Université J.F. CHAMPOLLION Albi

Apprentissage supervisé

Licence d'informatique S6

Thierry Montaut

Apprentissage supervisé

- On parle d'apprentissage supervisé lorsque la méthode d'apprentissage de la fonction f nécessite l'utilisation de données pour lesquelles ont connaît exactement le résultats. Dit autrement, d'un ensemble de couples $\{(x_1, y_1), ...(x_n, y_n)\}$ pour lesquels $y_i = f(x_i)$.
- L'apprentissage supervisé est l'un des types les plus courants et les plus efficaces d'apprentissage automatique.
- Les données x sont appelées observations, descripteurs, caractéristiques (variables, attributes, features). Les résultats y sont appelés étiquettes (labels, target, outcomes)

Rappel : Différents problèmes d'apprentissage

On peut toujours considérer qu'à partir d'un ensemble X, on cherche à apprendre pour tout x dans X la valeur $y = f(x)$ que prend une fonction f en x.

Quand on cherche à apprendre la fonction

$$f : \begin{pmatrix} X & \to & Y \\ x & \mapsto & y = f(x) \end{pmatrix}$$

On distingue les problèmes d'apprentissage selon la nature de l'ensemble Y :

- Si Y est un ensemble fini, on parle de problème de **classification**. La fonction à prédire est un classificateur.
- Dans le cas particulier fréquent où Y n'a que deux valeurs on parle également de classificateur binaire, de prédicteur ou de fonction de décision.
- Si $Y \subseteq \mathbb{R}$, on parle de problème de **régression**.

Cadre expérimental

Théorème

Il n'existe pas de baguette magique permettant de résoudre tous les problèmes d'apprentissage (c'est le "no free lunch" theorem). Il est donc toujours nécessaire, pour un problème donné, de tester plusieurs modèles et de les comparer à l'aide d'un cadre expérimental rigoureux et de critères de performances standardisés.

Espace des hypothèses

1. Pour poser un problème d'apprentissage, on commence par décider du type de fonctions de modélisation que nous allons considérer et des paramètres de ces fonctions. C'est l'espace des hypothèses.
2. On pourra ainsi chercher à classifier les données en les séparant par des droites, des hyperplans, des cercles, des ellipses, des parallélépipèdes, des polynômes de degrés n. On doit alors :
3. décider d'un moyen de vérifier la qualité de l'hypothèse.
4. décider d'une mesure de l'erreur commise par la fonction de modélisation (la fonction de coût ou loss function)
5. trouver une méthode permettant d'optimiser la fonction de modélisation en déterminant les valeurs des paramètres qui minimisent cette erreur.

Généralisation et surapprentissage

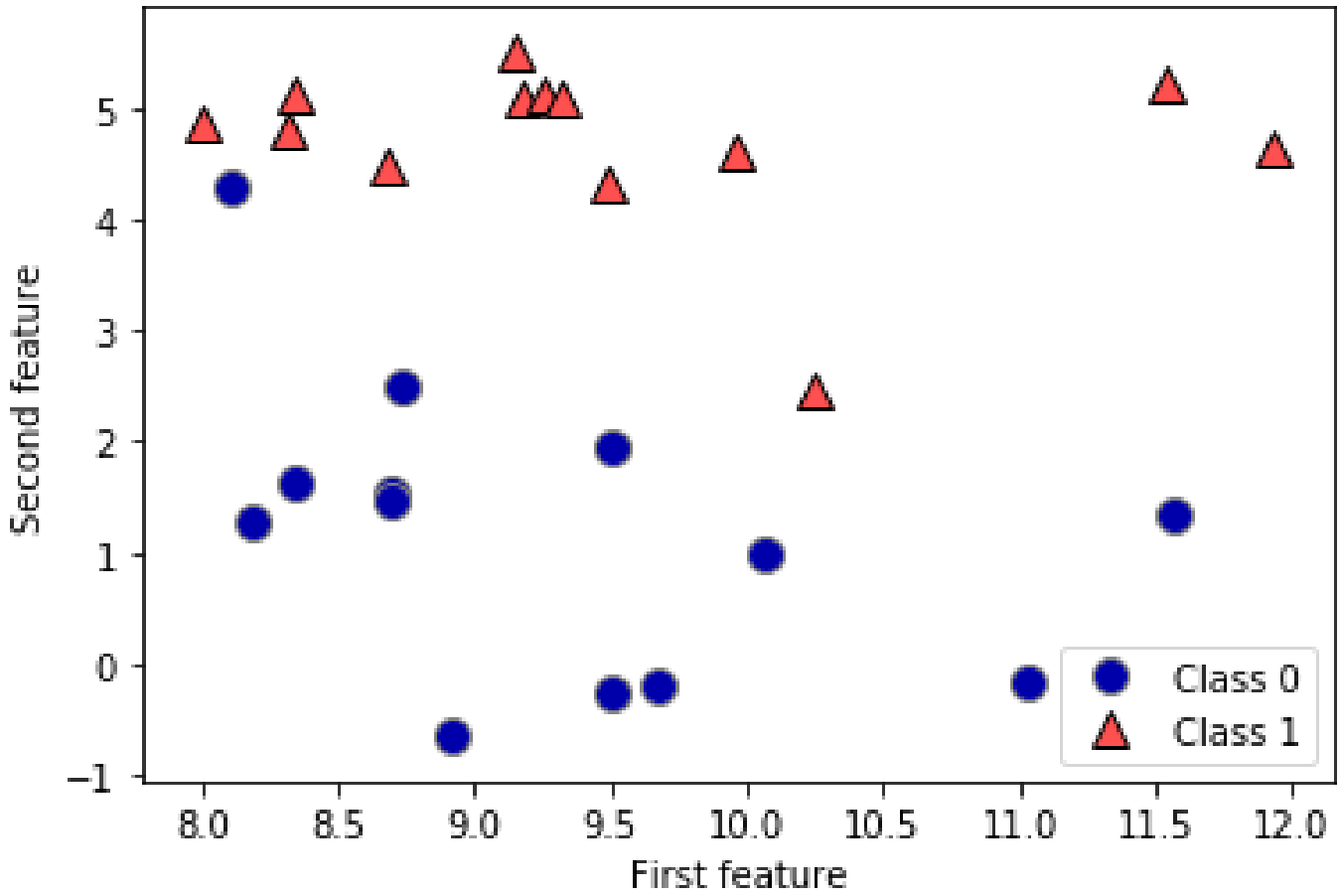
1. Dans l'apprentissage supervisé, nous souhaitons construire un modèle à partir d'un jeu de **données d'apprentissage** pour lesquelles on connaît la valeur de f, puis être capable d'en déduire la valeur de f sur des **données de test** pour lesquelles on ne connaît pas le résultat.
2. La première phase est **l'apprentissage du modèle**, la seconde est la phase de **généralisation ou de prédiction**.
3. Un bon modèle n'est pas un modèle qui fonctionne bien sur les données d'apprentissage mais qui se généralise bien à de nouvelles données de test.

Généralisation et surapprentissage

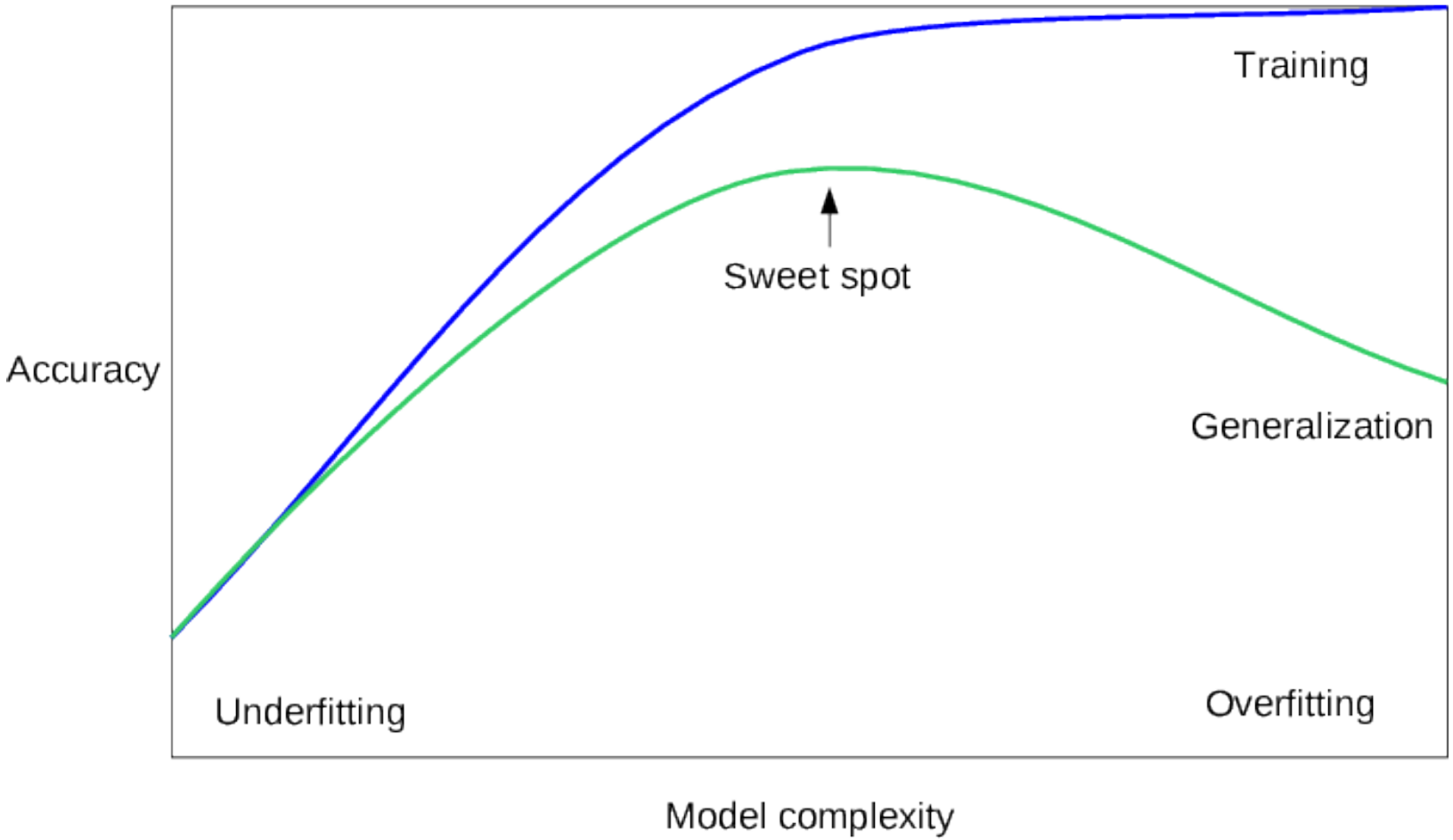
1. On commencera donc toujours pas scinder aléatoirement les données dont on dispose en deux (souvent 80% pour l'apprentissage et 20% pour le test).
2. L'optimisation sera effectuée sur les données d'apprentissage mais les performances du modèle seront calculées sur **le taux de bonnes prédiction sur les données de test.**
3. On peut être tenté de construire des modèles complexes collant parfaitement aux données d'apprentissage (même si elle contiennent des données exubérantes et rares (il est toujours possible d'être aussi exacte que l'on souhaite sur les données d'apprentissage). Mais cela se fait souvent au détriment de la capacité de généralisation à de nouvelles données. C'est le phénomène de **surapprentissage.**
4. On dira au contraire d'un modèle trop simple ne permettant pas de bonne généralisation du fait de sa simplicité qu'il **sous-apprend.**

Surapprentissage

Comment classifier au mieux le jeu de donné *forge* ?



On cherche donc un compromis entre la complexité du modèle lors de l'apprentissage et sa capacité de généralisation, ce qui nécessite une phase d'optimisation.



Validation croisée

1. La séparation d'un jeu de données en jeu d'entraînement (training set) et jeu de test (test set) peut être effectuée aléatoirement.
2. Pour vérifier la stabilité de l'erreur de généralisation on répète plusieurs fois cette phase de partitionnement et on en mesure la moyenne et la variance.
3. On peut également partitionner les données en N parties S_k et pour tout k dans $[\![1, N]\!]$:
   - entraîner le modèle sur toutes les parties sauf S_k
   - mesurer l'erreur de généralisation sur S_k.
4. Cette méthode est appelée la validation croisée (cross validation).

Courbe d'apprentissage

1. Il est important de se demander si le nombre de données d'apprentissage est suffisant ou si, au contraire, on pourrait encore améliorer le modèle en disposant de davantage de données.
2. La **courbe d'apprentissage** donne le score obtenu par une validation croisée en fonction du nombre de données dont on dispose (exprimée comme fraction des données disponibles).
3. On peut considérer que le nombre de données est suffisant si la courbe d'apprentissage semble a atteint une valeur asymptotique (un plateau).

Critères de performance

Il existe de nombreuses façons d'évaluer la performance prédictive d'un modèle d'apprentissage, qui dépendent en général de la nature du modèle. Pour un problème de régression on utilise une des trois métriques suivantes :

1. La moyenne des erreurs absolues

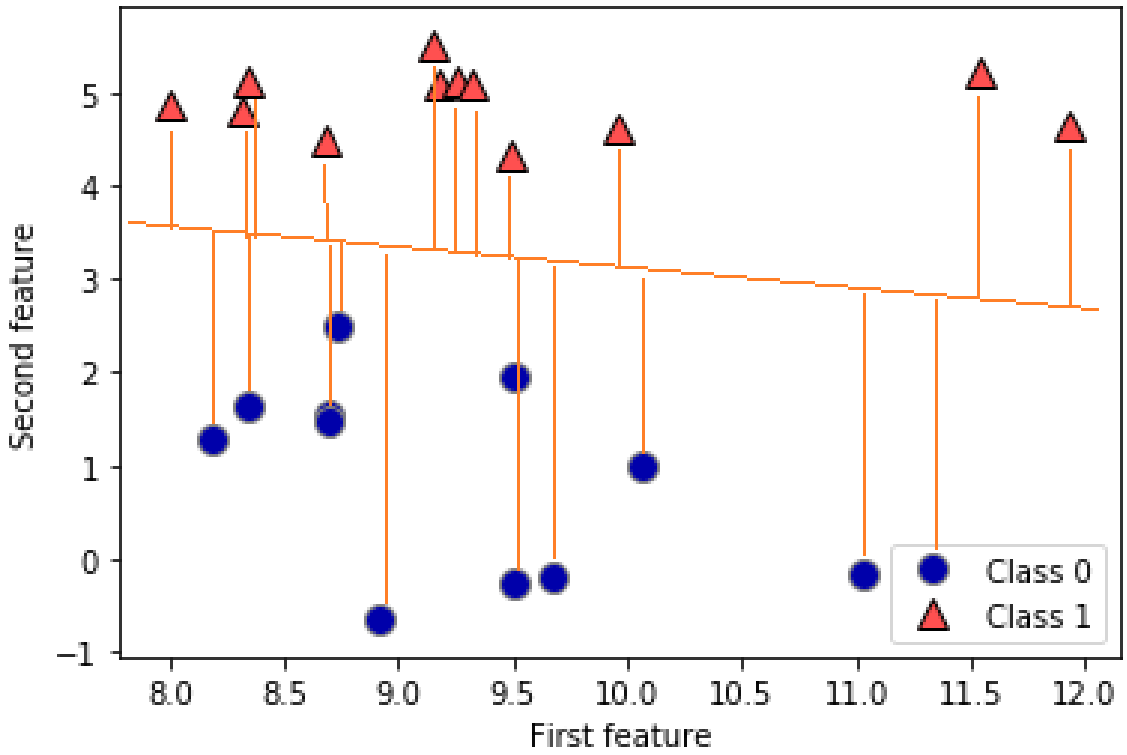
$$MAE = \frac{1}{n}\sum_{i=1}^{N} |y_i - f(x_i)|$$

2. La moyenne des erreurs quadratique (amplifie les erreurs importantes)

$$MSE = \frac{1}{n}\sum_{i=1}^{N} (y_i - f(x_i))^2$$

3. La médiane des erreurs absolues (élimine les points aberrants)

$$MedAE = \frac{1}{n}\sum_{i=1}^{N} |y_i - f(x_i)|$$


Second feature
First feature
Class 0
Class 1
5
4
3
2
1
0
−1
8.0
8.5
9.0
9.5
10.0
10.5
11.0
11.5
12.0

Critères de performance

1. Pour un problème de classification, on utilise simplement le taux d'erreurs de classifications exprimé en pourcentage (accuracy).
2. Pour un classificateur binaire, on utilise fréquemment la matrice de confusion :

	Classe réelle	
	0	1
0	vrais négatifs(VN)	faux négatifs(FN)
1	faux positifs (FP)	vrais positifs (VP)

3. La **sensibilité** est le taux de vrais positifs parmi les réellement positifs : $s = \frac{TP}{TP+FN}$.
4. La **précision** est le taux de vrais positifs parmi les positifs prédits : $s = \frac{TP}{TP+FP}$.
5. La **spécificité** est le taux de vrais négatifs parmi les réellement négatifs : $s = \frac{TN}{TN+FP}$.

Retenir

1. Les trois ingrédients d'un problème d'apprentissage supervisé sont :
   - L'espace des hypothèses
   - La fonction de coût (la mesure de l'erreur)
   - L'algorithme d'optimisation des paramètres du modèle
2. Les préoccupation majeures dans le choix d'un modèle sont :
   - Sa capacité de généralisation à des données différentes de celles qui l'ont entraîné
   - Ne pas chercher à trop coller aux données d'apprentissage (sur-apprendre)