

Méthodes d'apprentissage

IFT603

Régression linéaire

Par
Pierre-Marc Jodoin
/
Hugo Larochelle

1

Apprentissage supervisé

RAPPEL

Deux sortes d'apprentissage supervisé

- **Classification** : la cible est un indice de classe $t \in \{1, \dots, K\}$
 - Exemple : reconnaissance de caractères
 - ✓ \vec{x} : vecteur des intensités de tous les pixels de l'image
 - ✓ t : identité du caractère
- **Régression** : la cible est un nombre réel $t \in \mathbb{R}$
 - Exemple : prédiction de la valeur d'une action à la bourse
 - ✓ \vec{x} : vecteur contenant l'information sur l'activité économique de la journée
 - ✓ t : valeur d'une action à la bourse le lendemain

2

2

Apprentissage supervisé

RAPPEL

Deux sortes d'apprentissage supervisé

➤ **Classification** : la cible est un indice de classe $t \in \{1, \dots, K\}$

- Exemple : reconnaissance de caractères
 - ✓ \vec{x} : vecteur des intensités de tous les pixels de l'image
 - ✓ t : identité du caractère

➤ **Régression** : la cible est un nombre réel $t \in \mathbb{R}$

- Exemple : prédiction de la valeur d'une action à la bourse
 - ✓ \vec{x} : vecteur contenant l'information sur l'activité économique de la journée
 - ✓ t : valeur d'une action à la bourse le lendemain

3

3

Régression linéaire

• Le modèle de **régression linéaire** est le suivant :

$$y_{\vec{w}}(\vec{x}) = w_0 + w_1x_1 + w_2x_2 + \dots + w_dx_d$$

$$\text{où } \vec{x} = (x_1, x_2, \dots, x_d)^T$$

• La prédiction correspond donc à

- Une **droite** pour $d=1$
- Un **plan** pour $d=2$
- Un **hyperplan** pour $d>2$

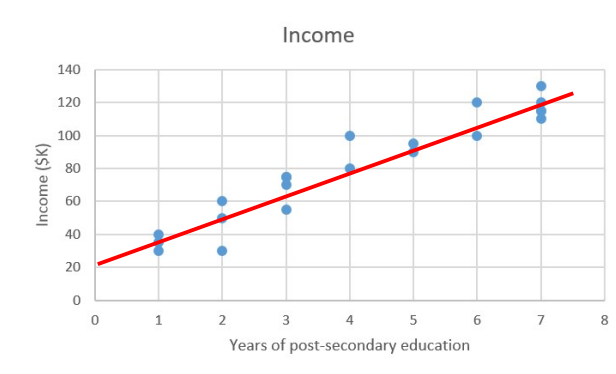
4

4

Régression linéaire 1D

Exemple

$$y_{\bar{w}}(x) = w_0 + w_1x$$



Credit : medium.com/machine-learning-for-humans/supervised-learning-740383a2feab

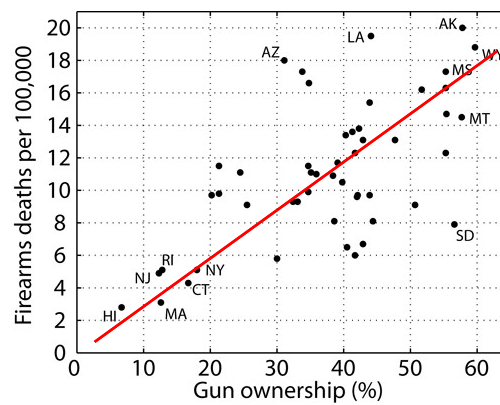
5

5

Régression linéaire 1D

Exemple

$$y_{\bar{w}}(x) = w_0 + w_1x$$



Source: <http://election.princeton.edu/2012/12/22/scientific-americans-gun-error/>

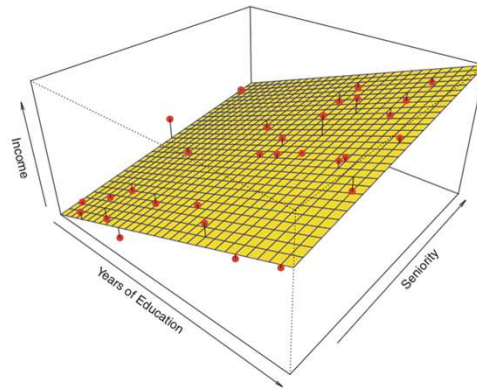
6

6

Régression linéaire 2D

Exemple

$$y_{\vec{w}}(\vec{x}) = w_0 + w_1x_1 + w_2x_2$$



Credit : sphweb.bumc.bu.edu/otlt/MPH-Modules/BS/R/R5_Correlation-Regression/R5_Correlation-Regression4.html

7

7

Régression linéaire

$$y_{\vec{w}}(\vec{x}) = w_0 + w_1x_1 + w_2x_2 + \dots + w_dx_d$$

Biais

poids

8

8

Régression linéaire

$$y_{\vec{w}}(\vec{x}) = w_0 + w_1 x_1 + w_2 x_2 + \dots + w_d x_d$$

$$y_{\vec{w}}(\vec{x}) = \vec{w}^T \vec{x}'$$

$$\text{où } \vec{x}' = (1, x_1, x_2, \dots, x_d)^T$$

9

9

Régression linéaire

Produit scalaire

Par simplicité, nous écrivons

$$y_{\vec{w}}(\vec{x}) = \vec{w}^T \vec{x}$$

10

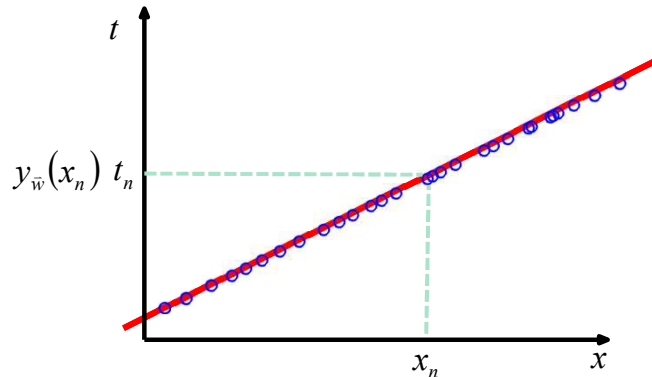
10

Problème à résoudre

Soit un ensemble d'apprentissage :

$$D = \{(x_1, t_1), (x_2, t_2), \dots, (x_N, t_N)\}$$

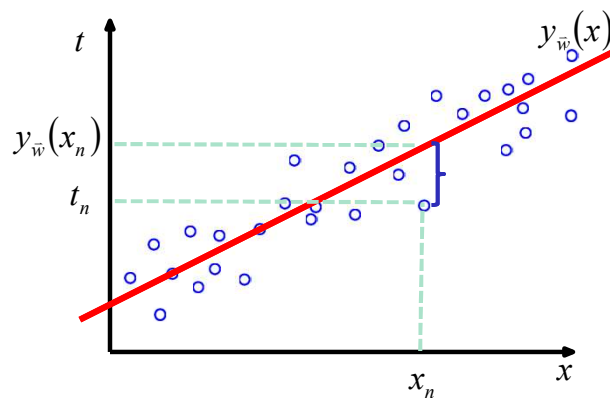
Idéalement, on souhaiterait trouver un modèle tel que $y_{\bar{w}}(x_i) = t_i$



11

Problème à résoudre

Malheureusement, dans la vraie vie, les données sont **bruitées**

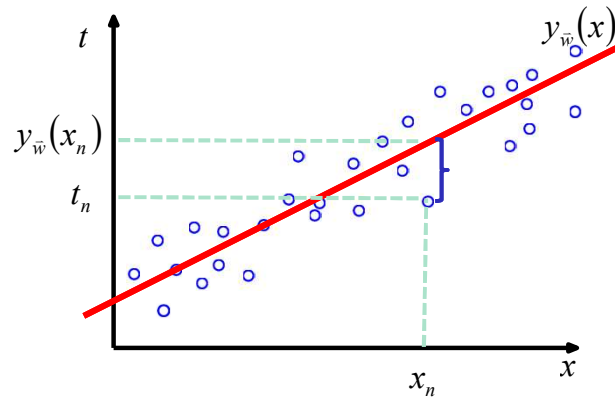


Dans ce cas, le but est de trouver un modèle qui **fait le moins d'erreurs possible**.

12

Problème à résoudre

$$\vec{w} = \arg \min_{\vec{w}} \sum_{n=1}^N (y_{\vec{w}}(x_n) - t_n)^2$$



13

Problème à résoudre

$$\vec{w} = \arg \min_{\vec{w}} \sum_{n=1}^N (y_{\vec{w}}(x_n) - t_n)^2$$

Il est bien connu en technique d'apprentissage que cette solution est **optimale** lorsque le **bruit est gaussien**.

C'est même une question d'entrevue!

14

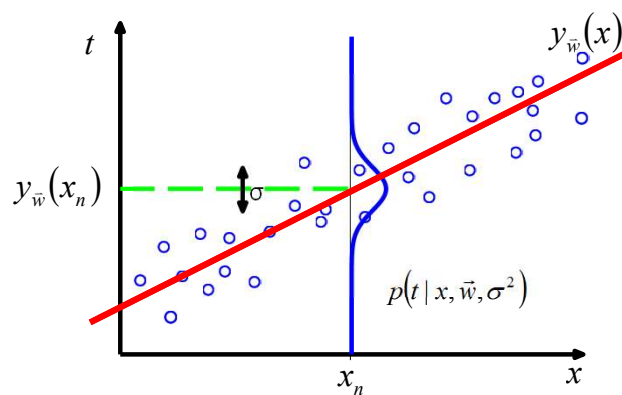
Régression et maximum de vraisemblance

15

15

Formulation probabiliste

Loi gaussienne conditionnelle



16

Formulation probabiliste

Pour entraîner le modèle $y_{\bar{w}}(x)$ nous passerons par une formulation probabiliste :

$$p(t | x, \bar{w}, \sigma^2) = N(t | y_{\bar{w}}(x), \sigma^2)$$

➤ Revient à supposer que les **cibles** sont des **versions bruitées** du vrai modèle

$$t_n = y_{\bar{w}}(x_n) + \varepsilon$$

Bruit gaussien de moyenne 0
et de variance σ^2

17

Maximum de vraisemblance

Soit notre **ensemble d'entraînement**

$$D = (X, T)$$

où

$$X = \{\bar{x}_1, \dots, \bar{x}_N\} \text{ et } \bar{x}_i \in R^d$$

$$T = \{t_1, \dots, t_N\}$$

et la fonction de **probabilités dont les données sont issues**

$$p(T | X, \bar{w}, \sigma^2)$$

Le **maximum de vraisemblance** s'exprime comme

$$\bar{w} = \arg \max_{\bar{w}} p(T | X, \bar{w}, \sigma^2)$$

Connue

Inconnue

18

Maximum de vraisemblance

$$\begin{aligned}\bar{w} &= \arg \max_{\bar{w}} p(T | X, \bar{w}, \sigma^2) \\ &= \arg \max_{\bar{w}} p(t_1, \dots, t_N | \bar{x}_1, \dots, \bar{x}_N, \bar{w}, \sigma^2)\end{aligned}$$

En supposant que les données sont i.i.d

$$\begin{aligned}\bar{w} &= \arg \max_{\bar{w}} \prod_{n=1}^N p(t_n | \bar{x}_n, \bar{w}, \sigma^2) \\ &= \arg \max_{\bar{w}} \prod_{n=1}^N N(t_n | y_{\bar{w}}(\bar{x}_n), \sigma^2) \\ &= \arg \max_{\bar{w}} \prod_{n=1}^N \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y_{\bar{w}}(\bar{x}_n) - t_n)^2}{2\sigma^2}}\end{aligned}$$

19

Maximum de vraisemblance

$$\begin{aligned}\bar{w} &= \arg \max_{\bar{w}} \ln \left(\prod_{n=1}^N \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y_{\bar{w}}(\bar{x}_n) - t_n)^2}{2\sigma^2}} \right) \\ &= \arg \max_{\bar{w}} \sum_{n=1}^N \ln \left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y_{\bar{w}}(\bar{x}_n) - t_n)^2}{2\sigma^2}} \right) \\ &= \arg \max_{\bar{w}} N \ln \left(\frac{1}{\sqrt{2\pi}\sigma} \right) + \sum_{n=1}^N \ln \left(e^{-\frac{(y_{\bar{w}}(\bar{x}_n) - t_n)^2}{2\sigma^2}} \right) \\ &= \arg \max_{\bar{w}} \sum_{n=1}^N -\frac{(y_{\bar{w}}(\bar{x}_n) - t_n)^2}{2\sigma^2}\end{aligned}$$

Indépendant de \bar{w}

20

Maximum de vraisemblance

$$\bar{w} = \arg \max_{\bar{w}} \sum_{n=1}^N \frac{(y_{\bar{w}}(\bar{x}_n) - t_n)^2}{\cancel{\bar{x}_n^2}}$$

$$\bar{w} = \arg \min_{\bar{w}} \sum_{n=1}^N (y_{\bar{w}}(\bar{x}_n) - t_n)^2$$

Et puisque $y_{\bar{w}}(\bar{x}) = \bar{w}^T \bar{x}$ (voir quelques pages précédentes)

$$\bar{w} = \arg \min_{\bar{w}} \sum_{n=1}^N (\bar{w}^T \bar{x}_n - t_n)^2$$

21

Maximum de vraisemblance

$$\bar{w} = \arg \min_{\bar{w}} \underbrace{\sum_{n=1}^N (\bar{w}^T \bar{x}_n - t_n)^2}_{E_D(\bar{w})}$$

Le « meilleur » \bar{w} est celui pour lequel le **gradient est nul**

$$\nabla_{\bar{w}} E_D(\bar{w}) = \sum_{n=1}^N (\bar{w}^T \bar{x}_n - t_n) \bar{x}_n^T = 0$$

$$\bar{w}^T \sum_{n=1}^N \bar{x}_n \bar{x}_n^T - \sum_{n=1}^N t_n \bar{x}_n^T = 0$$

22

Maximum de vraisemblance

$$\vec{w}^T \sum_{n=1}^N \vec{x}_n \vec{x}_n^T - \sum_{n=1}^N t_n \vec{x}_n^T = 0$$

En **isolant** \vec{w} , on obtient que

$$\vec{w}_{MV} = (X^T X)^{-1} X^T T$$

où

$$X = \begin{pmatrix} 1 & x_{1,1} & \cdots & x_{1,d} \\ 1 & x_{2,1} & \cdots & x_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N,1} & \cdots & x_{N,d} \end{pmatrix} \quad T = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix}$$

23

En résumé

Maximiser la vraisemblance de **données gaussiennes**

$$\vec{w} = \arg \max_{\vec{w}} \prod_{n=1}^N \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y_w(\vec{x}_n) - t_n)^2}{2\sigma^2}}$$

Équivaut à minimiser la **somme de l'erreur au carré**

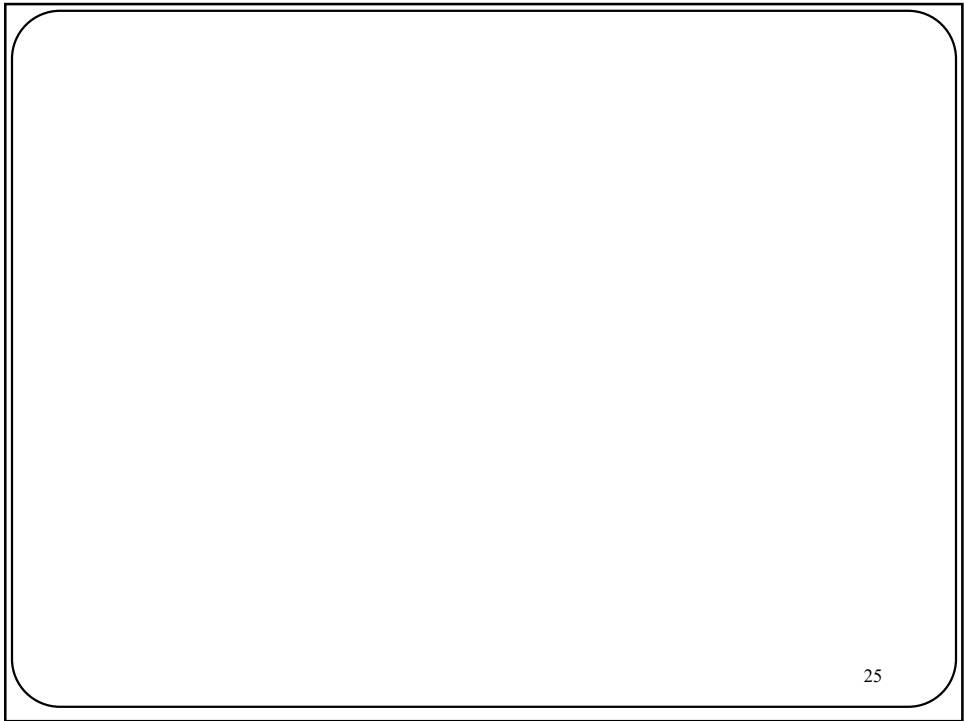
$$\vec{w} = \arg \min_{\vec{w}} \sum_{n=1}^N (\vec{w}^T \vec{x}_n - t_n)^2$$

Très important à comprendre!

Et en **forçant à zéro le gradient**, on obtient la solution

$$\vec{w}_{MV} = (X^T X)^{-1} X^T T$$

24



25

Maximum *a posteriori* (MAP)

Cherche les meilleurs paramètres \vec{w} en **maximisant la probabilité a posteriori**

Inconnue → **Connues**

$$\begin{aligned} \vec{w} &= \arg \max_{\vec{w}} p(\vec{w} | X, T, \sigma^2) \\ &= \arg \max_{\vec{w}} \frac{p(T | X, \vec{w}, \sigma^2) p(\vec{w})}{P(X, T, \sigma^2)} \quad \Rightarrow \text{Par le théorème de Bayes} \\ &= \arg \max_{\vec{w}} p(T | X, \vec{w}, \sigma^2) p(\vec{w}) \quad \leftarrow \text{Constante par rapport à } \vec{w} \end{aligned}$$

26

Maximum *a posteriori* (MAP)

On va émettre l'hypothèse que les données X, T ainsi que les paramètres \vec{w} sont iid de **distributions gaussiennes**

$$\begin{aligned}\vec{w} &= \arg \max_{\vec{w}} p(T | X, \vec{w}, \sigma^2) p(\vec{w}) \\ &= \arg \max_{\vec{w}} \prod_{n=1}^N N(t_n | y_{\vec{w}}(\vec{x}_n), \vec{w}, \sigma^2) N(\vec{w} | 0, \alpha^2)\end{aligned}$$

Moyenne nulle

$$N(t_n | y_{\vec{w}}(\vec{x}_n), \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y_{\vec{w}}(\vec{x}_n) - t_n)^2}{2\sigma^2}}$$

$$N(\vec{w} | 0, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|} e^{-\frac{\vec{w}^T \Sigma^{-1} \vec{w}}{2}}$$

27

Maximum *a posteriori* (MAP)

Cherche les meilleurs paramètres \vec{w} en **maximisant la probabilité *a posteriori***

$$\begin{aligned}\vec{w} &= \arg \max_{\vec{w}} \ln \left[\prod_{n=1}^N N(t_n | y_{\vec{w}}(x_n), \sigma^2) N(\vec{w} | 0, \Sigma) \right] \\ &= \arg \max_{\vec{w}} \sum_{n=1}^N \ln \left[N(t_n | y_{\vec{w}}(x_n), \sigma^2) N(\vec{w} | 0, \Sigma) \right] \\ &= \arg \max_{\vec{w}} \sum_{n=1}^N \ln \left[\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(t_n - y_{\vec{w}}(x_n))^2}{2\sigma^2}} \right] + \ln \left[\frac{1}{(2\pi)^{d/2} |\Sigma|} e^{-\frac{\vec{w}^T \Sigma^{-1} \vec{w}}{2}} \right] \\ &= \arg \max_{\vec{w}} \sum_{n=1}^N \left[-\frac{(t_n - y_{\vec{w}}(x_n))^2}{2\sigma^2} - \frac{\vec{w}^T \Sigma^{-1} \vec{w}}{2} \right] + \ln \left[\frac{1}{\sqrt{2\pi}\sigma} \right] + \ln \left[\frac{1}{(2\pi)^{d/2} |\Sigma|} \right]\end{aligned}$$

28

Maximum *a posteriori* (MAP)

Cherche les meilleurs paramètres \bar{w} en maximisant la probabilité *a posteriori*

$$\bar{w} = \arg \max_{\bar{w}} \sum_{n=1}^N - \frac{(t_n - y_{\bar{w}}(x_n))^2}{\cancel{\sigma^2}} - \frac{\bar{w}^T \Sigma^{-1} \bar{w}}{\cancel{\sigma^2}} + \ln \left[\frac{1}{\sqrt{2\pi}\sigma} \right] + \ln \left[\frac{1}{(2\pi)^{d/2} |\Sigma|} \right]$$

Constante par rapport à \bar{w}

De plus, comme on ne connaît généralement pas Σ , on suppose qu'elle est isotropique

$$\Sigma = \begin{pmatrix} \alpha^2 & 0 & \dots & 0 \\ 0 & \alpha^2 & & \vdots \\ \vdots & & \ddots & \\ 0 & 0 & \dots & \alpha^2 \end{pmatrix} = \alpha^2 I$$

29

Maximum *a posteriori* (MAP)

Cherche les meilleurs paramètres \bar{w} en maximisant la probabilité *a posteriori*

$$\begin{aligned} \bar{w} &= \arg \max_{\bar{w}} \sum_{n=1}^N - \frac{(t_n - y_{\bar{w}}(x_n))^2}{\sigma^2} - \frac{\bar{w}^T \Sigma^{-1} \bar{w}}{\sigma^2} \\ &= \arg \max_{\bar{w}} \sum_{n=1}^N - \frac{(t_n - y_{\bar{w}}(x_n))^2}{\sigma^2} - \frac{\bar{w}^T \bar{w}}{\alpha^2} \\ &= \arg \min_{\bar{w}} \sum_{n=1}^N (t_n - y_{\bar{w}}(x_n))^2 + \lambda \bar{w}^T \bar{w} \quad \text{où } \lambda = \frac{\sigma^2}{\alpha^2} \end{aligned}$$

30

Maximum *a posteriori* (MAP)

Cherche les meilleurs paramètres \vec{w} en **maximisant la probabilité a posteriori**

$$\vec{w} = \arg \min_{\vec{w}} \sum_{n=1}^N (t_n - y_{\vec{w}}(x_n))^2 + \lambda \vec{w}^T \vec{w}$$

NOTE

Formule également connue sous le nom de
« **régression de Ridge** »

Voir sklearn pour une implémentation simple

scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html

31

Maximum *a posteriori* (MAP)

$$\vec{w} = \arg \min_{\vec{w}} \underbrace{\sum_{n=1}^N (t_n - y_{\vec{w}}(x_n))^2 + \lambda \vec{w}^T \vec{w}}_{E_D(\vec{w})}$$

$E_D(\vec{w})$

Les meilleurs paramètres sont ceux qui correspondent au gradient nul

$$\nabla_{\vec{w}} E_D(\vec{w}) = 0$$

32

Maximum *a posteriori* (MAP)

Puisque $y_{\vec{w}}(\vec{x}) = \vec{w}^T \vec{x}$ (voir quelques pages précédentes)

$$E_D(\vec{w}) = \sum_{n=1}^N (t_n - \vec{w}^T \vec{x})^2 + \lambda \vec{w}^T \vec{w}$$

En forçant le **gradient à zéro** $\nabla E_D(W) = 0$ on peut démontrer que

$$W_{\text{MAP}} = (X^T X + \lambda I)^{-1} X^T T$$

Cette preuve est sujette à devoir...

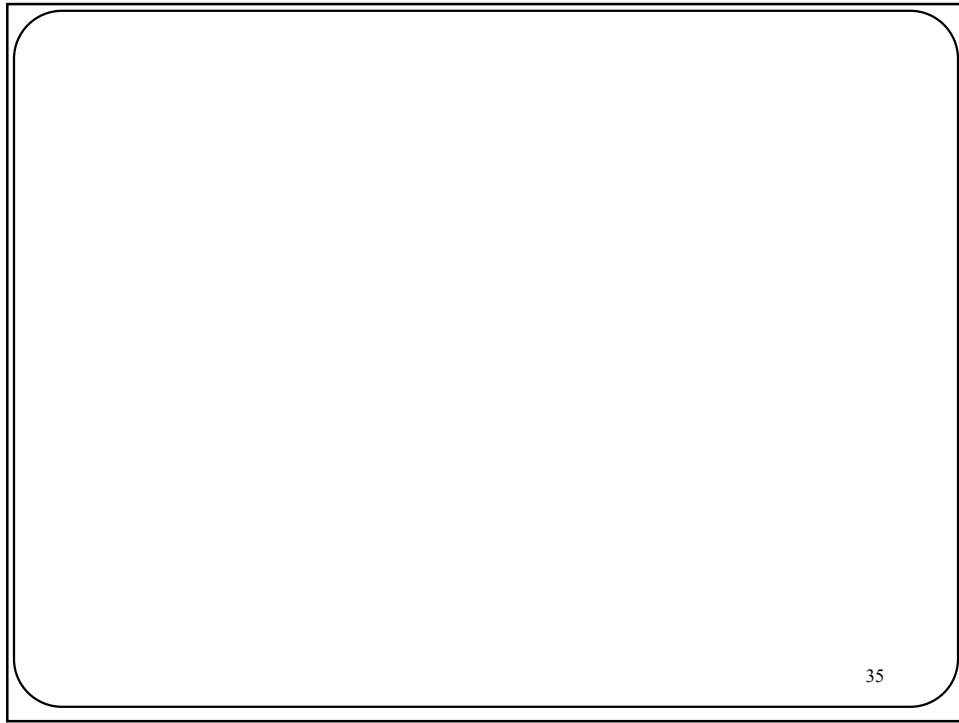
33

Maximum *a posteriori* (MAP)

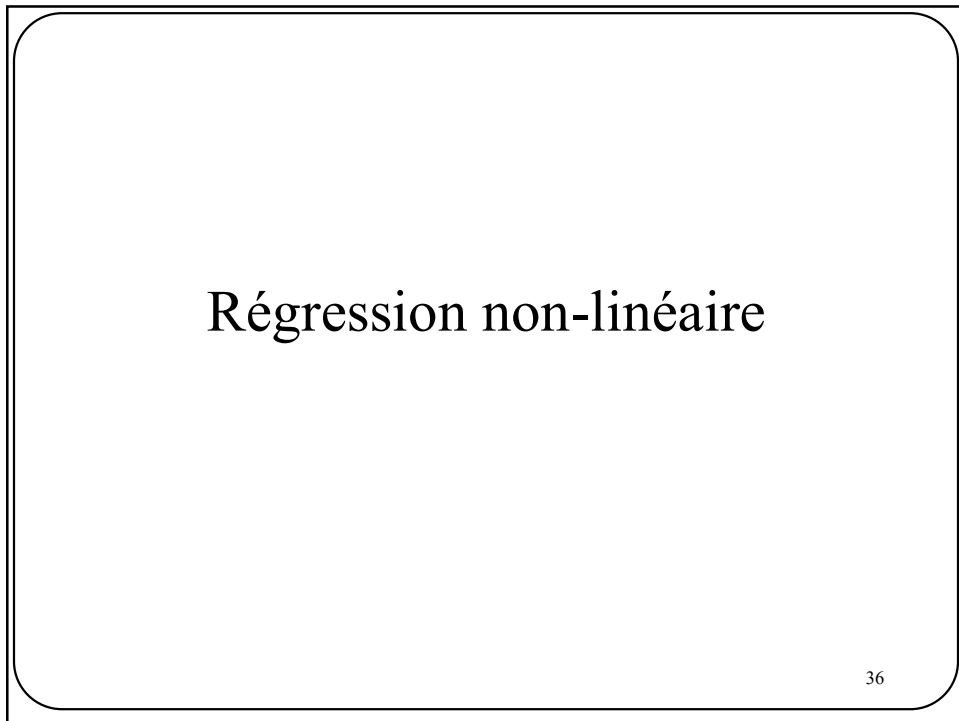
$$W_{\text{MAP}} = (X^T X + \lambda I)^{-1} X^T T$$

- Le terme de régularisation $\lambda \frac{W^T W}{2}$ est souvent appelé **weight decay**
- La régression avec un **weight decay** est souvent appelé **régression de Ridge**
- On retrouve le **maximum de vraisemblance** lorsque $\lambda = 0$
- Permet de réduire le **sur-apprentissage** lorsque $\lambda > 0$

34



35



36

Régression linéaire

- Le modèle de **régression linéaire** est le suivant :

$$y_{\vec{w}}(\vec{x}) = w_0 + w_1x_1 + w_2x_2 + \dots + w_dx_d$$

$$\text{où } \vec{x} = (x_1, x_2, \dots, x_d)$$

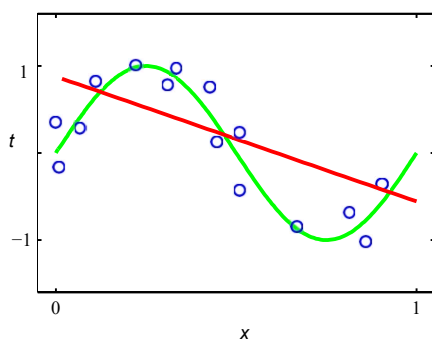
- **Problème**

- Un modèle linéaire est souvent **pas assez flexible** pour bien représenter les données

37

37

Exemple sous-apprentissage



— Modèle dont sont issues les données

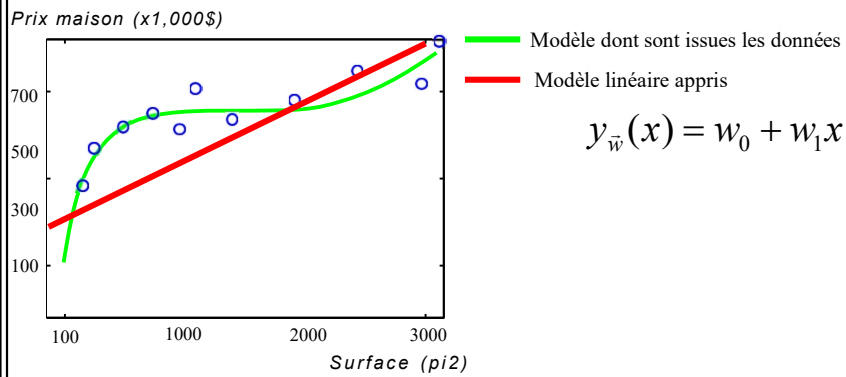
— Modèle linéaire appris

$$y_{\vec{w}}(x) = w_0 + w_1x$$

38

38

Exemple sous-apprentissage



39

39

Fonctions de base

Solution: on va projeter les données dans un **espace plus grand**, là où les données sont **distribuées linéairement**.

=> régression sur des données à M dimensions au lieu de d dimensions ($M > d$)

$$\phi: R^d \rightarrow R^M$$

40

40

Fonctions de base

Exemple: au lieu de faire une régression linéaire 1D,
=> faire une régression linéaire en 4D

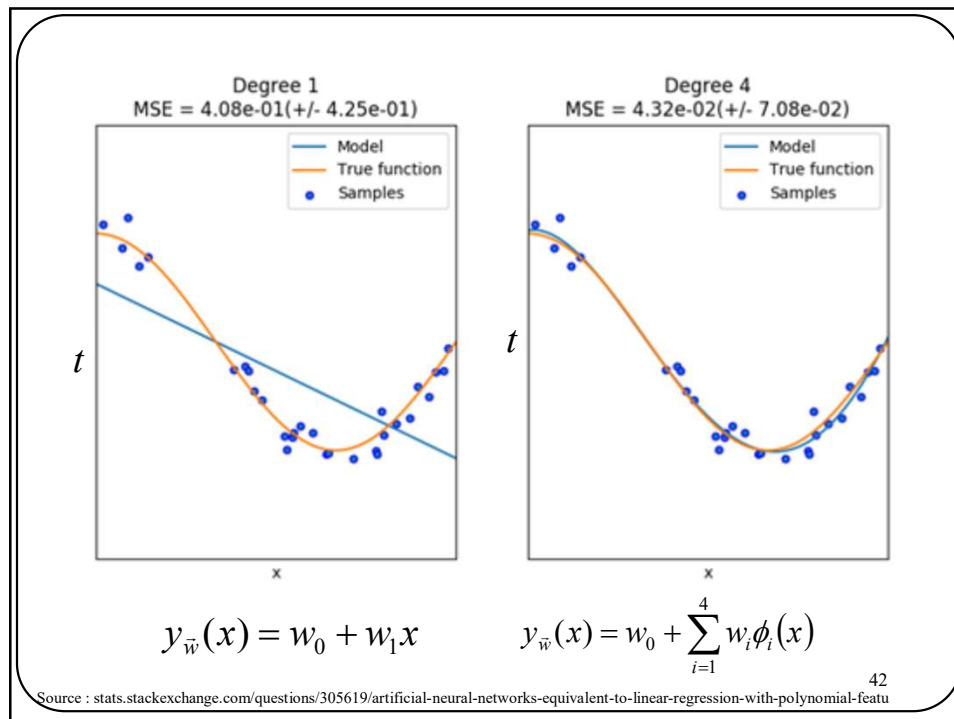
$$\phi(x) \rightarrow (x, x^2, x^3, x^4)$$

$$y_{\bar{w}}(x) = w_0 + w_1x \quad \rightarrow \quad y_{\bar{w}}(x) = w_0 + w_1x + w_2x^2 + w_3x^3 + w_4x^4$$

$$= w_0 + \sum_{i=1}^4 w_i\phi_i(x)$$

41

41



42

Fonctions de base

De façon plus générale

$$y_{\vec{w}}(\vec{x}) = w_0 + \sum_{i=1}^M w_i \phi_i(\vec{x})$$

où les $\phi_i(\vec{x})$ sont des **fonctions de base** (*basis functions*)

- Cas particulier : $\phi_i(\vec{x}) = x_i$ et $M = d + 1$

43

43

Fonctions de base

Pour **simplifier la notation**, on va supposer que $\phi_0(\vec{x}) = 1$ afin d'inclure le **biais** dans la sommation

$$y_{\vec{w}}(\vec{x}) = \sum_{i=0}^{M-1} w_i \phi_i(\vec{x})$$

hyperparamètre

paramètre

Fonction de base

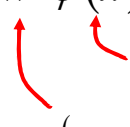
44

44

Fonctions de base

Pour **simplifier la notation**, on va supposer que $\phi_0(\vec{x}) = 1$
afin d'inclure le **biais** dans la sommation

$$\begin{aligned}y_{\vec{w}}(\vec{x}) &= \sum_{i=0}^{M-1} w_i \phi_i(\vec{x}) \\ &= \vec{w}^T \vec{\phi}(\vec{x})\end{aligned}$$


 (w_0, \dots, w_{M-1}) $(\phi_0(\vec{x}), \dots, \phi_{M-1}(\vec{x}))^T$

45

45

Fonctions de base

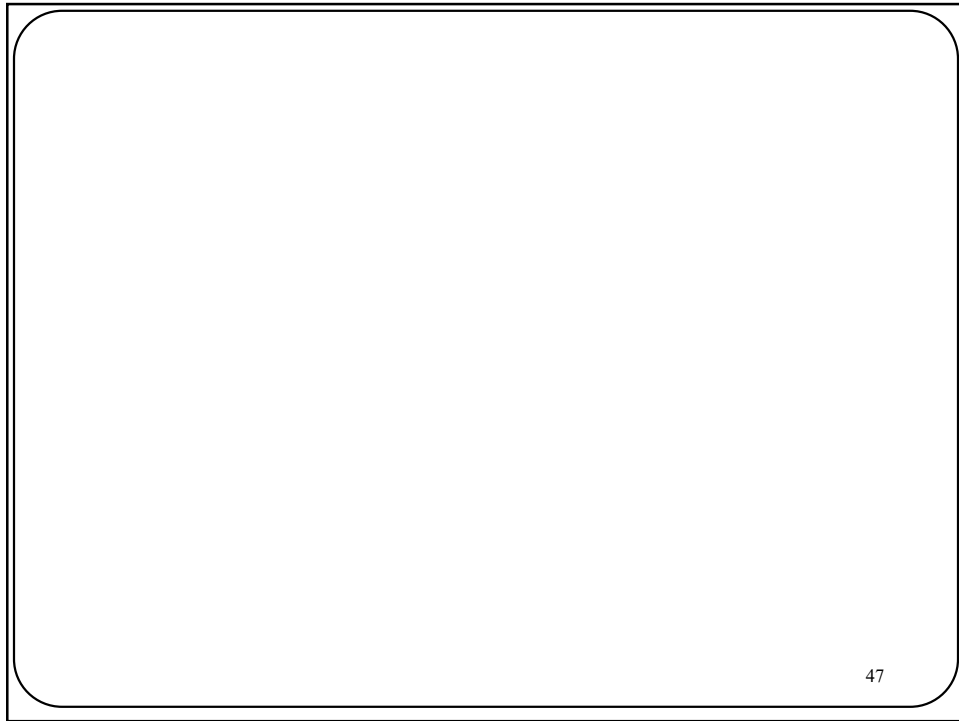
Une des fonctions de base les plus fréquentes est la **fonction polynomiale**

$$\phi_i(x) = x^i$$

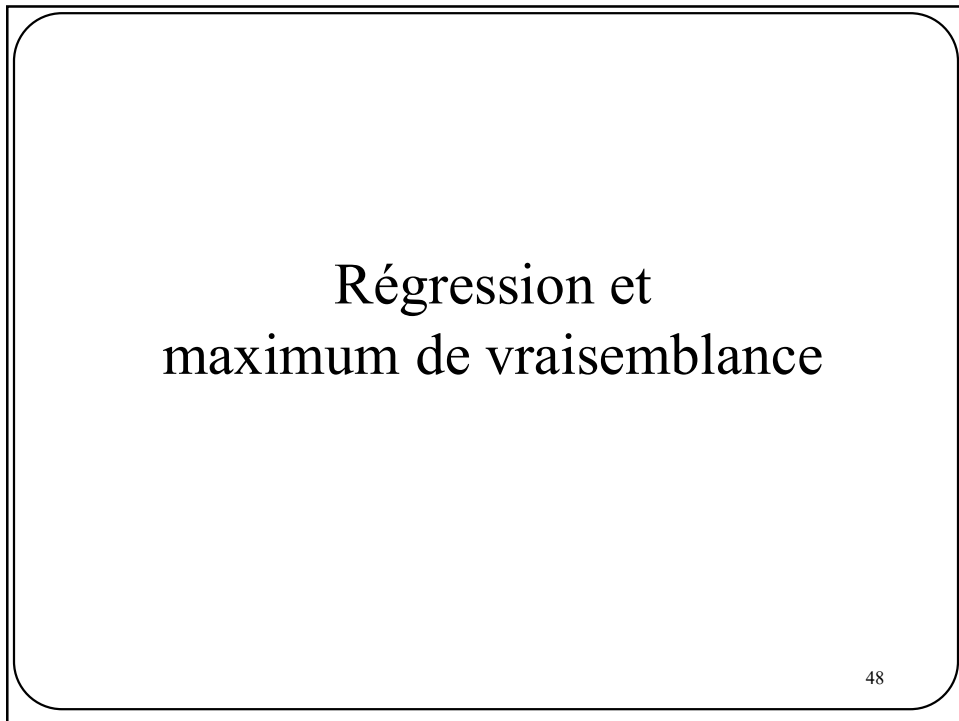
=> Régression polynomiale

46

46



47

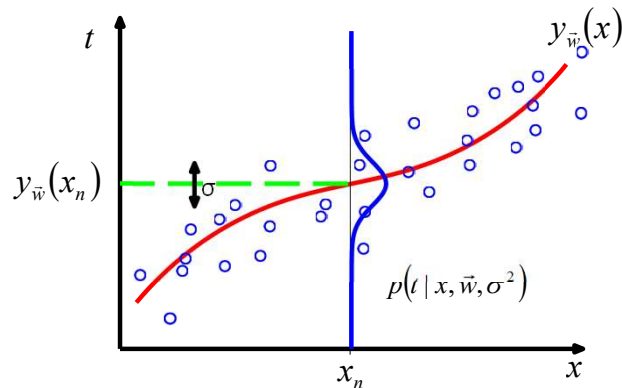


48

Formulation probabiliste

Loi gaussienne conditionnelle

Comme auparavant, on suppose ici que les données sont corrompues par un **bruit gaussien**.



49

Maximum de vraisemblance

Suivant le même processus que précédemment, on obtient que

$$\vec{w} = \arg \min_{\vec{w}} \underbrace{\sum_{n=1}^N \left(\vec{w}^T \vec{\phi}(\vec{x}_n) - t_n \right)^2}_{E_D(\vec{w})}$$

Et ici aussi, le « meilleur » \vec{w} est celui pour lequel le **gradient est nul**

$$\nabla_{\vec{w}} E_D(\vec{w}) = \sum_{n=1}^N \left(\vec{w}^T \vec{\phi}(\vec{x}_n) - t_n \right) \vec{\phi}(\vec{x}_n)^T = 0$$

50

Maximum de vraisemblance

$$\vec{w}^T \sum_{n=1}^N \vec{\phi}(\vec{x}_n) \vec{\phi}(\vec{x}_n)^T - \sum_{n=1}^N t_n \vec{\phi}(\vec{x}_n)^T = 0$$

En isolant \vec{w} , on obtient que

$$\vec{w}_{MV} = (\Phi^T \Phi)^{-1} \Phi^T T$$

où

$$\Phi = \begin{pmatrix} \phi_0(\vec{x}_1) & \phi_1(\vec{x}_1) & \cdots & \phi_{M-1}(\vec{x}_1) \\ \phi_0(\vec{x}_2) & \phi_1(\vec{x}_2) & \cdots & \phi_{M-1}(\vec{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\vec{x}_N) & \phi_1(\vec{x}_N) & \cdots & \phi_{M-1}(\vec{x}_N) \end{pmatrix} \quad T = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix}$$

51

Maximum *a posteriori* (MAP)

Encore une fois, en suivant les mêmes étapes qu'auparavant, la solution au maximum *a posteriori* s'exprime sous la forme suivante

$$\vec{w} = \arg \min_{\vec{w}} \underbrace{\sum_{n=1}^N \frac{(t_n - \vec{w}^T \vec{\phi}(\vec{x}_n))^2}{2}}_{E_D(\vec{w})} + \lambda \frac{\vec{w}^T \vec{w}}{2}$$

Formule également connue sous le nom de
« régression de Ridge »

NOTE

Exemple pour une fonction de base polynomiale

https://scikit-learn.org/stable/auto_examples/linear_model/plot_polynomial_interpolation.html

52

Maximum *a posteriori* (MAP)

Ici aussi on obtient la solution optimale en forçant le **gradient à zéro**

$$\nabla E_D(\vec{w}) = 0$$

Et ainsi obtenir

$$\vec{w}_{\text{MAP}} = (\Phi^T \Phi + \lambda I)^{-1} \Phi^T T$$

Cette preuve est sujette à devoir...

53

Maximum *a posteriori* (MAP)

$$\vec{w}_{\text{MAP}} = (\Phi^T \Phi + \lambda I)^{-1} \Phi^T T$$

- Le terme de régularisation $\lambda \frac{\vec{w}^T \vec{w}}{2}$ est souvent appelé **weight decay**
- La régression avec un **weight decay** est souvent appelé **régression de Ridge**
- On retrouve le **maximum de vraisemblance** lorsque $\lambda = 0$
- Permet de réduire le **sur-apprentissage** lorsque $\lambda > 0$

54

Régression avec prédictions multiples

55

55

Régression avec prédiction simple

RAPPEL

$$D = (X, T)$$

où

$$X = \{\vec{x}_1, \dots, \vec{x}_N\} \text{ et } \vec{x}_i \in R^d$$

$$T = \{t_1, \dots, t_N\}$$

Exemple: prédiction du prix d'une maison (d=1)

x : Surface (pi2) t : prix maison

250	89,000\$
554	197,000\$
710	261,000\$
...	...
2890	681,000\$

56

56

Régression avec prédiction simple

RAPPEL

$$D = (X, T)$$

où

$$X = \{\vec{x}_1, \dots, \vec{x}_N\} \text{ et } \vec{x}_i \in R^d$$

$$T = \{t_1, \dots, t_N\}$$

Exemple: prédiction du prix d'une maison (d=2)

\vec{x} : Surface (pi2); âge de la maison (années) t : prix maison

(250, 45)	89,000\$
(554, 90)	197,000\$
(710, 12)	261,000\$
...	...
(2890, 51)	681,000\$

57

57

Régression avec prédictions multiples

$$D = (X, T)$$

où

$$X = \{\vec{x}_1, \dots, \vec{x}_N\} \text{ et } \vec{x}_i \in R^d$$

$$T = \{\vec{t}_1, \dots, \vec{t}_N\} \text{ et } \vec{t}_i \in R^K$$

Exemple: prédiction de plusieurs éléments d'une maison (d=2, K=3)

\vec{x} : Surface (pi2); âge de la maison (années) \vec{t} : prix maison; coût chauffage; taxes

(250, 45)	(89,000\$, 720\$, 1231\$)
(554, 90)	(197,000\$, 1301\$, 1711\$)
(710, 12)	(261,000\$, 1445\$, 1199\$)
...	...
(2890, 51)	(681,000\$, 3789\$, 2998\$)

58

58

Régression avec prédictions multiples

Le modèle doit maintenant **prédire un vecteur**

$$y_{\mathbf{W}}(\vec{x}) = \mathbf{W}^T \vec{\phi}(\vec{x})$$

Où \mathbf{W} est une matrice $M \times K$

Chaque ligne de \mathbf{W} peut être vue comme un vecteur W_k du modèle $y_{\vec{w}_k}(\vec{x}) = \vec{w}_k^T \vec{\phi}(\vec{x})$ pour la k^e cible

59

59

Régression avec prédictions multiples

Si on suppose encore une fois un modèle de bruit gaussien

$$p(\vec{t} | \vec{x}, \mathbf{W}, \sigma^2) = N(\vec{t} | \vec{y}_{\mathbf{W}}(\vec{x}), \sigma^2)$$

On peut montrer que la solution du **maximum de vraisemblance** est

$$\mathbf{W}_{\text{ML}} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{T}$$

Et la solution du **maximum a posteriori** est

$$\mathbf{W}_{\text{MAP}} = (\Phi^T \Phi + \lambda \mathbf{I})^{-1} \Phi^T \mathbf{T}$$

Où \mathbf{T} est une matrice $N \times K$

60

60

Résumé régression linéaire

Paramètre **Fonction de base**

- **Modèle** : $y_{\vec{w}}(\vec{x}) = \sum_{i=0}^{M-1} w_i \phi_i(\vec{x}) = \vec{w}^T \vec{\phi}(\vec{x})$

- Entraînement par **maximum de vraisemblance**: $\vec{w}_{MV} = (\Phi^T \Phi)^{-1} \Phi^T T$

- Entraînement par **maximum a posteriori**: $\vec{w}_{MAP} = (\Phi^T \Phi + \lambda I)^{-1} \Phi^T T$

- **Hyper-paramètres** : M et λ

61