MAP553 Regression

Chapter 7: Biased regression

Contents

7.1. Introduction : orthogonal matrix X setting 1
7.2. Ridge regression 2
7.3. LASSO regression 6
7.4. Elastic-net regression 8
7.5. PLS regression (Partial least square) 9
7.6. Cross validation 11
7.2.1. Holdout methout 11
7.2.2. Cross-validation 12

In the previous chapters, we showed how to set a model, to select a model and to validate this model. But we are limited to the case $p \leq n$. When the matrix X is not a full rank matrix, the model is not identifiable anymore and to select a solution, we choose some constraints (asee previous chapter).

There are procedures that dispense with the assumption of full rank. These same procedures allow you to process the $p > n$ setting (one of the important issues of today). In order to introduce these other adapted estimation procedures, let's take a closer look at the case of orthogonal matrix X.

7.1. Introduction : orthogonal matrix X setting

➦ Let us consider the linear regression model ($p \leq n$)

$$Y = X\beta + \varepsilon$$

where X is a $n \times p$ full rank matrix, $\mathbb{E}[\varepsilon] = 0_n$ and $\mathbb{V}\text{ar}[\varepsilon] = \sigma^2 \mathbb{I}_n$, then

$$\boxed{\widehat{\beta} = (X^T X)^{-1} X^T Y \qquad \mathbb{V}\text{ar}[\widehat{\beta}] = \sigma^2 \mathbb{I}_p \qquad \text{and} \quad \mathbb{E}\|\widehat{\beta} - \beta\|^2 = \sigma^2 p.}$$

➦ Now set $p = n$ and X a $n \times n$ full rank orthogonal matrix, *i.e.* $X^T X = X X^T = \mathbb{I}_n$ then

$$\boxed{\widehat{\beta} = X^T Y \qquad \mathbb{V}\text{ar}[\widehat{\beta}] = \sigma^2 \mathbb{I}_n \qquad \mathbb{E}\|\widehat{\beta} - \beta\|^2 = \sigma^2 n \quad \text{and} \quad \widehat{\beta}_k = P_{X_k} Y.}$$

➦ Consider now two nested models, say $[m_0]$ and $[m_1] \subseteq [n] = [p]$ such that

$$[m_1] = [m_0] \cup \{X_k \notin [m_0]\},$$

Choosen between the model $[m_1]$ and $[m_0]$ is equivalent to select or not the regressor X_k. We saw in chapter 5 that we select X_k if for some positive constant q, $S > q$ where

$$S = \frac{SCR(m_0) - SCR(m_1)}{SCR(m_1)/(n - |m_1|)} = \frac{\|P_{[X_k]}Y\|^2}{\widehat{\sigma}^2_{m_1}} \approx \frac{\|P_{[X_k]}Y\|^2}{\widehat{\sigma}^2}$$

with $\widehat{\sigma}^2_{m_1} = \frac{SCR(m_1)}{(n-|m_1|)}$.

➦ Therefore, in the setting of orhtogonal matrix $S \approx \widehat{\beta}^2_k / \widehat{\sigma}^2$. And, we keep/select the regressor X_k iff

$$|\widehat{\beta}_k| > q\widehat{\sigma}.$$

In other words, we select the regressor X_k if the estimation of the associated parameter β_k is larger than the level of noise times up tp a multiplicative constant : *"we do not only estimate noise", "the signal is louder than the level of noise"*.

7.2. Ridge regression

If the full-rank assumption is not satisfied, the matrix $X = [X_1, \cdots, X_p]$ is such that

$$\exists k \in \{1, \cdots, p\}, \quad X_k = \sum_{j \neq k} \alpha_j X_j.$$

Note that the projection of Y into th space $[X] = span\{X_1, \cdots, X_p\}$ denoted $\widehat{Y} = P_{[X]}Y$ is unique but its decomposition on $[X]$ is not, *i.e.* there is an infinite solution for $\widehat{\beta}$ such that

$$\widehat{Y} = X\widehat{\beta}.$$

To counter the problem of identifiability of the model, we have already set constraints (see previous chapters). Let's now introduce methods adapted to the rank deficiency.

First of all some basic notations and reminders:

- By language abuse, we write $X^T X \geq 0$ to signify that $X^T X$ is positive, *i.e.*

$$\forall v \in \mathbb{R}^p_*, \quad v^T X^t X v \geq 0.$$

Therefore the eigenvalues of X^TX denoted $\Lambda := \Lambda_{X^TX}$ are such that

$$\Lambda_1 \geq \cdots \geq \Lambda_p \geq 0.$$

Moreover, if the determinant of X^TX is null ($\text{Det}(X^TX) = 0$) then

$$\exists j, \quad s.t.\ \Lambda_j = 0 \quad \text{and} \quad \forall r \geq j, \quad \Lambda_r = 0.$$

- Let $\lambda > 0$, then the eigenvectors of X^TX are equal to the eigenvectors of $(X^TX + \lambda\mathbb{I}_p)$:

$$\Lambda_{X^TX+\lambda\mathbb{I}_p} = \lambda + \Lambda_{X^TX}.$$

By appling the previous remaning, and by replacing $(X^TX)^{-1}$ by $(X^TX + \lambda\mathbb{I}_p)^{-1}$ in the definition of the least squares estimator, we obtain an unique estimator of β stable, called Ridge estimator:

$$\widehat{\beta^R}(\lambda) = (X^TX + \lambda\mathbb{I}_p)^{-1}X^TY.$$

Proposition 1

Let $\lambda > 0$, then the Ridge estimator[a] is such that

$$\widehat{\beta^R}(\lambda) = (X^TX + \lambda\mathbb{I}_p)^{-1}X^TY = \arg\min_{\beta\in\mathbb{R}^p} ||Y - X\beta||^2 + \lambda||\beta||^2.$$

*The parameter λ is called the **tuning parameter**.*

[a] Hoerl and Kennard, 1970

Proof : Admitted.

Comments:

- ☛ If $\lambda > 0$, then $(X^TX + \lambda\mathbb{I}_p)^{-1}$ is well defined.
- ☛ Let consider the conditions of existance of $\widehat{\beta}$ the ordinary least squares estimator (OLSE), then

$$\boxed{\widehat{\beta^R}(\lambda) = (X^TX + \lambda\mathbb{I}_p)^{-1}X^TX\widehat{\beta}.}$$

- ☛ The Ridge estimator $\widehat{\beta^R}(\lambda)$ can be seen as a **penalized least squares estimator**.

Proposition 2

Let $\lambda > 0$, then

1. $\mathbb{E}[\widehat{\beta^R}(\lambda)] = \beta - \lambda(X^TX + \lambda\mathbb{I}_p)^{-1}\beta$
2. $\mathbb{V}ar[\widehat{\beta^R}(\lambda)] = \sigma^2(X^TX + \lambda\mathbb{I}_p)^{-1}X^TX(X^TX + \lambda\mathbb{I}_p)^{-1}$

3. *Let $X^TX = P\text{Diag}(\Lambda)\text{P}^{\text{T}}$, where P is a matrix such that $P^T = P^{-1}$ and $\|P^T\beta\|^2 = \|\beta\|^2$, then the mean square error (MSE) is equal to*

$$\mathbb{E}\|\widehat{\beta^R}(\lambda) - \beta\|_2^2 = \sum_{j=1}^{p} \frac{\sigma^2\Lambda_j + \lambda^2[P^T\beta]_j^2}{(\lambda + \Lambda_j)^2}$$

Proof : We prove the proposition when the conditions of existance of the OLSE $\widehat{\beta}$ are satisfied.

1. Let $\widehat{\beta^R}(\lambda) = (X^TX + \lambda\mathbb{I}_p)^{-1}X^TX\widehat{\beta}$, where $\widehat{\beta}$ is such that

$$\mathbb{E}[\widehat{\beta}] = \beta \quad \text{and} \quad \mathbb{V}\text{ar}[\widehat{\beta}] = \sigma^2(X^TX)^{-1}$$

Then,

$$\begin{aligned}
\mathbb{E}[\widehat{\beta^R}(\lambda)] &= (X^TX + \lambda\mathbb{I}_p)^{-1}X^TX\mathbb{E}[\widehat{\beta}] \\
&= (X^TX + \lambda\mathbb{I}_p)^{-1}X^TX\beta \\
&= (X^TX + \lambda\mathbb{I}_p)^{-1}(X^TX + \lambda\mathbb{I}_p - \lambda\mathbb{I}_p)\beta \\
&= (X^TX + \lambda\mathbb{I}_p)^{-1}(X^TX + \lambda\mathbb{I}_p)\beta - \lambda(X^TX + \lambda\mathbb{I}_p)^{-1}\beta \\
&= \beta - \lambda(X^TX + \lambda\mathbb{I}_p)^{-1}\beta
\end{aligned}$$

Moreover

$$\beta - \mathbb{E}[\widehat{\beta^R}(\lambda)] = \lambda(X^TX + \lambda\mathbb{I}_p)^{-1}\beta. \tag{1}$$

2. As $\mathbb{V}\text{ar}[\widehat{\beta}] = \sigma^2(X^TX)^{-1}$, it comes

$$\begin{aligned}
\mathbb{V}\text{ar}[\widehat{\beta^R}(\lambda)] &= (X^TX + \lambda\mathbb{I}_p)^{-1}X^TX\mathbb{V}\text{ar}[\widehat{\beta}]X^TX(X^TX + \lambda\mathbb{I}_p)^{-1} \\
&= \sigma^2(X^TX + \lambda\mathbb{I}_p)^{-1}X^TX(X^TX)^{-1}X^TX(X^TX + \lambda\mathbb{I}_p)^{-1} \\
&= \sigma^2(X^TX + \lambda\mathbb{I}_p)^{-1}X^TX(X^TX + \lambda\mathbb{I}_p)^{-1}
\end{aligned}$$

3. First recall that for any estimator $\widehat{\theta}$ of an parameter $\theta \in \mathbb{R}^p$

$$\mathbb{E}\left[(\theta - \widehat{\theta})(\theta - \widehat{\theta})^T\right] = \left(\theta - \mathbb{E}[\widehat{\theta}]\right)\left(\theta - \mathbb{E}[\widehat{\theta}]\right)^T + \mathbb{V}\text{ar}[\widehat{\theta}]. \tag{2}$$

Applying (2) to $\widehat{\beta^R}(\lambda)$, we get

$$\mathbb{E}\left[(\beta - \widehat{\beta^R}(\lambda))(\beta - \widehat{\beta^R}(\lambda))^T\right] = \left(\beta - \mathbb{E}[\widehat{\beta^R}(\lambda)]\right)\left(\beta - \mathbb{E}[\widehat{\beta^R}(\lambda)]\right)^T + \mathbb{V}\text{ar}[\widehat{\beta^R}(\lambda)].$$

Moreover, by (1)

$$\begin{aligned}
\mathbb{E}\left[(\beta - \widehat{\beta^R}(\lambda))(\beta - \widehat{\beta^R}(\lambda))^T\right] &= \left(\lambda(X^TX + \lambda\mathbb{I}_p)^{-1}\beta\right)\left(\lambda(X^TX + \lambda\mathbb{I}_p)^{-1}\beta\right)^T + \mathbb{V}\text{ar}[\widehat{\beta^R}(\lambda)] \\
&= \lambda^2(X^TX + \lambda\mathbb{I}_p)^{-1}\beta\beta^T(X^TX + \lambda\mathbb{I}_p)^{-1} + \sigma^2(X^TX + \lambda\mathbb{I}_p)^{-1}X^TX(X^TX + \lambda\mathbb{I}_p)^{-1} \\
&= (X^TX + \lambda\mathbb{I}_p)^{-1}\left(\lambda^2\beta\beta^T + \sigma^2X^TX\right)(X^TX + \lambda\mathbb{I}_p)^{-1} \qquad (3)
\end{aligned}$$

From now, we denote by $\text{diag}(\Lambda)$ and $\text{diag}\left(1/(\Lambda+\lambda)\right)$ the $p \times p$ diagonal matrices whose diagonal elements are the respectively $1/(\Lambda_i+\lambda)$ and $\Lambda_i+\lambda$, with Λ_i the eigenvalues of the matrix X^TX. Then, let us decompose $X^TX = P\text{diag}(\Lambda)P^T$, it comes

$$X^TX = P\text{diag}(\Lambda)P^T \quad \text{and} \quad (X^TX+\lambda\mathbb{I}_p)^{-1} = P\text{diag}\left(1/(\Lambda+\lambda)\right)P^T$$

Then, as $P^TP = PP^T = \mathbb{I}_p$ and by using equation (3) becomes

$$\begin{aligned}
\mathbb{E}\left[(\beta-\widehat{\beta}^R(\lambda))(\beta-\widehat{\beta}^R(\lambda))^T\right] &= (X^TX+\lambda\mathbb{I}_p)^{-1}\left(\lambda^2\beta\beta^T+\sigma^2X^TX\right)(X^TX+\lambda\mathbb{I}_p)^{-1} \\
&= P\text{diag}\left(1/(\Lambda+\lambda)\right)P^T\left(\lambda^2\beta\beta^T+\sigma^2P\text{diag}(\Lambda)P^T\right)P\text{diag}\left(1/(\Lambda+\lambda)\right)P^T \\
&= P\text{diag}\left(1/(\Lambda+\lambda)\right)\left(\lambda^2P^T\beta\beta^TP+\sigma^2P^TP\text{diag}(\Lambda)PP^T\right)\text{diag}\left(1/(\Lambda+\lambda)\right)P^T \\
&= P\text{diag}\left(1/(\Lambda+\lambda)\right)\left(\lambda^2P^T\beta\beta^TP+\sigma^2\text{diag}(\Lambda)\right)\text{diag}\left(1/(\Lambda+\lambda)\right)P^T \qquad (4)
\end{aligned}$$

We are now ready to prove 3. of the proposition 2

$$\begin{aligned}
\mathbb{E}\left[\|\widehat{\beta}^R(\lambda)-\beta\|_2^2\right] &= \mathbb{E}\left[\text{Tr}\left(\|\widehat{\beta}^R(\lambda)-\beta\|_2^2\right)\right] = \mathbb{E}\left[\text{Tr}\left((\beta-\widehat{\beta}^R(\lambda))^T(\beta-\widehat{\beta}^R(\lambda))\right)\right] \\
&= \mathbb{E}\left[\text{Tr}\left((\beta-\widehat{\beta}^R(\lambda))(\beta-\widehat{\beta}^R(\lambda))^T\right)\right] = \text{Tr}\left[\mathbb{E}\left((\beta-\widehat{\beta}^R(\lambda))(\beta-\widehat{\beta}^R(\lambda))^T\right)\right]
\end{aligned}$$

By using (4), we have as $P^TP = \mathbb{I}_p$

$$\begin{aligned}
\mathbb{E}\left[\|\widehat{\beta}^R(\lambda)-\beta\|_2^2\right] &= \text{Tr}\left[\mathbb{E}\left((\beta-\widehat{\beta}^R(\lambda))(\beta-\widehat{\beta}^R(\lambda))^T\right)\right] \\
&= \text{Tr}\left[P\text{diag}\left(1/(\Lambda+\lambda)\right)\left(\lambda^2P^T\beta\beta^TP+\sigma^2\text{diag}(\Lambda)\right)\text{diag}\left(1/(\Lambda+\lambda)\right)P^T\right] \\
&= \text{Tr}\left[P^TP\text{diag}\left(1/(\Lambda+\lambda)\right)\left(\lambda^2P^T\beta\beta^TP+\sigma^2\text{diag}(\Lambda)\right)\text{diag}\left(1/(\Lambda+\lambda)\right)\right] \\
&= \text{Tr}\left[\text{diag}\left(1/(\Lambda+\lambda)\right)\left(\lambda^2P^T\beta\beta^TP+\sigma^2\text{diag}(\Lambda)\right)\text{diag}\left(1/(\Lambda+\lambda)\right)\right] \\
&= \text{Tr}\left[\text{diag}\left(1/(\Lambda+\lambda)^2\right)\left(\lambda^2P^T\beta\beta^TP+\sigma^2\text{diag}(\Lambda)\right)\right] \\
&= \text{Tr}\left[\text{diag}\left(1/(\Lambda+\lambda)^2\right)\left(\lambda^2P^T\beta\beta^TP\right)\right]+\text{Tr}\left[\text{diag}\left(1/(\Lambda+\lambda)^2\right)\left(\sigma^2\text{diag}(\Lambda)\right)\right] \\
&= \text{Tr}\left[\text{diag}\left(\lambda^2/(\Lambda+\lambda)^2\right)\left(P^T\beta\beta^TP\right)\right]+\text{Tr}\left[\text{diag}\left(\sigma^2\Lambda/(\Lambda+\lambda)^2\right)\right] \\
&= \sum_{j=1}^p\frac{\lambda^2[P^T\beta]_j^2}{(\Lambda_j+\lambda)^2}+\sum_{j=1}^p\frac{\sigma^2\Lambda_j}{(\Lambda_j+\lambda)^2} = \sum_{j=1}^p\frac{\lambda^2[P^T\beta]_j^2+\sigma^2\Lambda_j}{(\Lambda_j+\lambda)^2} \qquad \square
\end{aligned}$$

Comments:

- From propisition 2, it comes that the Ridge estimator is an biased estinator. Anyway, it may be better from the MSE point of view. Indeed if λ increases the variance of the Ridge estimator decreases.
- If $\lambda \to +\infty$ then $\widehat{\beta}^R(\lambda) \to 0$, it means no regressor is selected. Moreover the MSE is

$$\mathbb{E}\|\widehat{\beta}^R(\lambda)-\beta\|_2^2 \to \|\beta\|^2.$$

- ☛ If $\lambda \to 0$ then $\widehat{\beta^R}(\lambda) \to \widehat{\beta}$: all regressors are selected. The MSE is in this case the one of the OLSE
$$\mathbb{E}\|\widehat{\beta^R}(\lambda) - \beta\|_2^2 \to \sigma^2 \mathrm{Tr}((\mathrm{X}^{\mathrm{T}}\mathrm{X})^{-1}).$$
- ☛ If $X^T X = \mathbb{I}_p$ (orthogonal matrix X) then
$$\widehat{\beta^R}(\lambda) = \frac{X^T Y}{1+\lambda} = \frac{\widehat{\beta}}{1+\lambda}.$$
The values of $\widehat{\beta_k}$ the OLSE are decreasing with λ. Ridge estimator belongs to the "*shrinkage estimators*", it shrinks the coefficients but keeps all variables.
- ☛ In general, Ridge regression is interesting when eigenvalues Λ_j are small.
- ☛ The hyperprameter λ has to be calibrated (see section 7.6.).

Differents packages

- Function `lm.ridge` of the package `MASS`.
- Function `train( ,method='glmnet',`$\alpha = 0$`)` of the package `caret`.

Verify in the documentation if the regressors and/or the response variable are re-centered and/or standardizeds. In `glmnet`, by default, it is both done and the output coefficient are in the original scale!

7.3. LASSO regression

The Ridge estimator can be viewed as a penalized estimator (the least square plus a ℓ_2 penality). Now let's introduce the LASSO estimator (Least Absolute Shrinkage and Selection Operator) which is also a penalized estimator (the least squares plus penality ℓ_1).

Definition 1
Let $\lambda > 0$, LASSO[a] estimator denoted by $\widehat{\beta^L}(\lambda)$ is defined as follows
$$\widehat{\beta^L}(\lambda) = \arg\min_{\beta \in \mathbb{R}^p} \|Y - X\beta\|^2 + \lambda\|\beta\|_1$$
where $\|\beta\|_1 = \sum_{j=1}^p |\beta_j|$.

[a] Tibshirani, 1996

Comments:

- ☛ In general, there is no closed form for the LASSO estimator.
- ☛ But, this is a convex optimization problem and there are efficient approximation algorithms.

☛ In the setting of orthogonal matrix X, we can derive a closed form. This is the aimof proposition 3.

Proposition 3 *Let $\lambda > 0$, if $XTX = \mathbb{I}_p$ then the LASSO estimator is equal to*

$$\widehat{\beta}_j^L(\lambda) = \text{sign}(X_j^T Y)\left[|X_j^T Y| - \lambda/2\right]_+,$$

where $[x]_+ = x$ if $x \geq 0$ and 0 otherwise.

Proof : Let as exercice.

Comments:

☛ For any $j = 1, \cdots, p$, the quantity $|X_j^T Y|$ an indicator of the correlation between the regressor X_j and the response Y.

☛ Note that if $\lambda/2 > |X_j^T Y|$ then $\widehat{\beta}_j^L(\lambda) = 0$, so the regressor X_j is removed from the model. Therefore, if

$$\lambda/2 > \|X^T Y\|_\infty = \max_j |X_j^T Y| \Rightarrow \widehat{\beta}^L(\lambda) = 0.$$

So there is an infinity of λ values such that $\widehat{\beta}^L(\lambda) = 0$.

☛ Let $k \in \{1, \cdots, p\}$ such that

$$\|X^T Y\|_\infty = \max_j |X_j^T Y| = |X_k^T Y|,$$

i.e. the regressor X_k is the most correlated to Y. So as soon as $\lambda/2$ goes below the threshold $\|X^T Y\|_\infty$, a first explanatory variable is retained, the regressor X_k.

☛ In practice, the λ parameter must be calibrated (see section 7.6.).

☛ Note that there is no theory on tests and confidence intervals for Ridge and Lasso estimators.

Conclusion comments:

☛ The Ridge estimator prevents overfitting. But it does not make the variable selection, it only shrinkes the β coefficients but keeps all regressors.

☛ The LASSO estimator also shrinks coefficients but puts some coefficients to zeros. It makes variables selection.

Differents packages/functions

- Function `l1ce` with the package `lasso2`.
- Function `cv.lars` with the package `lars`.
- Function `train( ,method='glmnet',`$\alpha = 1$`)` of the package `caret`.

7.4. Elastic-net regression

Definition 2

• *Let $\lambda_1 > 0$ and $\lambda_2 > 0$, the Elastic-Net[a] estimator denoted by $\widehat{\beta}^{EN}(\lambda_1, \lambda_2)$ is defined as follows*

$$\widehat{\beta}^{EN}(\lambda_1, \lambda_2) = \arg\min_{\beta \in \mathbb{R}^p} \|Y - X\beta\|^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2.$$

• *Let $\lambda > 0$, the Elastic-Net estimator can also be defined as follows*

$$\widehat{\beta}^{EN}(\lambda, \alpha) = \arg\min_{\beta \in \mathbb{R}^p} \|Y - X\beta\|^2 + \lambda \left(\alpha \|\beta\|_1 + (1-\alpha)\|\beta\|_2^2\right),$$

where $\alpha \in [0, 1]$.

[a] Zou and Hastie, 2005

Conclusion comments:

- ☛ As for the LASSO, there is no closed form for the Elastic-net, but the optimisation problem is convex and there exist efficient approximation algorithms.
- ☛ The Elastic-net regression is a mix of the Ridge regression and the LASSO regression.
- ☛ As for Ridge and LASSO regressions, there is no theory on tests and confidence intervals for the Elastic-net regression.
- ☛ The Ridge estimator prevents overfitting. But it does not make the variable selection, it only shrinkes the β coefficients but keeps all regressors.
- ☛ The LASSO estimator also shrinks coefficients with some shrunk to zeros. It helps with features selection.
- ☛ Here again, hyperprameters λ and α have to be calibrated (see section 7.6.).

Differents packages/functions

- Function `elasctic.net` with the package .
- Function `train( ,method='glmnet',`α `=, `λ `=)` of the package `caret`.

7.5. PLS regression (Partial least square)

The principle of PSL regression is "close" to principal component regression (PCA). The purpose of this method is to introduce new regressors $t^{(1)}, \cdots, t^{(k)}$ such that :

- They are linear combinations of the departure regressors (which will have previously centered and renormalized). We assume $\overline{Y} = \overline{X}_j = 0$ and $s_Y^2 = s_{X_j}^2 = 1$.
- They are 2 by 2 orthogonal

$$\forall i \neq i' \quad \text{we have} \quad < t^{(i)}, t^{(i')} >= 0.$$

- They are ranked in order of importance by taking as their criterion their link with the variable Y.

PSL algorithm

Step 1

➤ We set $X^{(1)} = X$ and $Y^{(1)} = Y$.

➤ Calculation of the first PLS component $t^{(1)}$: for $w \in \mathbb{R}^p$

$$t^{(1)} = \underset{t=X^{(1)}w,\ \|w\|_2=1}{\arg\max} < t, Y^{(1)} >$$

➤ Then, regress $Y^{(1)}$ on $t^{(1)}$ such that $Y^{(1)} = r_1 t^{(1)} + \widehat{\varepsilon}_1$,

- with $r_1 \in \mathbb{R}$, the coefficient of the orthogonal projection on $t^{(1)}$:

$$r_1 t^{(1)} = P_{[t^{(1)}]} Y^{(1)} \quad \text{and} \quad r_1 = (t^{(1)\,T} t^{(1)})^{-1} t^{(1)\,T} Y^{(1)}$$

- and the residual $\widehat{\varepsilon}_1$ which corresponds to the part not explained by $t^{(1)}$

$$\widehat{\varepsilon}_1 = P_{[t^{(1)}]^\perp} Y^{(1)}$$

Step k

➤ We set $X^{(k)} = P_{[t^{(k-1)}]^\perp} X^{(k-1)}$ *i.e.* the part of $X^{(k-1)}$ that was not used in the first step to explain and $Y^{(k)} = P_{[t^{(k-1)}]^\perp} Y^{(k-1)} = \widehat{\varepsilon}_{k-1}$.

➤ Calculation of the $k-th$ PLS component $t^{(k)}$: for $w \in \mathbb{R}^p$

$$t^{(k)} = \underset{t=X^{(k)}w,\ \|w\|_2=1}{\arg\max} < t, Y^{(k)} >$$

➤ Then, regress $Y^{(k)}$ on $t^{(k)}$ such that $Y^{(k)} = r_k t^{(k)} + \widehat{\varepsilon}_k$,

- with $r_k \in \mathbb{R}$, the coefficient of the orthogonal projection on $t^{(k)}$:

$$r_k t^{(k)} = P_{[t^{(k)}]} Y^{(k)}$$

- and the residual $\widehat{\varepsilon_k}$ which corresponds to the part not explained by $t^{(k)}$

$$\widehat{\varepsilon_k} = P_{[t^{(k)}]^\perp} Y^{(k)}$$

Comments:

☛ Here again, the hyperprameter k has to be calibrated (see section 7.6.).

☛ The PLS components are orthogonal to each other by construction, indeed $t^{(j)}$ is a linear combination of the columns of $X^{(j)}$

$$X^{(j)} = P_{[t^{(j-1)}]^\perp} X^{(j-1)}$$

Therefore, $X^{(j)} \in \text{span}(t^{(1)}, \cdot, t^{(j-1)})$ and $t^{(j)} \perp \{t^{(1)}, \cdot, t^{(j-1)}\}$.

☛ Note that the $t^{(j)}$ are chosen according to the maximum empirical correlation with Y, the variables X and Y being centered.

Theorem 1 *The PLS model is written as follows :*

$$\begin{aligned} Y &= P_{[t^{(1)}]} Y^{(1)} + \cdots + P_{[t^{(k)}]} Y^{(k)} + \widehat{\varepsilon_k} \\ &= r_1 t^{(1)} + \cdots + r_k t^{(k)} + \widehat{\varepsilon_k}, \end{aligned}$$

where $\widehat{\varepsilon_k} = P_{[t^{(k}]^\perp} Y^{(k)} = P_{\text{span}(t^{(1)}, \cdot, t^{(j-1)})} Y$.

Proof : Admited or let in exercice.

Differents packages/functions

- Function `plsr` with the package `pls`.
- Function `train( ,method='pls')` of the package `caret`.

Here the regressors and/or the response variable are re-centered and/or standardizeds by adding the following option in the command `preProc = c("center", "scale")`.

7.6. Cross validation

7.2.1. Holdout methout

If we use the full dataset to fit $\widehat{\beta}$, we will have no guarantee as to the quality of prediction $X_{new}\widehat{\beta}$ on a new dataset. It would not be smart to use the data twice (once to adjust the model and again to estimate the accuracy of the predictions) as this would reward the overfitting.

If, moreover, fitting $\widehat{\beta}$ requires to choose hyperparameters (tuning parameter λ for example), again it would be necessary to have a third data set to fix these hyperparameters.

Ideally, we would have 3 datasets

- A *train*-dataset for learning/training $(x_l^T, y_l)_l$. To compute $\widehat{\beta}_l$ or $(\widehat{\beta}_l(\lambda))_\lambda$.
- A *validation*-dataset to validate/fix the "optimal" hyperparameters, say λ_{opt} according to a criterion and find the optimal procedure $\widehat{\beta}_l(\lambda_{opt})$.
- A *test*-dataset to evaluate how well $X_t\widehat{\beta}_l(\lambda_{opt})$ predicted the observations Y_t in this *test*-dataset. We valuate this accuracy by calculating a performance score of the model on the test sample, for example the predicted residual error sum of squares PRESS

Splitting the original dataset into 3 sub-samples could be the solution. Unfortunately, datas are rare and expensive. The *validation*-sample can be avoided by doing $cross - validation$ on the *train*-sample.

Holdout method

In practice, we divide the original sample $(x_i^T, y_i)_{i=1,\cdots,n}$ into two sub-samples : the *train* and the *test*.

- A *training*-sample of size n_{learn} for learning/training $(x_l^T, y_l)_l$: to build the model and to estimate the parameters (and the hyperparamter) say $\widehat{\beta}_l$ (or $\widehat{\beta}_l(\lambda_{opt})$).
- A *test*-sample of size n_{test} for validation $(x_t^T, y_t)_t$: It is an independent sample from the same population as the *training*-sample. The *test*-sample allows to validate that the estimated model responds to validation as well as during learning. The error is estimated by calculating a performance score of the model on the test sample, for example the predicted residual error sum of squares PRESS. Set $\widehat{Y}_t = X_t\widehat{\beta}_l(= X_t\widehat{\beta}_l(\lambda_{opt}))$, then

$$\text{PRESS} = ||\widehat{Y}_t - Y_t||^2 = ||X_t\widehat{\beta}_l - Y_t||^2 = \sum_{i=1}^{n_{test}} (\widehat{Y}_i - Y_i)^2,$$

where the estimation of β have been calculated with the *train*-sample dataset.

Comments:

- ☛ When dividing the sample by 2, the proportion is (70%,30%).
- ☛ When dividing the sample by 3, the proportion is (60%,20%,20%).
- ☛ Careful attention should be paid to the choice of sub-samples, choosing them in an intelligent way so that each sample represents at best the entire sample.

7.2.2. Cross-validation

• If we need to select the "optimal" hyperparamter(s), we can do it on the *train*-data by cross-validation. And, for a upper bound $\overline{\lambda}$ of λ, the optimal λ_{opt} is the one that will minimize the criterion

$$\lambda_{opt} = \arg \min_{\lambda \in [0, \overline{\lambda}]} \text{Crit}(\lambda).$$

K-fold cross validation

• We divide the *train*-sample $(x_i^T, y_i)_{i=1,\cdots,n_{train}}$ into k sub-samples $(x_i^T, y_i)_{i=1,\cdots,n_{val_k}}$ of size n_{val_k}.

1. Then, we select one (k sub-sample of size n_{val_k}) among the K sub-samples as the *validation*-sample.
2. The remaining $K-1$ samples constitute the *training*-sample.
3. The performance score is calculated by the PRESS

$$\text{PRESS}^k(\lambda) = \|\widehat{Y}_{v_k}(\lambda) - Y_{v_k}\|^2 = \|X_{v_k}\widehat{\beta}_l(\lambda) - Y_{t_v}\|^2 = \sum_{i=1}^{n_{val_k}} (\widehat{Y}_i(\lambda) - Y_i)^2,$$

where $\widehat{Y}_t = X_t\widehat{\beta}_l(\lambda)$ and $\widehat{\beta}_l(\lambda)$ have been calculated with the $(k-1)$ *train*-sample dataset.

• Operations (1., 2., and 3.) are repeated by selecting another *validation*-sample among the $K-1$ remaining samples that have not yet been used to validate the model. Then, we repeat the full operation, K times (each sub-sample has been used exactly once).

• The average of the K-PRESS is finally calculated to estimate the prediction error

$$\text{Crit}(\lambda) := \frac{1}{K}\sum_{k=1}^{K} \text{PRESS}^k(\lambda) = \frac{1}{K}\sum_{k=1}^{K} \|X_{v_k}\widehat{\beta}_l(\lambda) - Y_{v_k}\|^2 = \frac{1}{K}\sum_{k=1}^{K}\sum_{i=1}^{n_{val_k}} (\widehat{Y}_i(\lambda) - Y_i)^2$$

Leave-one-out cross validation (LOOCV)

This is a particular case of the second method where $k = n_{train}$, *i.e.* we learn with $(n_{train} - 1)$ observations then we validate the model on the n_{train}-*th* observation and we repeat the full operation n_{train} times and

$$\text{Crit}(\lambda) := \text{PRESS}_{CV}(\lambda) = \frac{1}{n_{train}} \sum_{i=1}^{n_{train}} (\widehat{Y}_i(\lambda) - Y_i)^2 = \frac{1}{n_{train}} \sum_{i=1}^{n_{train}} (x_i^T \widehat{\beta}_{(-i)}(\lambda) - Y_i)^2,$$

where $\widehat{\beta}_{(-i)}(\lambda)$ have been calculated with the *train*-sub-sample where the observation (x_i^T, Y_i) has been removed.

Comment:

☛ If we denote by $H(\lambda) := X_t(X_t^T X_t + \lambda \mathbb{I}_p)^{-1} X_t^T$ the projector in Ridge setting, it comes that

$$x_i^T \widehat{\beta}_{(-i)}(\lambda) - Y_i = \frac{x_i^T \widehat{\beta}(\lambda) - Y_i}{1 - H_{ii}} = \frac{\widehat{\varepsilon}_i(\lambda)}{1 - H_{ii}},$$

where $\widehat{\beta}(\lambda)$ have been compute with all the *train*-sample and H_{ii} is the $i - th$ diagonal element of H. Then, we can write

$$\text{PRESS}_{CV}(\lambda) = \frac{1}{n_{train}} \sum_{i=1}^{n_{train}} \left(\frac{\widehat{\varepsilon}_i(\lambda)}{1 - H_{ii}(\lambda)} \right)^2.$$

☛ The standard (*LOOCV*) gives a very precise estimate of the optimal parameters. Unfortunately, its implementation is limited in practice by a high numerical cost. We prefer the *generalized cross validation* (Carefull, nothing to do with *(LOOCV)*) where $1 - H_{ii}(\lambda)$ is replaced by $(1 - \text{Tr}(H(\lambda))/n)$.

Generalized cross-validation (GCV)

• In **our Ridge setting**, we calculate the estimators $\left(\widehat{\beta}_t^R(\lambda)\right)_\lambda$ with this *train*-sample $(x_t^T, y_t)_t$ of size n_{train} such that

$$\widehat{\beta}_t^R(\lambda) = (X_t^T X_t + \lambda \mathbb{I}_p)^{-1} X_t^T Y_t \quad \text{and} \quad \widehat{Y}_t^R(\lambda) = H(\lambda) Y_t$$

with $H(\lambda) = X_t(X_t^T X_t + \lambda \mathbb{I}_p)^{-1} X_t^T$ the projector in Ridge setting for the *train*-sample.

• We define the Generalized cross-validation criterion as follows

$$\text{GCV}(\lambda) := \frac{\|Y_t - \widehat{Y}_t^R(\lambda)\|^2 / n_{train}}{(\text{Tr}(\mathbb{I}_n - H(\lambda))/n_{train})^2} = \frac{1}{n_{train}} \sum_{i=1}^{n_{train}} \left(\frac{Y_i - \widehat{Y}_i^R(\lambda)}{1 - \text{Tr}(H(\lambda))/n_{train}} \right)^2$$