# La régression linéaire

Mon premier modèle



Rafik LACHAAL

# Sommaire

- Le principe de régression linéaire univariée.
- La régression multivariée.
- Les métriques de performances
- La régression polynomiale.
- Interaction entre variables
- La régression régularisée.

- Mise en pratique

# Le principe de régression linéaire univariée

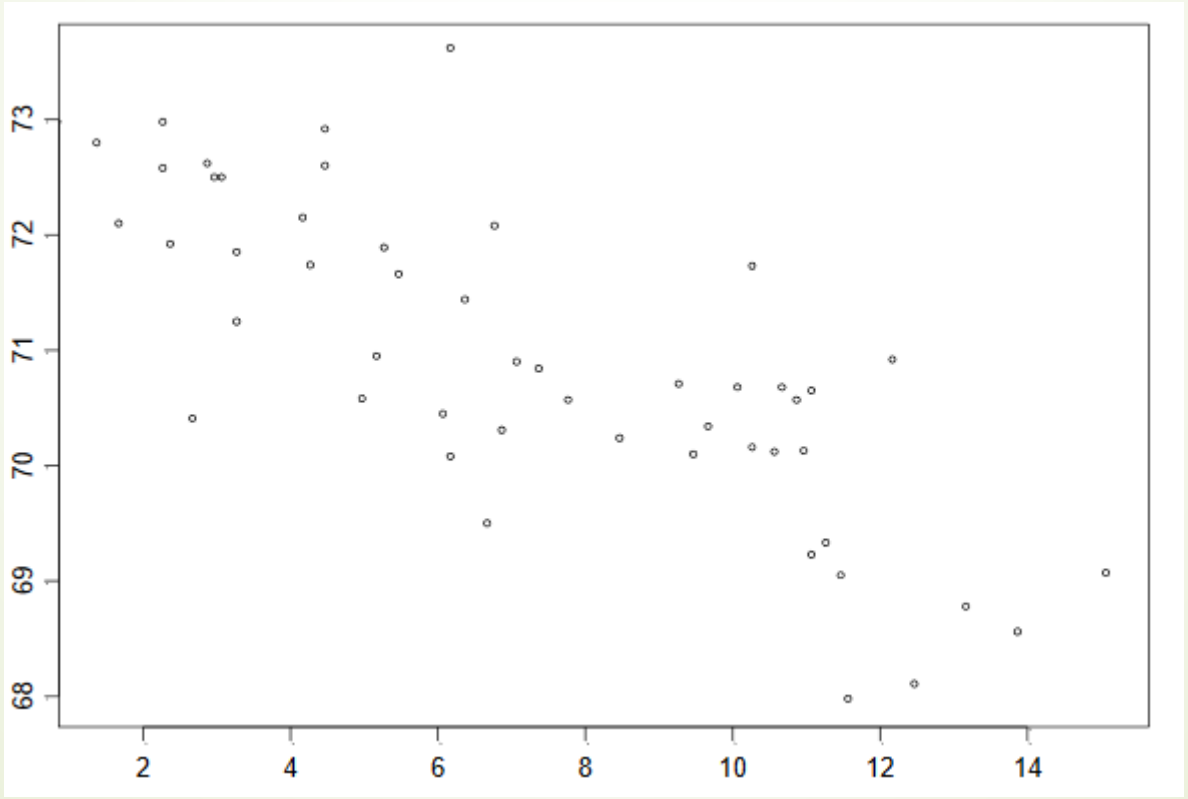
3 étapes pour passer des données brut au meilleur modèle :

- La définition de la fonction hypothèse
- La construction d'une fonction coût
- La minimisation de cette fonction de coût

# Le principe de régression linéaire univariée

Le modèle dépend d'une unique variable explicative nommé X

**Exemple** : existe un lien entre le nombre de meurtre et l'espérance de vie



# Le principe de régression linéaire univariée

- **la fonction hypothèse**

$$valeur\ d'entrée\ x \xrightarrow{hypothèse\ h} valeur\ de\ sortie\ y$$

Dans le cas de la régression linéaire l'hypothèse h sera de la forme :

$$h(X) = \theta_0 + \theta_1 X$$

En représentation matriciel $\theta_0\ et\ \theta_1$ forme un vecteur : $\Theta = \begin{pmatrix} \theta_0 \\ \theta_1 \end{pmatrix}$

On doit donc trouver le meilleur couple $(\theta_0, \theta_1)$ tel que h(x) soit proche de y

# Le principe de régression linéaire univariée



- **construction d'une fonction coût**

La somme des erreurs unitaire pour les $x_i$ est définis par :

$$\sum_{i=1}^{m}(y_i - \mathrm{h}(x_i))^2$$

La fonction de coût est alors définit en normant cette somme par le nombre m de point dans la base d'apprentissage :

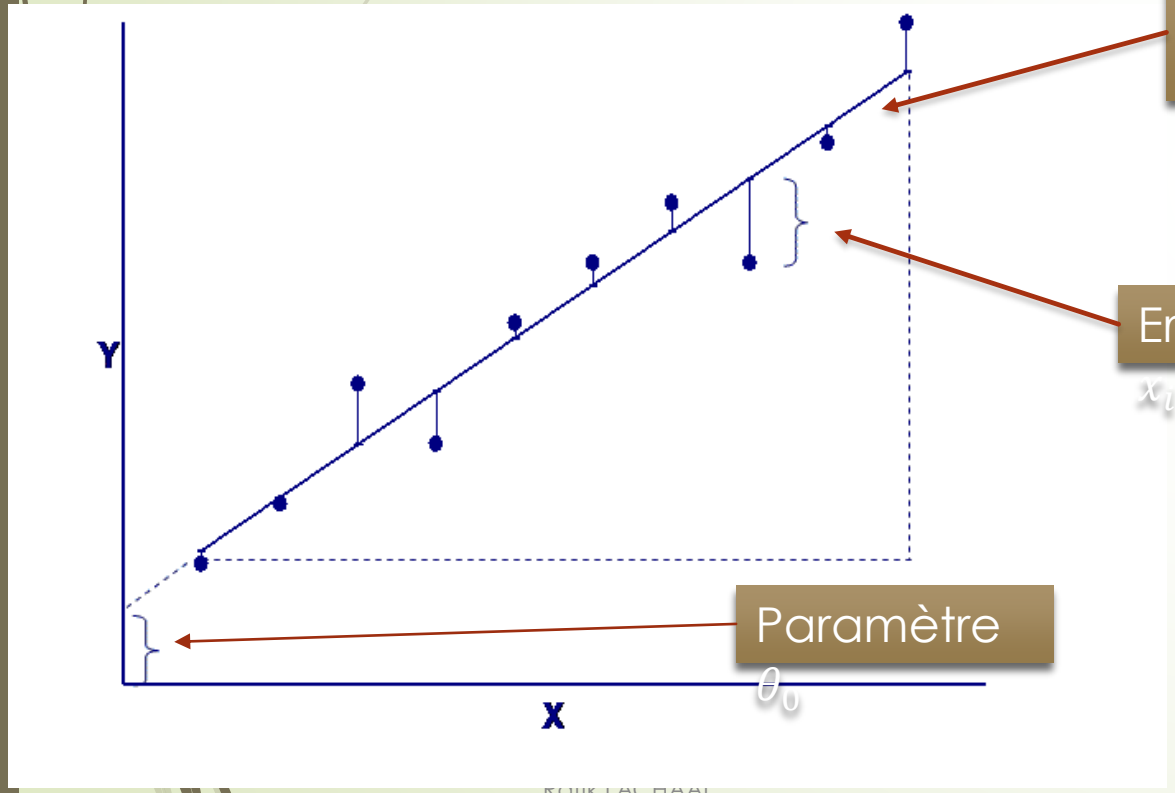
$$j(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(\mathrm{h}(x_i) - y_i)^2$$

Cela revient à écrire :

$$j(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(\theta_0 + \theta_1 x_i - y_i)^2$$

# Le principe de régression linéaire univariée

- **construction d'une fonction coût**




$\theta_0$ : L'ordonnée à l'origine
$\theta_1$ : Le coefficient directeur

# Le principe de régression linéaire univariée

- **Minimiser la fonction de coût :**

Ainsi trouver les meilleurs paramètres $(\theta_0, \theta_1)$ revient à minimiser la fonction $j(\theta_0, \theta_1)$.

Méthode de résolution, **la descente de gradient** :

- Itération 0 : initialisation d'un couple $(\theta_0, \theta_1)$
- Itérer jusqu'à convergence :

$$\theta_j := \theta_j - \alpha \frac{d}{d\theta_j} j(\theta_0, \theta_1) \quad \text{pour j = 1 et 0}$$

A chaque itération on choisi la meilleur pente pour la fonction j pour ce diriger itération après itération vers le minimum de notre fonction.

$\alpha$ est le « learning rate » : plus il est grand et plus le pas est grand entre 2 itérations.

# La régression multivariée.

- Même principe que la régression univarié à la différence que la fonction h admet plusieurs variables en entrées.
- Pour n variables d'entrées la fonction h prend la forme :

$$h(X) = \theta_0 + \theta_1 X_1 + \theta_2 X_2 + \cdots + \theta_n X_n$$

Les données d'entrées se présente sous la forme d'une matrice :

$$\begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{bmatrix}$$

# La régression multivariée.

- La fonction coût est une généralisation de celle vu pour la régression univarié :

$$j(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(\mathrm{h}(x_i) - y_i)^2$$

- Pour minimiser le coût in utilise encore la descente de gradient :
  - Itération 0 : initialisation d'un couple $(\theta_0, \theta_1)$
  - Itérer jusqu'à convergence :

$$\theta_j := \theta_j - \alpha \frac{d}{d\theta_j} j(\theta_0, \theta_1) \quad \text{pour j = 0, …, n}$$

# La régression multivariée.

- **Attention :**

Lorsqu'on utilise la descente de gradient pour la régression multivarié la différence d'échelle entre les variables risque de ne pas faire converger l'algorithme de résolution.

**Exemple** : Si on cherche à prédire le prix d'appartement parisien en fonction de leurs superficies et du nombre pièces on aura :

- Pour la superficie des valeur comprise entre 20 et 200
- Pour le nombre de pièces des valeurs comprise entre 1 et 8

Soit à peut prés un rapport 20

# La régression multivariée.

- **Pour palier ce problème :**

On met les variables à la même échelle par exemple entre -1 et 1.

On parle de **normalisation** (ou scaling).

En pratique avec Scikit-learn :

*Sklearn.preprocessing.StandardScaler*

(Donne à toute les varibles une moyenne nulle et une variance de 1)

*Sklearn.preprocessing.MixMaxScaler*

(Donne au variables entre 0 et 1)

# La régression multivariée.

- Autre méthode de résolution :
  - Les moindres carrés ordinaires
  - Les moindres carrées généralisés
  - Le maximum de vraisemblance

# Mesures de performance

Les mesures que l'on utilise pour évaluer la qualité d'un modèle de régression :

- La valeur observée d'une série à prédire ($y_i$)
- La valeur prédite par le modèle pour cette même valeur observée ($\widehat{y_i}$)
- Prévision naïve de référence qui est la moyenne de la valeur observée ($\bar{y}$)

# Mesures de performance

Elles permettent de calculer pour tout $i$ des $m$ observations :

- L'erreur de prédiction du modèle : $y_i - \hat{y}_i$
- L'erreur de prédiction naïve : $y_i - \bar{y}$

C'est avec cela que l'on définit les indicateurs de performance du modèle.

# Mesures de performance

- **L'erreur moyenne absolue** (MAE, Mean Absolute Error) **:**

$$\frac{1}{m}\sum_{i=1}^{m}|y_i - \hat{y}_i|$$

- **La racine carré de la moyenne du carré des erreurs** (RMSE, Root Mean Squared Error) :

$$\sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}$$

Par rapport à MAE, RMSE permet de punir plus sévèrement les grandes erreurs.

# Mesures de performance

- **Le coefficient de détermination** ($R^2$) :

$$1 - \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{m}(y_i - \bar{y})^2}$$

  - Permet d'avoir une **idée général** de la performance du modèle. Il mesure l'adéquation entre le modèle et les données.
  - Sa valeur est comprise entre 0 et 1, 0 indiquant une adéquation nul et 1 une adéquation parfaite.
  - Valable seulement pour la régression linéaire.

# Mesures de performance

- **Le critère d'information d'Akaike** (AIC) :

Formulation dans le cas d'erreur distribuées normalement :

$$-2\log\left(\sum_{i=1}^{m}(y_i - \hat{y}_i)^2\right) + 2(k+1)$$

Avec k le nombre de paramètres du modèle.

Alternative au $R^2$ pour les modèles non linéaire.

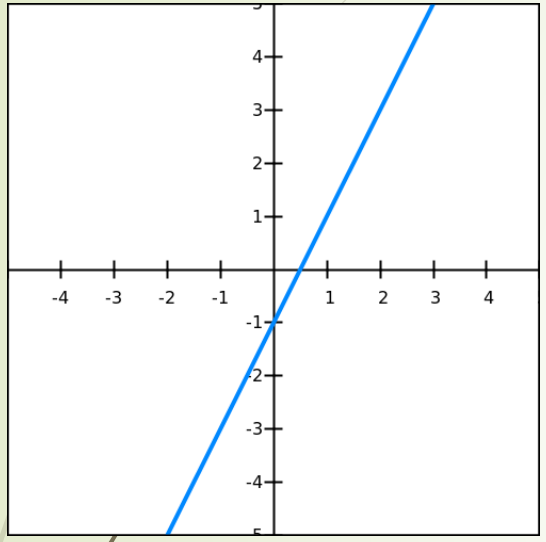
Meilleur est un modèles, plus petit est l'AIC.

# La régression polynomiale

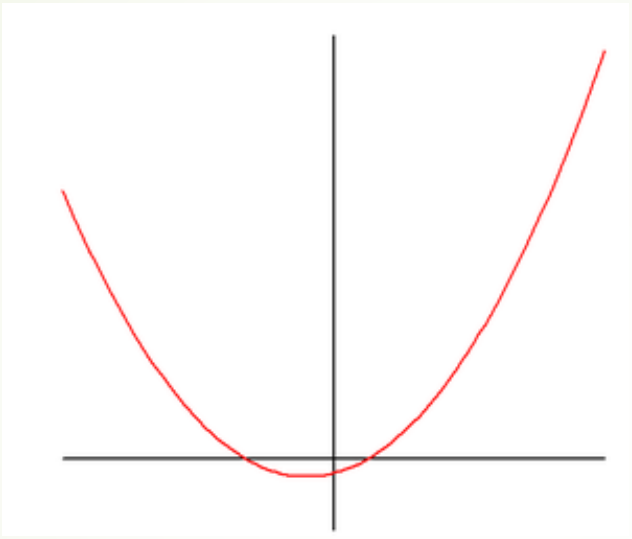
- La régression polynomiale est une forme particulière de la régression multivariée.

- Elle permet de lier les variables par un polynôme de dégrées k, c'est ce qui permet d'introduire la non linéarité dans les relation entre variables.

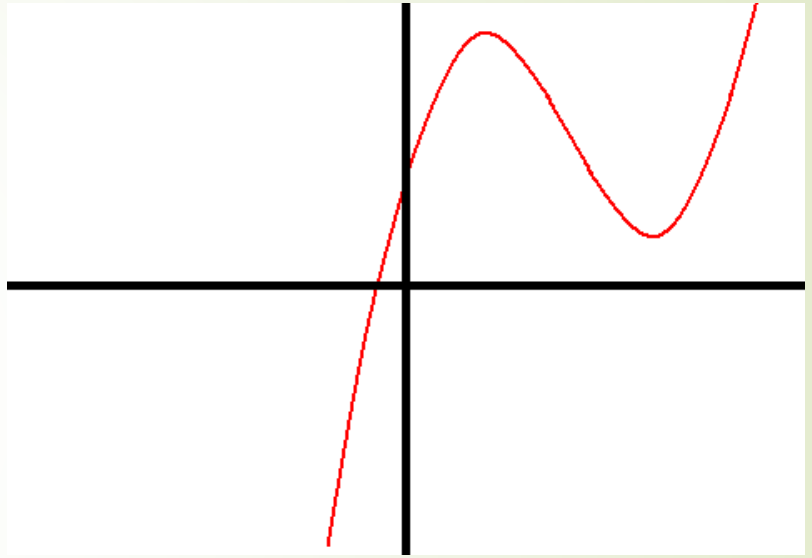
# La régression polynomiale




Polynôme d'ordre 1
de la forme :
$y = a + bx$



Polynôme d'ordre 2
de la forme :
$y = a + bx + cx^2$



Polynôme d'ordre 3 de
la forme :
$y = a + bx + cx^2 + dx^3$

Intuitivement chaque nouvel ordre va permettre d'ajouter un pli à la courbe.

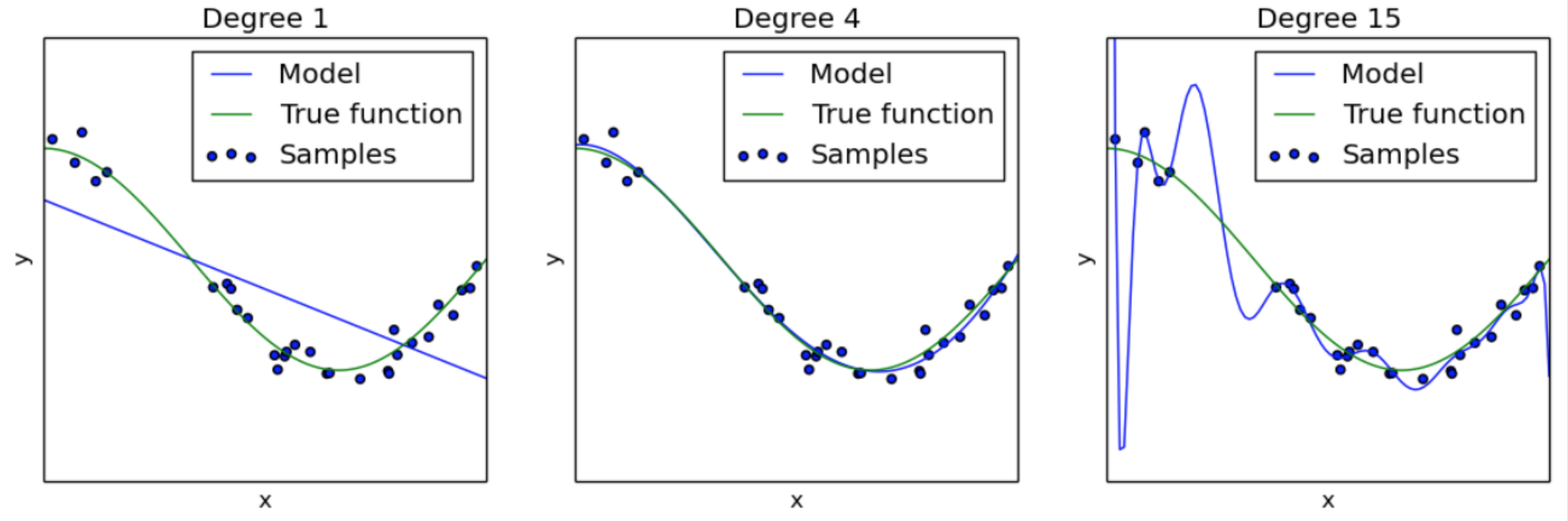
# La régression polynomiale

- On évalues chaque variable en l'associant à tout les degrés polynomiaux de 1 à k. Chacun de ces polynôme a son propre coéfficient.
- Exemple, un polynôme de degré 2 à deux variables explicatives s'écrira :

$$h(X) = \theta_0 + \theta_1 X_1 + \theta_2 X_2 + \theta_1 X_1^2 + \theta_2 X_2^2$$

- En pratique cela revient à faire une régression multivarié en ajoutant deux colonnes à sont jeux de données prenant les valeurs de $X_1^2$ et $X_2^2$.
- **Attention** : polynôme trop grand = risque de sur-apprentissage.

# La régression polynomiale



# Interaction entre variables



La possibilité existe que l'effet d'une variable x1, ou de x2, ou... de $x_n$ ne soit pas constant, mais varie en fonction des valeurs prises par une des autres variables indépendantes introduite dans le modèle.

- Par exemple, que l'effet de x1diffère selon la valeur prise par x2.
- On dit dans ce cas **qu'il y a interaction entre x1et x2**

Exemple :

On cher a expliquer le niveau de revenu en fonction de et du niveau d'éducation.

On peut l'hypothèse que l'effet positif de l'âge sur le revenu est plus fort pour les personnes avec un niveau de formation plus élevé, car celles-ci, au fur et à mesure qu'elles avancent en âge, peuvent mieux tirer parti des opportunités de promotion et bénéficient davantage de la règle d'ancienneté.

On suppose donc qu'il y a interaction entre l'âge et l'éducation.

# Interaction entre variables

Comment tester une telle hypothèse ?

Notre modèle de base serait :

$$h(X) = \theta_0 + \theta_1 X_1 + \theta_2 X_2$$

Avec $X_1$étant l'âge et $X_2$ étant l'éducation (nombre d'années d'étude après le bac).

Notre hypothèse stipule que :

$$\theta_1 = C + DX_2$$

Ce sui revient a écrire que :

$$h(X) = \theta_0 + (C + DX_2)X_1 + \theta_2 X_2$$
$$h(X) = \theta_0 + CX_1 + DX_2X_1 + \theta_2 X_2$$

Au final on abouti à :

$$h(X) = \theta_0 + \theta_1 X_1 + \theta_2 X_2 + \theta_3 X_2 X_1$$

# Interaction entre variables et régression polynomial

Concrètement :

```
from sklearn.preprocessing import PolynomialFeatures
```

La transformation PolynomialFeatures créée de nouvelles features en multipliant les variables les unes avec les autres.

Pour le degré deux et trois features a, b, c, on obtient les nouvelles features : 1, a, b, c, $a^2$, ab, ac, $b^2$, bc, $c^2$.

# La régression régularisée

- Problème avec la régression multivarié :
  Pas de contrainte sur les paramètres du modèles, donc on peut obtenir des paramètres avec de grande valeurs. En conséquence de faibles changement dans les données (même d'arrondie) peuvent produire des modèles très différents.
- Pour compenser ce problème on ajoute une fonction de pénalité $P(\lambda, \theta)$ dans la fonction de coût  qui va gérer la pénalité selon un paramètre $\lambda$ :

$$j(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (\mathrm{h}(x_i) - y_i)^2 + P(\lambda, \theta)$$

# La régression régularisée

- Méthodes de régularisation :
  - La régression ridge
  - Le LASSO
  - ElasticNet

Ces trois méthodes sont aussi résolu par la descente de gradient.

# La régression régularisée

- **La régression ridge** :
  La fonction de pénalité est basé sur la norme dite $l_2$ que l'on note $\|\Theta\|_{l_2}$ :

$$\|\Theta\|_{l_2} = \sqrt{\Theta_1^2 + \cdots + \Theta_n^2}$$

On à donc la fonction coût :

$$j(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(\mathrm{h}(x_i) - y_i)^2 + \lambda\sum_{j=1}^{n}\Theta_j^2$$

Cette mesure de pénalité permet de limité l'instabilité lié à des variables explicatives trop corrélées entre elles.

# La régression régularisée

- **Le LASSO** :

Cette fois ci on utilise la norme $l_1$ que l'on note $\|\Theta\|_1$ :

$$\|\Theta\|_{l_2} = |\Theta_1| + \cdots |\Theta_2|$$

On à donc la fonction coût :

$$j(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(\mathrm{h}(x_i) - y_i)^2 + \lambda\sum_{j=1}^{n}|\Theta_j|$$

La propriété du LASSO et qu'il a la possibilité de fixer les coefficients à 0 contrairement au ridge ou ceux-ci peuvent proche de zéro mais jamais être nul.

# La régression régularisée

- **EslaticNet** = ridge + LASSO

Avantage : On peut faire face au variables corrélées (ridge) et réduire le nombre de variables. La fonction cout est :

$$j(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(\mathrm{h}(x_i) - y_i)^2 + \lambda\sum_{j=1}^{n}\left(\frac{1}{2}(1-\alpha)\Theta_j^2 + \alpha\left|\Theta_j\right|\right)$$

Le paramètre $\alpha$ est compris entre 0 et 1 et permet de définir l'équilibre entre ridge et LASSO.

- LASSO : $\alpha$ = 1
- ridge : $\alpha$ = 0