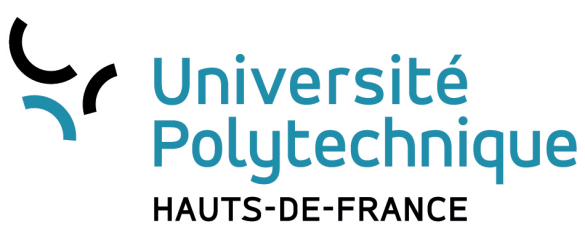


Apprentissage Statistique Automatique I

Régression linéaire

Andrés F. López-Lopera
Université Polytechnique Hauts-de-France (UPHF)

Thèmes

1. Régression linéaire
    - Régression linéaire simple
    - Régression linéaire multiple
2. Décomposition en valeurs singulières
    - Définition
    - SVD et moindres carrés
    - Analyse du biais, de la variance et du risque par la SVD
    - Stabilité numérique
3. Réduction de la dimension
    - Régression sur les composantes principales
    - Régression des moindres carrés partiels

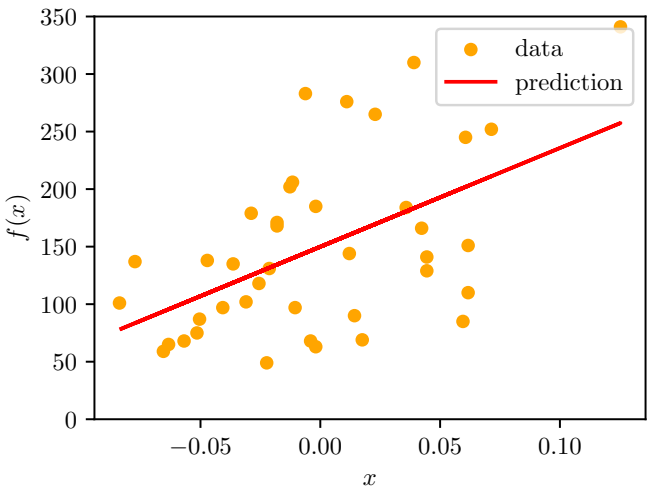
Régression linéaire

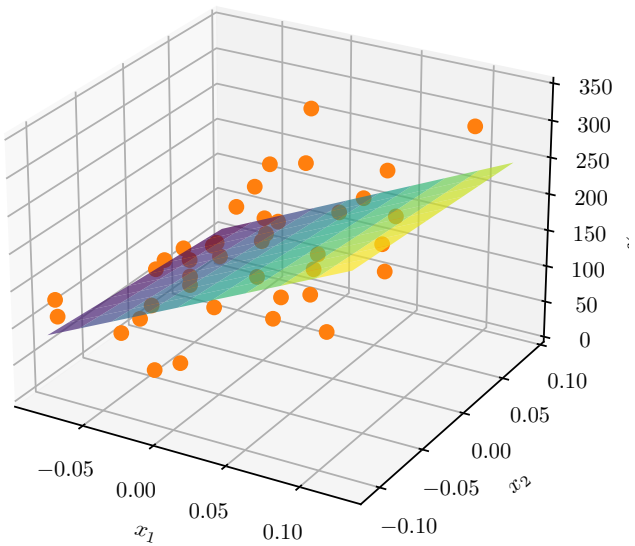
Régression linéaire

· Soit

$$f : \begin{cases} \mathbb{R}^d \to \mathbb{R} \\ x_1, \ldots, x_d \mapsto f(x_1, \ldots, x_d) \end{cases}$$

· La **régression linéaire** est une approche simple pour l'apprentissage supervisé, supposant que y dépend linéairement de $x_1, \ldots, x_d$





· Bien que simple, elle est utile à la fois conceptuellement et en pratique

Régression linéaire

· Dans un cadre d'apprentissage, on cherche à créer une règle de régression dans la classe

$$\mathcal{F}_L := \left\{ f(x) = \beta_0 + \sum_{j=1}^{d} \beta_j x_j, \beta = (\beta_0, \ldots, \beta_d) \in \mathbb{R}^{d+1} \right\}, \tag{1}$$

avec $x = (x_1, \ldots, x_d)^\top \in \mathbb{R}^d$

· On supposera qu'il existe β (inconnu) t.q.

$$y(x) = \beta_0 + \sum_{j=1}^{d} \beta_j x_j + \varepsilon, \tag{2}$$

où ε est un bruit additif centré, *i.e.*, $\mathbb{E}(\varepsilon) = 0$. Par exemple, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$

Régression linéaire simple

· On pose un modèle de la forme

$$y(x) = \beta_0 + \beta_1 x + \varepsilon \tag{3}$$

· Le modèle précédent corresponde bien à l'équation d'une droite avec

- β_0 : ordonnée à l'origine (*intercept*)
- β_1 : pente (*slope*)

· $\beta = (\beta_0, \beta_1)$ sont les **coefficients** du **modèle linéaire**

· Étant estimé $\widehat{\beta} = (\widehat{\beta}_0, \widehat{\beta}_1)$, on peut ensuite prédire y pour une valeur de x

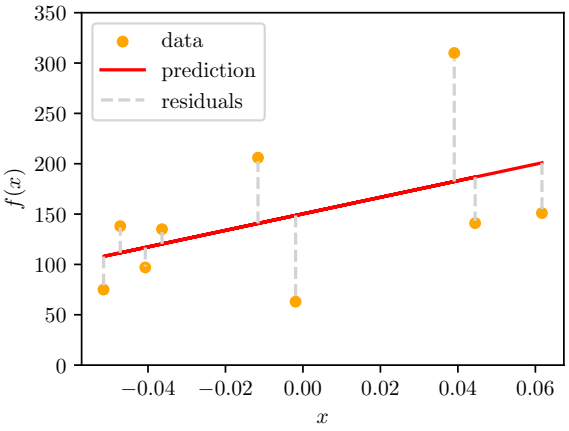
$$\widehat{y}(x) = \widehat{\beta}_0 + \widehat{\beta}_1 x \tag{4}$$

Méthode de moindres carrés

· Supposons une base de données $(x, y) = (x_i, y_i)_{1 \leq i \leq n}$ avec $x_i \in \mathbb{R}$ et $y_i := f(x_i) \in \mathbb{R}$

· Soit $\widehat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1 x_i$ la prédiction de y sur la i-ème valeur de X

· Pour estimer (β_0, β_1), on chercher à minimiser la somme des carrés résiduels (SSE, en anglais)



$$\widehat{\beta} = (\widehat{\beta}_0, \widehat{\beta}_1) = \underset{\beta \in \mathbb{R}^2}{\operatorname{argmin}} \frac{1}{2} \sum_{i=1}^{n} (y_i - \widehat{y}_i)^2, \quad (5)$$

ce qui donne **[exercice]**

$$\widehat{\beta}_0 = \overline{y} - \widehat{\beta}_1 \overline{x},$$

$$\widehat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{n} (x_i - \overline{x})^2} = \frac{\operatorname{cov}(x, y)}{\operatorname{var}(x)}$$

avec $\overline{z} = \frac{1}{n} \sum_{i=1}^{n} z_i$

Solution. Conditions nécessaires d'optimalité (CNO) du premier ordre :

- $\dfrac{\partial \mathrm{SSE}}{\partial \beta_0} = 0$:

$$\frac{\partial \mathrm{SSE}}{\partial \beta_0} = \frac{1}{2}\frac{\partial}{\partial \beta_0}\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i)^2 = -\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i) = 0,$$

d'où on obtient

$$\widehat{\beta}_0 = \frac{1}{n}\left[\sum_{i=1}^{n}(y_i - \beta_1 x_i)\right] = \frac{1}{n}\sum_{i=1}^{n} y_i - \beta_1 \frac{1}{n}\sum_{i=1}^{n} x_i = \overline{y} - \beta_1 \overline{x} \qquad (6)$$

- $\dfrac{\partial \mathrm{SSE}}{\partial \beta_1} = 0$. En sachant (6),

$$\frac{\partial \mathrm{SSE}}{\partial \beta_1} = \frac{1}{2}\frac{\partial}{\partial \beta_1}\sum_{i=1}^{n}(y_i - \widehat{\beta}_0 - \beta_1 x_i)^2 = \sum_{i=1}^{n}(y_i - \overline{y} + \beta_1 \overline{x} - \beta_1 x_i)(\overline{x} - x_i) = 0,$$

d'où on obtient

$$\widehat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{n}(x_i - \overline{x})^2}$$

Critères d'évaluation de la précision du modèle linéaire

Erreur Quadratique Moyenne (*Mean Squared Error*, MSE) :

$$\text{MSE} = \frac{\text{SSE}}{n}, \qquad \text{avec SSE} = \sum_{i=1}^{n} (y_i - \widehat{y}_i)^2 \tag{7}$$

Erreur Quadratique Moyenne Standardisée (*Standardized MSE*, SMSE) :

$$\text{SMSE} = \frac{\text{MSE}}{\text{SST}}, \qquad \text{avec SST} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \bar{y})^2, \tag{8}$$

où SST est la somme des carrés totale

Fraction de Variance Expliquée :

$$R^2 = \frac{\text{SST} - \text{MSE}}{\text{SST}} = 1 - \text{SMSE} \tag{9}$$

Le R^2 est également connu comme le coefficient de corrélation entre Y et X

Rem : Le R^2 devient Q^2 s'il est calculé que sur une base de test

***F*-statistique :**

$$F_{stat} = \left(\frac{n - d - 1}{d} \right) \frac{\text{SST} - \text{SSE}}{\text{SSE}} = \left(\frac{n - d - 1}{d} \right) \frac{R^2}{1 - R^2} \tag{10}$$

sklearn.metrics et sklearn.feature_selection

sklearn.metrics

- SSE : mean_squared_error
- R^2 : r2_score

sklearn.feature_selection

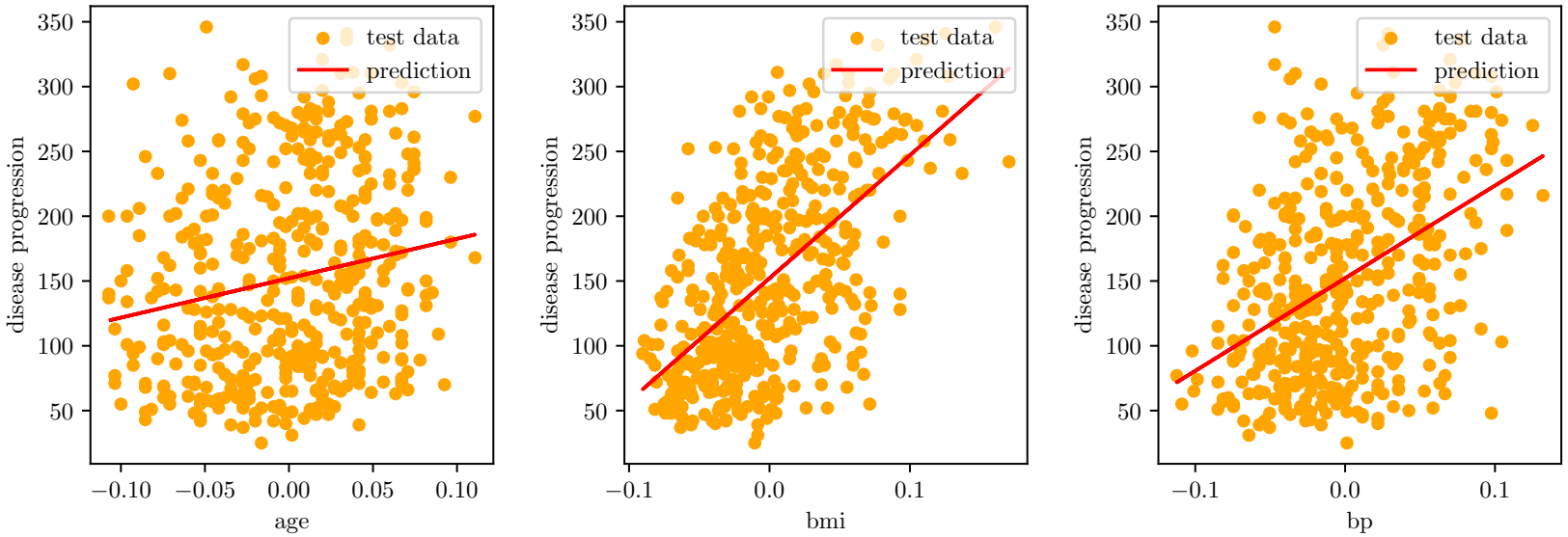
- F_{stat}, p_{val} : f_regression

Plus de détails :

- https://scikit-learn.org/1.5/modules/model_evaluation.html
- https://scikit-learn.org/1.5/modules/feature_selection.html

Cas de test - diabètes (sklearn) [Pedregosa et al., 2011]

Caractéristiques : $n = 442$, $d = 3$



x_i	β_0	β_1	MSE (↓)	SMSE (↓)	R^2 ($\rightarrow 1$)	F_{stat} (↑)	p_{value}
âge	152.13	304.18	5720.55	0.93	0.04	1.58	7.06×10^{-5}
bmi	152.13	949.44	3890.46	0.66	0.34	22.59	3.47×10^{-42}
bp	152.13	714.74	4774.10	0.81	0.19	10.43	1.65×10^{-22}

- bmi : indice de masse corporelle (*body mass index*)
- bp : tension artérielle (*blood pressure*)

Régression linéaire multiple

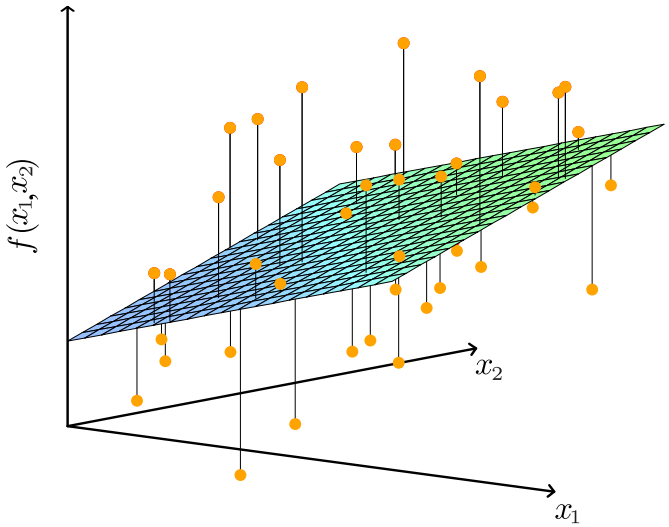
· On pose un modèle de la forme

$$y(x) = \beta_0 + \sum_{j=1}^{d} \beta_j x_j + \varepsilon, \qquad (11)$$

avec $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$

· A partir de $\widehat{\beta} = (\widehat{\beta}_0, \ldots, \widehat{\beta}_d)$, on peut ensuite prédire y avec

$$\widehat{y}(x) = \widehat{\beta}_0 + \sum_{j=1}^{d} \widehat{\beta}_j x_j \qquad (12)$$



· Comme pour $d = 1$, on estime β en minimisant la SSE $= \frac{1}{2} \sum_{i=1}^{n} (y_i - \widehat{y}_i)^2$

· L'estimation dépend d'une inversion de matrice produit qui est possible de calculer en utilisant un logiciel de statistique !

· De même, var$(\widehat{\beta}_j)$, t-test de nullité, test de Fisher, ...

Régression linéaire multiple

· Pour une démonstration rapide, considérons une base de données $(X, y) = (x_i, y_i)_{1 \leq i \leq n}$ avec $x_i = (x_{i,1}, \ldots, x_{i,d}) \in \mathbb{R}^d$ et $y_i \in \mathbb{R}$

· De façon matricielle, on a $y = X\beta$ avec

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad \beta = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_d \end{bmatrix}, \quad \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_{1,1} & \ldots & x_{1,d} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \ldots & x_{n,d} \end{bmatrix}.$$

· Alors, par minimisation du critère des moindres carrés :

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{d} \beta_j x_{i,j} \right)^2 = \|y - X\beta\|^2 = y^\top y - 2\beta^\top X^\top y + \beta^\top X^\top X\beta$$

· Par dérivation matricielle (sous l'hypothèse $X^\top X$ est une matrice de plein rang), on obtient :

$$\widehat{\beta} = (X^\top X)^{-1} X^\top y,$$

dont la solution correspond bien à un minimum car la matrice hessienne $2X^\top X$ est semi définie-positive

Cas de test - diabètes (sklearn)

Caractéristiques : $n = 442$, $d = 3$

Variable	β_0	β_1	β_2	β_3	MSE	SMSE	Q^2
(âge,bmi,bp)	152.13	25.99	788.78	394.13	**3580.33**	**0.60**	**0.40**
âge	152.13	304.18	–	–	5720.55	0.93	0.04
bmi	152.13	–	949.44	–	3890.46	0.66	0.34
bp	152.13	–	–	714.74	4774.10	0.81	0.19

- bmi : indice de masse corporelle (*body mass index*)
- bp : tension artérielle (*blood pressure*)

· Que pourrait-on conclure du tableau ?

Cas de test - diabètes (sklearn)

Caractéristiques : $n = 442$, $d = 3$

Variable	β_0	β_1	β_2	β_3	MSE	SMSE	Q^2
(âge,bmi,bp)	152.13	25.99	788.78	394.13	**3580.33**	**0.60**	**0.40**
âge	152.13	304.18	–	–	5720.55	0.93	0.04
bmi	152.13	–	949.44	–	3890.46	0.66	0.34
bp	152.13	–	–	714.74	4774.10	0.81	0.19

- bmi : indice de masse corporelle (*body mass index*)
- bp : tension artérielle (*blood pressure*)

· Que pourrait-on conclure du tableau ?

- Interprétation de β_j comme l'effet moyen sur y d'un accroissement de x_j d'une unité (lorsque tous les autres prédicteurs sont fixés)
- Impossible de faire des affirmation en terme de **causalité**. Comment doit-il s'interpréter $\beta_1 = 25.99$?

Des questions importantes

1. Y a t-il au moins un des $x_1, \ldots, x_d$ utile pour prédire y ?
2. Sont-ils vraiment tous utiles ?
3. Comment le modèle s'ajuste aux données ?
4. Avec une nouvelle valeur de x_*, quelle réponse doit-on prédire ? Précision de la prédiction ?

Des questions importantes

1. Y a t-il au moins un des $x_1, \ldots, x_d$ utile pour prédire y ?
3. Comment le modèle s'ajuste aux données ?

- On peut utiliser la F-statistique (et/ou les autres critères d'évaluation de la précision)

Variable	MSE ($\downarrow$)	SMSE ($\downarrow$)	Q^2 ($\rightarrow 1$)	F_{stat} ($\uparrow$)
(âge,bmi,bp)	**3580.33**	**0.60**	**0.40**	**28.28**
âge	5720.55	0.93	0.04	1.58
bmi	3890.46	0.66	0.34	22.59
bp	4774.10	0.81	0.19	10.43

Cas de test - diabètes (sklearn)

Des questions importantes

2. Sont-ils vraiment tous utiles ?

Choix de co-variables

I. Approche complète : Comparer les modèles linéaires avec tous les sous-ensembles possibles de co-variables (Souvent 2^p trop grand)
II. Approche séquentielle que ne parcourt que certains sous-ensembles.

Algorithm Sélection progressive

Données d'entré : (X, y), tol $\epsilon > 0$

$D^{(0)} = \{1, \ldots, d\}$, $X_* = ()$
for $i = 1, \ldots, d$ **do**
  for $j \in D_{i-1}$ **do**
    $e_j^2 = \mathrm{SSE}(\mathcal{F}_L(X_* \cup_c X_{:,j}), y)$
  $j_* = \mathrm{argmin}_{j \in D^{(i-1)}} e_j^2$
  $D^{(i)} = D^{(i-1)} \setminus j_*$
  $X_* = X_* \cup_c X_{:,j_*}$
  if critère $\leq \epsilon$ **then**
    End

Algorithm Sélection rétrograde

Données d'entré : (X, y), tol $\epsilon > 0$

$D^{(0)} = \{1, \ldots, d\}$
for $i = 1, \ldots, d$ **do**
  for $j \in D_{i-1}$ **do**
    $D_{-j} = D^{(i-1)} \setminus j$
    $p_j = p\text{-val}(\mathcal{F}_L(X_{:,D_{-j}}), y)$
  $j_* = \mathrm{argmax}_{j \in D^{(i-1)}} p_j$
  $D^{(i)} = D^{(i-1)} \setminus j_*$
  $X = X_{:,D^{(i)}}$
  if critère $\leq \epsilon$ **then**
    End

Des questions importantes

4. Avec une nouvelle valeur de x_*, quelle réponse doit-on prédire ? Précision de la prédiction ?

- Prédictions :

$$\widehat{y}(x) = \widehat{\beta}_0 + \sum_{j=1}^{d} \widehat{\beta}_j x_j,$$

  pour tout $x \in \mathbb{R}^d$!

- Précision de la prédiction : MSE, SMSE, Q^2, ...

Extensions

Variables qualitatives. On crée des nouvelles variables binaires (autant que nécessaire pour représenter les modalités) :

$$x_j = \begin{cases} 1 & \text{si la condition est satisfaite,} \\ 0 & \text{sinon} \end{cases}$$

Interactions de variables. On crée des nouvelles variables données par des produits de variables d'intérêt :

$$y(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 \underbrace{(x_1 x_2)}_{x_3}$$

Effets non linéaires. On crée des nouvelles variables données par la puissance de la variable d'intérêt :

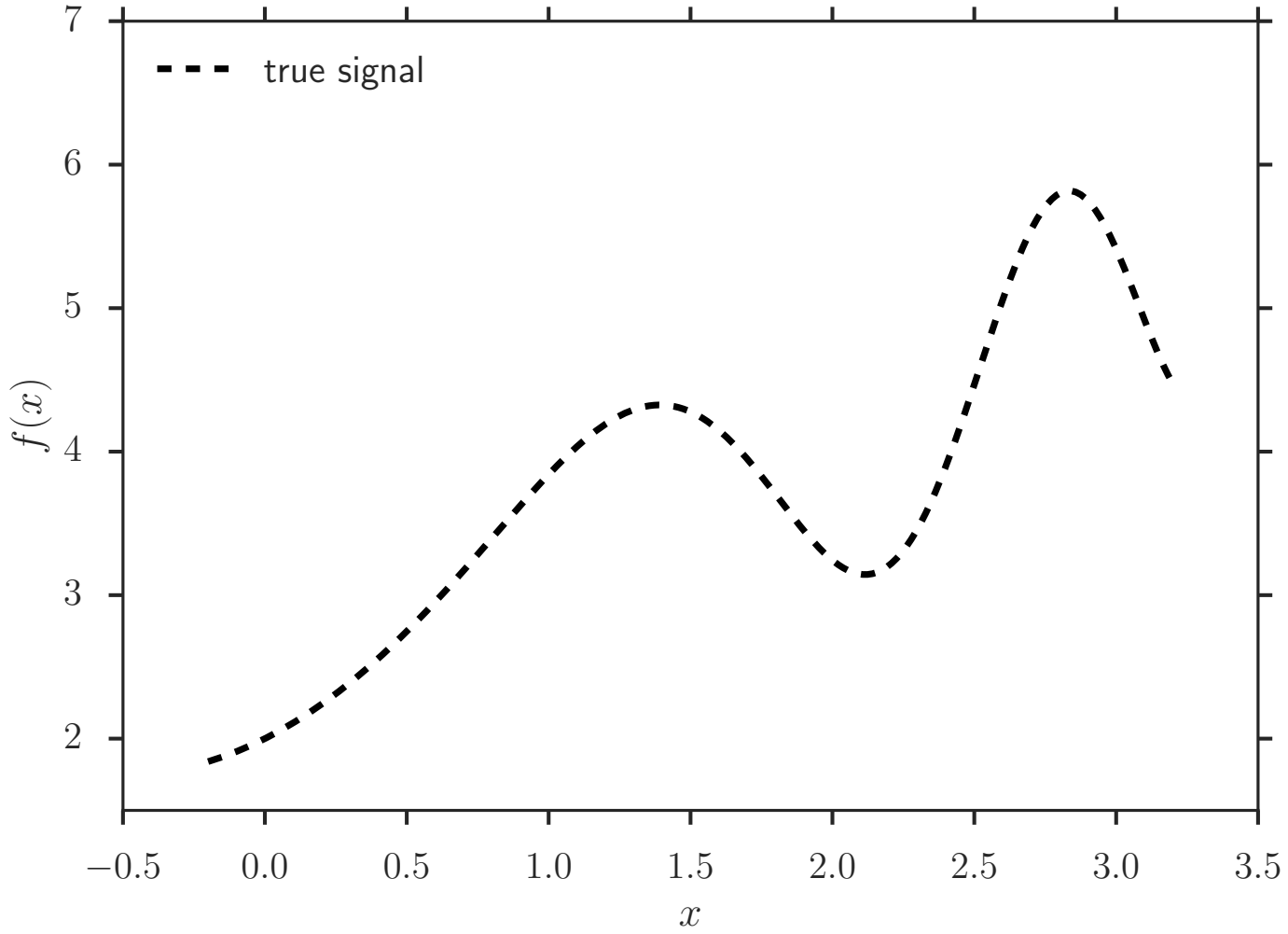
$$y(x) = \beta_0 + \sum_{j=1}^{p} \beta_j x^p + \varepsilon \quad \text{(régression polynomiale d'ordre } p\text{)}$$

En général, on peut envisager une projection sur un ensemble de fonctions de base $\phi_1, \ldots, \phi_p : \mathbb{R}^d \to \mathbb{R}$

$$y(x) = \beta_0 + \sum_{j=1}^{p} \beta_j \phi_j(x) + \varepsilon$$

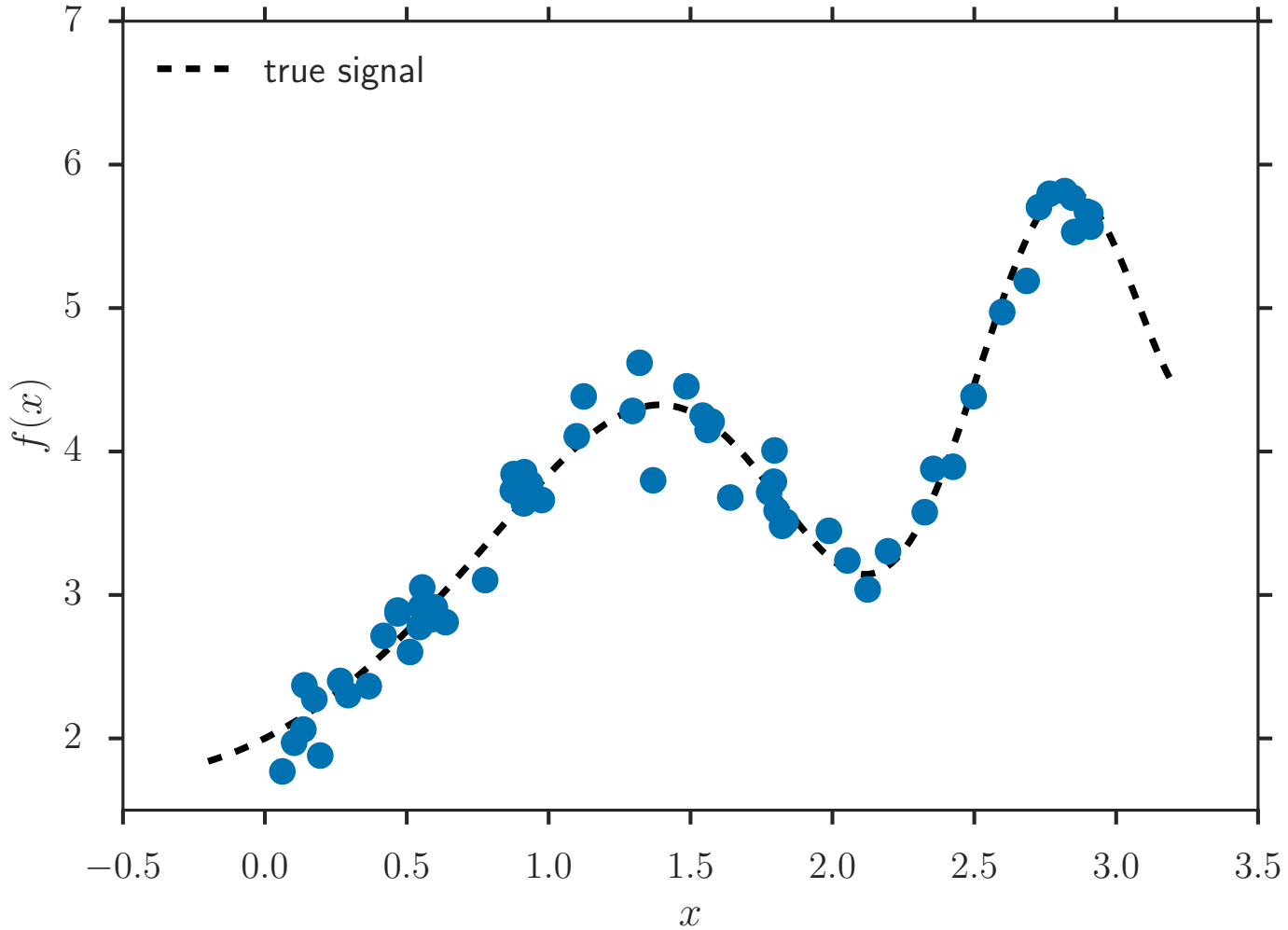
Effets non linéaires

Fonction cible : $f(x_i)$ for $i = 1, \ldots, n$

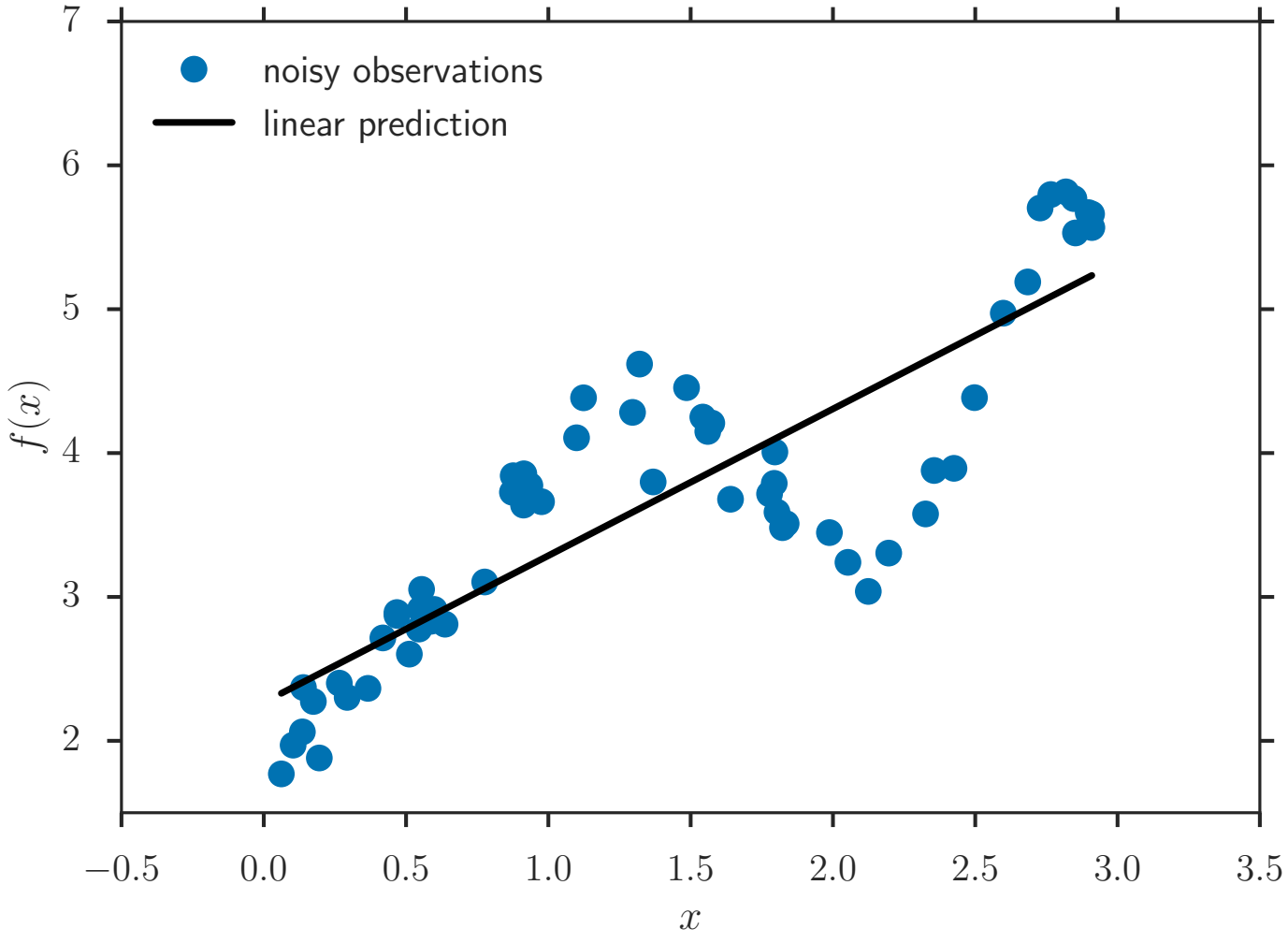


Effets non linéaires

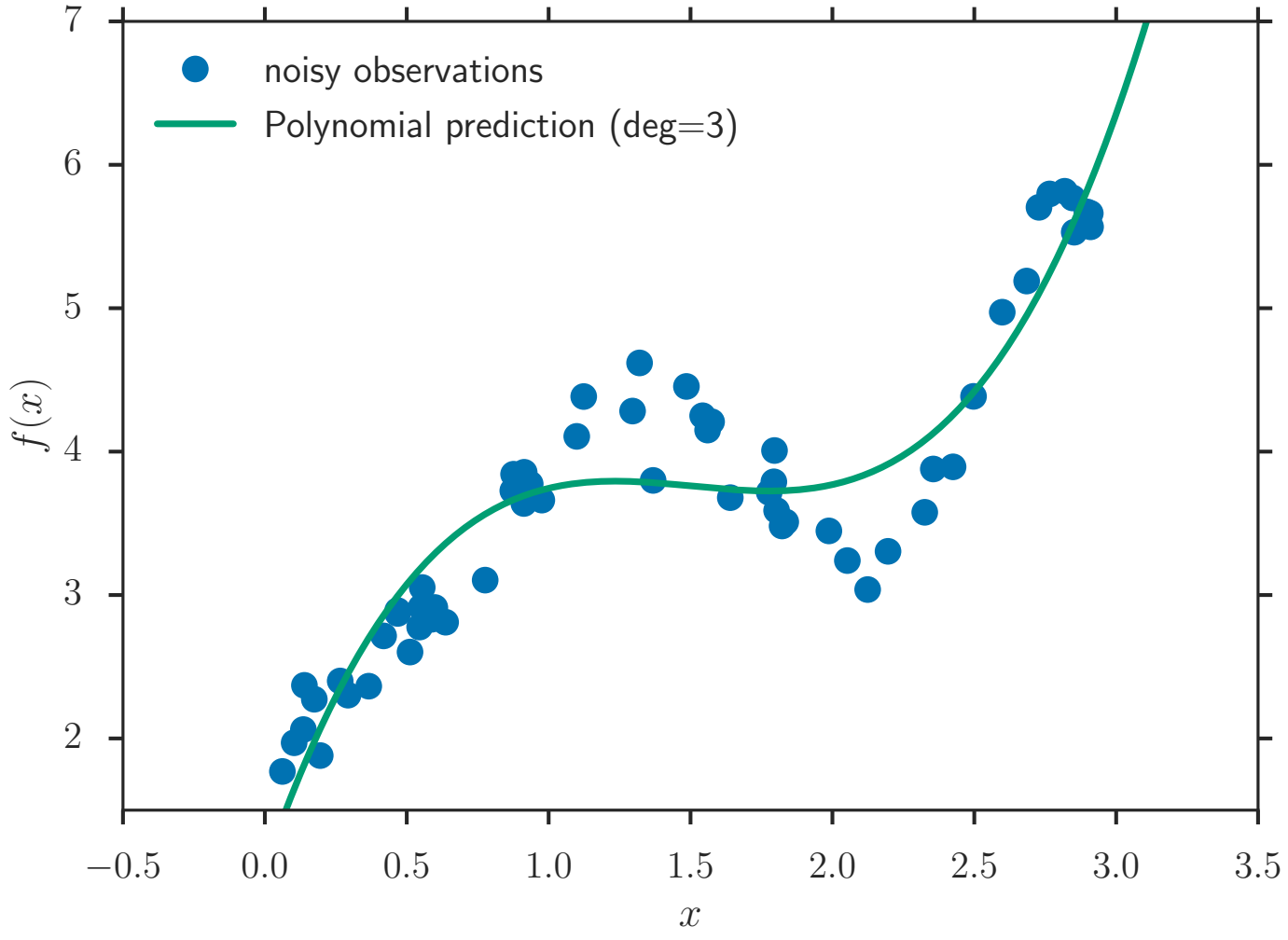
Observations bruitées : $y_i = f(x_i) + \varepsilon_i$ pour tout $i = 1, \ldots, n$



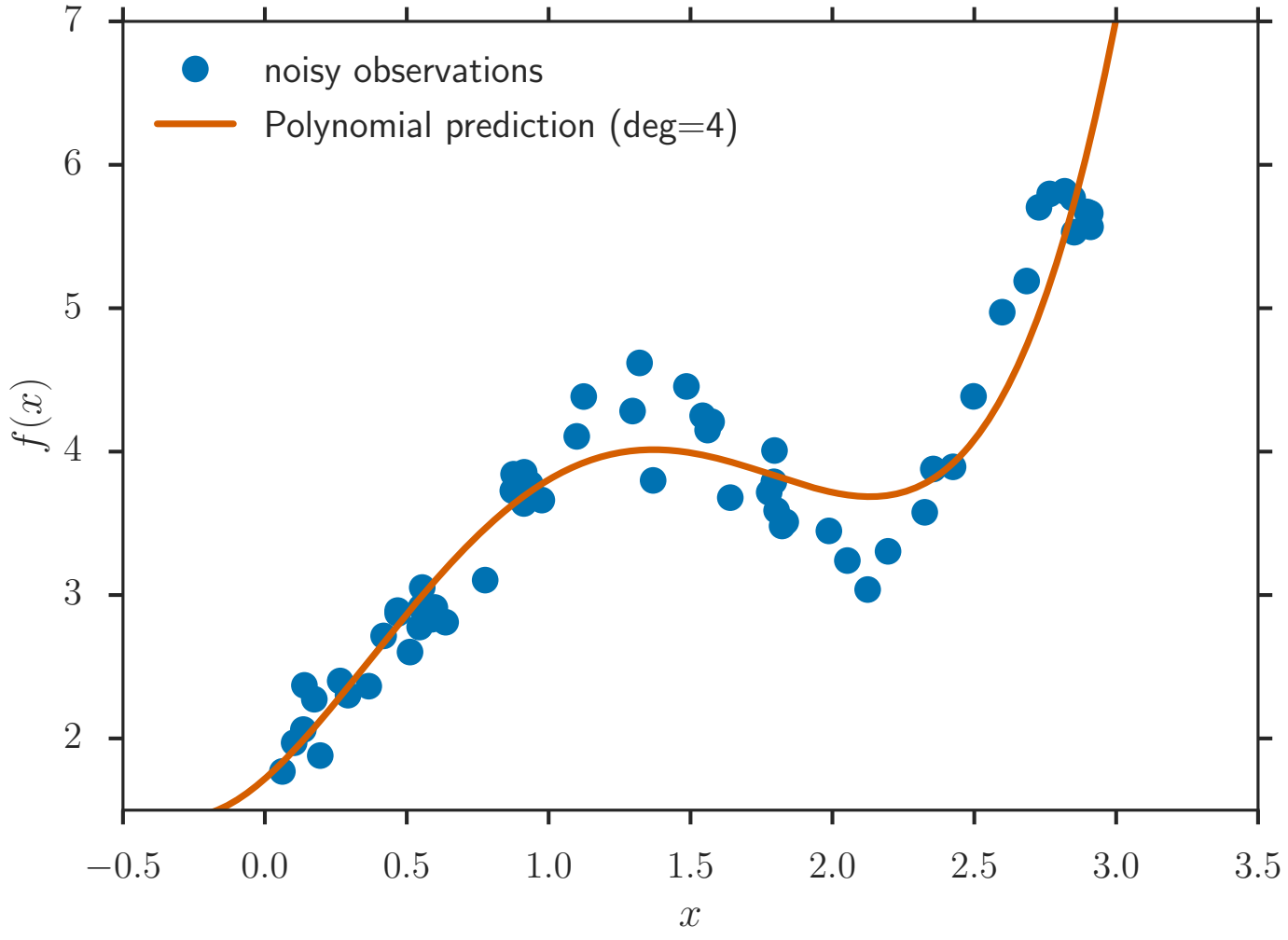
Régression linéaire



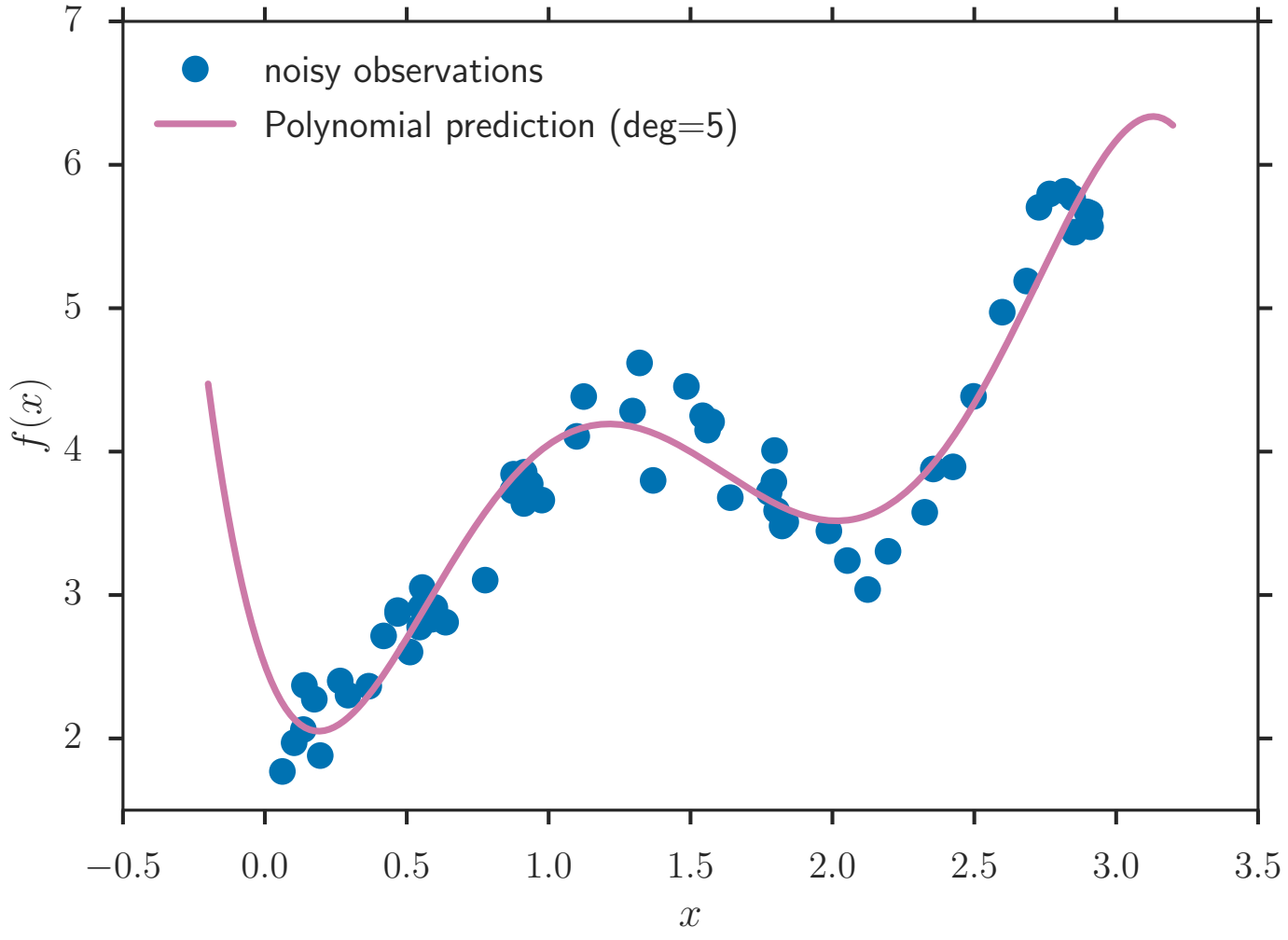
Régression polynomiale



Régression polynomiale

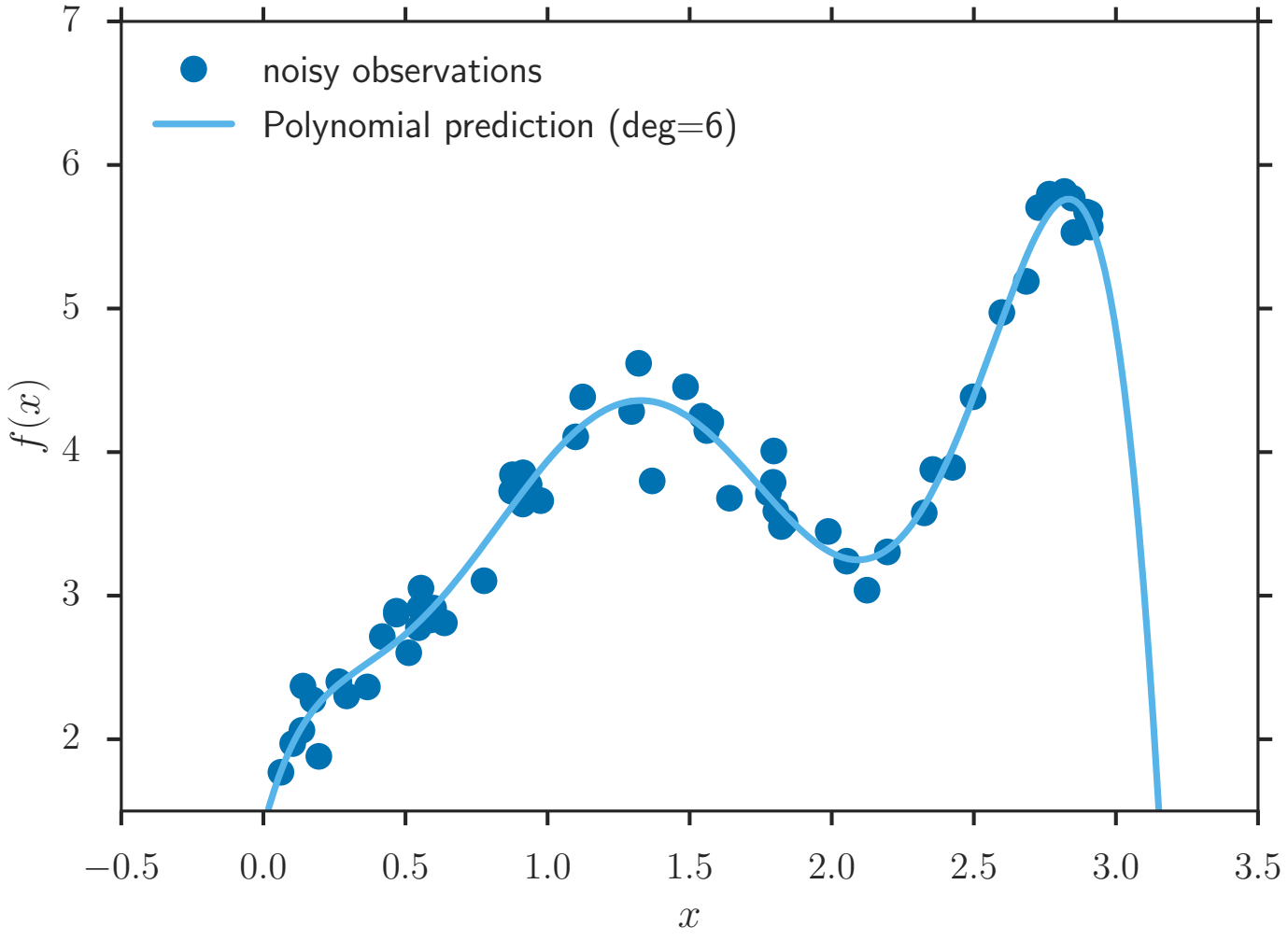


Régression polynomiale

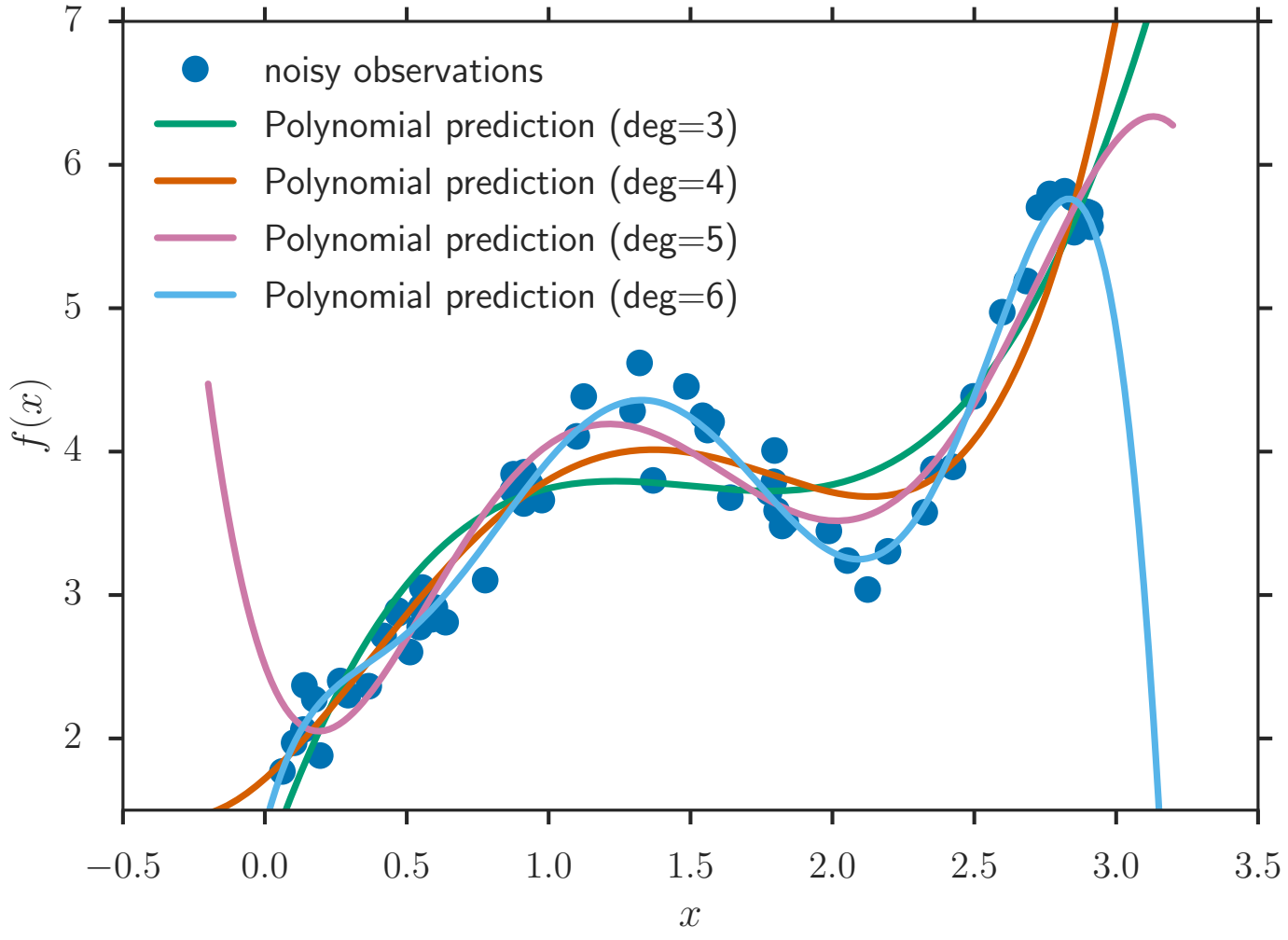


Effets non linéaires

Régression polynomiale



Régression polynomiale



Extensions

- **Ajustement régularisé :** régression Ridge, Lasso, Elastic Net, ...
- **Problèmes de classification :** régression logistique, support vector machine
- **Non-linéarité :** lissage à noyau, splines, modèles additifs généralisés, méthode de plus proches voisins
- **Interactions :** méthode basée sur des arbres, bagging, forêts aléatoires (random forests), et boosting (capturent aussi les non-linéarités)

Décomposition en valeurs singulières

Décomposition spectrale

Theorem (Théorème spectral [Golub and Van Loan, 2013])

Une matrice symétrique $S \in \mathbb{R}^{n\times n}$ est diagonalisable en base orthonormée, i.e., *il existe $\lambda_1 \geq \cdots \geq \lambda_n$ et une matrice orthogonale $U \in \mathbb{R}^{n\times n}$ t.q. :*

$$S = U\Sigma U^\top \quad \Leftrightarrow \quad SU = U\Sigma,$$

avec $\Sigma = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ et $\lambda_1 \geq \cdots \geq \lambda_n > 0$

· Si l'on écrit $U = [u_1, \ldots, u_n]$ cela signifie que :

$$S = \sum_{i=1}^{n} \lambda_i u_i u_i^\top, \quad \text{avec} \quad Su_i = \lambda_i u_i \text{ pour tout } i \in [\![1, n]\!]$$

Rappel : une matrice orthogonale $U \in \mathbb{R}^{n\times n}$ doit vérifier :

$$U^\top U = UU^\top = I \quad \Leftrightarrow \quad u_i^\top u_j = \langle u_i, u_j \rangle = \delta_{i,j}$$

· Les $\lambda_i \in \mathbb{R}^+$ sont les **valeurs propres** de S et les $u_i \in \mathbb{R}^n$ sont les **vecteurs propres** associés

Décomposition en valeurs singulières (*singular value decomposition, SVD*)

Theorem (Décomposition SVD [Golub and Van Loan, 2013])

Pour toute matrice $X \in \mathbb{R}^{n \times p}$*, il existe deux matrices orthogonales* $U = [u_1, \ldots, u_n] \in \mathbb{R}^{n \times n}$ *et* $V = [v_1, \ldots, v_p] \in \mathbb{R}^{p \times p}$*, t.q. :*

$$X = U \Sigma V^\top,$$

avec $\Sigma = \mathrm{diag}(s_1, \ldots, s_r)$*,* $s_1 \geq \cdots \geq s_r > 0$*, et* $r = \mathrm{rang}(X)$*.*

Démonstration. Diagonaliser $X^\top X$ [Golub and Van Loan, 2013]

· Les $s_j \in \mathbb{R}^+$ sont les **valeurs singulières** de X, et les u_j (resp. v_j) sont les **vecteurs singuliers** à gauche (resp. à droite)

Décomposition en valeurs singulières (*singular value decomposition, SVD*)

· Si $n < p$:

$$X = U\Sigma V^\top = [u_1, \ldots, u_n] \left[\begin{array}{ccc|c} s_1 & \cdots & 0 & \\ \vdots & \ddots & \vdots & 0_{n\times(p-n)} \\ 0 & \cdots & s_n & \end{array}\right] \begin{bmatrix} v_1^\top \\ \vdots \\ v_p^\top \end{bmatrix} = \sum_{i=1}^{n} s_i u_i v_i^\top$$

· Si $p < n$:

$$X = U\Sigma V^\top = [u_1, \ldots, u_n] \begin{bmatrix} s_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & s_p \\ \hline & 0_{(n-p)\times p} & \end{bmatrix} \begin{bmatrix} v_1^\top \\ \vdots \\ v_p^\top \end{bmatrix} = \sum_{i=1}^{p} s_i u_i v_i^\top$$

SVD réduite

· Si $n < p$:

$$X = U\Sigma V^\top = [u_1, \ldots, u_n] \begin{bmatrix} s_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & s_n \end{bmatrix} \begin{bmatrix} v_1^\top \\ \vdots \\ v_n^\top \end{bmatrix} = \sum_{i=1}^{n} s_i u_i v_i^\top$$

· Si $p < n$:

$$X = U\Sigma V^\top = [u_1, \ldots, u_p] \begin{bmatrix} s_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & s_p \end{bmatrix} \begin{bmatrix} v_1^\top \\ \vdots \\ v_p^\top \end{bmatrix} = \sum_{i=1}^{p} s_i u_i v_i^\top$$

SVD réduite

SVD réduite : on ne garde que les éléments utiles avec $r = \min(n, p)$:

$$X = U\Sigma V^\top = \sum_{i=1}^{r} s_i u_i v_i^\top = U_r \Sigma_r V_r^\top, \tag{13}$$

avec $s_i > 0$ pour tout $i \in [\![1, r]\!]$, et $U_r = [u_1, \ldots, u_r], V_r = [v_1, \ldots, v_r]$

SVD compacte : on ne garde que les $r = \text{rang}(X)$ valeurs singulières non-nulles

Rem : les u_i (resp. les $v_i^\top$) sont orthonormés et engendrent le même espace que celui engendré par les colonnes (resp. les lignes) de X

$$\text{vect}(x_1, \ldots, x_p) = \text{vect}(u_1, \ldots, u_r)$$

Propriété variationnelle de la plus grande valeur singulière

Rappel :

$$X = U\Sigma V^\top \quad \Rightarrow \quad \Sigma = U^\top X V$$

· La première valeur singulière s_1 s'obtient du problème d'optimisation :

$$s_1 = \begin{cases} \max\limits_{u\in\mathbb{R}^n, v\in\mathbb{R}^p} & u^\top X v \\ \text{s.c.} & \|u\|^2 = 1, \\ & \|v\|^2 = 1 \end{cases}$$

· Lagrangien :

$$\mathcal{L}(u,v) = u^\top X v - \lambda_1(\|u\|^2 - 1) - \lambda_2(\|v\|^2 - 1)$$

· Conditions nécessaires d'optimalité (CNO) :

$$\begin{cases} \nabla_u \mathcal{L} = Xv - 2\lambda_1 u = 0 \\ \nabla_v \mathcal{L} = X^\top u - 2\lambda_2 v = 0 \end{cases} \iff \begin{cases} Xv = 2\lambda_1 u \\ X^\top u = 2\lambda_2 v \end{cases} \Rightarrow \begin{cases} X^\top X v = \alpha v \\ X X^\top u = \alpha u \end{cases}$$

avec $\alpha = 4\lambda_1\lambda_2$, et donc v et u sont des vecteurs propres de $X^\top X$ et de $XX^\top$

Pseudo-inverse

· Considérons que $X \in \mathbb{R}^{n \times p}$ admet pour SVD $X = \sum_{i=1}^{r} s_i u_i v_i^\top$ avec $r = \text{rang}(X)$

· La **pseudo-inverse** de X, dénotée par $X^+ \in \mathbb{R}^{p \times n}$, est définie par :

$$\begin{aligned} X^+ := (U\Sigma V^\top)^{-1} &= (V^\top)^{-1}\Sigma^{-1}U^{-1} \\ &= V \ \text{diag}(1/s_1, \ldots, 1/s_r) \ U^\top \\ &= \sum_{i=1}^{r} \frac{1}{s_i} v_i u_i^\top \end{aligned}$$

· Si $X = \sum_{i=1}^{n} s_i u_i v_i^\top \in \mathbb{R}^{n \times n}$ est inversible alors $X^+ = X^{-1}$

SVD et numérique

· Les fonctions SVD et pseudo-inverse sont disponibles dans les librairies numériques classiques, par exemple Numpy

· **SVD :** `U, s, V = np.linalg.svd(X)`

  • $\Sigma = \text{diag}(s)$ (`X = U.dot(np.diag(s).dot(V))`)

  • Variantes compactes ou non par l'option : `full_matrices=True/False`

· **Pseudo-inverse :** `np.linalg.pinv(X)`

Retour sur les moindres carrés

- Partons de la SVD $X = U\Sigma V^\top = \sum_{i=1}^{r} s_i u_i v_i^\top$:

$$\|X\beta - y\|^2 = \left\|U\Sigma V^\top \beta - y\right\|^2$$

Retour sur les moindres carrés

· Partons de la SVD $X = U\Sigma V^\top = \sum_{i=1}^{r} s_i u_i v_i^\top$:

$$
\begin{aligned}
\|X\beta - y\|^2 = \left\|U\Sigma V^\top \beta - y\right\|^2 &= \left\|U\Sigma V^\top - UU^\top y\right\|^2 \\
&= \left\|\sum_{i=1}^{r} s_i u_i v_i^\top \beta - \sum_{i=1}^{n} u_i u_i^\top y\right\|^2 \\
&= \left\|\sum_{i=1}^{r} u_i (s_i v_i^\top \beta - u_i^\top y) - \sum_{i=r+1}^{n} u_i u_i^\top y\right\|^2 \\
&= \left\|\sum_{i=1}^{r} u_i (s_i v_i^\top \beta - u_i^\top y)\right\|^2 + \left\|\sum_{i=r+1}^{n} u_i u_i^\top y\right\|^2 \\
&= \sum_{i=1}^{r} \left(s_i v_i^\top \beta - u_i^\top y\right)^2 + \sum_{i=r+1}^{n} (u_i^\top y)^2
\end{aligned}
$$

Retour sur les moindres carrés (suite)

$$\|X\beta - y\|^2 = \sum_{i=1}^{r} (s_i v_i^\top \beta - u_i^\top y)^2 + \sum_{i=r+1}^{n} (u_i^\top y)^2 \geq \sum_{i=r+1}^{n} (u_i^\top y)^2$$

· L'égalité s'obtient si

$$\beta = \sum_{i=1}^{r} \frac{\langle y, u_i \rangle}{s_i} v_i = \sum_{i=1}^{r} \frac{v_i u_i^\top}{s_i} y,$$

or $X^+ = \sum_{i=1}^{r} \frac{1}{s_i} v_i u_i^\top$

· Ainsi une solution des moindres carrés peut s'écrire:

$$\widehat{\beta} = X^+ y \in \operatorname*{argmin}_{\beta \in \mathbb{R}^p} \frac{1}{2} \|X\beta - y\|^2$$

Rem : l'ensemble de toutes les solutions est:

$$\left\{ X^+ y + \sum_{i=r+1}^{p} \alpha_i v_i, (\alpha_{r+1}, \ldots, \alpha_p) \in \mathbb{R}^{p-r} \right\},$$

où $X^+ y$ est la solution de norme $\|\cdot\|$ minimale

Analyse du biais par la SVD

· Sous l'hypothèse de bruit "blanc" (*i.e.*, $\mathbb{E}(\varepsilon) = 0$):

$$\mathbb{E}(\widehat{\beta}) = \mathbb{E}(X^+ y) = \sum_{i=1}^{r} v_i v_i^\top \beta^\star = \Pi_l \beta^\star$$

- Π_l : projecteur sur l'espace des lignes de X

$$\Pi_l = \sum_{i=1}^{r} v_i v_i^\top = X^+ X$$

- Π_c : projecteur sur l'espace des colonnes de X

$$\Pi_c = \sum_{i=1}^{r} u_i u_i^\top = X X^+$$

Rem : si $r = \text{rang}(X) = n$ on retrouve que les conditions d'optimalité sont sans biais : $\mathbb{E}(\widehat{\beta}) = \beta^\star$

Analyse du biais par la SVD

· Sous l'hypothèse de bruit "blanc" (*i.e.*, $\mathbb{E}(\varepsilon) = 0$):

$$\mathbb{E}(\widehat{\beta}) = \mathbb{E}(X^+ y) = \sum_{i=1}^{r} v_i v_i^\top \beta^\star = \Pi_l \beta^\star$$

- Π_l : projecteur sur l'espace des lignes de X

$$\Pi_l = \sum_{i=1}^{r} v_i v_i^\top = X^+ X$$

- Π_c : projecteur sur l'espace des colonnes de X

$$\Pi_c = \sum_{i=1}^{r} u_i u_i^\top = X X^+$$

Rem : si $r = \text{rang}(X) = n$ on retrouve que les conditions d'optimalité sont sans biais : $\mathbb{E}(\widehat{\beta}) = \beta^\star$

Démonstration.

$$\mathbb{E}(\widehat{\beta}) = \mathbb{E}(X^+ [X\beta^\star + \varepsilon]) = \sum_{i=1}^{r} \frac{1}{s_i} v_i u_i^\top \sum_{j=1}^{r} s_j u_j v_j^\top \beta^\star = \sum_{i=1}^{r} v_i v_i^\top \beta^\star = \Pi_l \beta^\star$$

Analyse du biais, de la variance et du risque par la SVD

· Hypothèse : bruit blanc et homoscédastique (*i.e.*, $\mathbb{E}(\varepsilon) = 0$, $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 I_n$)

$$\mathrm{cov}(\widehat{\beta}) = \mathrm{cov}(X^+ y) = \sum_{i=1}^{r} \frac{\sigma^2}{s_i^2} v_i v_i^\top$$

Rem : si $\mathrm{rang}(X) = n$, on retrouve $\mathrm{cov}(\widehat{\beta}) = \sigma^2 (X^\top X)^{-1}$

Analyse du biais, de la variance et du risque par la SVD

· Hypothèse : bruit blanc et homoscédastique (*i.e.*, $\mathbb{E}(\varepsilon) = 0$, $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 I_n$)

$$\mathrm{cov}(\widehat{\beta}) = \mathrm{cov}(X^+ y) = \sum_{i=1}^{r} \frac{\sigma^2}{s_i^2} v_i v_i^\top$$

Rem : si $\mathrm{rang}(X) = n$, on retrouve $\mathrm{cov}(\widehat{\beta}) = \sigma^2 (X^\top X)^{-1}$

Démonstration.

$$\mathrm{cov}(\widehat{\beta}) = \mathrm{cov}(X^+ [X\beta^\star + \varepsilon]) = \mathbb{E}(X^+ \varepsilon\varepsilon^\top (X^+)^\top) = X^+ \mathbb{E}(\varepsilon\varepsilon^\top)(X^+)^\top = \sum_{i=1}^{r} \frac{\sigma^2}{s_i^2} v_i v_i^\top$$

Analyse du risque de prédiction par la SVD

· Hypothèse : modèle homoscédastique (*i.e.*, $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 I_n$) et que X est de plein rang

$$R_{pred}(y, \widehat{y}) = \mathbb{E}\|X\beta^\star - X\widehat{\beta}\|_2^2 = \sigma^2 \,\mathrm{rang}(X)$$

Analyse du risque de prédiction par la SVD

· Hypothèse : modèle homoscédastique (*i.e.*, $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 I_n$) et que X est de plein rang

$$R_{pred}(y, \widehat{y}) = \mathbb{E}\|X\beta^\star - X\widehat{\beta}\|_2^2 = \sigma^2 \operatorname{rang}(X)$$

Démonstration.

$$\begin{aligned}
R_{pred}(y, \widehat{y}) &= \mathbb{E}\left((\widehat{\beta} - \beta^\star)^\top X^\top X\,(\widehat{\beta} - \beta^\star)\right) \\
&= \mathbb{E}\left((X^+[X\beta^\star + \varepsilon] - \beta^\star)^\top X^\top X\,(X^+[X\beta^\star + \varepsilon] - \beta^\star)\right) \\
&= \mathbb{E}\left((X^+\varepsilon + [\Pi_l - I_p]\beta^\star)^\top X^\top X\,(X^+\varepsilon + [\Pi_l - I_p]\beta^\star)\right) \\
&= \mathbb{E}\left((X^+\varepsilon)^\top X^\top X\,(X^+\varepsilon)\right) + \underbrace{(\beta^\star)^\top(\Pi_l - I_p)^\top X^\top X\,(\Pi_l - I_p)\beta^\star}_{=0} \\
&= \mathbb{E}\left((X\,X^+\varepsilon)^\top X\,X^+\varepsilon\right) \\
&= \mathbb{E}\left(\operatorname{tr}(X\,X^+\varepsilon\,(X\,X^+\varepsilon)^\top)\right) \qquad (a^\top b = \operatorname{tr}(ba^\top)) \\
&= \operatorname{tr}(\Pi_c \mathbb{E}\left(\varepsilon\varepsilon^\top\right)\Pi_c^\top) \\
&= \sigma^2 \operatorname{tr}(\Pi_c\Pi_c^\top) = \sigma^2 \operatorname{rang}(\Pi_c) = \sigma^2 r = \sigma^2 \operatorname{rang}(X)
\end{aligned}$$

Quelques mots de stabilité numérique

· Prenons $\widehat{\beta} = X^{+}y$ comme solution des moindres carrés.

· Supposons qu'on observe maintenant $y + \Delta$ où $\|\Delta\| \ll \|y\|$

· Alors l'estimateur des moindres carrés pour $y + \Delta$ par X donne

$$\widehat{\beta}^{\Delta} = X^{+}(y + \Delta)$$

$$\widehat{\beta}^{\Delta} = \widehat{\beta} + X^{+}\Delta$$

$$\widehat{\beta}^{\Delta} = \widehat{\beta} + \sum_{i=1}^{r} \frac{1}{s_i} v_i u_i^{\top} \Delta$$

Rem : Noter l'influence des "petites" valeurs singulières

Prochains cours

· Remèdes possibles contre les mauvais “conditionnements”

- Régulariser le spectre / les valeurs singulières
- Contraindre les coefficients de $\widehat{\beta}$ à n’être pas trop grands

· Une solution rendant ces deux points de vue équivalents : *Ridge Regression* / Régularisation de Tychonoff

Réduction de la dimension

Rappel sur l'analyse en composante principale (ACP)

· Soit $X = [x_1^\top, \ldots, x_n^\top]^\top \in \mathbb{R}^{n \times p}$ la matrice des variables explicatives

· L'ACP (de niveau $k \in \mathbb{N}$) consiste à effectuer la SVD de X, et à ne garder que les k axes principaux pour représenter le nuage

$$X = \sum_{i=1}^{k} s_i u_i v_i^\top$$

· On appelle **axes principaux** (ou **axes factoriels**) les vecteurs $v_1, \ldots, v_k \in \mathbb{R}^p$. En général $k \ll p$ (e.g., $k = 2$, pour une visualisation planaire)

· Les nouvelles variables $c_i = Xv_i$ sont appelées **composantes**

Rem : On doit recentrer les points pour qu'ils aient une moyenne nulle $X \leftarrow [(x_1 - \overline{x}_n)^\top, \ldots, (x_n - \overline{x}_n)^\top]^\top = X - 1_n \overline{x}_n^\top$ (on peut aussi mettre à l'échelle pour avoir un écart-type similaire par caractéristique (*feature*))

Régression sur les composantes principales

· On applique d'abord une ACP sur X afin d'obtenir les nouvelles variables

$$z_i = Xv_i,$$

pour tout $i \in \{1, \ldots, k\}$

· Ensuite, on considère le modèle linéaire sur $([z_1, \ldots, z_k] \in \mathbb{R}^{n \times k}, y \in \mathbb{R}^n)$:

$$y = \bar{y}1_n + \sum_{i=1}^{k} \theta_i z_i$$

· L'estimateur $\widehat{\theta} \in \mathbb{R}^{k+1}$ s'obtient comme suit **[exercice]**

$$\widehat{\theta}_i = \frac{\langle z_i, y \rangle}{\langle z_i, z_i \rangle}$$

· Car $z_1, \ldots, z_k$ s'obtiennent par des combinaisons linéaires de X, alors

$$\widehat{\beta}_k = \sum_{i=1}^{k} \widehat{\theta}_i v_i$$

Rem : Si $k = p$, on revient sur l'estimation des moindres carrés

Régression des moindres carrés partiels (*Partial Least Squares, PLS*)

· PLS est une méthode itérative permettant la réduction de la dimension

Algorithm Partial Least Squares [Hastie et al., 2009]

Données d'entré : (X, y), nombre d'axes PLS $k \ll p$

1: Recentrer et standardiser X

2: Définir $\widehat{y}^{(0)} = \bar{y}1_n$ et $x_j^{(0)} = x_j$ pour tout $j \in \{1, \dots, p\}$

3: **for** $i = 1, \dots, k$ **do**

4: $z_i = \sum_{j=1}^{p} \widehat{\phi}_{i,j} x_j^{(i-1)}$ où $\widehat{\phi}_{i,j} = \langle x_j^{(i-1)}, y \rangle$

5: $\widehat{y}^{(i)} = \widehat{y}^{(i-1)} + \widehat{\theta}_i z_i$ avec $\widehat{\theta}_i = \dfrac{\langle z_i, y \rangle}{\langle z_i, z_i \rangle}$

6: Orthogonaliser chaque $x_j^{(i-1)}$ par rapport à z_i :

$$x_j^{(i)} = x_j^{(i-1)} - \frac{\langle z_i, x_j^{(i-1)} \rangle}{\langle z_i, z_i \rangle} z_i$$

7: Car $z_1, \dots, z_k$ sont linéaire para rapport à x_j, alors

$$\widehat{y} = \widehat{y}^{(p)} = \bar{y}1_n + \sum_{i=1}^{k} \widehat{\theta}_i z_i$$

· Si $k = p$, on trouve une solution équivalente à celle des moindres carrés

Régression des moindres carrés partiels (*Partial Least Squares, PLS*)

PCA

$$\begin{aligned} \max_{\alpha \in \mathbb{R}^p} \quad & \text{var}(X\alpha) \\ \text{s.c.} \quad & \|\alpha\|^2 = 1, \\ & \alpha^\top S v_\ell = 0, \ \forall\, \ell = 1, \ldots, i-1 \end{aligned}$$

PLS

$$\begin{aligned} \max_{\alpha \in \mathbb{R}^p} \quad & \text{corr}^2(y, X\alpha)\, \text{var}(X\alpha) \\ \text{s.c.} \quad & \|\alpha\|^2 = 1, \\ & \alpha^\top S v_\ell = 0, \ \forall\, \ell = 1, \ldots, i-1 \end{aligned}$$

· S est la matrice de covariance empirique de x_j

· Les conditions $\alpha^\top S v_\ell = 0$ rassurent que $z_i = X\alpha$ n'est pas corrélée avec les combinaisons linéaires précédentes $z_\ell = X v_\ell$

Références

G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, 4 edition, 2013.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer, New York, second edition, 2009.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.