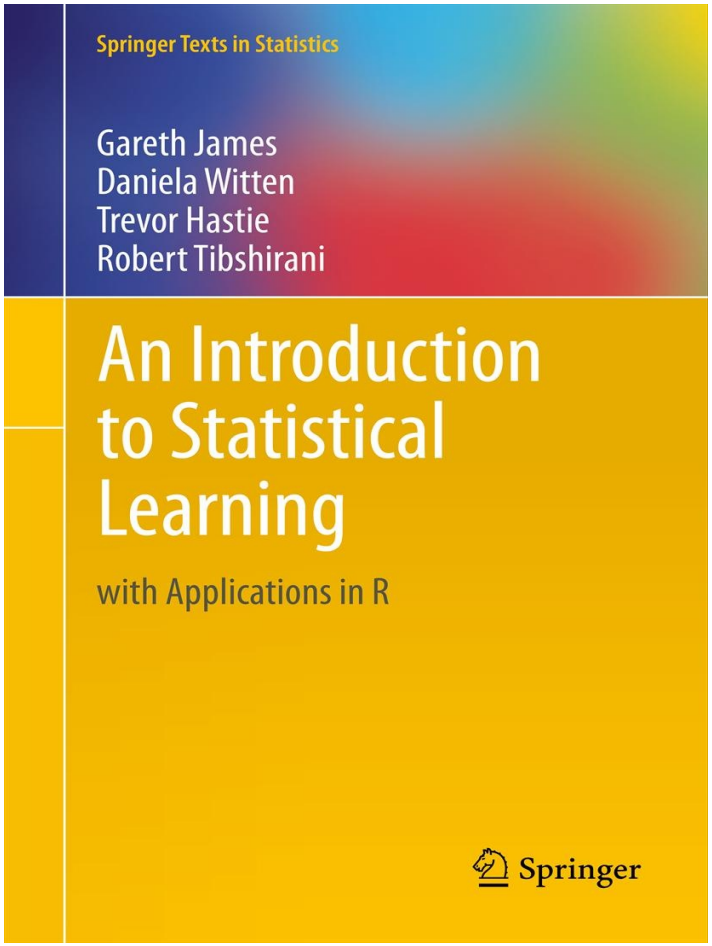
Université Paris-Saclay & CESP Inserm-1018

Déroulement

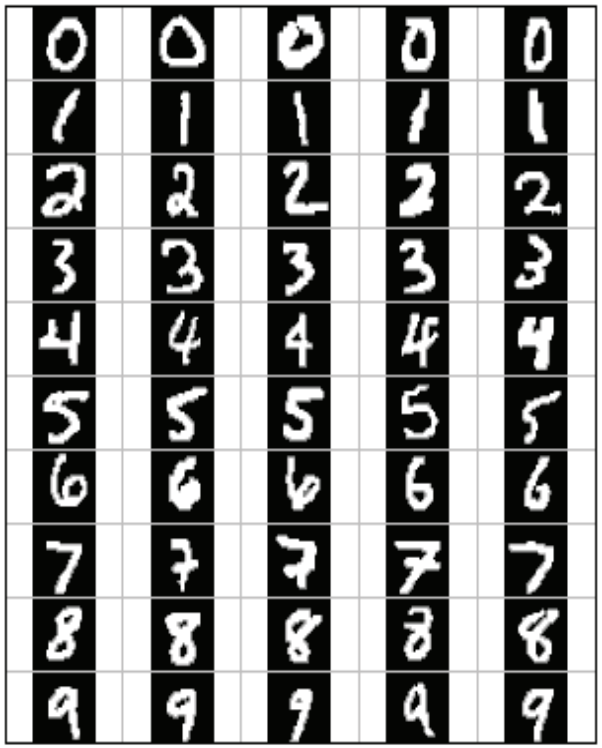
- Introduction + TP : notion de famille de modèles + phénomène de sur-apprentissage + choix de la meilleure complexité par validation croisée.
- Famille des arbres de décision : arbre de régression + algorithme CART + élagage + arbre de classification.
- Famille de modèles dits ensemblistes 1 : forêts aléatoires + différentes librairies
- Famille de modèles dits ensemblistes 2 : gradient boosting + différentes librairies
- Si le temps le permet : familles ensemblistes pour l'analyse de survie

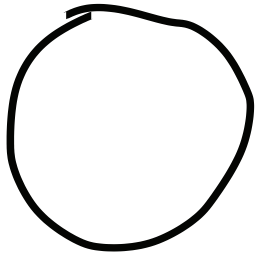
Les classiques





Problème d'apprentissage





- Reconnaissance de chiffres manuscrits ? 0, 1, 2 ... ?

Problèmes d'apprentissage statistique

- Identifier les facteurs de risque d'un cancer
- Prédire si une personne est sujette aux problèmes cardiaques, à partir de mesures cliniques, son régime et des données démographiques
- Personnaliser un système de détection de spam email
- Lecture de codes postaux écrits à la main
- Classification d'échantillons de tissus dans différents types de cancer, en fonction de données d'expression de gènes
- Établir une relation entre salaires et variables démographiques
- Distinguer des races de chiens sur des images
- Classer des dossiers médicaux par actes

Question

- Sur 4 601 mails, on a pu identifier 1813 spams.
- On a également mesuré sur chacun de ces mails la présence ou absence de 57 mots.

Peut-on construire à partir de ces données une méthode de classification automatique de mail en spam ou pas ?

Réprésentation du problème

- La plupart de ces problèmes peuvent être appréhendés dans un contexte de régression : on cherche à expliquer une variable Y par d'autres variables dites explicatives $X_1, \ldots, X_p$.

- Lorsque la variable à expliquer est quantitative, on parle de régression.
- Lorsqu'elle est qualitative, on parle de discrimination ou classification supervisée.

Régression

- Un échantillon i.i.d d'apprentissage $(X_1, Y_1), \ldots, (X_n, Y_n)$ d'une loi conjointe $\mathbb{P}$ **inconnue** sur $\mathbb{R}^p \times \mathbb{R}$.
- Objectif : Prédire ou expliquer la variable Y à partir d'une nouvelle observation X.
- Méthode : construire une règle de prédiction (ou régression)

$$m : \mathbb{R}^p \mapsto \mathbb{R}.$$

- Soit $\ell : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}_+$ une fonction de perte (i.e, $\ell(y, y') = 0$ et $\ell(y, y') > 0$ pour $y \neq y'$), par exemple

$$\ell(y, y') = |y - y'|^q$$

(perte absolue si $q = 1$ et perte quadratique $q = 2$).

Risque ou erreur de généralisation

- Le risque ou erreur de généralisation d'une règle de décision (ou prédiction) m est défini par

$$\mathcal{R}_{\mathbb{P}}(m) = \mathbb{E}_{(X,Y)}\Big[\ell\big(Y, m(X)\big)\Big].$$

La fonction de régression

- Pour la fonction de perte quadratique $\ell(y, y') = (y - y')^2$, on sait montrer qu'on a
$$m^*(x) = \mathbb{E}\big[Y|X = x\big]$$
appelé fonction de régression qui minimise le risque associé à la perte quadratique.
- Pour toute autre fonction m, on a

$$\mathbb{E}\left[\big(Y - m^*(X)\big)^2\right] \leq \mathbb{E}\left[\big(Y - m(X)\big)^2\right]$$

.

Preuve (Pythagore)

$$\mathbb{E}\left[\Big(Y-m(X)\Big)^2\right] = \mathbb{E}\left[\Big(Y-\mathbb{E}(Y|X)\Big)^2\right]+\mathbb{E}\left[\Big(\mathbb{E}(Y|X)-m(X)\Big)^2\right] = a+b$$

La classification binaire

- Un échantillon i.i.d d'apprentissage $(X_1, Y_1), \ldots, (X_n, Y_n)$ d'une loi conjointe $\mathbb{P}$ **inconnue** sur $\mathbb{R}^p \times \{0, 1\}$.
- Objectif : Prédire ou expliquer la variable Y à partir d'une nouvelle observation X.
- Méthode : construire une règle classification (ou décision)

$$g : \mathbb{R}^p \mapsto \{0, 1\}.$$

- La fonction de perte binaire $\ell(y, y') = \mathbf{1}_{\{y \neq y\}}$.
  - Risque associé à g : taux de mauvais classement

$$\mathcal{R}_{\mathbb{P}}(g) = \mathbb{E}\Big[\ell\big(g(X), Y\big)\Big] = \mathbb{P}\Big(g(X) \neq Y\Big).$$

La règle de Bayes

- Un champion appelé règle de Bayes

$$g^*(x) = \begin{cases} 1 & \text{si} \quad \eta(x) \geq \frac{1}{2} \\ 0 & \text{sinon,} \end{cases}$$

où $\eta(x) = \mathbb{P}(Y = 1 | X = x)$.

- Quelque soit la règle de décision g, nous avons

$$\mathcal{R}_{\mathbb{P}}(g^*) = \mathbb{P}\Big(g^*(X) \neq Y\Big) \leq \mathbb{P}\Big(g(X) \neq Y\Big) = \mathcal{R}_{\mathbb{P}}(g).$$

Règle de Bayes : théorème

Pour toute règle de classification $g : \mathcal{X} \mapsto \mathcal{Y}$, pour la fonction de perte binaire, nous avons

$$\mathcal{R}(g) - \mathcal{R}(g^*) = \mathbb{E}_X \left[\mathbf{1}\Big\{ g(X) \neq g^*(X) \Big\} \Big| 2\eta(X) - 1 \Big| \right].$$

Début de preuve

Rappel :

$$\mathbb{E}_{X,Y} h(X, Y) = \mathbb{E}_X \mathbb{E}_{Y|X} h(X, Y).$$

suite de la preuve : voir notes

Proposition

$$\mathcal{R}^* = \mathcal{R}(g^*) = \mathbb{E}_X \left[\min \left\{ \eta(X), 1 - \eta(X) \right\} \right]$$

Problème majeur !!

- Problème: m^* est inconnu en pratique. Il faut construire une fonction de régression $\widehat{m}_n$ à partir des données $(X_1, Y_1), \ldots, (X_n, Y_n)$, tel que

$$\widehat{m}_n(x) \approx m^*(x).$$

- Problème: g^* est inconnue en pratique. Il faut construire une règle $\widehat{g}_n$ à partir des données $(X_1, Y_1), \ldots, (X_n, Y_n)$, tel que

$$\widehat{g}_n(x) \approx g^*(x).$$

Un candidat naturel

- À partir des expressions de m^* et g^*, proposer deux estimateurs intuitifs.

Décomposition de l'erreur

Pour tout estimateur $\widehat{m}_n(x)$ de $m^*(x)$ à x fixé, nous avons

$$\begin{aligned}
\mathbb{E}\left[\left(m^*(x) - \widehat{m}_n(x)\right)^2\right] &= \left[m^*(x)\right]^2 - 2m^*(x)\mathbb{E}\left[\widehat{m}_n(x)\right] \\
&\quad + \mathbb{E}\left[\left(\widehat{m}_n(x)\right)^2\right] \\
&= \left[m^*(x) - \mathbb{E}\left(\widehat{m}_n(x)\right)\right]^2 \\
&\quad + \mathbb{E}\left[\left(\widehat{m}_n(x)\right)^2\right] - \left[\mathbb{E}\left(\widehat{m}_n(x)\right)\right]^2 \\
&= \Big(\text{biais}\Big)^2 + \text{Var}\left[\widehat{m}_n(x)\right]
\end{aligned}$$

Notations

- On s'intéresse au cas où on cherche à expliquer une variable qualitative Y par p variables explicatives $X_1, \ldots, X_p$.
- Y est à valeurs dans un ensemble discret fini de modalités qui peuvent être numérotées par des les indices $\{1, 2, \ldots, K\}$ et les variables $X_1, \ldots, X_p$ peuvent être qualitatives et/ou quantitatives.
- Néanmoins, pour présenter les méthodes, on se restreint au cas où Y est à 2 modalités (0 et 1).

Évaluation de la qualité de prédiction

L'objectif est de construire à partir des données

$$\mathcal{D}_n = \Big\{ (X_1, Y_1), \ldots, (X_n, Y_n) \Big\}, \quad X_i = (X_{i1}, \ldots, X_{ip}),$$

une (fonction de régression ou une règle de classification) $\widehat{f}(\cdot) = \widehat{f}(\cdot, \mathcal{D}_n)$.

- Avant d'utiliser les prévisions que $\widehat{f}(\cdot)$ fournit, il faut évaluer son risque.
- Bien souvent on dispose de plusieurs règles d'apprentissage (différents choix d'hyperparamètres, différentes méthodes), comment choisir la meilleure méthode et ses paramètres ?

Évaluation de la qualité de prédiction

Risque en régression (*Mean Squared Error*)

$$MSE = \mathbb{E}\Big[\big(Y - \widehat{f}(X)\big)^2\Big]$$

ou en classification (probabilité de mauvais classement)

$$\mathbb{P}\Big(Y \neq \widehat{f}(X)\Big).$$

On fera appel à des versions empiriques

$$\frac{1}{n}\sum_{i=1}^{n}\Big(Y_i - \widehat{f}(X_i)\Big) \quad \text{et} \quad \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{\left\{Y_i \neq \widehat{f}(X_i)\right\}}.$$

Évaluation de la qualité de prédiction

Plus un modèle est complexe (plus de paramètres) et flexible (s'adapte mieux aux données), plus son erreur d'apprentissage est faible. MAIS ce modèle pourra s'avérer mauvais dans un but de prédiction ou de généralisation (s'adapter à de nouvelles données).

- Minimiser l'erreur d'apprentissage conduit au sur-apprentissage.
- Comment obtenir un bon estimateur de l'erreur de prédiction ?
- Si l'on dispose de beaucoup de données, on partage le jeu de données en un ensemble d'apprentissage et un ensemble test :
  - le modèle est ajusté sur les données d'apprentissage : calcul de $\widehat{f}$.
  - le modèle est évalué sur les données test : calcul de l'erreur test.

Approche apprentissage/test

- $n = n_{train} + n_{test}$, estimation de $\widehat{f}$ à partir du jeu de données d'apprentissage $\left\{(x_i, y_i), i = 1, \ldots, n_{train}\right\}$

$$\text{Erreur de test} = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \left(y_i - \widehat{f}(x_i)\right)^2$$

Avantages :

- simple
- facile et rapide à implémenter

Inconvénients :

- l'erreur test peut être variable, dépendre du découpage des données
- l'erreur test peut sur-estimer l'erreur de prédiction en n'utilisant qu'une partie des données pour estimer f.

Méthodes pour éviter le sur-apprentissage

- Utiliser un jeu de données test de grande taille, rarement disponible.
- Corriger mathématiquement l'erreur d'apprentissage par pénalisation : AIC, BIC, Cp de Mallows,...
- On va s'intéresser à la technique très utilisée de validation croisée (*cross-validation*)

Validation croisée

- Méthode universelle : mise en oeuvre dans un cadre statistique général et pour la plupart des procédures d'estimation
- Principe : séparer les données d'apprentissage et de test; construire l'estimateur sur l'échantillon d'apprentissage et utiliser l'échantillon test pour calculer un risque de prédiction (*model assessment*). Répéter plusieurs fois et moyenner les risques de prédiction obtenus
- Technique très utilisée pour choisir les tuning parameters (*model selection*). En particulier : paramètre de complexité d'un arbre de décision.

Schéma de Validation croisée

- *K-fold CV* : partition des données en K sous-ensembles. Chaque sous-ensemble sert successivement d'échantillon test, le reste d'échantillon d'apprentissage. En pratique : K entre 5 et 10.
- *Leave-one-out* : n-fold CV
- *Leave-q-out* : chaque sous-ensemble de cardinal q est retiré comme échantillon test, le reste servant d'apprentissage

Évaluation/comparaison

La procédure complète partage en 3 les données :

apprentissage + validation (2/3 données) + test (1/3 données)

- sur les données d'apprentissage : calcul de l'estimateur (algorithme d'apprentissage)
- sur les données de validation : *model selection*, choix d'un meilleur modèle dans la classe de modèles considérée, par sélection de variables et/ou optimisation des hyper-paramètres de l'algorithme (*tuning parameters*).
- sur les données test : calcul de l'erreur de prédiction finale

Les 2 dernières étapes sont souvent réalisées par validation-croisée.

Table of Contents