Semaine 4 : Apprentissage non-supervisé et réduction de dimension

3 : Ouvertures en réduction de dimension

Laurent Risser
Ingénieur de Recherche CNRS

1 : Intro

Nous avons vu avec l'ACP comment réduire la dimension d'un problème d'apprentissage de manière ***non supervisée***, c'est-à-dire en ne tenant compte que des propriétés des variables d'entrée $X \in \mathbb{R}^{n*p}$.

Afin de diminuer la dimension de $X \in \mathbb{R}^{n*p}$ vers $X_d \in \mathbb{R}^{n*d}$ avec $d < p$, il existe aussi une multitude de méthodes ***supervisées***

→ Réduction de la dimension de X en tenant compte du fait que X est lié à des sorties Y telles que $Y \in \mathbb{R}^n$ ou bien $Y \in \{-1,1\}^n$, ou bien même $Y \in \{0,1,\dots,K\}^n$

Par exemple :

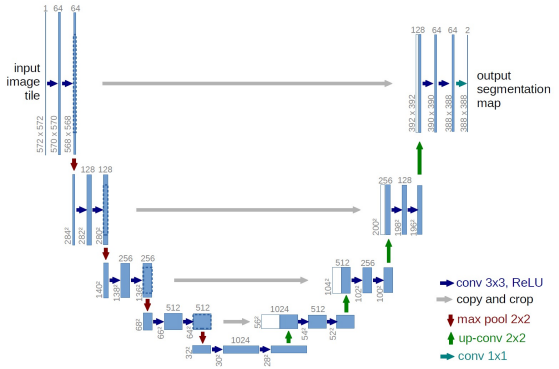
- **PLS** : Au lieu de résoudre $\widehat{v_1} = \arg \max_{v_1 \in \mathbb{R}^p, ||v_1||=1} \sum_{i=1}^{n} \langle X_i v_1, X_i v_1 \rangle$ (… et ainsi de suite) comme avec l'ACP, on résout

  $$\widehat{v_1} = \arg \max_{v_1 \in \mathbb{R}^p, ||v_1||=1} \sum_{i=1}^{n} \langle X_i v_1, Y_i \rangle$$

- **Régularisation LASSO :** On pénalise l'énergie minimisée avec la norme L_1 des paramètres d'un modèle. Les variables associées à des paramètres à zéro sont inutiles pour de futures prédictions. Par exemple :

  $$\hat{\theta} = \arg \min_{\theta \in \mathbb{R}^p} ||X\theta - Y||_2^2 + \lambda ||\theta||_1$$

- **Réseaux de neurones :** On réduit progressivement la dimension des couches de neurones et on considère que les données dans une couche de faible dimension sont les données réduites



1 : Intro

Nous avons vu avec l'ACP comment réduire la dimension d'un problème d'apprentissage de manière ***non supervisée***, c'est-à-dire en ne tenant compte que des propriétés des variables d'entrée $X \in \mathbb{R}^{n*p}$.

Afin de diminuer la dimension de $X \in \mathbb{R}^{n*p}$ vers $X_d \in \mathbb{R}^{n*d}$ avec $d < p$, il existe aussi une multitude de méthodes ***supervisées***

→ Réduction de la dimension de X en tenant compte du fait que X est lié à des sorties Y telles que $Y \in \mathbb{R}^n$ ou bien $Y \in \{-1,1\}^n$, ou bien même $Y \in \{0,1,\dots,K\}^n$

Par exemple :

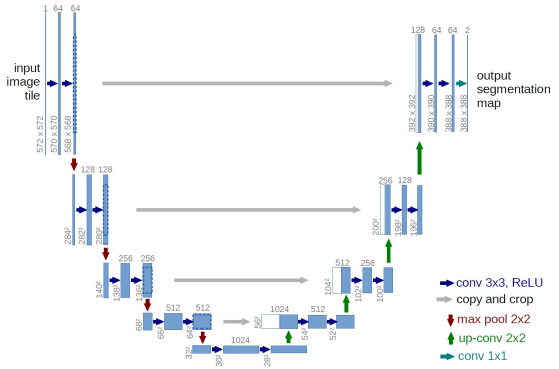
- **PLS** : Au lieu de résoudre $\widehat{v_1} = \arg \max_{v_1 \in \mathbb{R}^p, ||v_1||=1} \sum_{i=1}^{n} \langle X_i v_1, X_i v_1 \rangle$ (… et ainsi de suite) comme avec l'ACP, on résout

  $$\widehat{v_1} = \arg \max_{v_1 \in \mathbb{R}^p, ||v_1||=1} \sum_{i=1}^{n} \langle X_i v_1, Y_i \rangle$$

- **Régularisation LASSO :** On pénalise l'énergie minimisée avec la norme L_1 des paramètres d'un modèle. Les variables associées à des paramètres à zéro sont inutiles pour de futures prédictions. Par exemple :

  $$\hat{\theta} = \arg \min_{\theta \in \mathbb{R}^p} ||X\theta - Y||_2^2 + \lambda ||\theta||_1$$

- **Réseaux de neurones :** On réduit progressivement la dimension des couches de neurones et on considère que les données dans une couche de faible dimension sont les données réduites



Creusons ces méthodes !

On modélise souvent un problème de régression comme trouver la fonction f qui minimise le bruit ϵ_i sur n échantillons d'apprentissage (x_i, y_i) :

$$y_i = f(x_i) + \epsilon_i \,,$$

où f est une fonction inconnue et ϵ_i suit une loi Normale de moyenne nulle et d'écart type σ. Le but de la regression est alors de trouver une fonction $\hat{f}$ qui approxime au mieux f. Ceci se fait en fixant d'abord un modèle (linéaire, polynôme, arbre de décision, réseau de neurones, ...) puis en apprenant ses q paramètres à partir de ce que l'on connait, c'est à dire les (x_i, y_i). Le problème qui émerge naturellement est le suivant : Comment simultanément estimer f au mieux et tenir le moins possible compte du bruit ϵ sachant que les deux sont inconnus ? C'est la question clé du compromis biais-variance.

Plus formellement, on minimise l'esperance empirique de $(y - \hat{f}(x))^2$ sur les (x_i, y_i), c'est à dire l'erreur au carré moyenne (Mean Squared Error – MSE). Elle peut être décomposée sous cette forme :

$$\mathbb{E}\left[(y - \hat{f}(x))^2\right] = \underbrace{\mathbb{E}\left[\hat{f}(x) - f(x)\right]^2}_{biais[\hat{f}(x)]} + \underbrace{\mathbb{E}\left[\hat{f}(x)^2\right] - \mathbb{E}\left[\hat{f}(x)\right]^2}_{variance[\hat{f}(x)]} + \sigma^2 \qquad (3.1)$$

Cette représentation de la MSE peut être démontré en utilisant les relations suivantes :

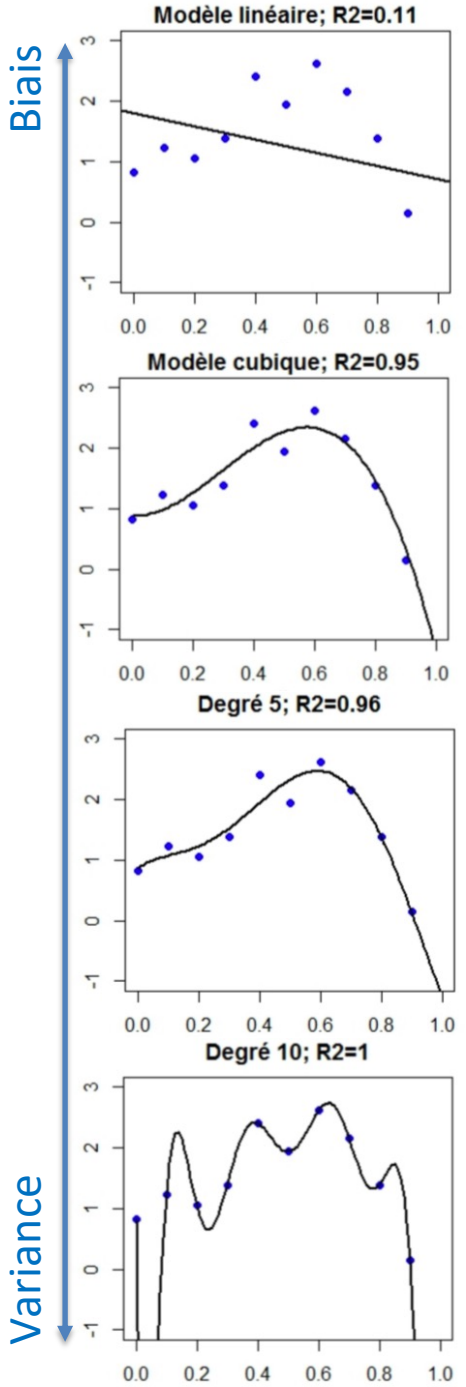
- $\mathbb{E}[f(x)] = f(x)$ car $f(x)$ est déterministe
- $\mathbb{E}[y] = \mathbb{E}[f(x) + \epsilon] = \mathbb{E}[f(x)] + \mathbb{E}[\epsilon] = \mathbb{E}[f(x)] = f(x)$
- $Var[\epsilon] = \mathbb{E}[\epsilon^2] + (\mathbb{E}[\epsilon])^2 = \mathbb{E}[\epsilon^2] = \sigma^2$
- $Var[y] = \mathbb{E}[(y - \mathbb{E}[y])^2] = \mathbb{E}[(y - f(x))^2] = \mathbb{E}[(f(x) + \epsilon - f(x))^2] = \sigma^2$

Plus formellement, on minimise l'esperance empirique de $(y - \hat{f}(x))^2$ sur les (x_i, y_i), c'est à dire l'erreur au carré moyenne (Mean Squared Error – MSE). Elle peut être décomposée sous cette forme :

$$\mathbb{E}\left[(y - \hat{f}(x))^2\right] = \underbrace{\mathbb{E}\left[\hat{f}(x) - f(x)\right]^2}_{biais[\hat{f}(x)]} + \underbrace{\mathbb{E}\left[\hat{f}(x)^2\right] - \mathbb{E}\left[\hat{f}(x)\right]^2}_{variance[\hat{f}(x)]} + \sigma^2 \qquad (3.1)$$

Plus intéressant ici, les différents termes d'Eq. (3.1) peuvent être interprétés comme suit :

- Le terme de biais $\mathbb{E}[\hat{f}(x) - f(x)]^2$ représente à quel point le modèle $\hat{f}$ approxime la fonction inconnue f.
- Le terme de variance $\mathbb{E}[\hat{f}(x)^2] - \mathbb{E}[\hat{f}(x)]^2 = Var[\hat{f}(x)]$ représent le niveau de variabilité de $\hat{f}$, sans tenir compte de f.
- Le terme σ^2 représente enfin le niveau de bruit dans les données (x_i, y_i), qui tout comme f est inconnu.



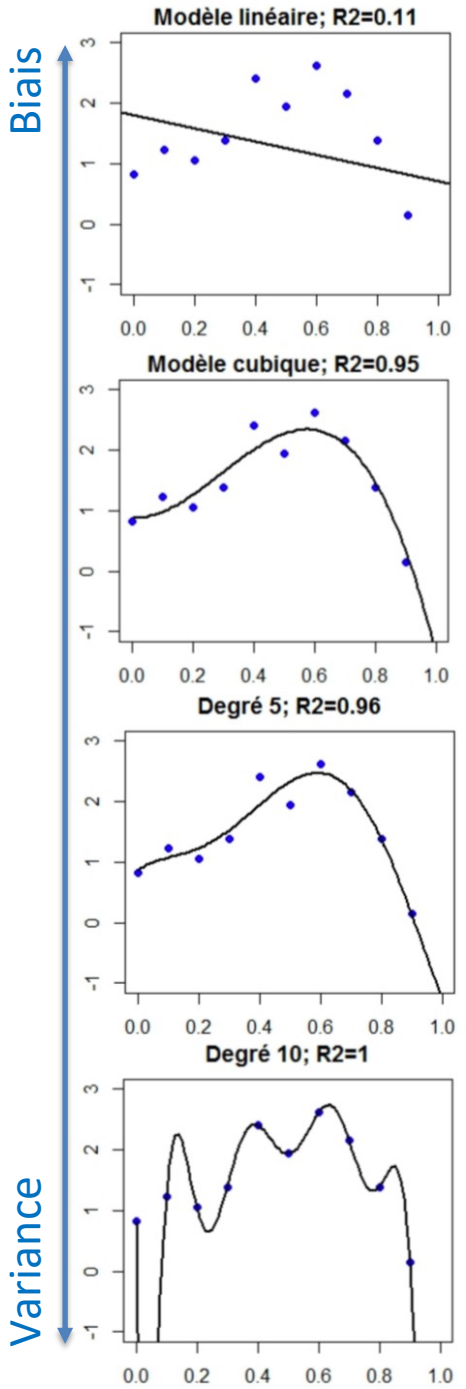
Plus formellement, on minimise l'esperance empirique de $(y - \hat{f}(x))^2$ sur les (x_i, y_i), c'est à dire l'erreur au carré moyenne (Mean Squared Error – MSE). Elle peut être décomposée sous cette forme :

$$\mathbb{E}\left[(y - \hat{f}(x))^2\right] = \underbrace{\mathbb{E}\left[\hat{f}(x) - f(x)\right]^2}_{biais[\hat{f}(x)]} + \underbrace{\mathbb{E}\left[\hat{f}(x)^2\right] - \mathbb{E}\left[\hat{f}(x)\right]^2}_{variance[\hat{f}(x)]} + \sigma^2 \qquad (3.1)$$

Plus intéressant ici, les différents termes d'Eq. (3.1) peuvent être interprétés comme suit :

- Le terme de biais $\mathbb{E}[\hat{f}(x) - f(x)]^2$ représente à quel point le modèle $\hat{f}$ approxime la fonction inconnue f.
- Le terme de variance $\mathbb{E}[\hat{f}(x)^2] - \mathbb{E}[\hat{f}(x)]^2 = Var[\hat{f}(x)]$ représent le niveau de variabilité de $\hat{f}$, sans tenir compte de f.
- Le terme σ^2 représente enfin le niveau de bruit dans les données (x_i, y_i), qui tout comme f est inconnu.

Pour une MSE (i.e. $\mathbb{E}[(y-\hat{f}(x))^2]$) donnée, un $\hat{f}$ représentera alors un compromis entre qualité d'approximation de f au niveau des observations $\{x_i\}_{i=1,\dots,n}$ et sa stabilité. Une trop grande qualité d'approximation au niveau des observations impliquera alors des fonctions $\hat{f}$ instables et ainsi moins généralisables en dehors des $\{x_i\}_{i=1,\dots,n}$ (sur-apprentissage). A contrario, des fonctions $\hat{f}$ trop stables captureront mal les relations entre les x_i et les y_i et auront de même un faible pouvoir prédictif.



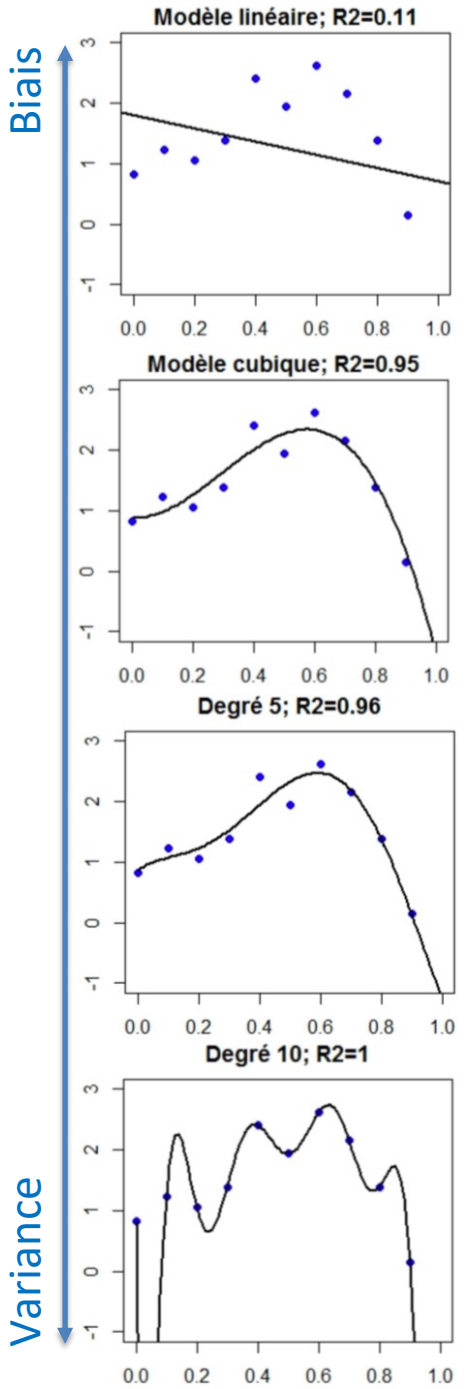
Plus formellement, on minimise l'esperance empirique de $(y - \hat{f}(x))^2$ sur les (x_i, y_i), c'est à dire l'erreur au carré moyenne (Mean Squared Error – MSE). Elle peut être décomposée sous cette forme :

$$\mathbb{E}\left[(y - \hat{f}(x))^2\right] = \underbrace{\mathbb{E}\left[\hat{f}(x) - f(x)\right]^2}_{biais[\hat{f}(x)]} + \underbrace{\mathbb{E}\left[\hat{f}(x)^2\right] - \mathbb{E}\left[\hat{f}(x)\right]^2}_{variance[\hat{f}(x)]} + \sigma^2 \qquad (3.1)$$

Plus intéressant ici, les différents termes d'Eq. (3.1) peuvent être interprétés comme suit :

- Le terme de biais $\mathbb{E}[\hat{f}(x) - f(x)]^2$ représente à quel point le modèle $\hat{f}$ approxime la fonction inconnue f.
- Le terme de variance $\mathbb{E}[\hat{f}(x)^2] - \mathbb{E}[\hat{f}(x)]^2 = Var[\hat{f}(x)]$ représent le niveau de variabilité de $\hat{f}$, sans tenir compte de f.
- Le terme σ^2 représente enfin le niveau de bruit dans les données (x_i, y_i), qui tout comme f est inconnu.

Trouver un bon compromis entre biais et variance pourra se faire en réduisant explicitement la dimension d'un modèle (Section 3.2) ou en régularisant l'estimation des paramètres d'un modèle (Section 3.3).

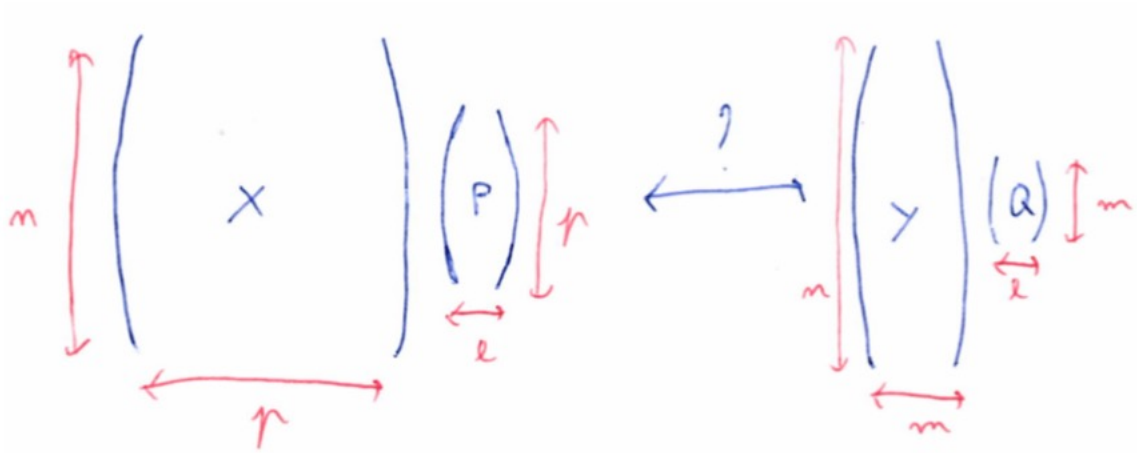


On a vu dans le cours de statistique que l'Analyse en Composantes Principales (ACP) était un outil essentiel pour explorer un ensemble d'observations $X_i = (x_i^1, \ldots, x_i^p)$, $i = 1, \ldots, n$ regroupées en ligne dans une matrice $\mathbf{X}$. L'ACP consiste en effet à maximiser la variance des projections des observations X_i, ce qui permet entre autres d'expliquer comment les variables interagissent entre elles. Plus spécifiquement, le 1er vecteur propre v_1 est celui qui maximise la variance des projections des X_i. En supposant que les X_i sont centrés (et idéalement réduits), cela signifie que

$$v_1 = \underset{v \text{ t.q. } |v|_2=1}{\arg\max} \sum_{i=1}^{n} (X_i v)^2$$

Le 2ème vecteur propre v_2 est choisi suivant le même principe, une fois enlevée l'influence de v_1 dans $\mathbf{X}$; et ainsi de suite.

L'idée de la méthode *Partial Least Squares* (PLS) est relativement similaire, mais maintenant on s'intéresse au lien entre $\mathbf{X}$ et une matrice $n \times m$ de réponses $\mathbf{Y}$. Pour chaque observation X_i de $\mathbf{X}$, la matrice $\mathbf{Y}$ contient une réponse Y_i en dimension m. Si $m = 1$, on a les mêmes données d'entrée que dans le cadre de la régression linéaire multiple (Section 2.2). L'approche d'analyse est cependant totalement différente : On cherche les transformations linéaires $\mathbf{P}$ et $\mathbf{Q}$ de $\mathbf{X}$ et de $\mathbf{Y}$ (si $m > 1$), respectivement, telles que : La 1ère colonne de $\mathbf{P}$ est celle qui projete les X_i de manière à séparer au mieux les y_i projetés par la première colonne de $\mathbf{Q}$; et ainsi de suite. Cette idée est shématisée ci-dessous :



2 : Méthode des Partial Least Squares (PLS)

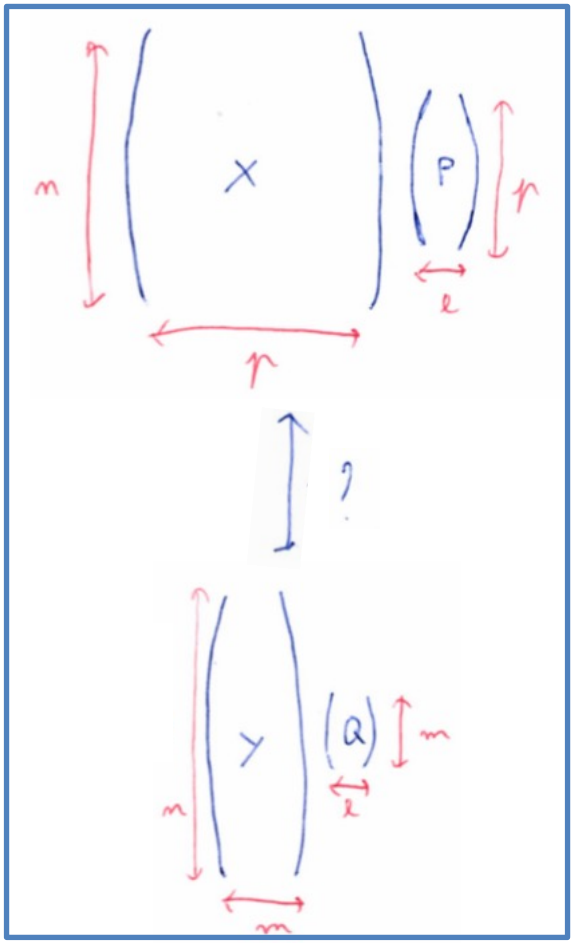
La méthode PLS (Partial Least Squares) repose toujours sur une hypothèse de modèle linéaire $\mathbf{Y} = \mathbf{X}\beta + \mathbf{U}$, où $\mathbf{U}$ modélise le bruit. L'approche utilisée pour estimer le lien entre $\mathbf{Y}$ et les variables explicatives de $\mathbf{X}$ est cependant différente de celle du modèle linéaire classique. En particulier, le modèle sur le bruit est totalement différent et va dépendre de la covariance entre des combinaisons linéaires de $\mathbf{X}$ et $\mathbf{Y}$.

Plus spécifiquement, on suppose :

$$\mathbf{X} = \mathbf{T}\mathbf{P}' + \mathbf{E}$$
$$\mathbf{Y} = \mathbf{U}\mathbf{Q}' + \mathbf{F}$$

où

- $\mathbf{X}$ est la matrice $n \times p$ de prédicteurs. Elle est supposée centrée/réduite,
- $\mathbf{Y}$ est la matrice $n \times m$ de réponses. Elle est supposée centrée/réduite,
- $\mathbf{P}$ et $\mathbf{Q}$ sont respectivement des matrices $p \times l$ et $m \times l$ de projection. Leurs colonnes sont orthonormés.
- $\mathbf{T}$ et $\mathbf{U}$ sont les projections de $\mathbf{X}$ et de $\mathbf{Y}$ respectivement par $\mathbf{P}$ et $\mathbf{Q}$. Elles sont de taille $n \times l$.
- $\mathbf{E}$ et $\mathbf{F}$ sont des termes d'erreur de même taille que $\mathbf{X}$ et $\mathbf{Y}$. Ils sont supposés *i.i.d.* et distribués suivant une loi normale.



Les projections de $\mathbf{X}$ et de $\mathbf{Y}$ dans $\mathbf{T}$ et $\mathbf{U}$ sont aussi toutes deux de même taille $n \times l$ avec $l \leq p$. La PLS consiste alors à calculer les projecteurs $\mathbf{P}$ et $\mathbf{Q}$ qui maximisent la covariance entre et $\mathbf{T}$ et $\mathbf{U}$. On dénote $\bar{\mathbf{T}}_j$ et $\bar{\mathbf{U}}_j$ la moyenne des valeurs des colonnes j de $\mathbf{T}$ et $\mathbf{U}$. On maximise alors $\sum_{j=1}^{l}\sum_{i=1}^{n}(\mathbf{T}_{ij} - \bar{\mathbf{T}}_j)(\mathbf{U}_{ij} - \bar{\mathbf{U}}_j)$.

Résolution pour *m=1*, c'est-à-dire $\mathbf{Y}$ est un vecteur colonne.

Alg. 1 Fonction $PLS1(\mathbf{X}, \mathbf{y}, l)$

$\mathbf{X}^{(0)} \leftarrow \mathbf{X}$
$\mathbf{w}^{(0)} \leftarrow \mathbf{X}^{'}\mathbf{y}/|\mathbf{X}^{'}\mathbf{y}|_2$.
for $k = 0, \ldots, l-1$ **do**
  $\mathbf{t}^{(k)} \leftarrow \mathbf{X}^{(k)}\mathbf{w}^{(k)}$
  $t_k \leftarrow \mathbf{t}^{(k)'}\mathbf{t}^{(k)}$
  $\mathbf{t}^{(k)} \leftarrow \mathbf{t}^{(k)}/t_k$
  $\mathbf{p}^{(k)} \leftarrow \mathbf{X}^{(k)'}\mathbf{t}^{(k)}$
  $q_k \leftarrow \mathbf{y}^{'}\mathbf{t}^{(k)}$
  if $q_k = 0$ **then**
    $l \leftarrow k$ et sort de la boucle **for** (toute la variabilité est capturée).
  end if
  if $k < (l-1)$ **then**
    $\mathbf{X}^{(k+1)} \leftarrow \mathbf{X}^{(k)} - t_k\mathbf{t}^{(k)}\mathbf{p}^{(k)'}$
    $\mathbf{w}^{(k+1)} \leftarrow \mathbf{X}^{(k+1)'}\mathbf{y}/|\mathbf{X}^{(k+1)'}\mathbf{y}|_2$
  end if
end for
$\mathbf{W}$ est la matrice composée des colonnes $\mathbf{w}^{(0)}, \mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(l-1)}$.
$\mathbf{P}$ est la matrice composée des colonnes $\mathbf{p}^{(0)}, \mathbf{p}^{(1)}, \ldots, \mathbf{p}^{(l-1)}$.
$\mathbf{q}$ est le vecteur composé des scalaires $q_0, q_1, \ldots, q_{l-1}$.
$\mathbf{B} \leftarrow \mathbf{W}(\mathbf{P}^{'}\mathbf{W})^{-1}\mathbf{q}$
$\mathbf{B_0} \leftarrow q_0 - \mathbf{P}^{(0)'}\mathbf{B}$
return $\mathbf{B}, \mathbf{B_0}$

Résolution pour *m=1*, c'est-à-dire $\mathbf{Y}$ est un vecteur colonne.

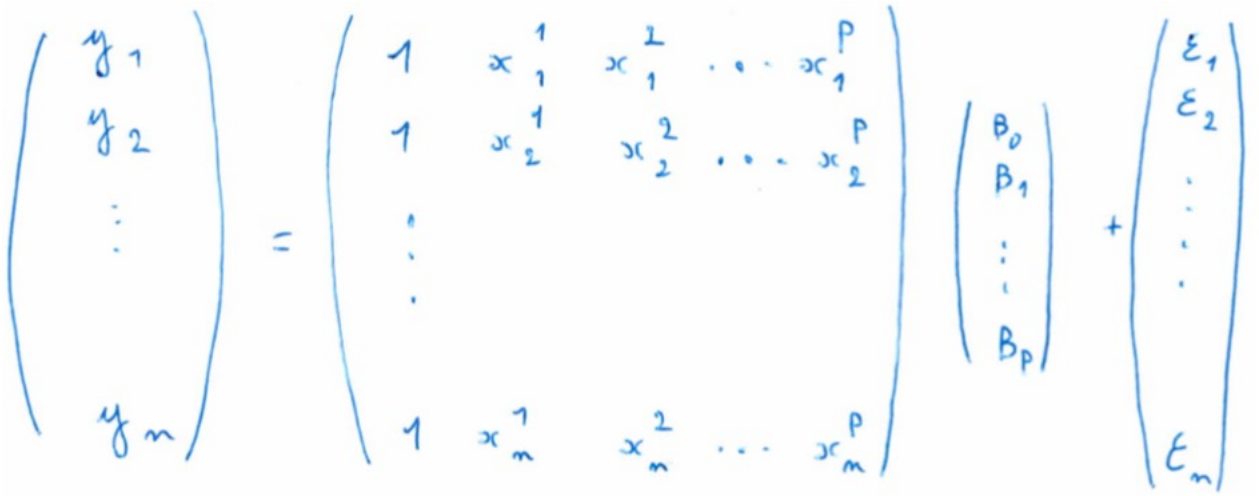
Alg. 1 Fonction $PLS1(\mathbf{X}, \mathbf{y}, l)$

$\mathbf{X}^{(0)} \leftarrow \mathbf{X}$
$\mathbf{w}^{(0)} \leftarrow \mathbf{X}'\mathbf{y}/|\mathbf{X}'\mathbf{y}|_2$.
for $k = 0, \ldots, l-1$ **do**
  $\mathbf{t}^{(k)} \leftarrow \mathbf{X}^{(k)}\mathbf{w}^{(k)}$
  $t_k \leftarrow \mathbf{t}^{(k)\prime}\mathbf{t}^{(k)}$
  $\mathbf{t}^{(k)} \leftarrow \mathbf{t}^{(k)}/t_k$
  $\mathbf{p}^{(k)} \leftarrow \mathbf{X}^{(k)\prime}\mathbf{t}^{(k)}$
  $q_k \leftarrow \mathbf{y}'\mathbf{t}^{(k)}$
  if $q_k = 0$ **then**
    $l \leftarrow k$ et sort de la boucle **for** (toute la variabilité est capturée).
  end if
  if $k < (l-1)$ **then**
    $\mathbf{X}^{(k+1)} \leftarrow \mathbf{X}^{(k)} - t_k\mathbf{t}^{(k)}\mathbf{p}^{(k)\prime}$
    $\mathbf{w}^{(k+1)} \leftarrow \mathbf{X}^{(k+1)\prime}\mathbf{y}/|\mathbf{X}^{(k+1)\prime}\mathbf{y}|_2$
  end if
end for
$\mathbf{W}$ est la matrice composée des colonnes $\mathbf{w}^{(0)}, \mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(l-1)}$.
$\mathbf{P}$ est la matrice composée des colonnes $\mathbf{p}^{(0)}, \mathbf{p}^{(1)}, \ldots, \mathbf{p}^{(l-1)}$.
$\mathbf{q}$ est le vecteur composé des scalaires $q_0, q_1, \ldots, q_{l-1}$.
$\mathbf{B} \leftarrow \mathbf{W}(\mathbf{P}'\mathbf{W})^{-1}\mathbf{q}$
$\mathbf{B_0} \leftarrow q_0 - \mathbf{P}^{(0)\prime}\mathbf{B}$
return $\mathbf{B}, \mathbf{B_0}$

Si « sparse PLS »: $p^{(k)} = p^{(k)} - \lambda_p sign(p^{(k)})$
et $p^{(k)} = 0$ si le signe est changé.

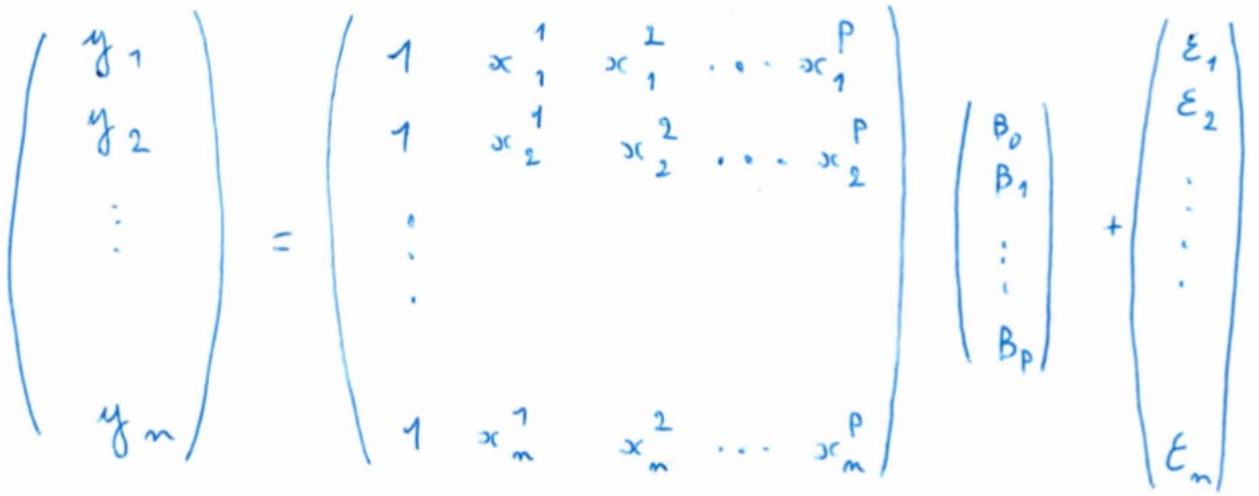
3 : Sélection de variable avec LASSO

Voyons maintenant le problème de la régression linéaire :
→ on estime des $Y_i \in \mathbb{R}$ à partir de $X_i \in \mathbb{R}^p$ avec un modèle linéaire



avec $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$

Voyons maintenant le problème de la régression linéaire :
→ on estime des $Y_i \in \mathbb{R}$ à partir de $X_i \in \mathbb{R}^p$ avec un modèle linéaire



avec $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$

L'estimation des β_i s'effectue alors en minimisant :

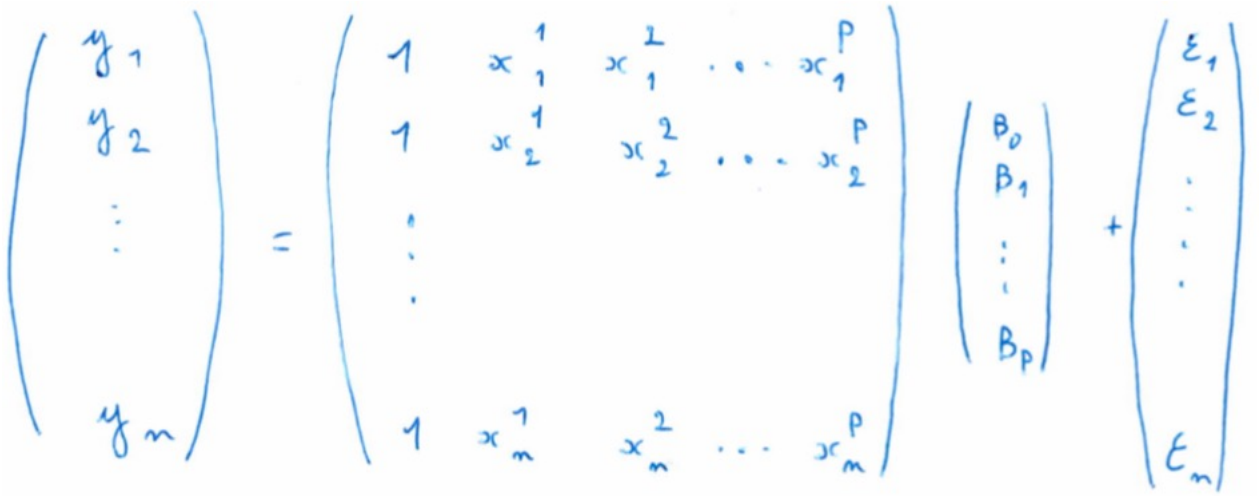
$$\sum_{i=1}^{n} (Y_i - \beta_0 - \beta_1 X_i^1 - \ldots - \beta_p X_i^p)^2 = ||\mathbf{Y} - \mathbf{X}\beta||^2$$

$$= \mathbf{Y}'\mathbf{Y} - 2\beta'\mathbf{X}'\mathbf{Y} + \beta'\mathbf{X}'\mathbf{X}\beta$$

Ce qui se résout analytiquement avec $\beta = (X'X)^{-1}X'Y$ … si $X'X$ est inversible !

3 : Sélection de variable avec LASSO

Voyons maintenant le problème de la régression linéaire :
→ on estime des $Y_i \in \mathbb{R}$ à partir de $X_i \in \mathbb{R}^p$ avec un modèle linéaire



avec $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$

L'estimation des β_i s'effectue alors en minimisant :

$$\sum_{i=1}^{n} (Y_i - \beta_0 - \beta_1 X_i^1 - \ldots - \beta_p X_i^p)^2 = ||\mathbf{Y} - \mathbf{X}\beta||^2$$

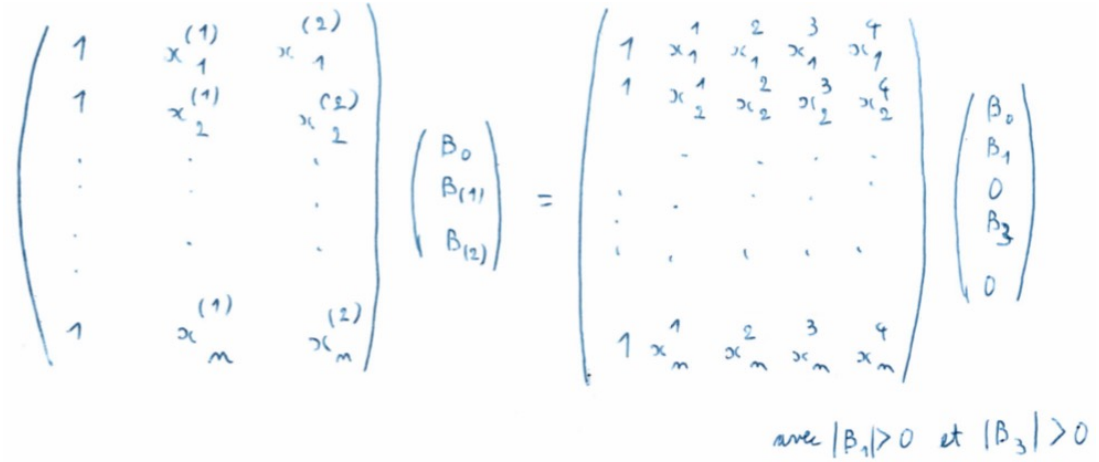
$$= \mathbf{Y}'\mathbf{Y} - 2\beta'\mathbf{X}'\mathbf{Y} + \beta'\mathbf{X}'\mathbf{X}\beta$$

Ce qui se résout analytiquement avec $\beta = (X'X)^{-1}X'Y$ … si $X'X$ est inversible !

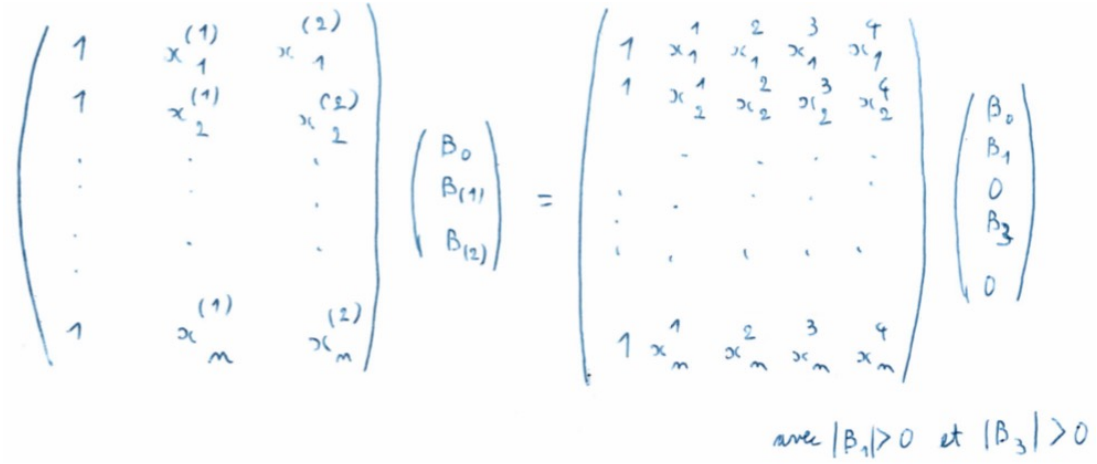
Motive de la sélection de modèle par critère (*AIC, …*), de la régularisation simple (*ridge*, …), ou de la régularisation parcimonieuse (*sparse*) avec par exemple LASSO !

3 : Sélection de variable avec LASSO

L'idée clé est d'estimer les β de manière à ce que les β_i les moins influents soient mis à zéro :



L'idée clé est d'estimer les β de manière à ce que les β_i les moins influents soient mis à zéro :



La méthode Lasso (Tibshirani, 1996) correspond à la minimisation d'un critère des moindres carrés avec une pénalité de type L_1 (et non L_2 comme dans la régression ridge). Soit $||\beta||_1 = \sum_{j=1}^{p} |\beta_j|$.

L'estimateur Lasso de β dans le modèle $\mathbf{Y} = \tilde{\mathbf{X}}\tilde{\beta} + \epsilon$ est alors défini par:

$$\widehat{\beta}_{LASSO} = \underset{\beta \in \mathbb{R}^{p+1}}{\arg\min} \left(\sum_{i=1}^{n} (Y_i - \sum_{j=0}^{p} X_i^{(j)} \beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right)$$

où λ est un paramètre positif. On peut montrer que ceci équivaut au problème de minimisation suivant

$$\widehat{\beta}_{LASSO} = \underset{\beta \in \mathbb{R}^p \text{ , } ||\beta||_1 < t}{\arg\min} ||\mathbf{Y} - \mathbf{X}\beta||^2$$

La solution obtenue est dite parcimonieuse (sparse en anglais), car elle comporte des coefficients nuls.

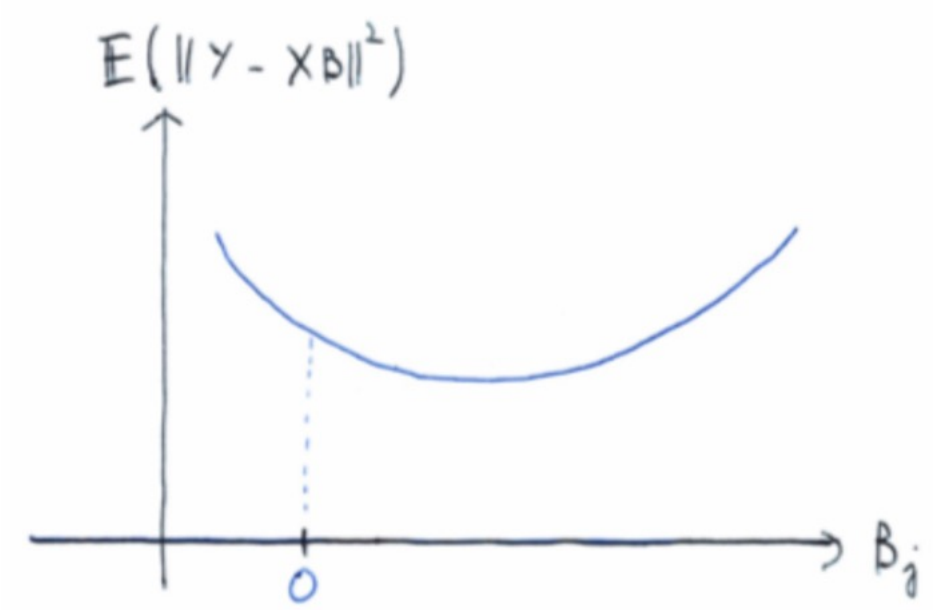
Pourquoi la pénalisation L_1 sélectionne-elle les variables ?

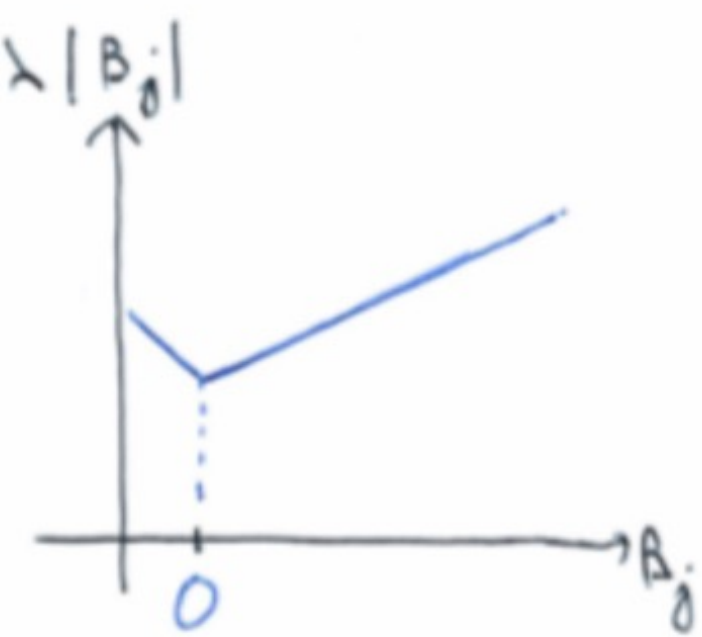
$$\widehat{\beta}_{L_1} = \arg\min_{\beta \in \mathbb{R}^p} f(\beta_1, \ldots, \beta_p) + \lambda \sum_{j=1}^{p} |\beta_j|$$

A l'état optimal, *i.e.* pour $\beta = \widehat{\beta}_{L_1}$, les gradients des fonctions optimisées sont nulles.

$$\frac{\partial f(\beta_1, \ldots, \beta_p)}{\partial \beta_j} = \lambda sign(\beta_j)$$

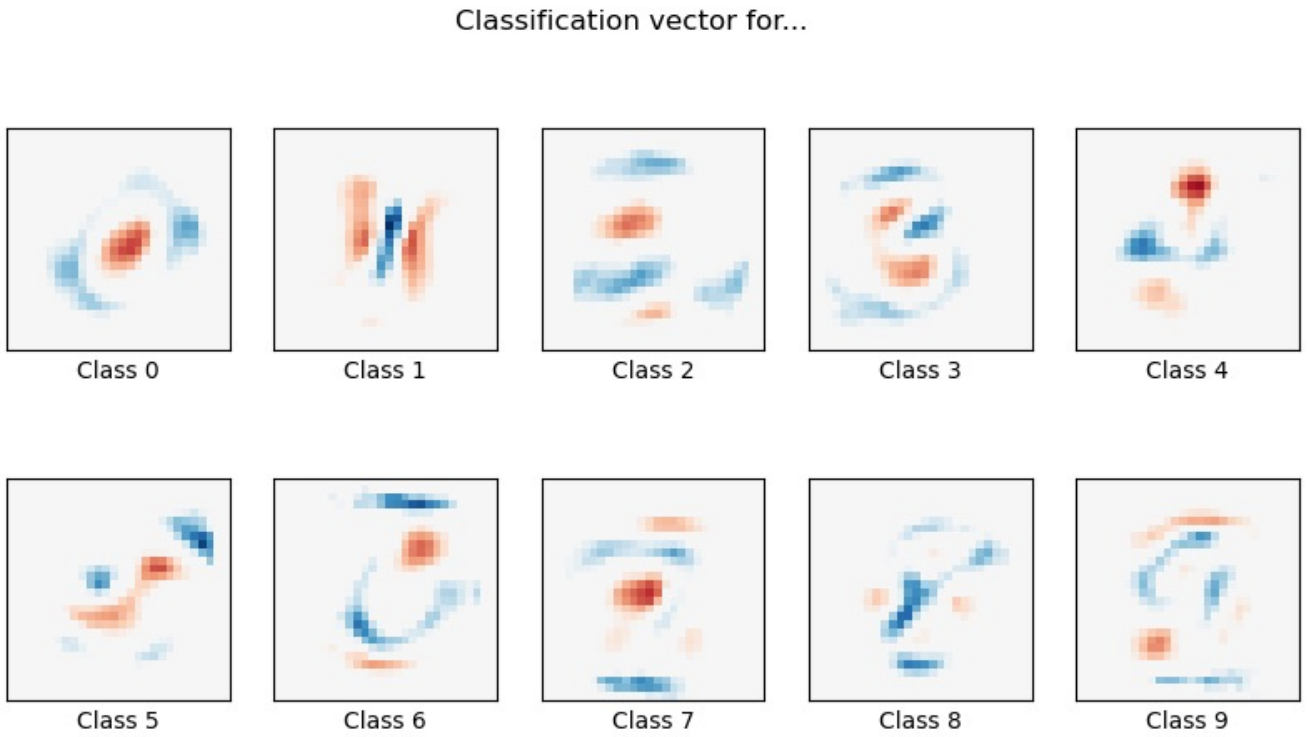
β_j est nul si $|\partial f(\ldots)/\partial \beta_j| < \lambda$ ce qui permet de ne sélectionner que les β_j ayant réellement une influence sur f.



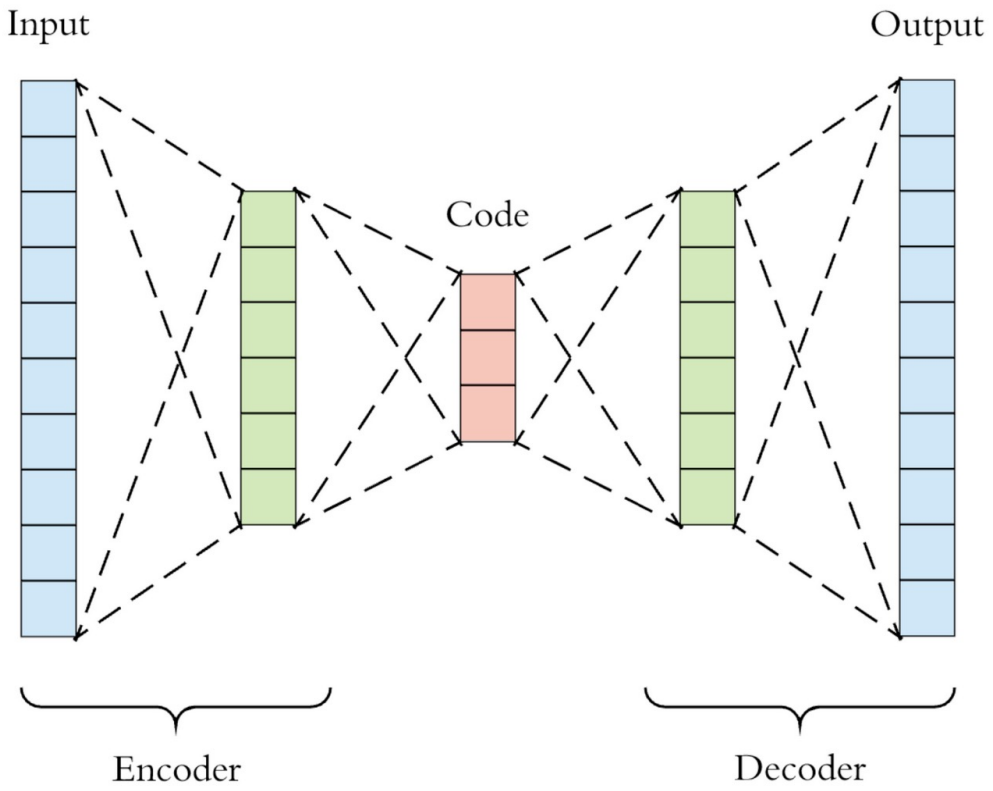


Application de la régularisation LASSO en régression logistique multi-classe

Représentation des β_i sous forme d'images pour détecter les {0,1,…, 9} dans les données MNIST :



Principe des auto-encodeurs :

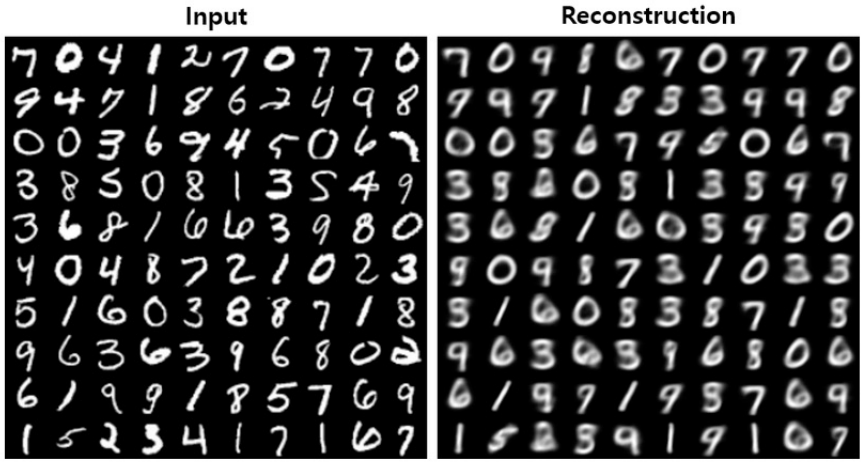


General architecture of an autoencoder

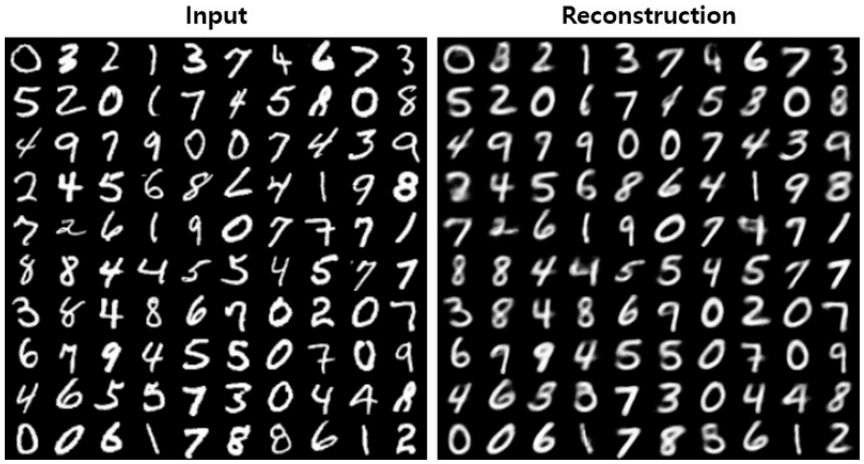
+ utilisation de couches de réseau convolutionelles

Principe des auto-encodeurs :

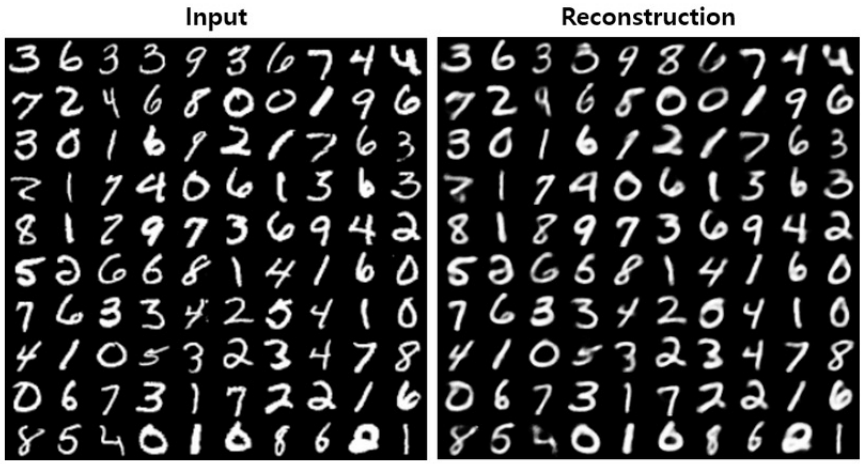
- 2-D latent space

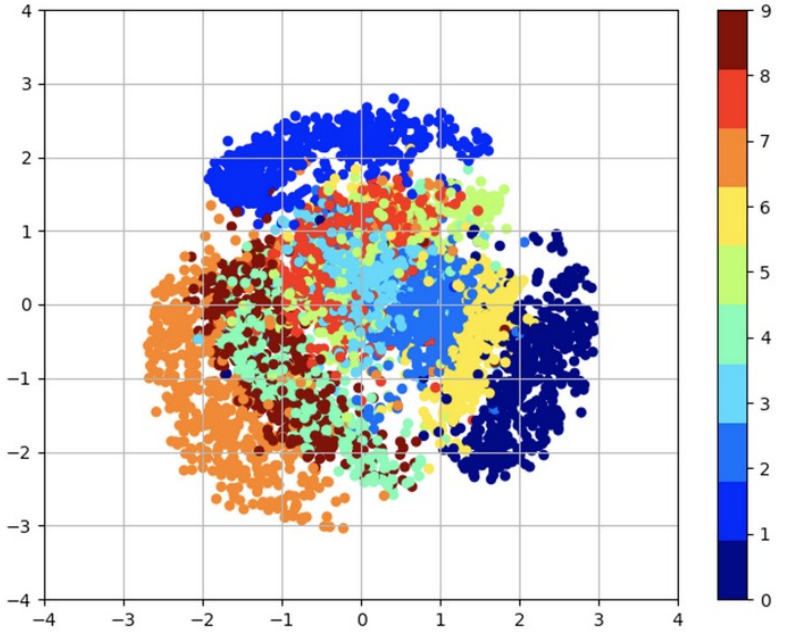


- 5-D latent space



- 20-D latent space





MERCI !