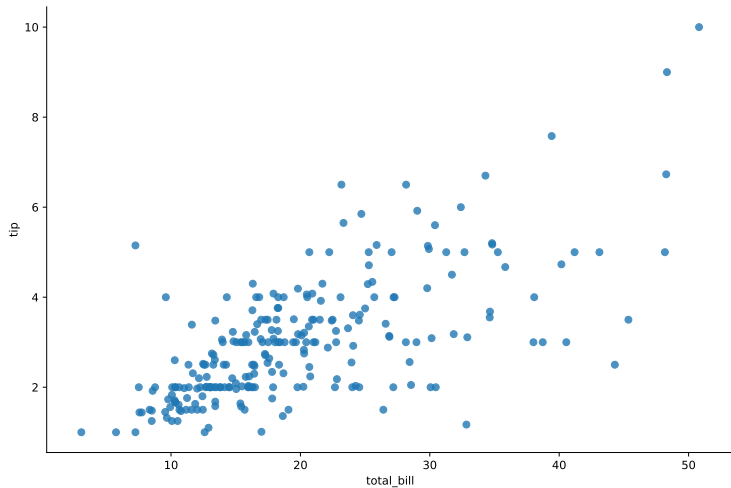


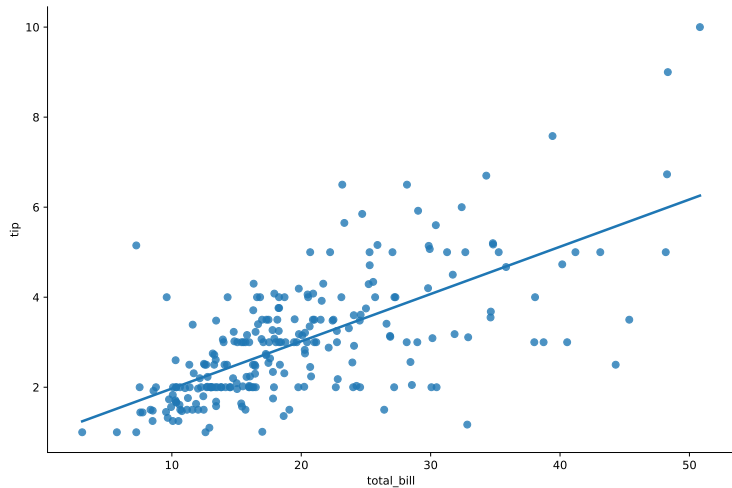
SD TSIA 204
Linear Models
Intro to linear models

Ekhiñe Irurozki
Télécom Paris

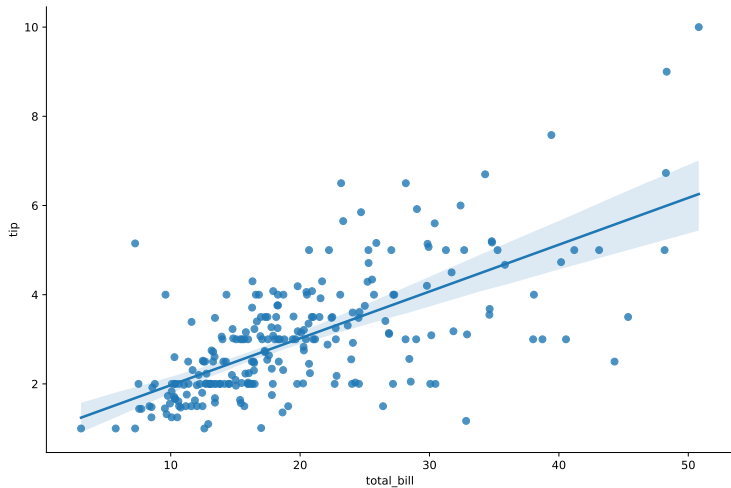
A 2D starting example



A 2D starting example



A 2D starting example



Notation interpretation

- ▶ $n = 244$
- ▶ $p = 1$
- ▶ y_i : tip let by the i -th customer
- ▶ x_i : total bill paid by the i -th customer
- ▶ y : the observation is the tips, dependent variable
- ▶ x : the feature/covariate, price of the bill, independent variable

Linear model / Linear regression hypothesis : assume that the price of the bill and the tip let are linearly correlated

Exo : use `describe()` from `Pandas` to get a rough data summary

Three questions to be covered : modeling, learning and predicting

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
# Generate example data
np.random.seed(42)
X = np.random.rand(20, 1)*10 # Independent variable
y = 2 * X + 3 + np.random.randn(20, 1) # Dependent variable
# Fit linear regression model
model = LinearRegression()
model.fit(X, y)
# Predict y values using the model
X_new = np.linspace(0, 10, 100).reshape(-1, 1)
y_pred = model.predict(X_new)
# Create a scatter plot of the data points
plt.scatter(X, y, label='Data Points')
# Plot the linear regression line
plt.plot(X_new, y_pred, color='red', label='Linear Regression Line')
plt.xlabel('X')
plt.ylabel('y')
```

Modeling I, the 1D case

Given a sample : (y_i, x_i) , for $i = 1, \dots, n$

Linear model or linear regression hypothesis assume :

$$y_i \approx \theta_0^* + \theta_1^* x_i$$

Model coefficients

- ▶ θ_0^* : intercept (unknown)
- ▶ θ_1^* : slope (unknown)

Rem: both parameters are unknown from the statistician

Data

- ▶ y is an **observation** or a variable to explain
- ▶ x is a **feature** or a covariate

Modeling II

Probabilistic model. Let us give a precise meaning to the sign \approx :

$$y_i = \theta_0^* + \theta_1^* x_i + \varepsilon_i,$$

$$\varepsilon_i \stackrel{i.i.d}{\sim} \varepsilon, \text{ for } i = 1, \dots, n$$

$$\mathbb{E}(\varepsilon) = 0$$

where i.i.d. means “independent and identically distributed”

Interpretation : $\varepsilon_i = y_i - \theta_0^* - \theta_1^* x_i$: represent the error between the theoretical model and the observations, represented by random variables ε_i centered (often referred to as **white noise**).

Rem: motivation for the random nature of the noise – measurement noise, transmission noise, in-population variability, etc.

Modeling III

$$y_i = \theta_0^* + \theta_1^* x_i + \varepsilon_i$$

We call

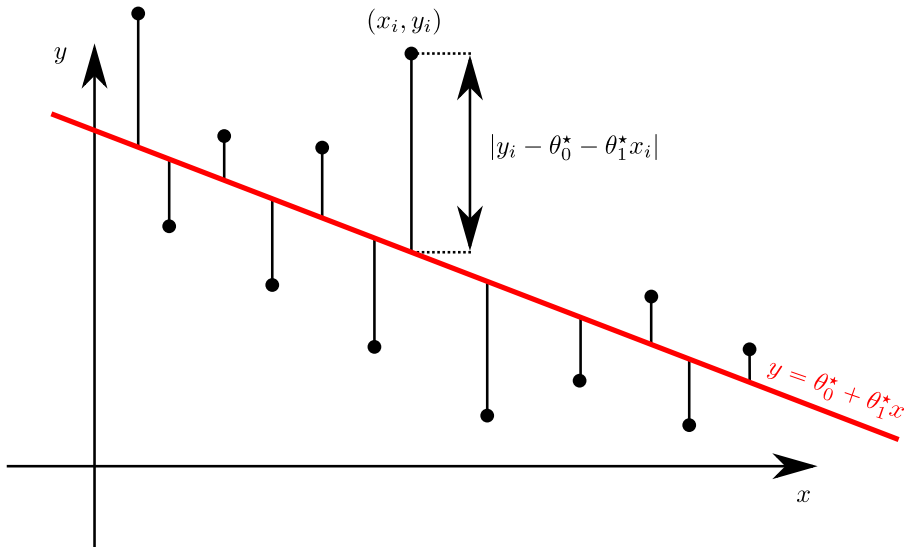
- **intercept** the scalar θ_0^* (■ ■ : *ordonnée à l'origine*)
- **slope** the scalar θ_1^* (■ ■ : *pente*)

Our **goal in the learning stage** is to estimate θ_0^* and θ_1^* (unknown) by $\hat{\theta}_0$ and $\hat{\theta}_1$ relying on observations (y_i, x_i) for $i = 1, \dots, n$

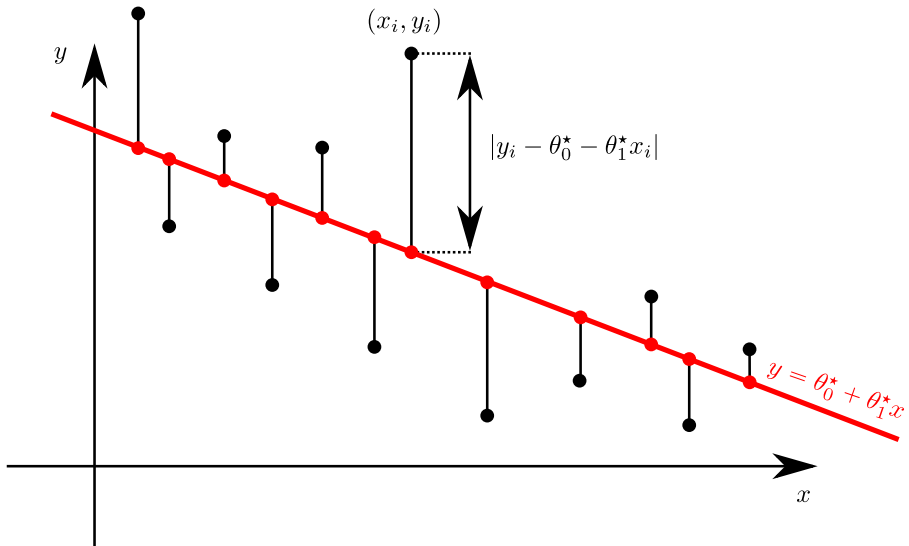
Rem: The “hat” notation is classical in statistics for referring to estimators

In **prediction time** $\hat{y}_i = \hat{\theta}_0 + \hat{\theta}_1 x_i$

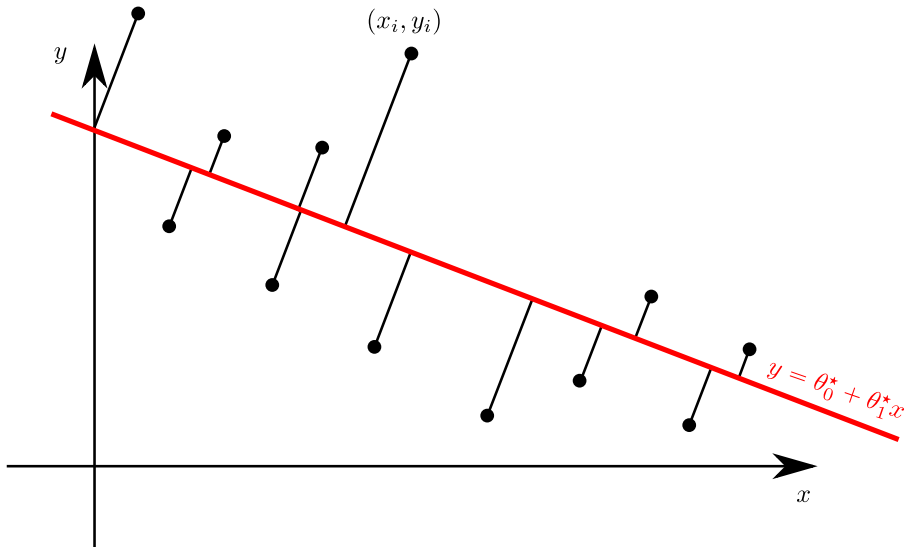
Least squares : visualization



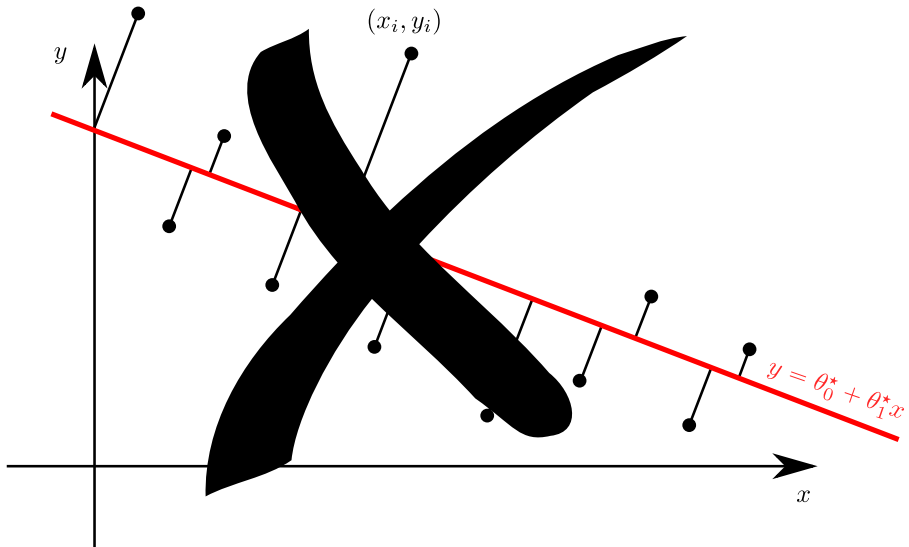
Least squares : visualization



(Total) Least squares : visualization



(Total) Least squares : visualization



Learning : mathematical formulation of Least squares

The **least squares** estimator is defined as :

$$(\hat{\theta}_0, \hat{\theta}_1) \in \operatorname{argmin}_{(\theta_0, \theta_1) \in \mathbb{R}^2} \sum_{i=1}^n (y_i - \theta_0 - \theta_1 x_i)^2$$

- ▶ Differentiate between θ^* , θ and $\hat{\theta}$!!!!
- ▶ it is also referred to as “ordinary least squares” (OLS)
- ▶ an original motivation for the squares is computational : first order conditions only require solving a linear system
- ▶ a solution always exists : minimizing a **coercive** continuous function
(coercive : $\lim_{\|x\| \rightarrow +\infty} f(x) = +\infty$)

Rem: write « $\in \operatorname{argmin}$ » as long as you do not know if the solution is unique

Least square authorship (controversial)



Figure – Adrien-Marie Legendre and Carl Friedrich Gauss

Historical / robust detour

The **least absolute deviation** (LAD) estimator reads :

$$(\hat{\theta}_0, \hat{\theta}_1) \in \operatorname{argmin}_{(\theta_0, \theta_1) \in \mathbb{R}^2} \sum_{i=1}^n |y_i - \theta_0 - \theta_1 x_i|$$

Rem: hard to compute without computer ; requires an optimization solver for non-smooth function (or a Linear Programming solver)

Rem: more robust to outliers (■ ■ : *données aberrantes*)

Least absolute deviation authorship

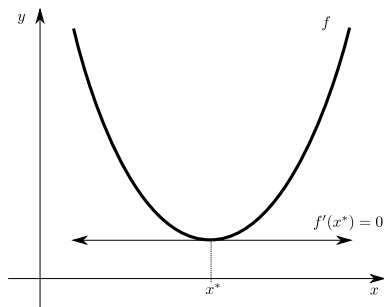


Figure – Ruđer Josip Bošković and Pierre-Simon de Laplace

Existence and uniqueness of the solution

Existence of a Local minimum : first order condition

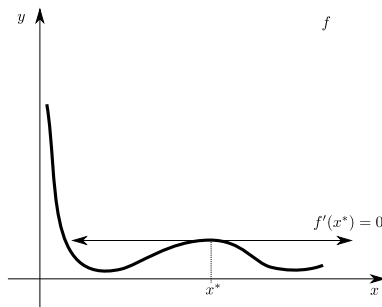
Fermat's rule Theorem If f is differentiable, then at a local minimum x^* the gradient of f vanishes at x^* , *i.e.* $\nabla f(x^*) = 0$.



Existence and uniqueness of the solution

Existence of a Local minimum : first order condition

Fermat's rule Theorem If f is differentiable, then at a local minimum x^* the gradient of f vanishes at x^* , *i.e.* $\nabla f(x^*) = 0$.

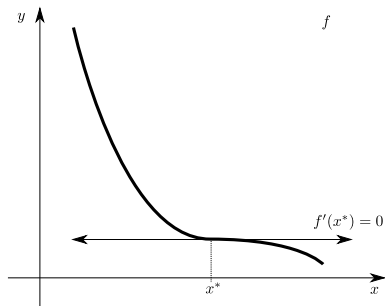


Rem: sufficient condition when f is strongly convex!

Existence and uniqueness of the solution

Existence of a Local minimum : first order condition

Fermat's rule Theorem If f is differentiable, then at a local minimum x^* the gradient of f vanishes at x^* , *i.e.* $\nabla f(x^*) = 0$.



Rem: sufficient condition when f is strongly convex!

The Hessian Matrix and Gradients

The **gradient** ∇f is a vector of first-order partial derivatives :

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$$

The **Hessian Matrix** \mathbf{H} of f is a square matrix of second-order partial derivatives :

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

The minimizer is unique when f is strictly convex

f is quadratic $\implies f$ is convex $\implies \nabla^2 f(\hat{\boldsymbol{\theta}})$ positive semi definite.

$\nabla^2 f(\hat{\boldsymbol{\theta}})$ positive definite \implies the minimizer is unique

Back to least squares

$$\hat{\boldsymbol{\theta}} = (\hat{\theta}_0, \hat{\theta}_1) \in \operatorname{argmin}_{(\theta_0, \theta_1) \in \mathbb{R}^2} \frac{1}{2} \sum_{i=1}^n (y_i - \theta_0 - \theta_1 x_i)^2$$

For least squares, minimize the function of two variables :

$$f(\theta_0, \theta_1) = f(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^n (y_i - \theta_0 - \theta_1 x_i)^2$$

First order condition / Fermat's rule :

$$\begin{cases} \frac{\partial f}{\partial \theta_0}(\hat{\boldsymbol{\theta}}) = \sum_{i=1}^n (y_i - \hat{\theta}_0 - \hat{\theta}_1 x_i) = 0 \\ \frac{\partial f}{\partial \theta_1}(\hat{\boldsymbol{\theta}}) = \sum_{i=1}^n (y_i - \hat{\theta}_0 - \hat{\theta}_1 x_i) x_i = 0 \end{cases}$$

Calculus continued

Usual mean notation : $\bar{x}_n = \frac{1}{n} \sum_{i=1}^n x_i$ and $\bar{y}_n = \frac{1}{n} \sum_{i=1}^n y_i$

With that, Fermat's rule states (dividing by n) :

$$\begin{cases} \frac{\partial f}{\partial \theta_0}(\hat{\theta}) = \sum_{i=1}^n (y_i - \hat{\theta}_0 - \hat{\theta}_1 x_i) = 0 \\ \frac{\partial f}{\partial \theta_1}(\hat{\theta}) = \sum_{i=1}^n (y_i - \hat{\theta}_0 - \hat{\theta}_1 x_i) x_i = 0 \end{cases}$$

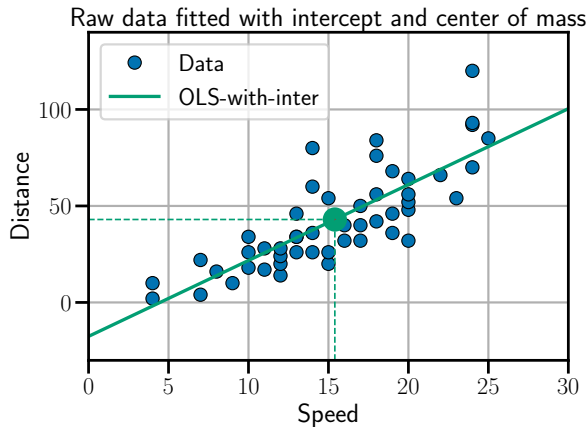
\Leftrightarrow

$$\begin{cases} \hat{\theta}_0 = \bar{y}_n - \hat{\theta}_1 \bar{x}_n & \text{(CNO1)} \\ \hat{\theta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x}_n)(y_i - \bar{y}_n)}{\sum_{i=1}^n (x_i - \bar{x}_n)^2} & \text{(CNO2)} \end{cases}$$

Exo : Show that the solution to the OLS is unique iff $Var(x) \neq 0$

Center of gravity and interpretation

$$(\text{CNO1}) \Leftrightarrow (\bar{x}_n, \bar{y}_n) \in \{(x, y) \in \mathbb{R}^2 : y = \hat{\theta}_0 + \hat{\theta}_1 x\}$$



► $\overline{speed} = 15.4$

► $\overline{dist} = 42.98$

► $\hat{\theta}_0 = -17.579095$ intercept (negative!)

► $\hat{\theta}_1 = 3.932409$ slope

Physical interpretation : the cloud of points' center of gravity belongs to the (estimated) regression line

Vector formulation

Notation : $\mathbf{x} = (x_1, \dots, x_n)^\top$ and $\mathbf{y} = (y_1, \dots, y_n)^\top$

$$(\text{CNO2}) \Leftrightarrow \hat{\theta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x}_n)(y_i - \bar{y}_n)}{\sum_{i=1}^n (x_i - \bar{x}_n)^2}$$

$$(\text{CNO2}) \Leftrightarrow \hat{\theta}_1 = \text{corr}_n(\mathbf{x}, \mathbf{y}) \cdot \frac{\sqrt{\text{var}_n(\mathbf{y})}}{\sqrt{\text{var}_n(\mathbf{x})}}$$

where $\text{corr}_n(\mathbf{x}, \mathbf{y}) = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x}_n)(y_i - \bar{y}_n)}{\sqrt{\text{var}_n(\mathbf{x})} \sqrt{\text{var}_n(\mathbf{y})}}$

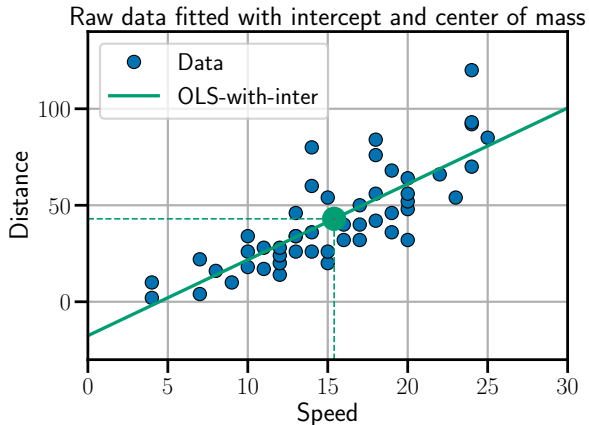
and $\text{var}_n(\mathbf{z}) = \frac{1}{n} \sum_{i=1}^n (z_i - \bar{z}_n)^2$ (for any $\mathbf{z} = (z_1, \dots, z_n)^\top$)

respectively **empirical correlation**, **empirical variances**

cars example

Braking distance for cars as a function of the speed

Line slope : $\text{corr}_n(\mathbf{x}, \mathbf{y}) \cdot \frac{\sqrt{\text{var}_n(\mathbf{y})}}{\sqrt{\text{var}_n(\mathbf{x})}} = 3.932409$.



Centering

Centered model :

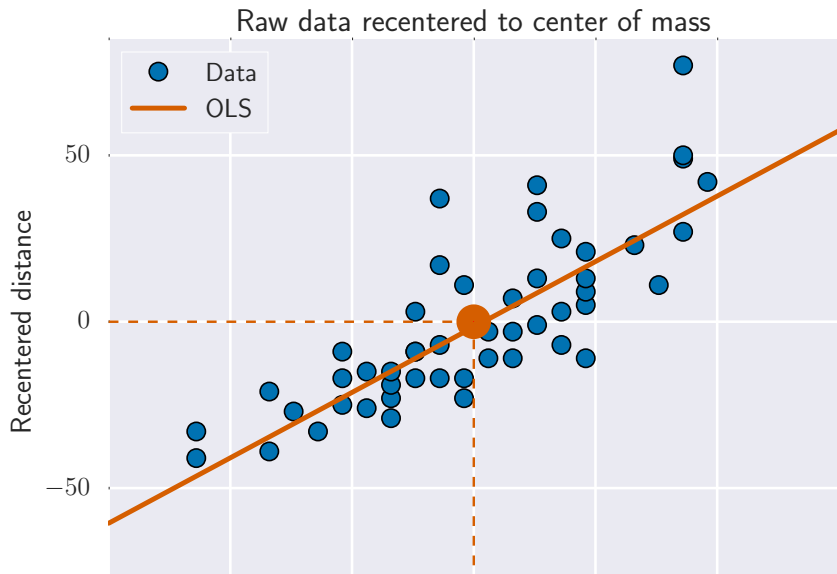
$$\text{Write for any } i = 1, \dots, n : \begin{cases} x'_i = x_i - \bar{x}_n \\ y'_i = y_i - \bar{y}_n \end{cases} \Leftrightarrow \begin{cases} \mathbf{x}' = \mathbf{x} - \bar{x}_n \mathbf{1}_n \\ \mathbf{y}' = \mathbf{y} - \bar{y}_n \mathbf{1}_n \end{cases}$$

and $\mathbf{1}_n = (1, \dots, 1)^\top \in \mathbb{R}^n$, then solving the OLS with $(\mathbf{x}', \mathbf{y}')$ leads to

$$\begin{cases} \hat{\theta}'_0 = 0 \\ \hat{\theta}'_1 = \frac{\frac{1}{n} \sum_{i=1}^n x'_i y'_i}{\frac{1}{n} \sum_{i=1}^n x'^2_i} \end{cases}$$

Rem: equivalent to choosing the cloud of points' center of mass as origin, *i.e.*
 $(\bar{x}'_n, \bar{y}'_n) = (0, 0)$

Centering (II)




Centering and interpretation

Consider the coefficient $\hat{\theta}'_1$ ($\hat{\theta}'_0 = 0$) for centered points \mathbf{y}', \mathbf{x}' , then :

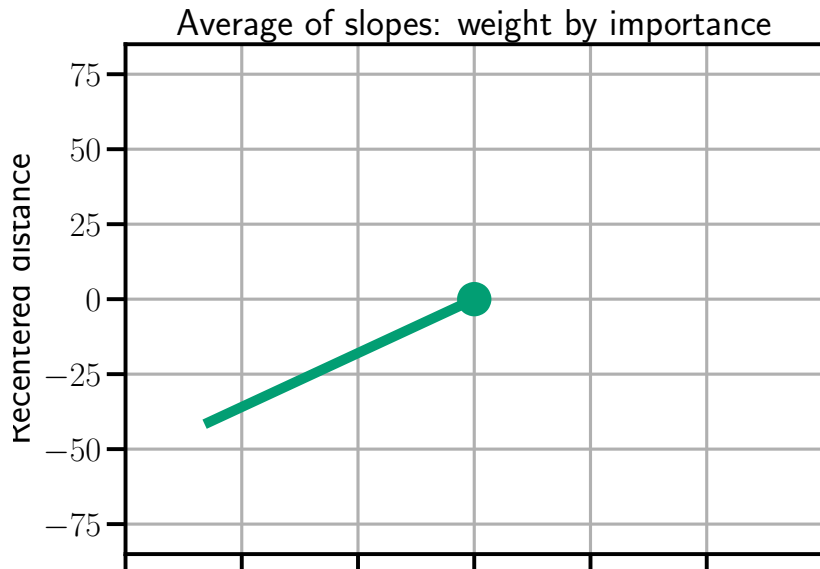
$$\hat{\theta}'_1 \in \operatorname{argmin}_{\theta_1} \sum_{i=1}^n (y'_i - \theta_1 x'_i)^2 = \operatorname{argmin}_{\theta_1} \sum_{i=1}^n x'^2_i \left(\frac{y'_i}{x'_i} - \theta_1 \right)^2$$

Interpretation : $\hat{\theta}'_1$ is a weighted average of the slopes $\frac{y'_i}{x'_i}$

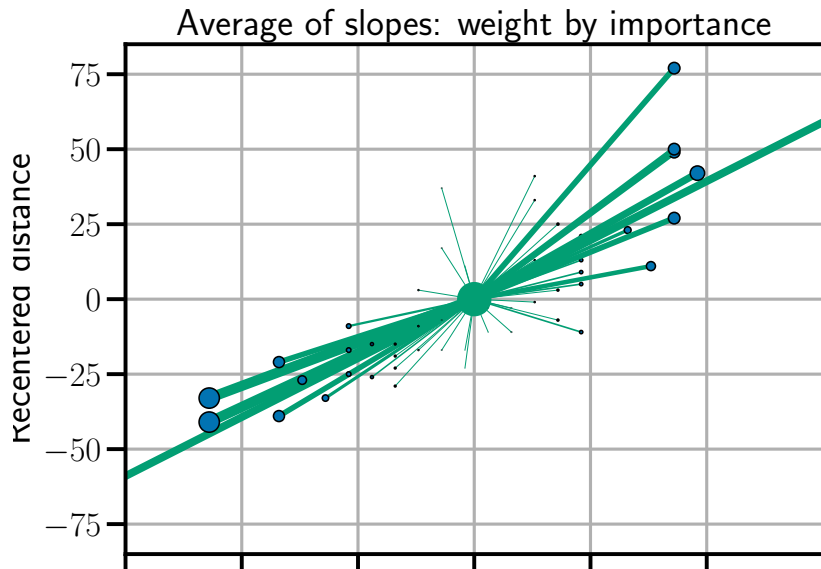
$$\hat{\theta}'_1 = \frac{\sum_{i=1}^n x'^2_i \frac{y'_i}{x'_i}}{\sum_{j=1}^n x'^2_j}$$

Influence of extreme points : weights proportional to x'^2_i ; connected to the **leverage** ( : *levier*) effect

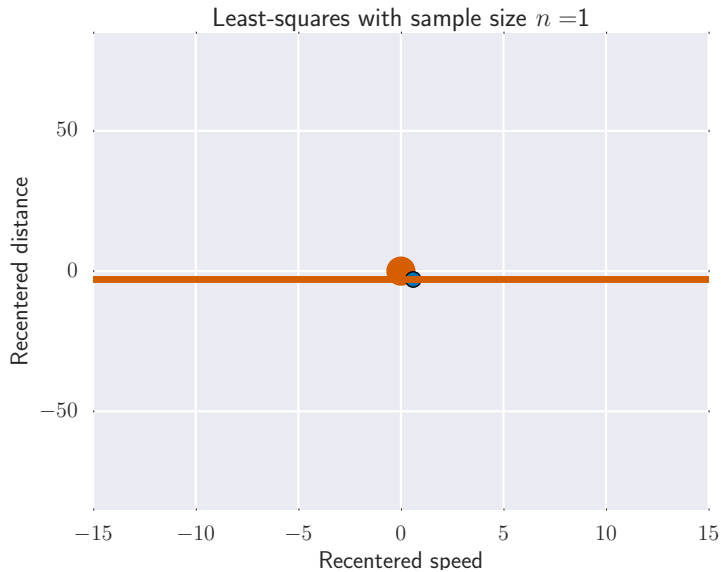
Extreme points – leverage effect



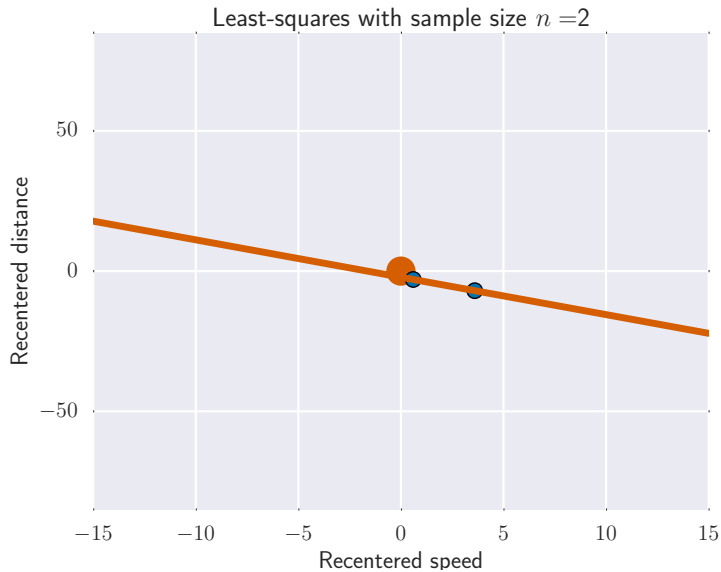
Extreme points – leverage effect



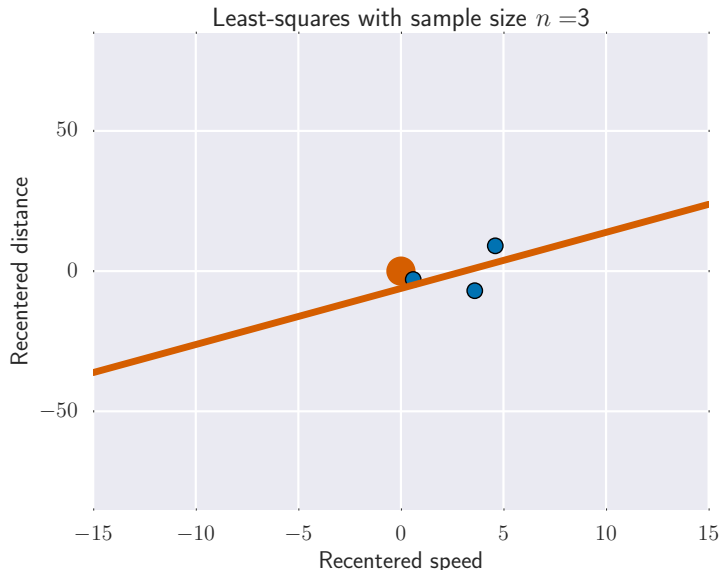
Extreme points – leverage effect (II)



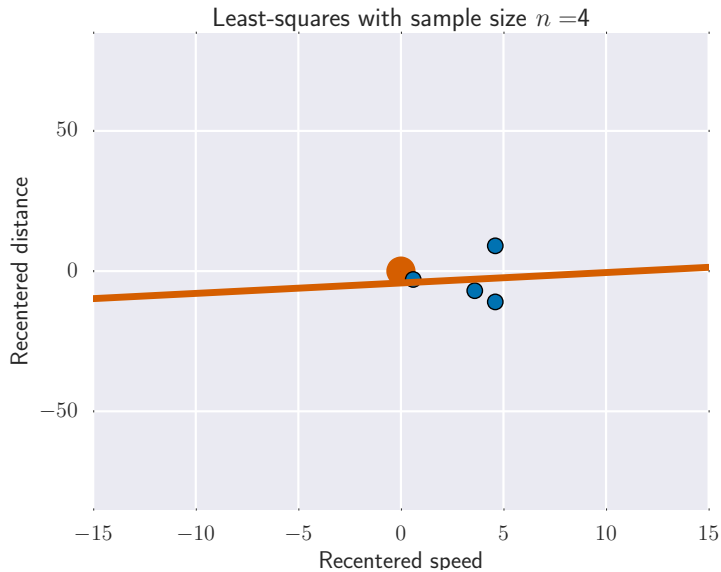
Extreme points – leverage effect (II)



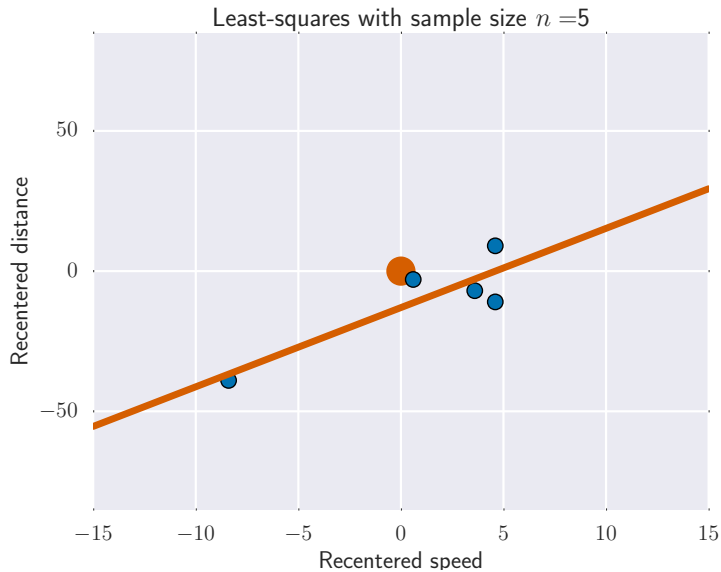
Extreme points – leverage effect (II)



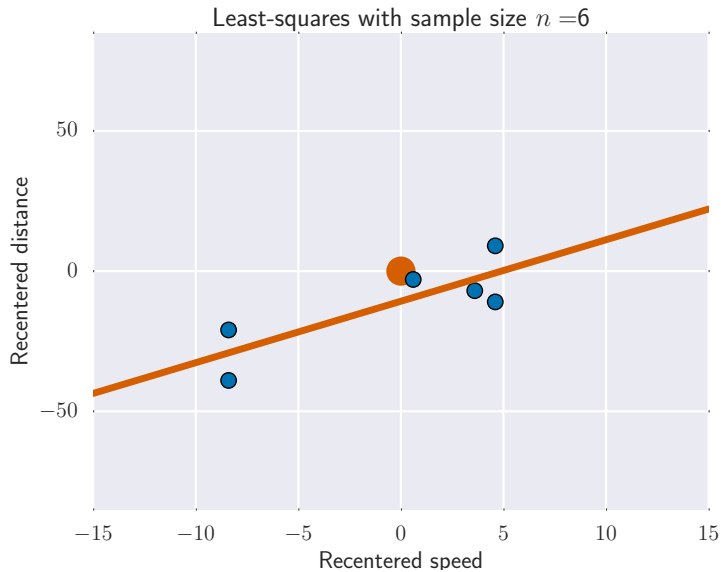
Extreme points – leverage effect (II)



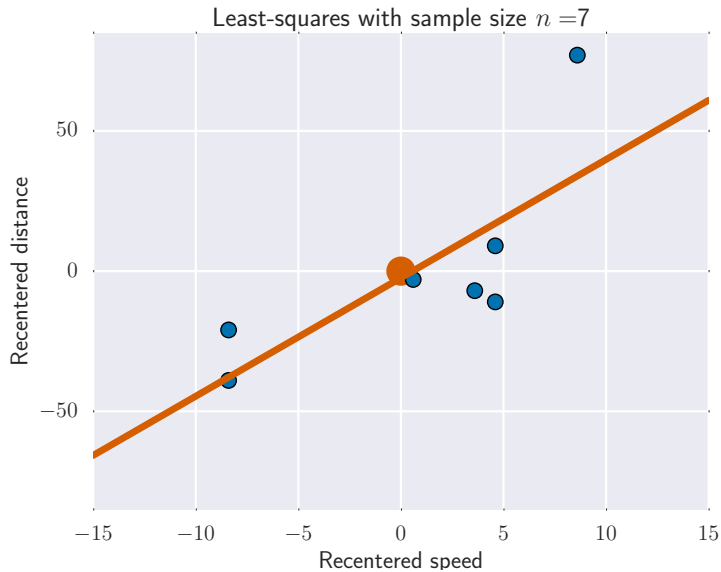
Extreme points – leverage effect (II)



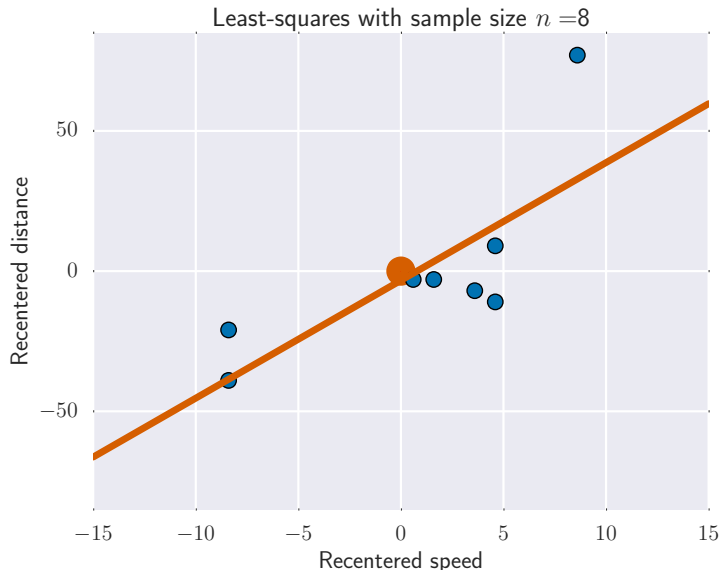
Extreme points – leverage effect (II)



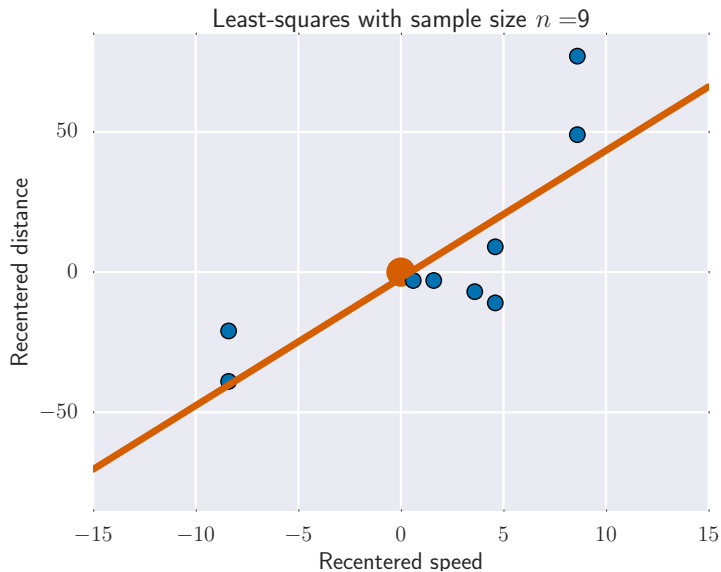
Extreme points – leverage effect (II)



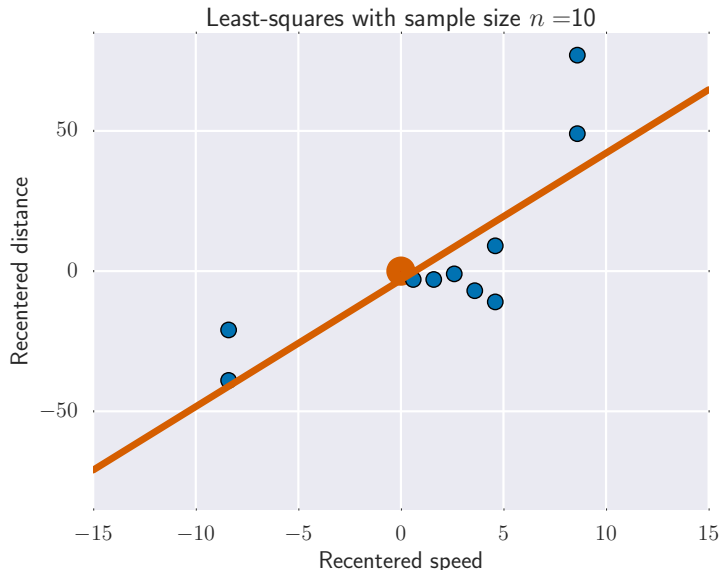
Extreme points – leverage effect (II)



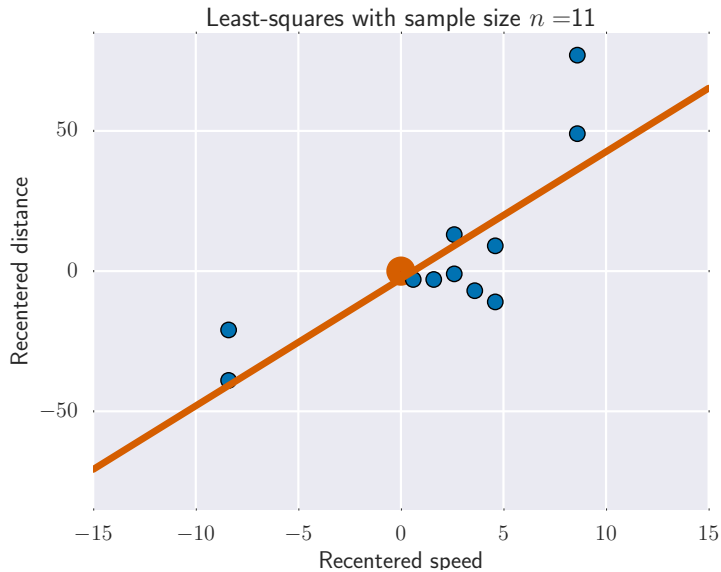
Extreme points – leverage effect (II)



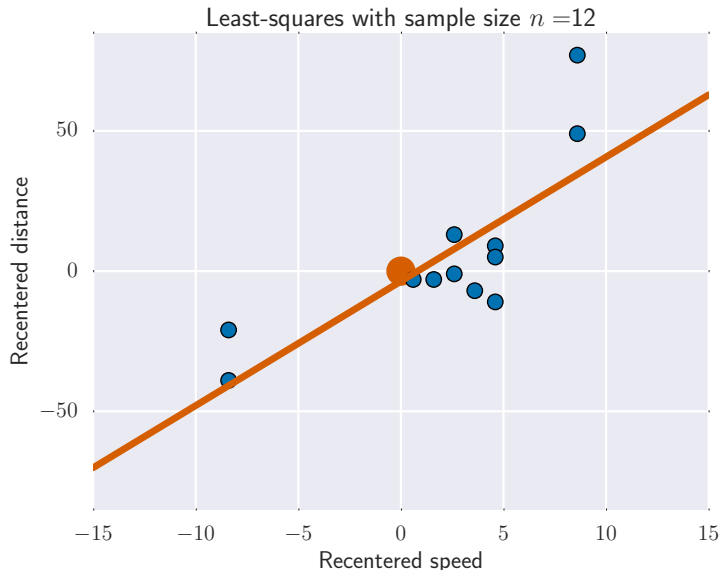
Extreme points – leverage effect (II)



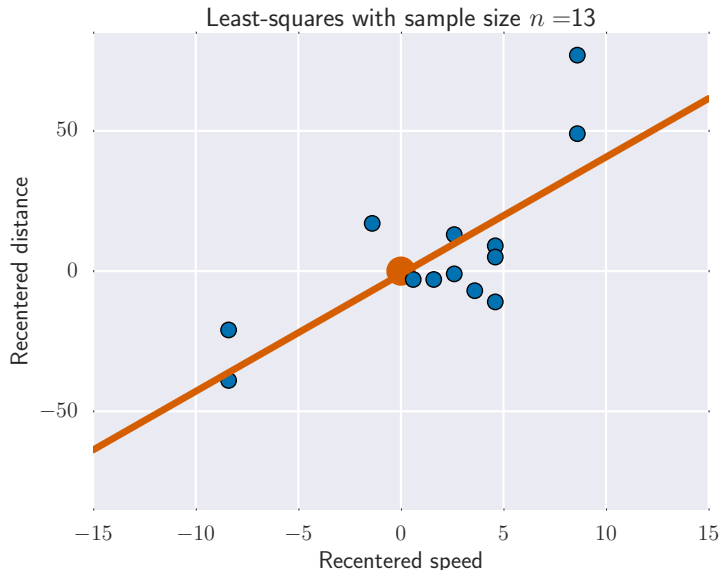
Extreme points – leverage effect (II)



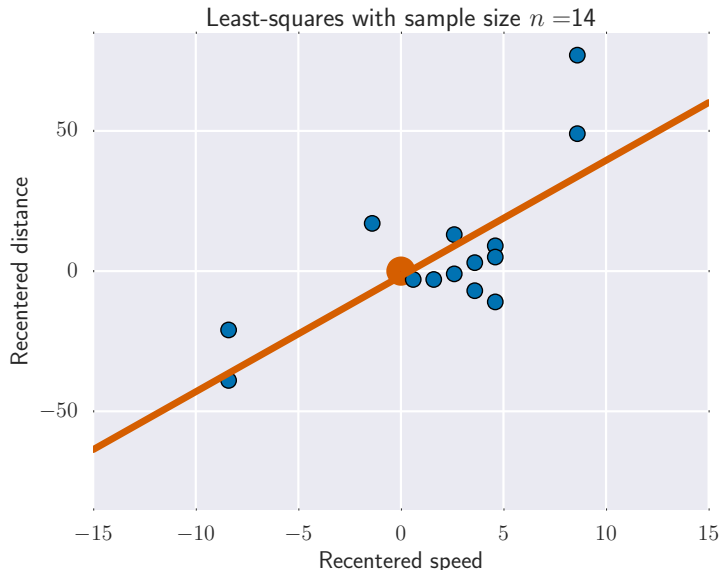
Extreme points – leverage effect (II)



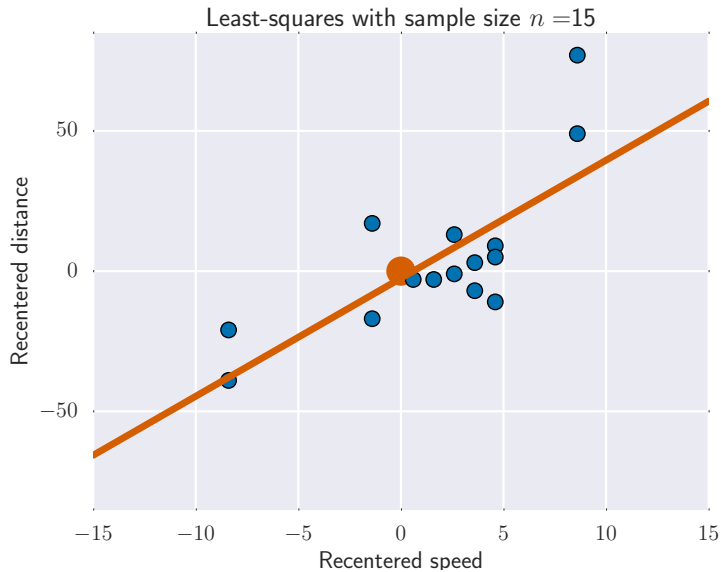
Extreme points – leverage effect (II)



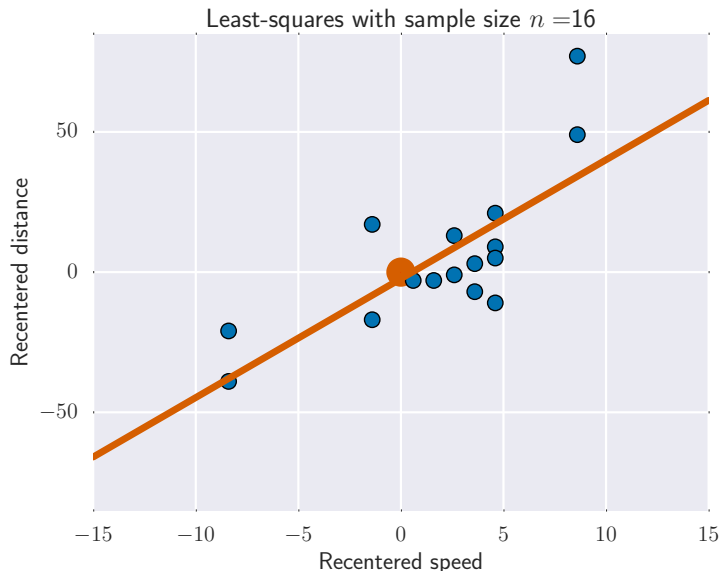
Extreme points – leverage effect (II)



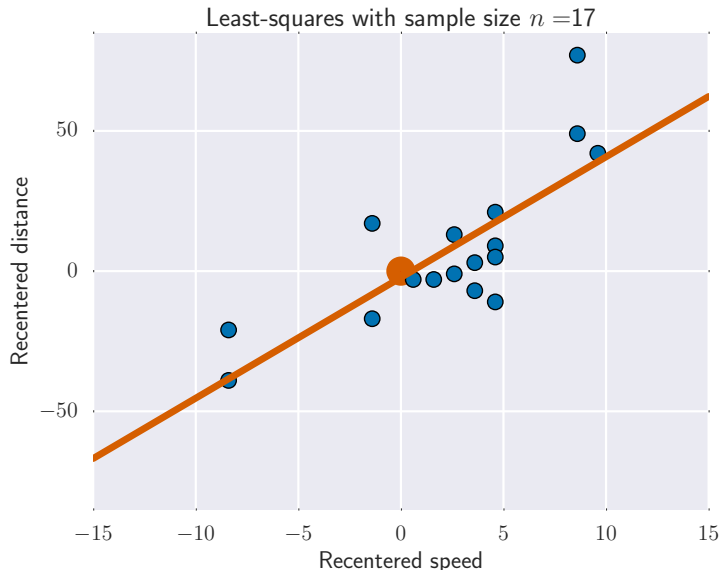
Extreme points – leverage effect (II)



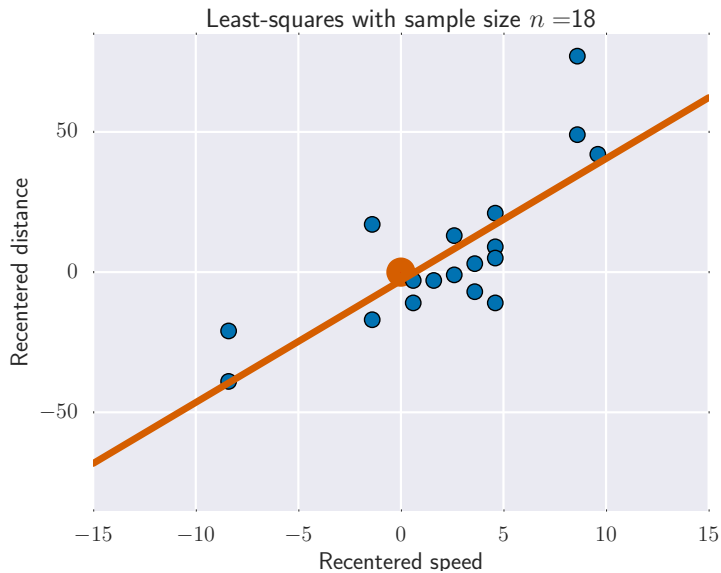
Extreme points – leverage effect (II)



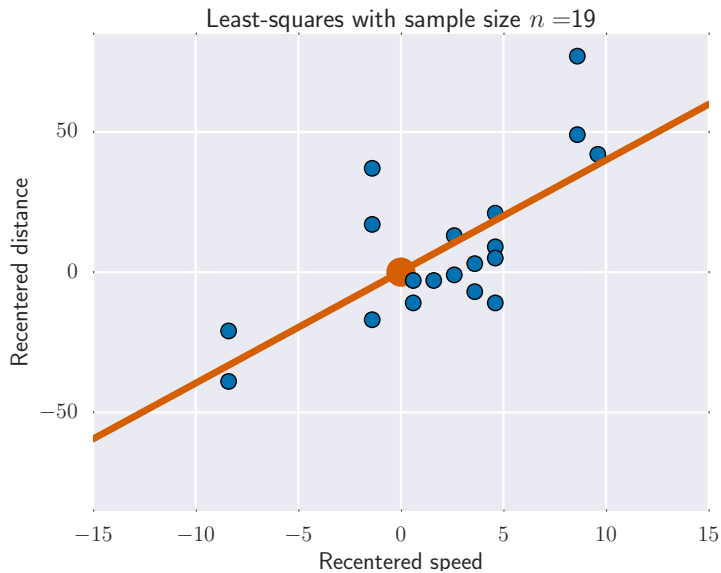
Extreme points – leverage effect (II)



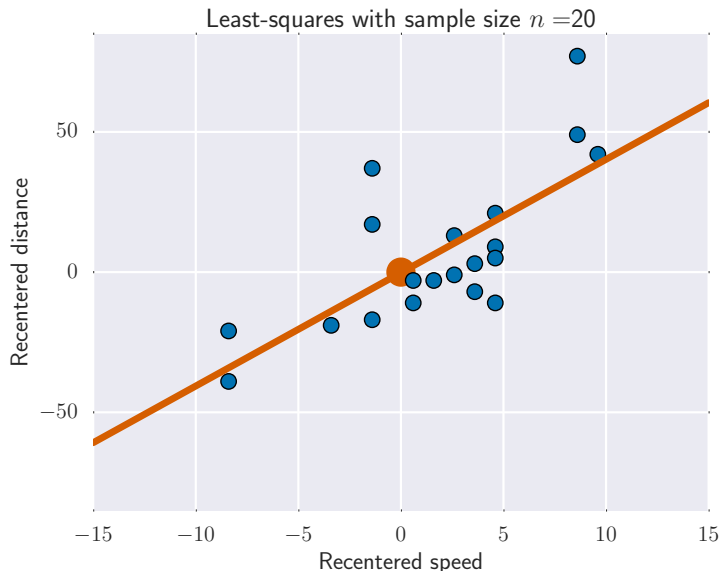
Extreme points – leverage effect (II)



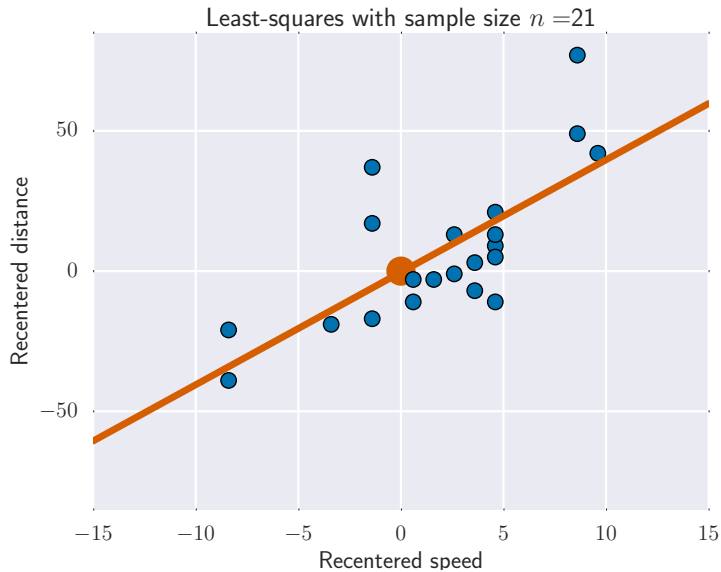
Extreme points – leverage effect (II)



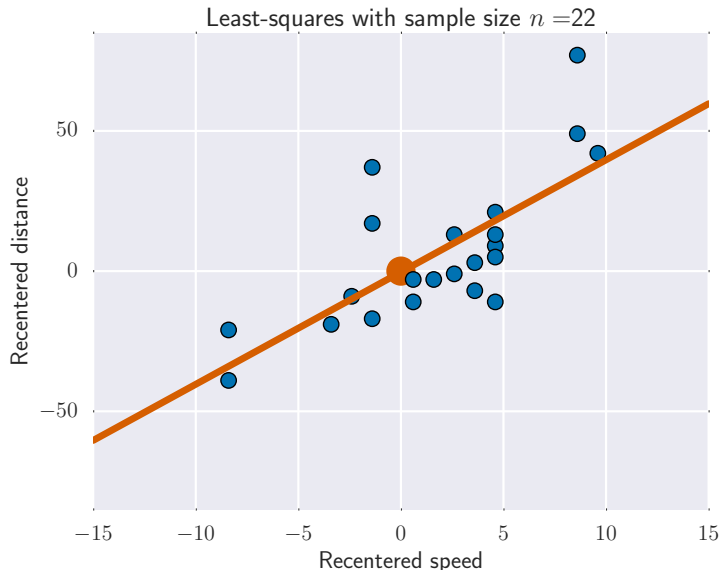
Extreme points – leverage effect (II)



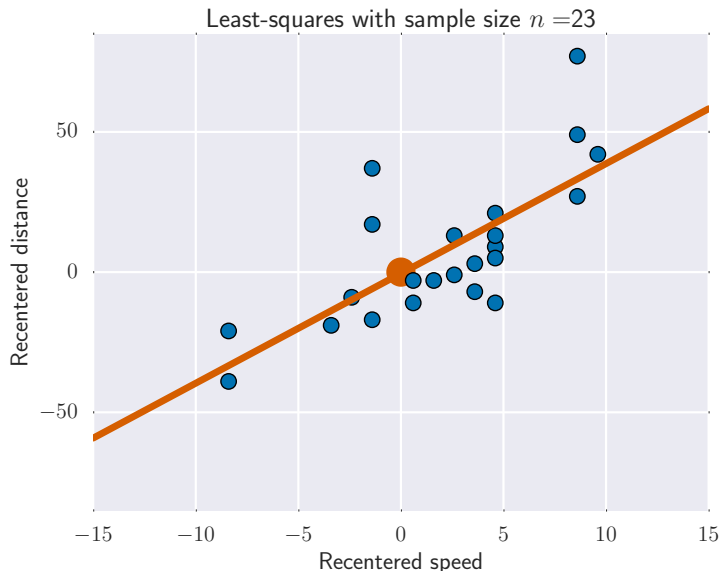
Extreme points – leverage effect (II)



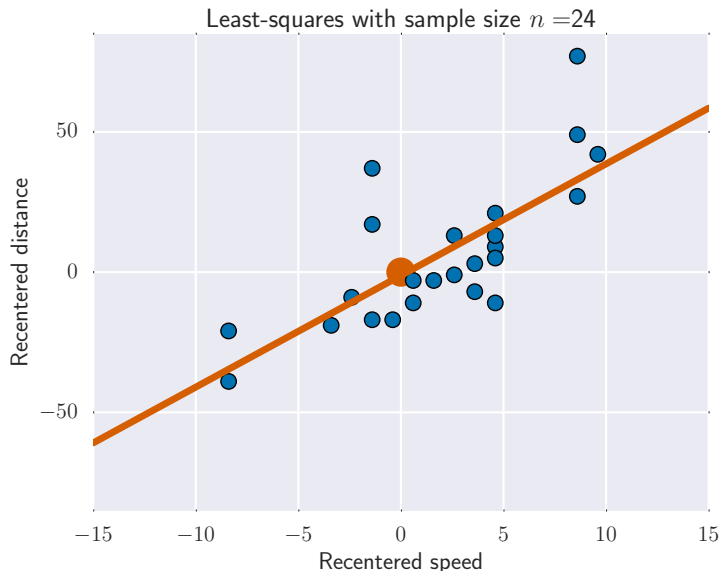
Extreme points – leverage effect (II)



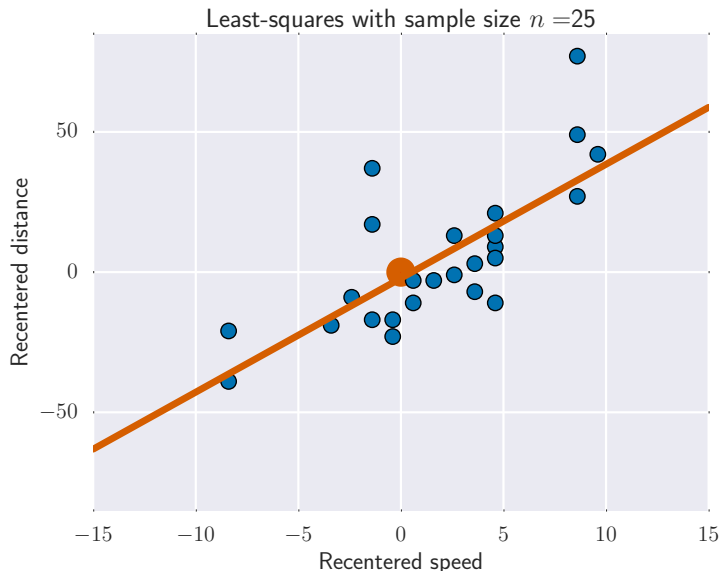
Extreme points – leverage effect (II)



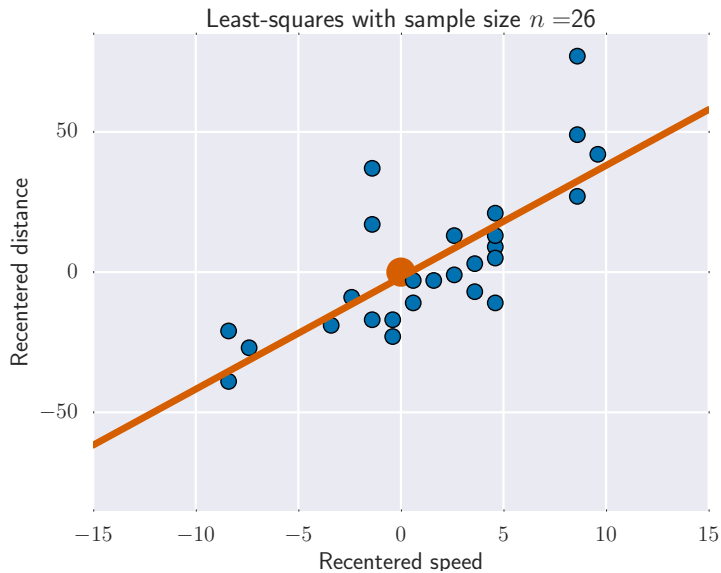
Extreme points – leverage effect (II)



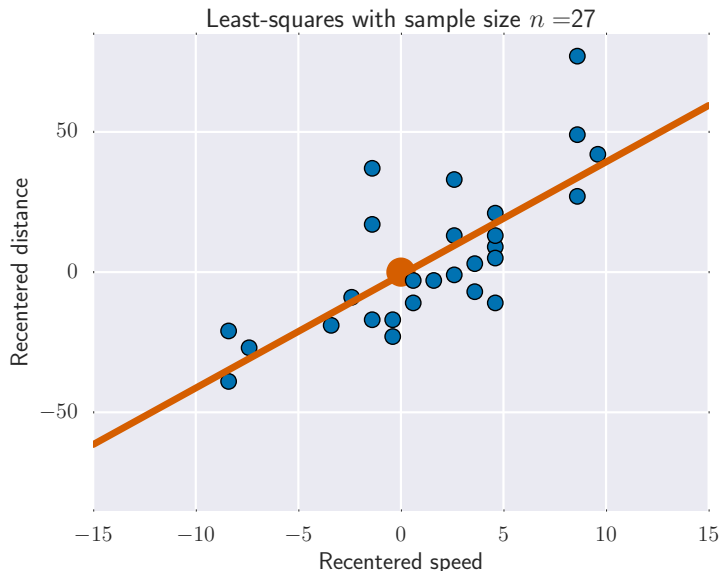
Extreme points – leverage effect (II)



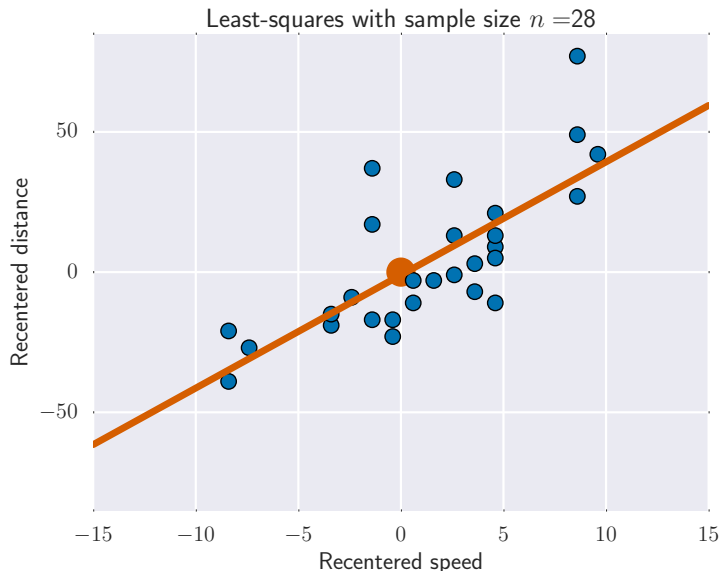
Extreme points – leverage effect (II)



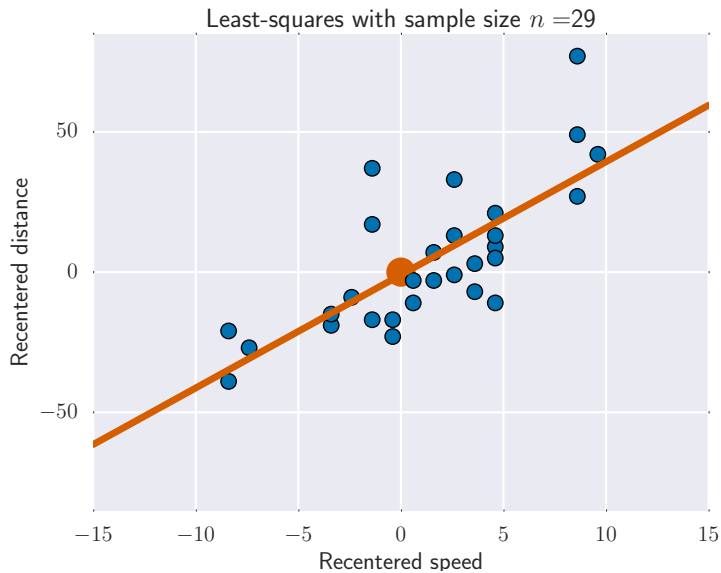
Extreme points – leverage effect (II)



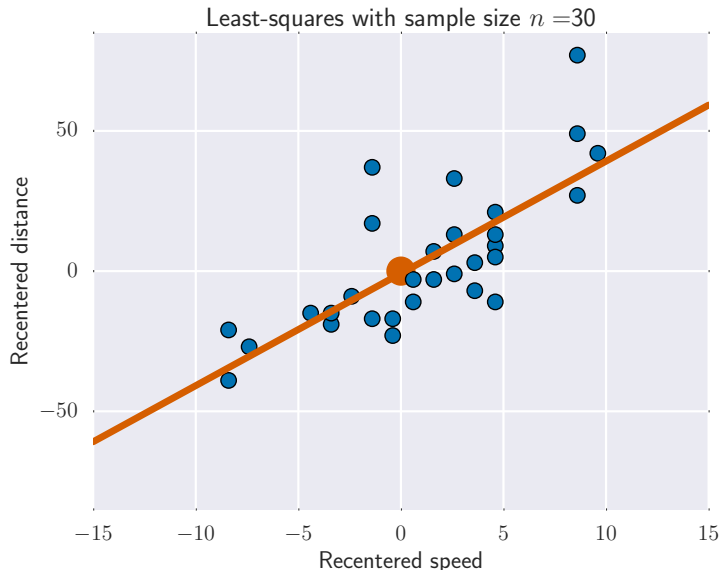
Extreme points – leverage effect (II)



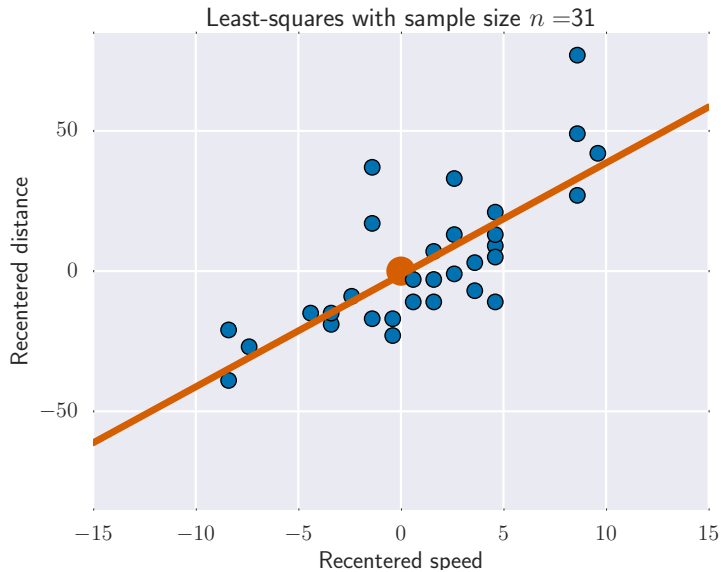
Extreme points – leverage effect (II)



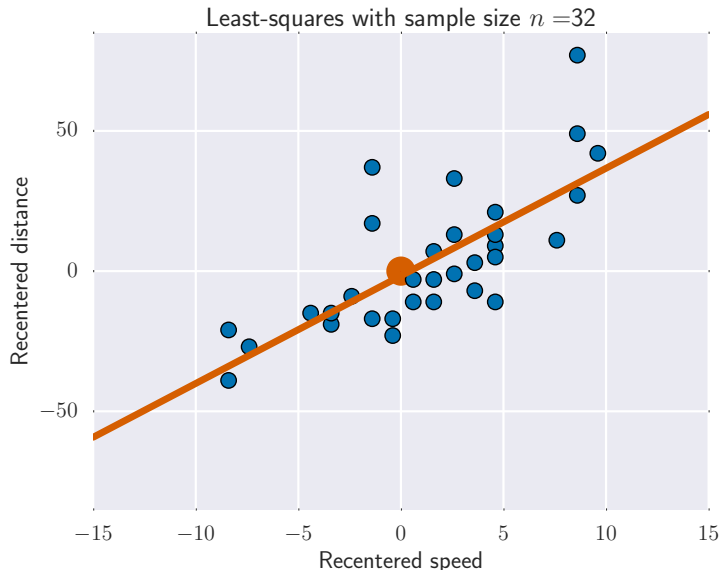
Extreme points – leverage effect (II)



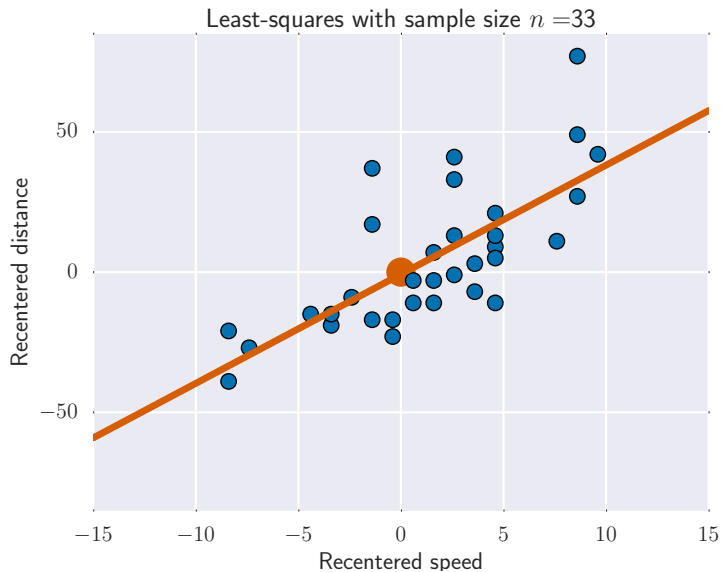
Extreme points – leverage effect (II)



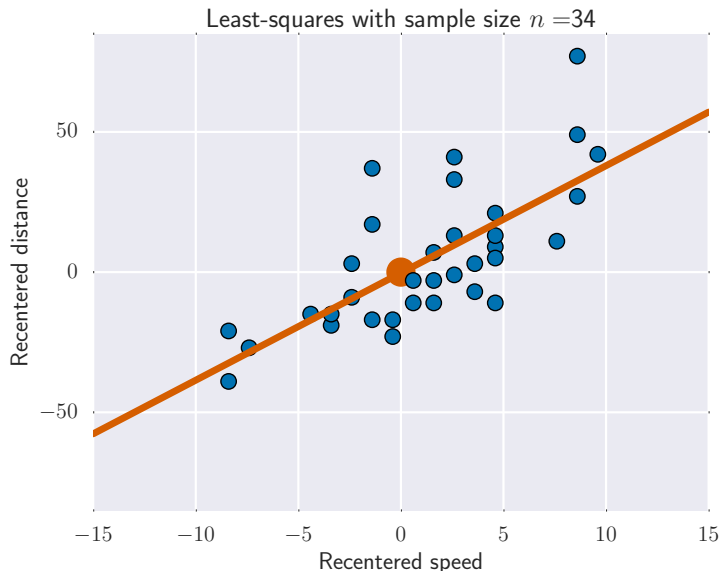
Extreme points – leverage effect (II)



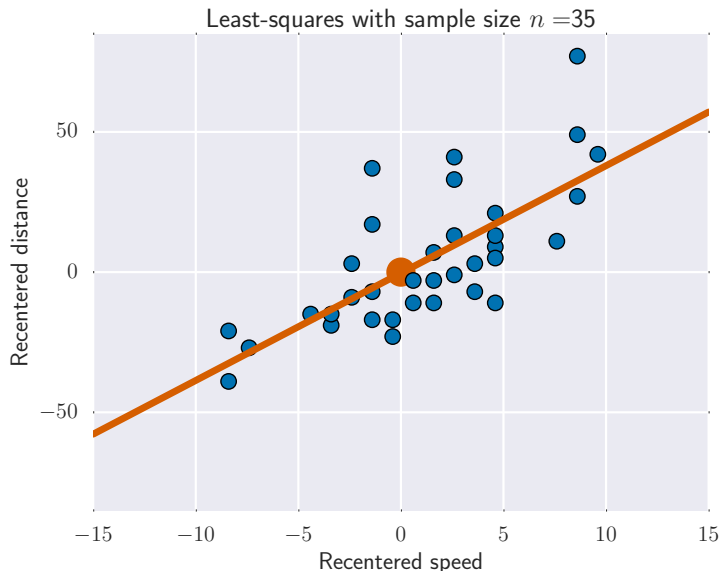
Extreme points – leverage effect (II)



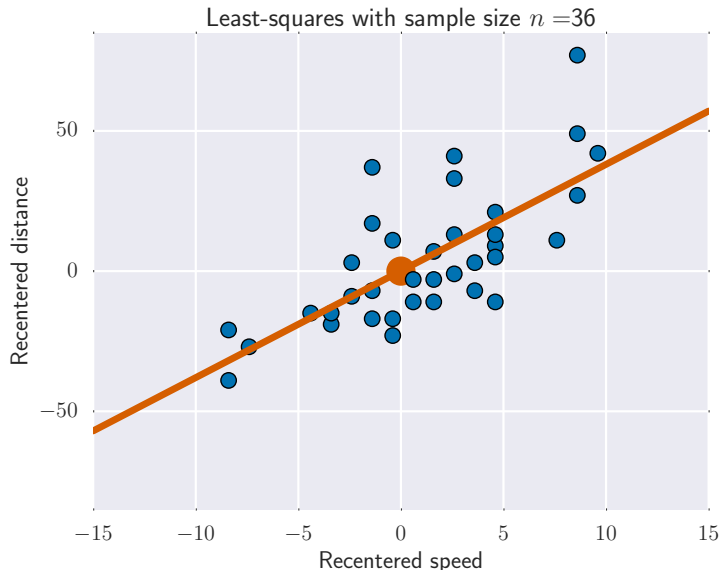
Extreme points – leverage effect (II)



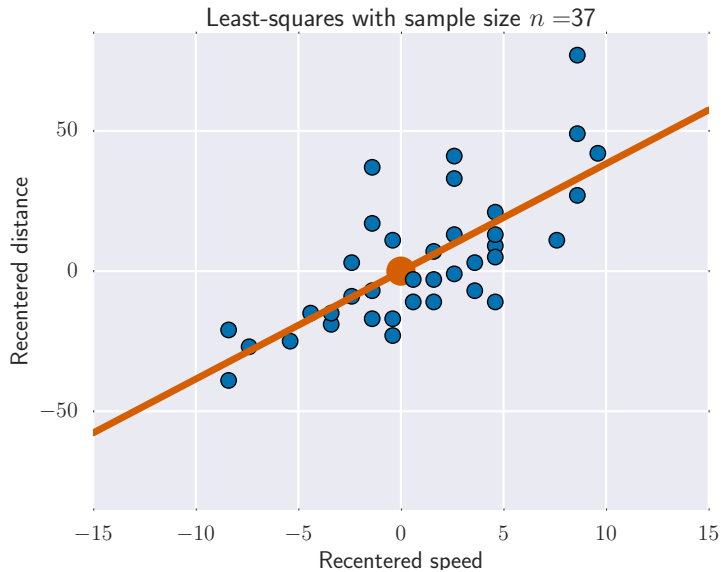
Extreme points – leverage effect (II)



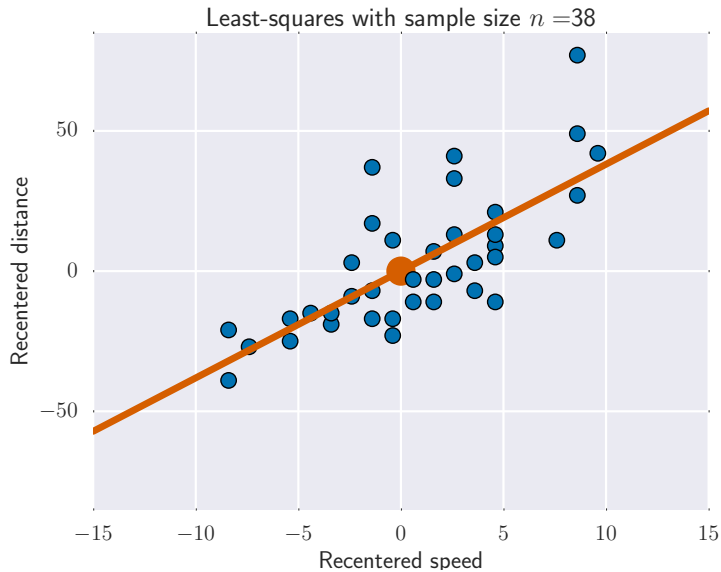
Extreme points – leverage effect (II)



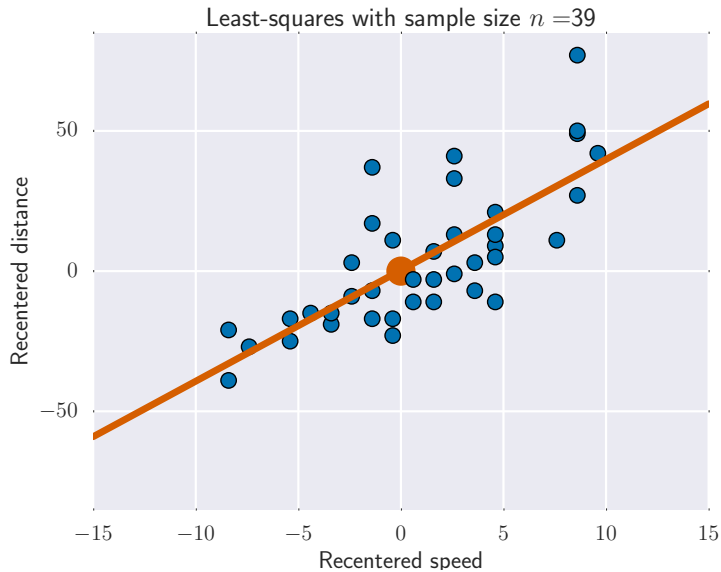
Extreme points – leverage effect (II)



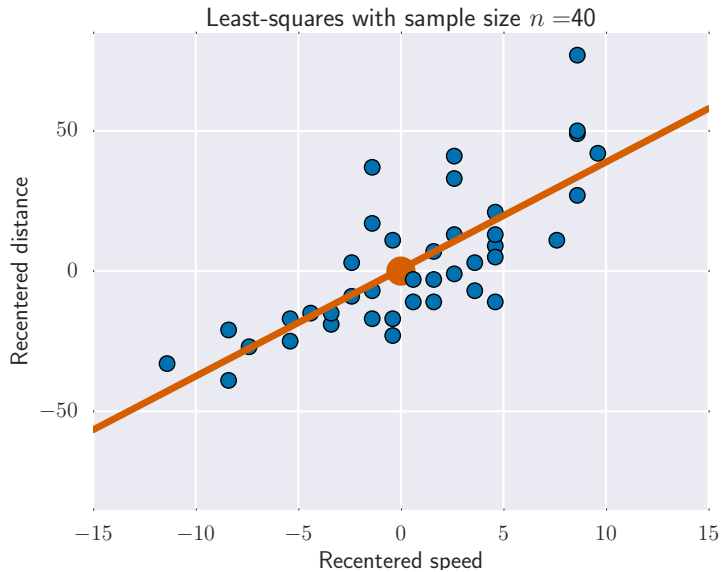
Extreme points – leverage effect (II)



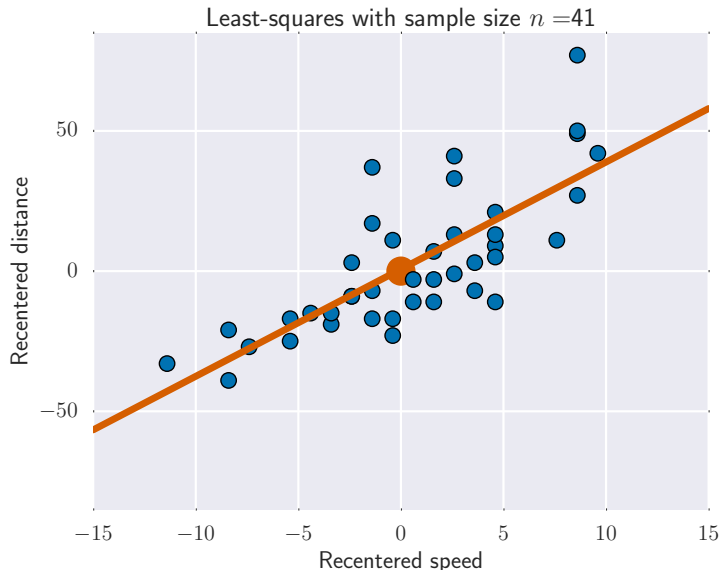
Extreme points – leverage effect (II)



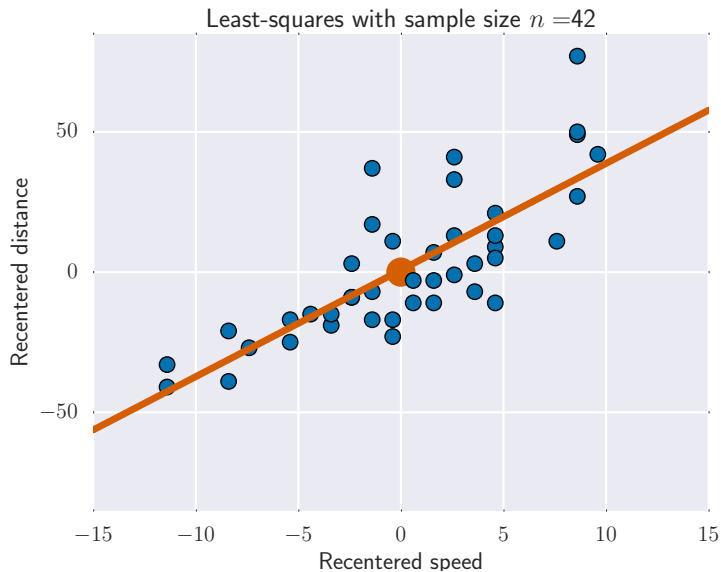
Extreme points – leverage effect (II)



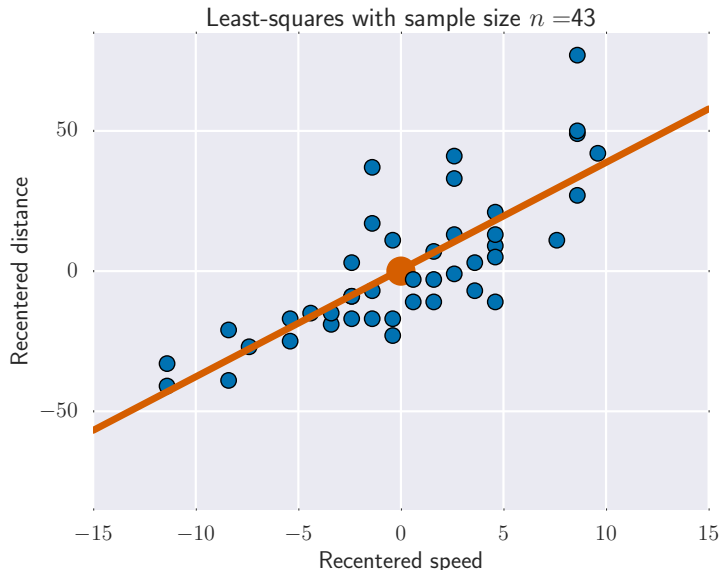
Extreme points – leverage effect (II)



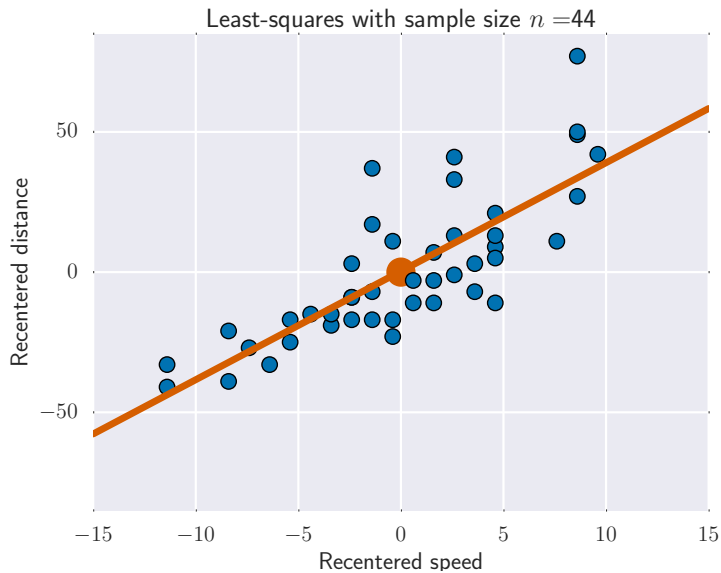
Extreme points – leverage effect (II)



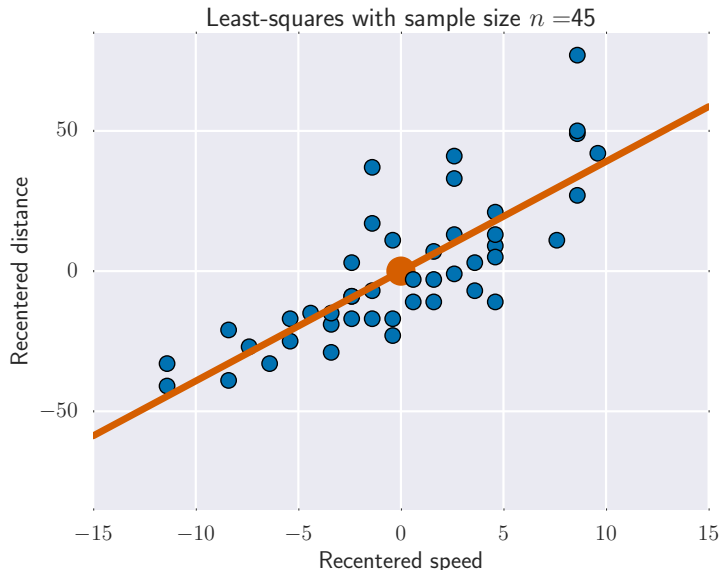
Extreme points – leverage effect (II)



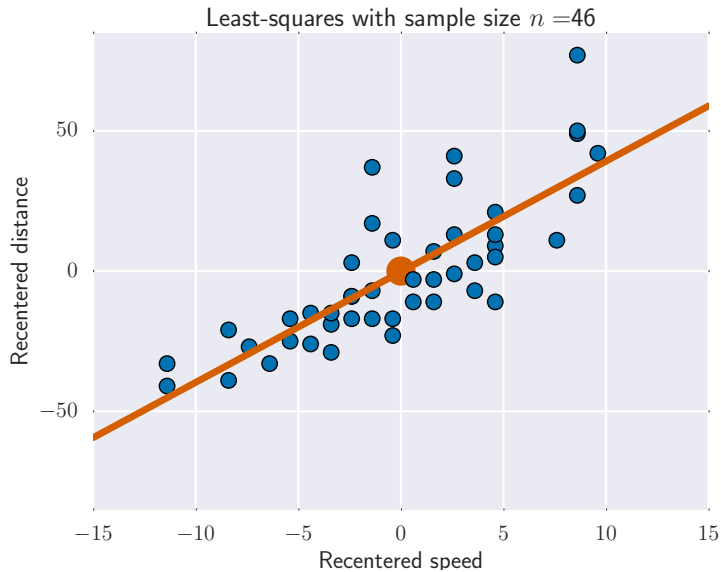
Extreme points – leverage effect (II)



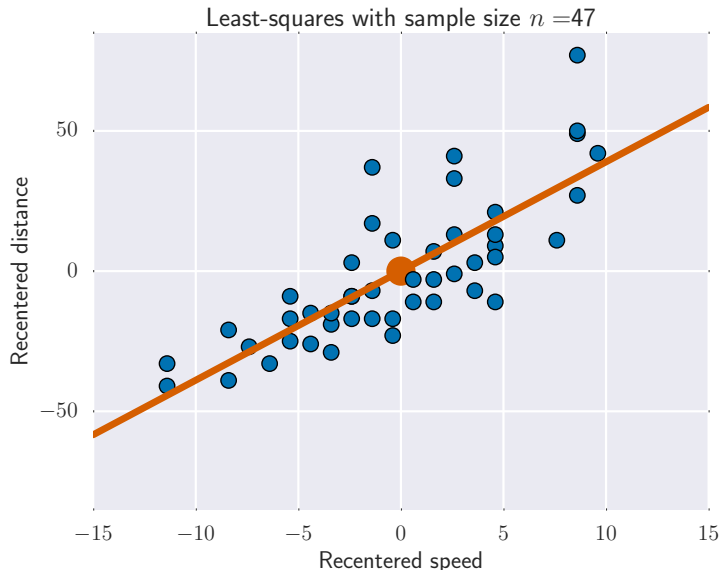
Extreme points – leverage effect (II)



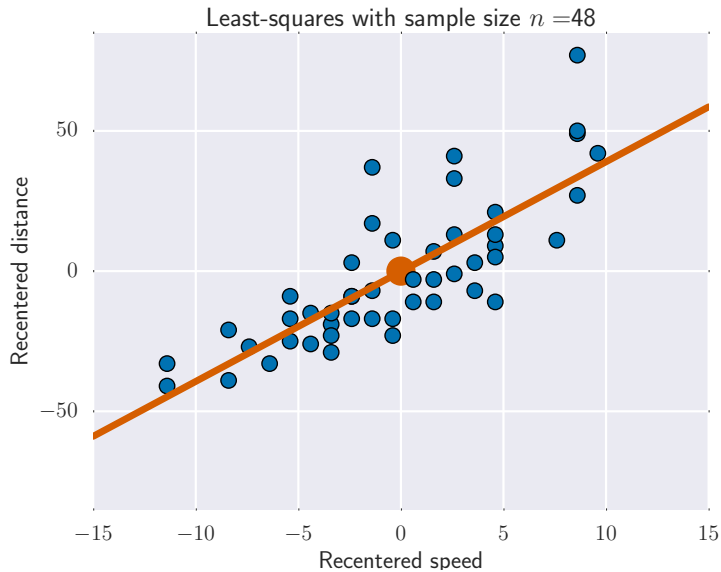
Extreme points – leverage effect (II)



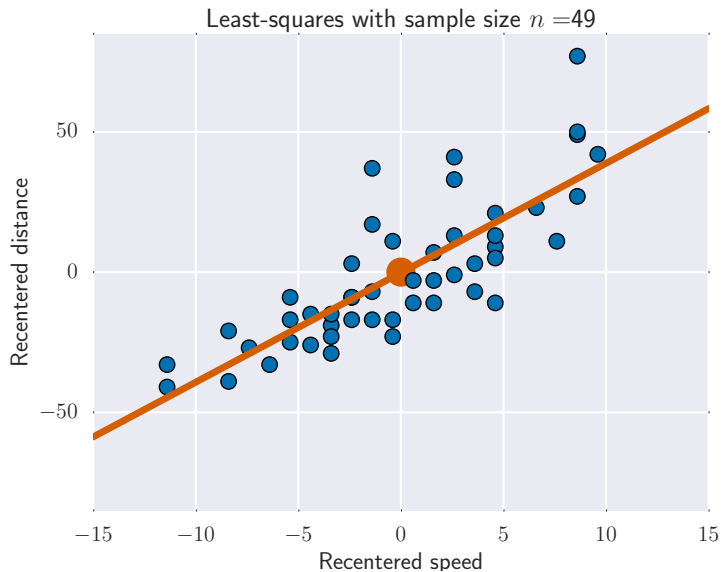
Extreme points – leverage effect (II)



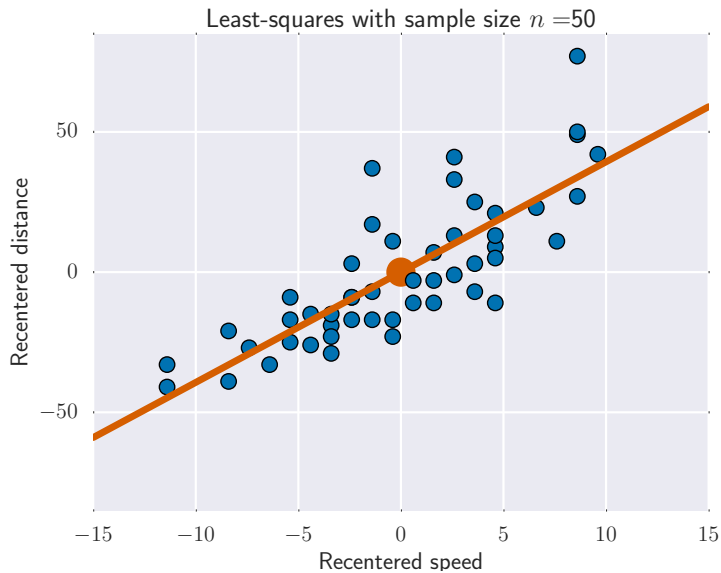
Extreme points – leverage effect (II)



Extreme points – leverage effect (II)



Extreme points – leverage effect (II)



Centering + scaling (standardization)

Centered-scaled model :

$$\forall i = 1, \dots, n : \begin{cases} x_i'' = (x_i - \bar{x}_n) / \sqrt{\text{var}_n(\mathbf{x})} \\ y_i'' = (y_i - \bar{y}_n) / \sqrt{\text{var}_n(\mathbf{y})} \end{cases} \Leftrightarrow \begin{cases} \mathbf{x}'' = \frac{\mathbf{x} - \bar{x}_n \mathbf{1}_n}{\sqrt{\text{var}_n(\mathbf{x})}} \\ \mathbf{y}'' = \frac{\mathbf{y} - \bar{y}_n \mathbf{1}_n}{\sqrt{\text{var}_n(\mathbf{y})}} \end{cases}$$

Solving OLS with $(\mathbf{x}'', \mathbf{y}'')$ then

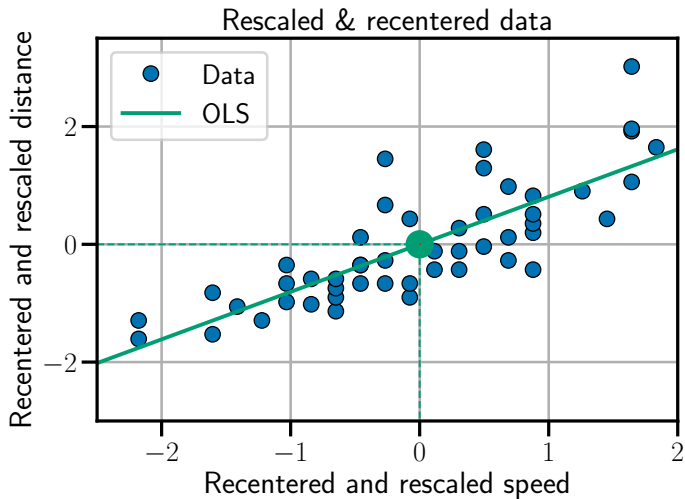
$$\begin{cases} \hat{\theta}_0'' = 0 \\ \hat{\theta}_1'' = \frac{1}{n} \sum_{i=1}^n x_i'' y_i'' \end{cases}$$

Rem: equivalent to choosing the points cloud center of mass as origin and normalize \mathbf{x} and \mathbf{y} to have unit **empirical norm** $\|\cdot\|_n$:

$$\|\mathbf{x}''\|_n^2 = \frac{1}{n} \sum_{i=1}^n (x_i'')^2 = 1$$

$$\|\mathbf{y}''\|_n^2 = \frac{1}{n} \sum_{i=1}^n (y_i'')^2 = 1$$

Centering + scaling



When/why preprocessing ?

Centering \mathbf{y} or using an intercept (or adding a constant feature) is equivalent

Rem: for sparse (■ ■ : *creux*) cases centering \mathbf{y} adding a constant feature could be preferred

Scaling features is important :

- ▶ if you want to interpret the coefficients' amplitude in regression (better solution : t-tests)
- ▶ if you want to penalize or regularize coefficients (*c.f.* Lasso, Ridge, etc.) a single scale is needed
- ▶ for computing reasons (*e.g.* store scaling to improve efficiency, etc.)

Rem: in practice centering/scaling is useful for **estimation** not so much for **prediction** (see next courses)

What happens with the logarithm scaling ?

Centering with Python

Use centering classes from `sklearn`, see preprocessing :

<http://scikit-learn.org/stable/modules/preprocessing.html>

```
from sklearn import preprocessing

scaler = preprocessing.StandardScaler().fit(X)

print(np.isclose(scaler.mean_, np.mean(X)))

print(np.array_equal(scaler.std_, np.std(X)))

print(np.array_equal(scaler.transform(X),
                    (X - np.mean(X)) / np.std(X)))

print(np.array_equal(scaler.transform([26]),
                    (26 - np.mean(X)) / np.std(X)))
```

Rem: most valuable with pipeline

<http://scikit-learn.org/stable/modules/pipeline.html>

Prediction

We call **prediction** function the function that associates an estimation of the variable of interest to a new sample. For least squares the prediction is given by :

$$\text{pred}(x_{n+1}) = \hat{\theta}_0 + \hat{\theta}_1 x_{n+1}$$

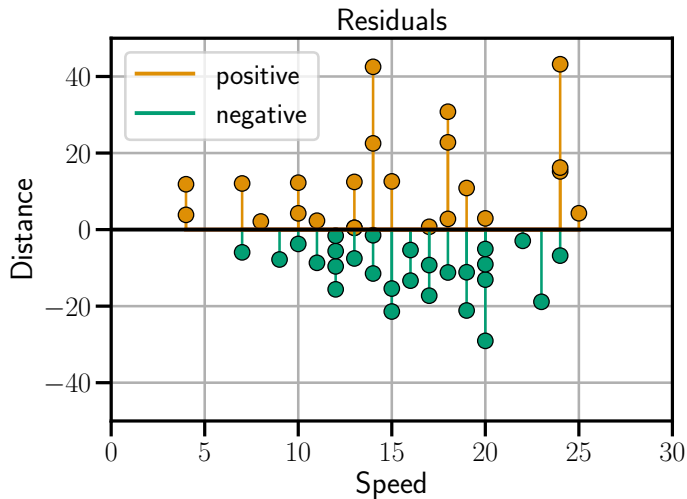
Rem: often written \hat{y}_{n+1} (implicit dependence on x_{n+1})

The **residual** : difference between observations and predicted values

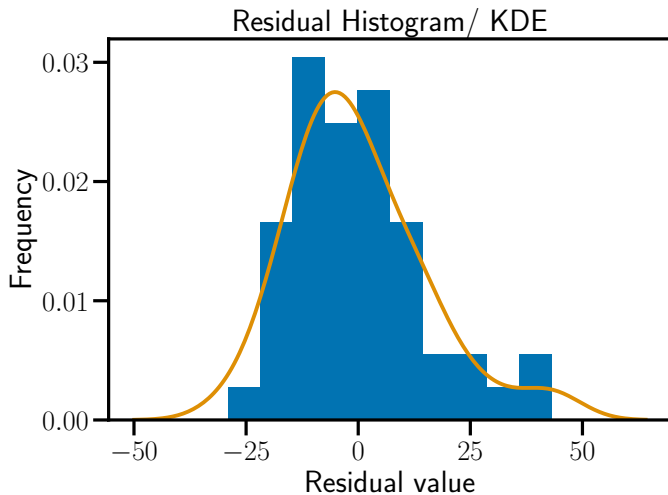
$$\epsilon_i = y_i - \text{pred}(x_i) = y_i - \hat{y}_i = y_i - (\hat{\theta}_0 + \hat{\theta}_1 x_i)$$

Rem: observable estimate of the unobservable statistical error

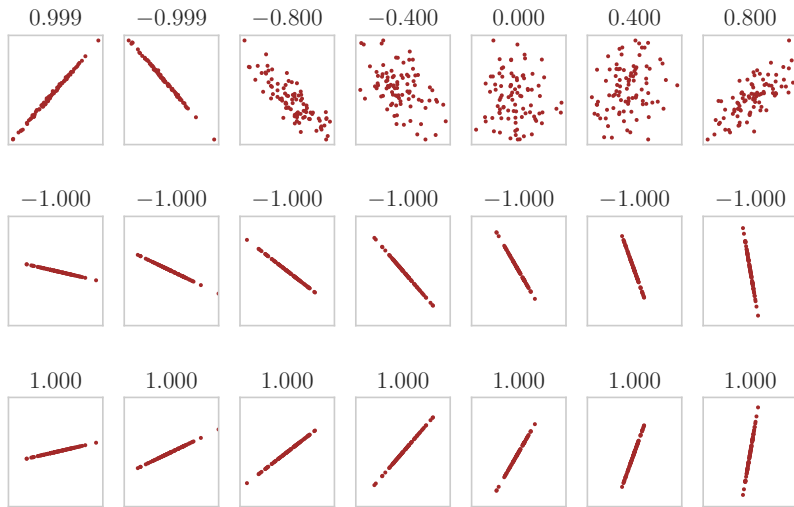
Residuals (on cars, heteroscedasticity)



Residual histograms

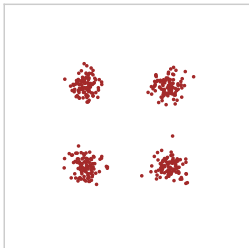


Correlation, variance and R^2

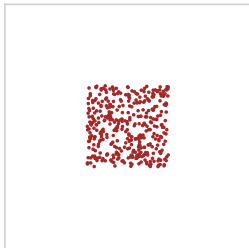


Correlation, variance and R^2

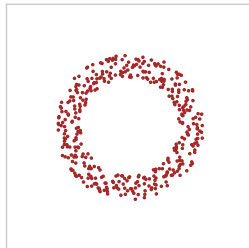
Corrélation = -0.021



Corrélation = 0.007



Corrélation = 0.011



Always visualize the data [https:](https://www.research.autodesk.com/publications/same-stats-different-graphs/)

[//www.research.autodesk.com/publications/same-stats-different-graphs/](https://www.research.autodesk.com/publications/same-stats-different-graphs/)

Least squares motivation

- ▶ Computing advantage : computationally heavy methods avoided before computers (*e.g.* iterative methods)
- ▶ Theoretical advantage : least square analysis easy under simple hypothesis
- ▶ Interpretability : how much does the regressor increase with the features

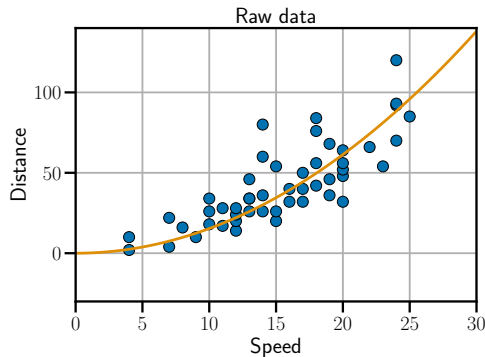
Example : under additive white Gaussian noise assumption *i.e.*, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ the maximum likelihood is equivalent to solving least squares to estimate (θ_0^*, θ_1^*)

Rem: for another noise model and/or to limit outliers influence one can solve (see *e.g.* `QuantReg` in `statsmodels`)

$$\hat{\theta} = (\hat{\theta}_0, \hat{\theta}_1) \in \operatorname{argmin}_{(\theta_0, \theta_1) \in \mathbb{R}^2} \sum_{i=1}^n |y_i - \theta_0 - \theta_1 x_i|$$

Discussion : toward multivariate cases

Physical laws (or your driving school memories) would lead to rather pick a **quadratic** model instead of a **linear** one : the OLS can be applied by choosing x_i^2 as features instead of x_i :

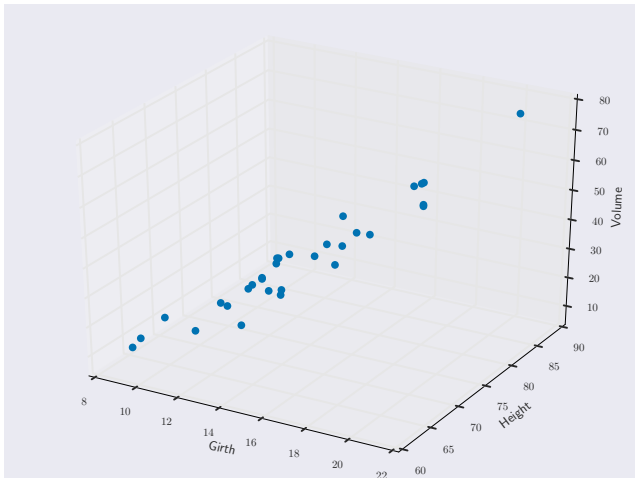


Web sites and books to go further

- ▶ Datascience in general : Blog + videos by Jake Vanderplas
<http://jakevdp.github.io/>
Homework for next lesson : watch the following videos <http://jakevdp.github.io/blog/2017/03/03/reproducible-data-analysis-in-jupyter/>
- ▶ A few [notebooks](#) of OLS with statsmodels
- ▶ [McKinney \(2012\)](#) about Python for statistics
- ▶ [Lejeune \(2010\)](#) about linear models (in French)
- ▶ Regression course by [B. Delyon](#) (in French, more technical)

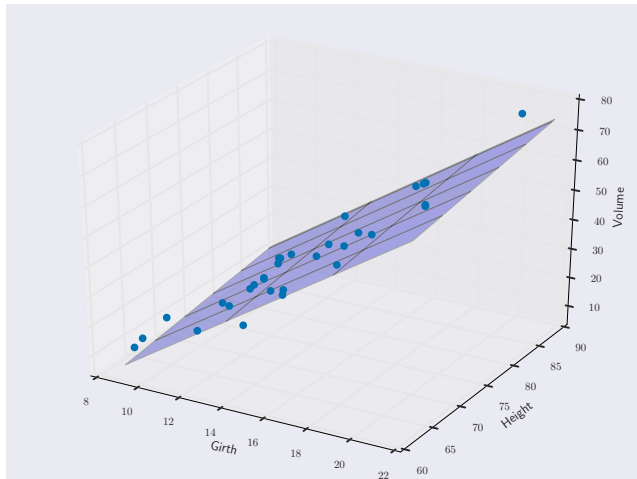
Toward multivariate models

Tree volume as a function of height / girth ( : *circonférence*)



Toward multivariate models

Tree volume as a function of height / girth (■ : *circonférence*)



Python commands

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

# Generate example data
...

# Fit linear regression model
model = LinearRegression()
model.fit(X, y)
```

Model

One observes p features $(\mathbf{x}_1, \dots, \mathbf{x}_p)$. Model in dimension p

$$y_i = \theta_0^* + \sum_{j=1}^p \theta_j^* x_{i,j} + \varepsilon_i$$

$$\varepsilon_i \stackrel{i.i.d}{\sim} \varepsilon, \text{ pour } i = 1, \dots, n$$

$$\mathbb{E}[\varepsilon] = 0$$

Rem: we assume (frequentist point of view) there exists a “true” parameter

$$\boldsymbol{\theta}^* = (\theta_0^*, \dots, \theta_p^*)^\top \in \mathbb{R}^{p+1}$$

Dimension p

Matrix model

$$\underbrace{\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} 1 & x_{1,1} & \dots & x_{1,p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \dots & x_{n,p} \end{pmatrix}}_X \underbrace{\begin{pmatrix} \theta_0^* \\ \vdots \\ \theta_p^* \end{pmatrix}}_{\boldsymbol{\theta}^*} + \underbrace{\begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}}_{\boldsymbol{\epsilon}}$$

Equivalently : $\boxed{\mathbf{y} = X\boldsymbol{\theta}^* + \boldsymbol{\epsilon}}$ (1)

Column notation : $X = (\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_p)$ with $\mathbf{x}_0 = \mathbf{1}_n = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$.

Line notation : $X = \begin{pmatrix} x_1^\top \\ \vdots \\ x_n^\top \end{pmatrix} = (x_1, \dots, x_n)^\top$

Matrix Notation and L_2 Norm

Matrix notation is a powerful way to represent mathematical operations involving vectors and matrices.

The **Inner Product** (dot product) of two vectors \mathbf{u} and \mathbf{v} is defined as :

$$\langle \mathbf{u}, \mathbf{v} \rangle = \sum_{i=1}^n u_i v_i = \mathbf{u}^T \cdot \mathbf{v}$$

Let \mathbf{A} be an $m \times n$ matrix and \mathbf{B} be an $n \times p$ matrix. The **matrix product** $\mathbf{C} = \mathbf{AB}$ is an $m \times p$ matrix with elements :

$$c_{ij} = \sum_{k=1}^n a_{ik} b_{kj}$$

The L_2 **Norm** (Euclidean norm) of a vector \mathbf{v} is defined as :

$$\|\mathbf{v}\|_2 = \sqrt{\sum_{i=1}^n v_i^2}$$

Matrix notation simplifies operations and equations involving vectors and matrices.

Vocabulary

$$\mathbf{y} = X\boldsymbol{\theta}^* + \boldsymbol{\epsilon}$$

- ▶ $\mathbf{y} \in \mathbb{R}^n$: observations vector
- ▶ $X \in \mathbb{R}^{n \times (p+1)}$: **design** matrix (with features as columns and a first column of 1s)
- ▶ $\tilde{X} \in \mathbb{R}^{n \times (p)}$: **reduced design** matrix (with features as columns and NO column of ones)
- ▶ $\boldsymbol{\theta}^* \in \mathbb{R}^{p+1}$: (unknown) **true** parameter to be estimated
- ▶ $\boldsymbol{\epsilon} \in \mathbb{R}^n$: noise vector

Vocabulary (and abuse of terms)

We call **Gram matrix** the matrix

$$X^{\top} X$$

whose general term is $[X^{\top} X]_{i,j} = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$

If the design matrix X is centered and scaled, the Gram matrix is proportional to the correlation between columns. $X^{\top} X$ is often referred to as the feature correlation matrix

Rem: when columns are scaled such that $\forall j \in \llbracket 0, p \rrbracket, \|\mathbf{x}_j\|^2 = n$, the Gramian diagonal is (n, \dots, n)

The vector $X^{\top} \mathbf{y} = \begin{pmatrix} \langle \mathbf{x}_0, \mathbf{y} \rangle \\ \vdots \\ \langle \mathbf{x}_p, \mathbf{y} \rangle \end{pmatrix}$ represents the correlation between the observations and the features

(Ordinary) Least squares

A least square estimator is any solution of the following problem :

$$\hat{\boldsymbol{\theta}} \in \operatorname{argmin}_{\boldsymbol{\theta} \in \mathbb{R}^{p+1}} \left(\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 \right)$$

$$\hat{\boldsymbol{\theta}} \in \operatorname{argmin}_{\boldsymbol{\theta} \in \mathbb{R}^{p+1}} \sum_{i=1}^n \left[y_i - \left(\theta_0 + \sum_{j=1}^p \theta_j x_{i,j} \right) \right]^2$$

$$\hat{\boldsymbol{\theta}} \in \operatorname{argmin}_{\boldsymbol{\theta} \in \mathbb{R}^{p+1}} \sum_{i=1}^n [y_i - \langle x_i, \boldsymbol{\theta} \rangle]^2$$

- Does the solution exist ? A solution always exists, as we are minimizing a coercive continuous function (**coercive** : $\lim_{\|x\| \rightarrow +\infty} f(x) = +\infty$)
- Is the solution unique ? not guaranteed

Exo how do we make the prediction ?

Row / column interpretation

Row interpretation

Let $\tilde{x}_1^\top, \dots, \tilde{x}_{p+1}^\top$ be the rows of X . The residuals are $r_i = y_i - \tilde{x}_i \boldsymbol{\theta}$ and the OLS is equivalent to minimizing the sum of squares residuals

Column interpretation

Let $\mathbf{x}_0, \dots, \mathbf{x}_p$ be the columns of X . Then $\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 = \|(\theta_0 \mathbf{x}_0, \dots, \theta_p \mathbf{x}_p) - \mathbf{y}\|_2^2$, so OLS is to find a linear combination of columns of X that is closest to \mathbf{y} .

Hilbert projection theorem (HPT)

Let $C \subset \mathbb{R}^d, Y \in \mathbb{R}^d$. Let $\hat{z} = \arg \min_{z \in C} \|Y - z\|_2^2$. Then \hat{z} always exists and is given by

$$\langle Y - \hat{z}, z \rangle = 0 \quad \forall z \in C$$

Hilbert projection theorem (HPT) and application to OLS

$$\hat{\boldsymbol{\theta}} \in \operatorname{argmin}_{\boldsymbol{\theta} \in \mathbb{R}^{p+1}} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2$$

Note $\operatorname{col}(X) = \operatorname{span}([\mathbf{x}_0, \dots, \mathbf{x}_p]) = \sum_{j=0}^p \mathbf{x}_j \theta_j = X\boldsymbol{\theta}$ OLS :

$$\widehat{W} \in \operatorname{argmin}_{W \in \operatorname{col}(X)} (\|\mathbf{y} - W\|_2^2)$$

$$\langle \mathbf{y} - \widehat{W}, W \rangle = 0$$

$$(\mathbf{y} - \widehat{W})^\top W = 0$$

$$(\mathbf{y} - \widehat{W})^\top X\boldsymbol{\theta} = 0$$

$$(\mathbf{y} - \widehat{W})^\top X = 0 \tag{2}$$

$$(\mathbf{y} - X\hat{\boldsymbol{\theta}})^\top X = 0$$

$$X^\top (\mathbf{y} - X\hat{\boldsymbol{\theta}}) = 0$$

$$X^\top X\hat{\boldsymbol{\theta}} = X^\top \mathbf{y}$$

OLS normal equations

The solution to the OLS problem is given by the solution to the normal equation

$$\text{Normal equation : } \boxed{X^{\top} X \hat{\boldsymbol{\theta}} = X^{\top} \mathbf{y}}$$

As a consequence,

- ▶ a solution always exists.
- ▶ its unique if the solution to the normal equations is unique

Hilbert projection theorem, geometric interpretation

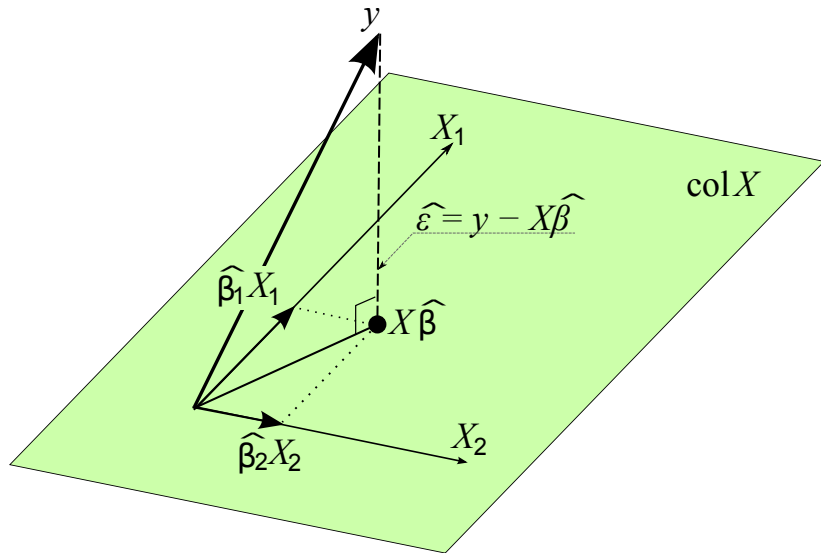


Figure — Source : Wikipedia

Least squares and uniqueness

Let $\hat{\boldsymbol{\theta}}$ be a solution of $\boxed{X^\top X \hat{\boldsymbol{\theta}} = X^\top \mathbf{y}}$

Non uniqueness : happens for non trivial kernel, *i.e.* when $\ker(X) = \{\boldsymbol{\theta} \in \mathbb{R}^{p+1} : X\boldsymbol{\theta} = 0\} \neq \{0\}$

Assume $\boldsymbol{\theta}_K \in \ker(X)$ with $\boldsymbol{\theta}_K \neq 0$, then

$$X(\hat{\boldsymbol{\theta}} + \boldsymbol{\theta}_K) = X\hat{\boldsymbol{\theta}}$$

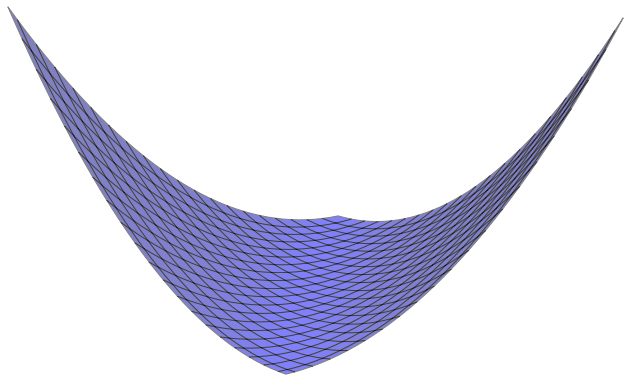
$$\text{and then } (X^\top X)(\hat{\boldsymbol{\theta}} + \boldsymbol{\theta}_K) = X^\top \mathbf{y}$$

Conclusion : the set of least squares solutions is an affine sub-space

$$\boxed{\hat{\boldsymbol{\theta}} + \ker(X)}$$

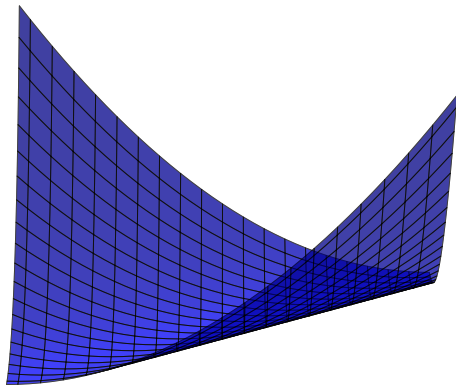
Optimization in \mathbb{R}^d

Convex case, $f(\boldsymbol{\theta}) = \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2$, where the set of minimizers is non-unique :



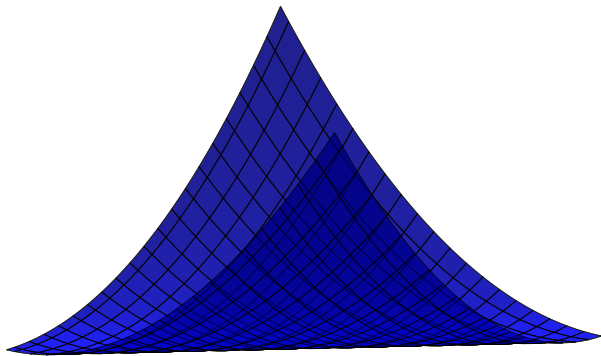
Optimization in \mathbb{R}^d

Convex case, $f(\boldsymbol{\theta}) = \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2$, where the set of minimizers is non-unique :



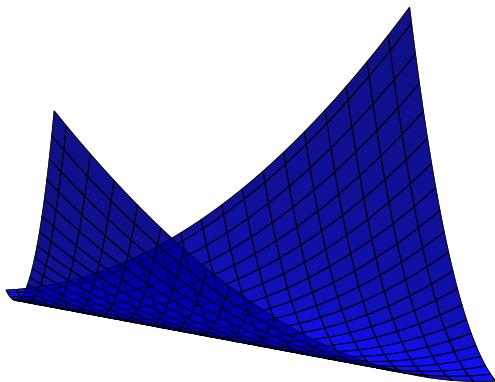
Optimization in \mathbb{R}^d

Convex case, $f(\boldsymbol{\theta}) = \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2$, where the set of minimizers is non-unique :



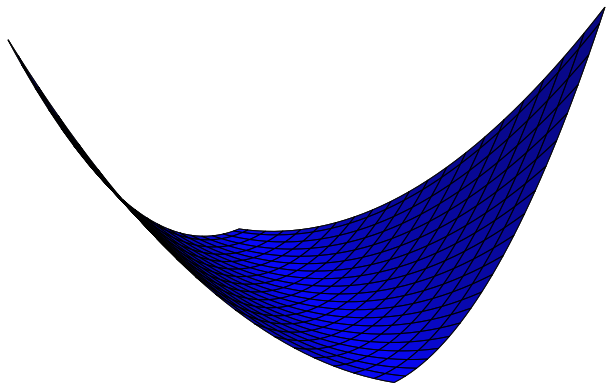
Optimization in \mathbb{R}^d

Convex case, $f(\boldsymbol{\theta}) = \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2$, where the set of minimizers is non-unique :



Optimization in \mathbb{R}^d

Convex case, $f(\boldsymbol{\theta}) = \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2$, where the set of minimizers is non-unique :



Interpretation for multivariate cases

Reminder : we write $X = (\mathbf{1}_n, \mathbf{x}_1, \dots, \mathbf{x}_p)$, the features being column-wise (each are of length n)

The property $\ker(X) = \{\boldsymbol{\theta} \in \mathbb{R}^{p+1} : X\boldsymbol{\theta} = 0\} \neq \{0\}$ means that there exists a linear dependence between the features $\mathbf{1}_n, \mathbf{x}_1, \dots, \mathbf{x}_p$,

Reformulation : $\exists \boldsymbol{\theta} = (\theta_0, \dots, \theta_p)^\top \in \mathbb{R}^{p+1} \setminus \{0\}$ s.t.

$$\theta_0 \mathbf{1}_n + \sum_{j=1}^p \theta_j \mathbf{x}_j = 0$$

Algebra reminder

Rank of a matrix : $\text{rank}(X) = \dim(\text{span}(\mathbf{1}_n, \mathbf{x}_1, \dots, \mathbf{x}_p))$; $\text{span}(\cdot)$: the space generated by \cdot .

Property : $\text{rank}(X) = \text{rank}(X^\top)$

Rank–nullity theorem :

- ▶ $\text{rank}(X) + \dim(\ker(X)) = p + 1$
- ▶ $\text{rank}(X^\top) + \dim(\ker(X^\top)) = n$

Property : $\boxed{\text{rank}(X) \leq \min(n, p + 1)}$

See [Golub and Van Loan \(1996\)](#) for details

Algebra reminder (continued)

Matrix inversion : A square matrix $A \in \mathbb{R}^{m \times m}$ is invertible

- ▶ if and only if its kernel is trivial : $\ker(A) = \{0\}$
- ▶ if and only if it is full rank $\text{rank}(A) = m$

OLS is unique iff $X^\top X$ is invertible

$$\Leftrightarrow \ker(X^\top X) = \{0\}$$

$$\Leftrightarrow \ker(X) = \{0\}$$

$$\Leftrightarrow X \text{ has full rank}$$

Exo: $\ker(X) = \ker(X^\top X)$

Non uniqueness : single feature case

Reminder :

$$X = \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix}$$

If $\ker(X) = \{\boldsymbol{\theta} \in \mathbb{R}^2 : X\boldsymbol{\theta} = 0\} \neq \{0\}$ there exists $(\theta_0, \theta_1) \neq (0, 0)$:

$$\begin{cases} \theta_0 + \theta_1 x_1 & = 0 \\ \vdots & \vdots & = \vdots \\ \theta_0 + \theta_1 x_n & = 0 \end{cases} \quad (\star)$$

1. If $\theta_1 = 0$: $(\star) \Rightarrow \theta_0 = 0$, so $(\theta_0, \theta_1) = (0, 0)$, **contradiction**
2. If $\theta_1 \neq 0$:
 - 2.1 If $\forall i, x_i = 0$ then $X = (\mathbf{1}_n, 0)$ and $\theta_0 = 0$
 - 2.2 Otherwise there exists $x_{i_0} \neq 0$ and $\forall i, x_i = -\theta_0/\theta_1 = x_{i_0}$, *i.e.* $X = [\mathbf{1}_n \quad x_{i_0} \cdot \mathbf{1}_n]$

Interpretation : $\mathbf{x}_1 \propto \mathbf{1}_n$, *i.e.* \mathbf{x}_1 is constant

Residuals and normal equation

$$\text{Residual(s) : } \hat{\boldsymbol{\varepsilon}} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - X\hat{\boldsymbol{\theta}} = (\text{Id}_n - H_X)\mathbf{y}$$

Proposition

$$\langle \hat{\boldsymbol{\varepsilon}}, X \rangle = 0_n$$

$$\langle \hat{\boldsymbol{\varepsilon}}, \hat{\mathbf{y}} \rangle = 0$$

$$\langle \hat{\boldsymbol{\varepsilon}}, \bar{\mathbf{y}}\mathbf{1}_n \rangle = 0$$

Rem: The Normal equation is $(X^\top X)\hat{\boldsymbol{\theta}} = X^\top \mathbf{y}$. It follows that
 $X^\top (X\hat{\boldsymbol{\theta}} - \mathbf{y}) = 0 \Leftrightarrow X^\top \hat{\boldsymbol{\varepsilon}} = 0 \Leftrightarrow \hat{\boldsymbol{\varepsilon}}^\top X = 0$

With $X = (\mathbf{1}_n, \mathbf{x}_1, \dots, \mathbf{x}_p)$, this can be rewritten

$$\forall j = 1, \dots, p : \langle \hat{\boldsymbol{\varepsilon}}, \mathbf{x}_j \rangle = 0 \text{ and } \bar{r}_n = 0$$

Interpretation : (1,2) residuals are \perp to features and (3) $\hat{\boldsymbol{\varepsilon}}$ is centered ($\sum \hat{\varepsilon}_i = 0$)

How good is our model? RSS and the determination coefficient R^2

The ratio of the variation explained by the model and the total variation of the data $R^2 = \frac{\|\hat{\mathbf{y}} - \bar{\mathbf{y}}\mathbf{1}_n\|^2}{\|\mathbf{y} - \bar{\mathbf{y}}\mathbf{1}_n\|^2}$ We can write also, by orthogonality :

$$\|\mathbf{y} - \bar{\mathbf{y}}\mathbf{1}_n\|^2 = \|\mathbf{y} - \hat{\mathbf{y}}\|^2 + \|\hat{\mathbf{y}} - \bar{\mathbf{y}}\mathbf{1}_n\|^2 \quad (3)$$

Reordering

$$\|\hat{\mathbf{y}} - \bar{\mathbf{y}}\mathbf{1}_n\|^2 = \|\mathbf{y} - \bar{\mathbf{y}}\mathbf{1}_n\|^2 - \|\mathbf{y} - \hat{\mathbf{y}}\|^2 \quad (4)$$

So

$$R^2 = 1 - \frac{\|\mathbf{y} - \hat{\mathbf{y}}\|^2}{\|\mathbf{y} - \bar{\mathbf{y}}\mathbf{1}_n\|^2} \quad (5)$$

Exo: Show that $0 \leq R^2 \leq 1$

Prediction

$$\text{Prediction vector : } \hat{\mathbf{y}} = X\hat{\boldsymbol{\theta}}$$

Rem: $\hat{\mathbf{y}}$ depends linearly on the observation vector \mathbf{y}

Rem: an **orthogonal projector** is a matrix H such that

1. H is symmetric : $H^\top = H$
2. H is idempotent : $H^2 = H$

Proposition Writing H_X the orthogonal projector onto the space span by the columns of X , one gets $\hat{\mathbf{y}} = H_X \mathbf{y}$

If X is full (column) rank, then $H_X = X(X^\top X)^{-1}X^\top$ is called the **hat matrix**

Exo: Show that H_X is an orthogonal projector

Prediction (continued)

If a new observation $x_{n+1} = (x_{n+1,1}, \dots, x_{n+1,p})$ is provided, the associated prediction is :

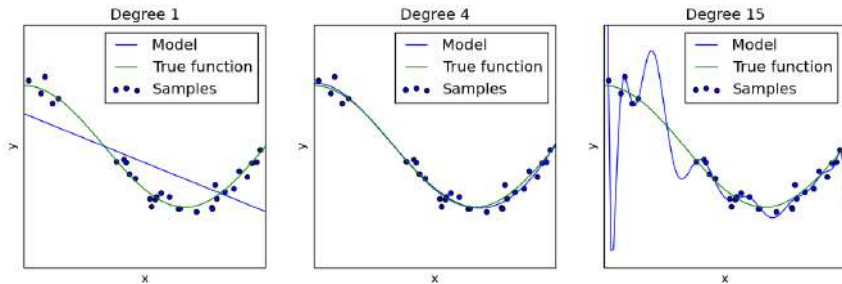
$$\hat{y}_{n+1} = \langle \hat{\boldsymbol{\theta}}, (1, x_{n+1,1}, \dots, x_{n+1,p})^\top \rangle$$

$$\hat{y}_{n+1} = \hat{\theta}_0 + \sum_{j=1}^p \hat{\theta}_j x_{n+1,j}$$

Rem: the normal equation ensures **equi-correlation** between observations and features :





$$\begin{aligned} (X^\top X) \hat{\boldsymbol{\theta}} &= X^\top \mathbf{y} \Leftrightarrow X^\top \hat{\mathbf{y}} = X^\top \mathbf{y} \\ &\Leftrightarrow \begin{pmatrix} \langle \mathbf{x}_0, \hat{\mathbf{y}} \rangle \\ \vdots \\ \langle \mathbf{x}_p, \hat{\mathbf{y}} \rangle \end{pmatrix} = \begin{pmatrix} \langle \mathbf{x}_0, \mathbf{y} \rangle \\ \vdots \\ \langle \mathbf{x}_p, \mathbf{y} \rangle \end{pmatrix} \end{aligned}$$

Polynomial regression and overfitting



Source : sklearn

References I

-  B. Delyon.
Régression, 2015.
<https://perso.univ-rennes1.fr/bernard.delyon/regression.pdf>.
-  G. H. Golub and C. F. van Loan.
Matrix computations.
Johns Hopkins University Press, Baltimore, MD, third edition, 1996.
-  M. Lejeune.
Statistiques, la théorie et ses applications.
Springer, 2010.
-  W. McKinney.
Python for Data Analysis : Data Wrangling with Pandas, NumPy, and IPython.
O'Reilly Media, 2012.

SD-TSIA204

Properties of Ordinary Least Squares

Ekhine Irurozki
Télécom Paris

Model I : The fixed design model

$$y_i = \theta_0^* + \sum_{k=1}^p \theta_k^* x_{i,k} + \varepsilon_i$$
$$x_i^\top = (1, x_{i,1}, \dots, x_{i,p}) \in \mathbb{R}^{p+1}$$
$$\varepsilon_i \stackrel{i.i.d}{\sim} \varepsilon, \text{ for } i = 1, \dots, n$$
$$\mathbb{E}(\varepsilon) = 0, \text{ Var}(\varepsilon) = \sigma^2$$

- ▶ x_i is deterministic
- ▶ σ^2 is called the noise level

Example :

- ▶ Physical experiment when the analyst is choosing the design *e.g.*, temperature of the experiment
- ▶ Some features are not random *e.g.*, time, location.

Model I with Gaussian noise : The fixed design Gaussian model

$$y_i = \theta_0^\star + \sum_{k=1}^p \theta_k^\star x_{i,k} + \varepsilon_i$$

$$x_i^\top = (1, x_{i,1}, \dots, x_{i,p}) \in \mathbb{R}^{p+1}$$

$$\varepsilon_i \stackrel{i.i.d}{\sim} \mathcal{N}(0, \sigma^2), \text{ for } i = 1, \dots, n$$

- ▶ Parametric model : specified by the two parameters $(\boldsymbol{\theta}, \sigma)$
- ▶ Strong assumption

Model II : The random design model

$$y_i = \theta_0^\star + \sum_{k=1}^p \theta_k^\star x_{i,k} + \varepsilon_i$$

$$x_i^\top = (1, x_{i,1}, \dots, x_{i,p}) \in \mathbb{R}^{p+1}$$

$$(\varepsilon_i, x_i) \stackrel{i.i.d}{\sim} (\varepsilon, x), \text{ for } i = 1, \dots, n$$

$$\mathbb{E}(\varepsilon|x) = 0, \text{ Var}(\varepsilon|x) = \sigma^2$$

Rem: here, the features are modelled as random (they might also suffer from some noise)

The ordinary least squares (OLS) estimator

$$\hat{\boldsymbol{\theta}} \in \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^{p+1}} \sum_{i=1}^n \left(y_i - \theta_0 - \sum_{k=1}^p \theta_k x_{i,k} \right)^2$$

How to deal with these two models ?

- ▶ The estimator is the same for both models
- ▶ The mathematics involved are different for each case
- ▶ The study of the fixed design case is easier as many closed formulas are available
- ▶ The two models lead to the same estimators of the variance σ^2

The OLS estimator, $\hat{\boldsymbol{\theta}} = (X^\top X)^{-1} X^\top Y$, how good it is?

$$\begin{aligned}\hat{\boldsymbol{\theta}} &= \boldsymbol{\theta} + (X^\top X)^{-1} X^\top \boldsymbol{\varepsilon} \\ \hat{\boldsymbol{\theta}} &\sim N(\boldsymbol{\theta}, \sigma^2 (X^\top X)^{-1})\end{aligned}\tag{1}$$

Its unbiased when $\mathbb{E}(\boldsymbol{\varepsilon}) = 0$

Its not very useful in practice since σ is not known

Exercise: Give the proof for Eq.(1). How is θ_i distributed?

Expectation and covariance

$$\begin{aligned}\mathbb{E}[\hat{\boldsymbol{\theta}}] &= \mathbb{E}[\boldsymbol{\theta}^* + (X^\top X)^{-1} X^\top \boldsymbol{\varepsilon}] \\ &= \boldsymbol{\theta}^* + (X^\top X)^{-1} X^\top \mathbb{E}[\boldsymbol{\varepsilon}] \\ &= \boldsymbol{\theta}^*\end{aligned}\tag{2}$$

Under model 1, whenever the matrix X has full rank, we have

$$\begin{aligned}\text{Cov}(\hat{\boldsymbol{\theta}}) &= \text{Cov}(\boldsymbol{\theta}^* + (X^\top X)^{-1} X^\top \boldsymbol{\varepsilon}) \\ &= \text{Cov}((X^\top X)^{-1} X^\top \boldsymbol{\varepsilon}) \\ &= ((X^\top X)^{-1} X^\top) \text{Cov}(\boldsymbol{\varepsilon}) ((X^\top X)^{-1} X^\top)^\top \\ &= (X^\top X)^{-1} X^\top \text{Cov}(\boldsymbol{\varepsilon}) X (X^\top X)^{-1} \\ &= (X^\top X)^{-1} X^\top \sigma^2 I X (X^\top X)^{-1} \\ &= \sigma^2 (X^\top X)^{-1} X^\top X (X^\top X)^{-1} = \sigma^2 (X^\top X)^{-1}\end{aligned}\tag{3}$$

Bias

Proposition: Under model I, whenever the matrix X has full rank, the least squares estimator is unbiased, i.e.,

$$\mathbb{E}(\hat{\boldsymbol{\theta}}) = \boldsymbol{\theta}^{\star}$$

Proof :

$$B = \mathbb{E}(\hat{\boldsymbol{\theta}}) - \boldsymbol{\theta}^{\star} = \mathbb{E}((X^{\top}X)^{-1}X^{\top}\mathbf{y}) - \boldsymbol{\theta}^{\star}$$

$$B = \mathbb{E}((X^{\top}X)^{-1}X^{\top}(X\boldsymbol{\theta}^{\star} + \boldsymbol{\varepsilon})) - \boldsymbol{\theta}^{\star}$$

$$B = (X^{\top}X)^{-1}X^{\top}X\boldsymbol{\theta}^{\star} + (X^{\top}X)^{-1}X^{\top}\mathbb{E}(\boldsymbol{\varepsilon}) - \boldsymbol{\theta}^{\star} = 0$$

The trace of a matrix

Let $A \in \mathbb{R}^{n \times n}$ denote a matrix. The **trace** of A is the sum of the diagonal elements of A and is denoted by $\text{tr}(A)$:

$$\text{tr}(A) = \sum_{i=1}^n A_{i,i}$$

Several properties :

- ▶ $\text{tr}(A) = \text{tr}(A^\top)$
- ▶ For any $A, B \in \mathbb{R}^{n \times n}$, and $\alpha \in \mathbb{R}$, $\text{tr}(\alpha A + B) = \alpha \text{tr}(A) + \text{tr}(B)$ (linearity)
- ▶ $\text{tr}(A^\top A) = \sum_{i=1}^n \sum_{j=1}^n A_{i,j}^2 := \|A\|_F^2$
- ▶ For any $A, B \in \mathbb{R}^{n \times n}$, $\text{tr}(AB) = \text{tr}(BA)$
- ▶ $\text{tr}(PAP^{-1}) = \text{tr}(A)$, hence if A is diagonalisable, the trace is the sum of the eigenvalues
- ▶ If H is an orthogonal projector $\text{tr}(H) = \text{rank}(H)$

Estimation risk $R(\boldsymbol{\theta}^\star, \hat{\boldsymbol{\theta}}) = \mathbb{E}\|\boldsymbol{\theta}^\star - \hat{\boldsymbol{\theta}}\|^2$

Under model 1, whenever the matrix X has full rank, we have

$$R(\boldsymbol{\theta}^\star, \hat{\boldsymbol{\theta}}) = \mathbb{E}\left[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^\star)^\top (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^\star)\right] = \sigma^2 \text{tr}((X^\top X)^{-1})$$

Proof :

$$\begin{aligned} R(\boldsymbol{\theta}^\star, \hat{\boldsymbol{\theta}}) &= \mathbb{E}\left[(\hat{\boldsymbol{\theta}} - \mathbb{E}\hat{\boldsymbol{\theta}})^\top (\hat{\boldsymbol{\theta}} - \mathbb{E}\hat{\boldsymbol{\theta}})\right] = \mathbb{E}\left[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^\star)^\top (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^\star)\right] \\ &= \mathbb{E}\left[((X^\top X)^{-1}X^\top(X\boldsymbol{\theta}^\star + \boldsymbol{\varepsilon}) - \boldsymbol{\theta}^\star)^\top ((X^\top X)^{-1}X^\top(X\boldsymbol{\theta}^\star + \boldsymbol{\varepsilon}) - \boldsymbol{\theta}^\star)\right] \\ &= \mathbb{E}\left[((X^\top X)^{-1}X^\top\boldsymbol{\varepsilon})^\top ((X^\top X)^{-1}X^\top\boldsymbol{\varepsilon})\right] = \mathbb{E}(\boldsymbol{\varepsilon}^\top X(X^\top X)^{-2}X^\top\boldsymbol{\varepsilon}) \\ &= \text{tr}[\mathbb{E}(\boldsymbol{\varepsilon}^\top X(X^\top X)^{-1}(X^\top X)^{-1}X^\top\boldsymbol{\varepsilon})] \text{ (thx to } \text{tr}(u^\top u) = u^\top u) \\ &= \mathbb{E}\left(\text{tr}\left[(X^\top X)^{-1}X^\top\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\top X(X^\top X)^{-1}\right]\right) \\ &= \text{tr}\left[(X^\top X)^{-1}X^\top\mathbb{E}(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\top)X(X^\top X)^{-1}\right] \\ &= \sigma^2 \text{tr}((X^\top X)^{-1}) \end{aligned}$$

Prediction risk (normalized) $R_{\text{pred}}(\boldsymbol{\theta}^*, \hat{\boldsymbol{\theta}}) = \mathbb{E} \|X\boldsymbol{\theta}^* - \hat{\mathbf{y}}\|^2/n$

Under model I, whenever the matrix X has full rank, we have

$$R_{\text{pred}}(\boldsymbol{\theta}^*, \hat{\boldsymbol{\theta}}) = \mathbb{E} \left[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*)^\top \left(\frac{X^\top X}{n} \right) (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*) \right] = \sigma^2 \frac{\text{rank}(X)}{n}$$

Because X has full rank, $\text{rank}(X) = p + 1$.

Proof : As before

$$\begin{aligned} n \cdot R_{\text{pred}}(\boldsymbol{\theta}^*, \hat{\boldsymbol{\theta}}) &= \mathbb{E} \left[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*)^\top (X^\top X) (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*) \right] \\ &= \mathbb{E} (\boldsymbol{\epsilon}^\top X (X^\top X)^{-1} (X^\top X) (X^\top X)^{-1} X^\top \boldsymbol{\epsilon}) \\ &= \mathbb{E} (\boldsymbol{\epsilon}^\top X (X^\top X)^{-1} X^\top \boldsymbol{\epsilon}) \\ &= \text{tr}[\mathbb{E}(\boldsymbol{\epsilon}^\top H \boldsymbol{\epsilon})] = \text{tr}[\mathbb{E}(\boldsymbol{\epsilon}^\top H^\top H \boldsymbol{\epsilon})] \\ &= \text{tr}[\mathbb{E}(H \boldsymbol{\epsilon} \boldsymbol{\epsilon}^\top H^\top)] = \text{tr}(H \mathbb{E}(\boldsymbol{\epsilon} \boldsymbol{\epsilon}^\top) H^\top) \\ &= \sigma^2 \text{tr}(H) = \sigma^2 \text{rank}(H) = \sigma^2 \text{rank}(X) \end{aligned}$$

Best linear unbiased estimator

Under the fixed design model, among all the unbiased linear estimators AY , $\hat{\theta}_n$ is the one with minimal variance, i.e.,

$$\text{cov}(\hat{\theta}_n) \leq \text{cov}(AY),$$

with equality if and only if $A = (X^T X)^{-1} X^T$.

proof First note that AY is unbiased if and only if $(A - (X^T X)^{-1} X^T) X \theta^* = 0$ for all θ^* , equivalently, $BX = 0$ with $B = (A - (X^T X)^{-1} X^T)$. Consequently, using that $E[\epsilon \epsilon^T] = \sigma^2 I_n$, $\text{cov}(BY, \hat{\theta}_n) = 0$. Then, just write

$$\begin{aligned} \text{cov}(AY) &= \text{cov}(BY + \hat{\theta}_n) \\ &= \text{cov}(BY) + \text{cov}(\hat{\theta}_n) \\ &= \sigma^2 B B^T + \text{cov}(\hat{\theta}_n) \geq \text{cov}(\hat{\theta}_n). \end{aligned}$$

The previous inequality is an equality if and only if $B = 0$.

Maximum Likelihood Estimation (MLE)

Explanation of the principle of maximum likelihood :

- ▶ Maximum Likelihood Estimation (MLE) is a widely used method to estimate unknown parameters.
- ▶ It is based on the idea of finding the parameter values that make the observed data most probable under a given statistical model.

Illustration of Maximum Likelihood Estimation (MLE)

MLE as finding the parameter value that maximizes likelihood :

- ▶ Consider a statistical model with unknown parameter θ and observed data X .
- ▶ The likelihood function $L(\theta; X)$ measures how probable the data is under the parameter θ as a product of their densities, $L(\theta; X) = \prod_{k=1}^n p(X_k; \theta)$.

- ▶ MLE seeks to find $\hat{\theta}$ that maximizes $L(\theta; X)$:
$$\hat{\theta} = \arg \max_{\theta} L(\theta; X)$$

Example : MLE for Coin Flip Model

Coin Flip Model : Probability of getting heads in a coin flip

- ▶ Model : Bernoulli
- ▶ Parameter : p_H (probability of getting heads, $0 \leq p_H \leq 1$)
- ▶ Fair coin : $p_H = 0.5$

Observations : "HH" (two heads in a row)

Likelihood for $p_H = 0.5$: $L(p_H = 0.5 \mid \text{HH}) = 0.5^2 = 0.25$

Likelihood for $p_H = 0.3$: $L(p_H = 0.3 \mid \text{HH}) = 0.3^2 = 0.09$

General Observation : For each observed value $s \in S$, we can calculate the corresponding likelihood as $\prod_{s \in S} p(s; \theta)$.

Note : Likelihoods need not integrate or sum to one over the parameter space.

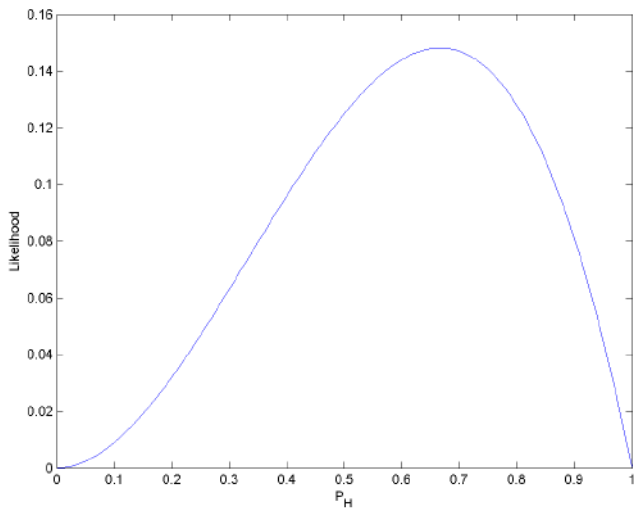


Figure – Likelihood function for different p_H values when we observe HHT

Definition of Likelihood Function and Log-Likelihood Function

Likelihood Function :

- ▶ Measures how well the observed data fit the model parameterized by θ .
- ▶ Denoted by $L(\theta; X)$, where θ is the parameter and X is the observed data.
- ▶ Provides a probability distribution for the observed data given the parameter.
- ▶ For independent and identically distributed random variables, it will be the product of univariate density functions :

$$L(\theta; X) = \prod_{k=1}^n p(X_k; \theta) .$$

Log-Likelihood Function :

- ▶ Definition : $\mathcal{L}(\theta; X) = \log L(\theta; X)$.
- ▶ Log-transform simplifies calculations and often leads to mathematical convenience.
- ▶ Useful for optimization techniques to find the MLE.
- ▶ The MLE can be found by maximizing the log-likelihood.

Log-Likelihood and Maximum

In practice, it is often convenient to work with the natural logarithm of the likelihood function, called the log-likelihood :

$$\mathcal{L}(\theta; \mathbf{y}) = \ln L_n(\theta; \mathbf{y}).$$

Since the logarithm is a monotonic function, the maximum of $\mathcal{L}(\theta; \mathbf{y})$ occurs at the same value of θ as does the maximum of \mathcal{L}_n . If $\mathcal{L}(\theta; \mathbf{y})$ is differentiable in Θ , the necessary conditions for the occurrence of a maximum (or a minimum) are :

$$\frac{\partial \mathcal{L}}{\partial \theta_1} = 0, \quad \frac{\partial \mathcal{L}}{\partial \theta_2} = 0, \quad \dots, \quad \frac{\partial \mathcal{L}}{\partial \theta_k} = 0.$$

MLE for Different Distributions. Exercise : give the proofs

Bernoulli Distribution : MLE for success probability p :

$$\hat{p} = \frac{\text{number of successes}}{\text{total trials}}$$

Normal Distribution : MLE for mean μ and variance σ^2 :

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^n x_i$$

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu})^2$$

Poisson Distribution : MLE for rate parameter λ : $\hat{\lambda} = \frac{1}{n} \sum_{i=1}^n x_i$

Exponential Distribution : MLE for rate parameter λ : $\hat{\lambda} = \frac{n}{\sum_{i=1}^n x_i}$

Multinomial Distribution : MLE for probabilities p_1, p_2, \dots, p_k of k categories in n trials : $\hat{p}_i = \frac{n_i}{n}$, where n_i is the count of category i

Poisson and Exponential Distributions

Poisson Distribution

- ▶ Discrete probability distribution.
- ▶ Models the number of rare events in a fixed interval.
- ▶ Parameter : λ (average rate of events).
- ▶ Probability mass function (PMF) :

$$P(X = k) = \frac{e^{-\lambda} \lambda^k}{k!}$$

- ▶ Mean : λ
- ▶ Variance : λ

Exponential Distribution

- ▶ Continuous probability distribution.
- ▶ Models the time between rare events.
- ▶ Parameter : λ (rate parameter).
- ▶ Probability density function (PDF) :

$$f(x|\lambda) = \begin{cases} \lambda e^{-\lambda x} & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

- ▶ Mean : $\frac{1}{\lambda}$
- ▶ Variance : $\frac{1}{\lambda^2}$

Estimation of the noise level

- ▶ An estimator of the noise level σ^2 is given by

$$\frac{1}{n} \|\mathbf{y} - X\hat{\boldsymbol{\theta}}\|_2^2$$

- ▶ Another estimator which is unbiased is defined by

$$\hat{\sigma}^2 = \frac{1}{n - \text{rank}(X)} \|\mathbf{y} - X\hat{\boldsymbol{\theta}}\|_2^2$$

To show that this estimator is unbiased we need to give more properties of the Hat matrix and Cochran's lemma

Properties of the Hat matrix

Rem: the Hat matrix is defined as $H = X(X^\top X)^{-1}X^\top$

Proposition:

1. H is an orthogonal projection matrix
2. $(I - H)$ is an orthogonal projection matrix
3. $HX = X$
4. $(I - H)X = 0$

Statistical background, χ_k^2 distribution

Let $Z \sim \mathcal{N}(0, 1)$, then the sum of their squares, $Q = \sum_{i=1}^k Z_i^2$, is distributed according to the chi-squared distribution with k degrees of freedom. This is denoted as $Q \sim \chi_k^2$. The chi-squared distribution has one parameter : a positive integer k that specifies the number of degrees of freedom (the number of random variables being summed, i s).

If $a \sim \chi_k^2$ then $\mathbb{E}[a] = k$ and $Var(a) = 2k$

Cochran's lemma

Let $\boldsymbol{\varepsilon} \sim N(0, \sigma^2 I)$ and $\hat{\sigma}^2 = \frac{1}{n-p-1} \|\mathbf{y} - X\hat{\boldsymbol{\theta}}\|_2^2$ and X full rank. Then

$$\begin{aligned}\hat{\boldsymbol{\theta}}_n \text{ and } \hat{\sigma}_n^2 \text{ are independent,} \\ \hat{\boldsymbol{\theta}}_n \sim N(\boldsymbol{\theta}^\star, \sigma^2 (X^T X)^{-1}), \\ (n-p-1) \left(\frac{\hat{\sigma}_n^2}{\sigma^2} \right) \sim \chi_{n-p-1}^2.\end{aligned}\tag{4}$$

Estimation of the noise level, $\hat{\sigma}^2$ is unbiased

Under model I, whenever the matrix X has full rank, we have

$$\mathbb{E}\hat{\sigma}^2 = \sigma^2$$

Proof sketch :

$$\|\mathbf{y} - \hat{\mathbf{y}}\|_2^2 = \mathbf{y}^\top (\text{Id}_n - H) \mathbf{y} = \boldsymbol{\varepsilon}^\top (\text{Id}_n - H) \boldsymbol{\varepsilon}$$

Gaussian case : if $\boldsymbol{\varepsilon}_i \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$, then $\|\mathbf{y} - \hat{\mathbf{y}}\|_2^2 \sim \chi^2$ à $n - \text{rank}(X)$ degrés de liberté

Exercise: Complete the proof

Heteroscedasticity

Model I and Model II are homoscedastic models, *i.e.*, we assume that the noise level σ^2 does not depend on x_i

Heteroscedastic Model : we allow σ^2 to change with the observation i , we denote by $\sigma_i^2 > 0$ the associated variance

$$\hat{\boldsymbol{\theta}} \in \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^{p+1}} \sum_{i=1}^n \left(\frac{y_i - \langle \boldsymbol{\theta}, x_i \rangle}{\sigma_i} \right)^2 = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^{p+1}} (y - X\boldsymbol{\theta})^\top \Omega (y - X\boldsymbol{\theta})$$

with $\Omega = \text{diag}(\frac{1}{\sigma_1^2}, \dots, \frac{1}{\sigma_n^2})$

Exercise: give a closed formula for $\hat{\boldsymbol{\theta}}$ when $X^\top \Omega X$ has full rank

Exercise: give a necessary and sufficient condition for $X^\top \Omega X$ to be invertible

Bias and variance

Proposition: Under model II, whenever the matrix $X = (\mathbf{x}_1, \dots, \mathbf{x}_n)^\top$ has full rank, we have

$$\mathbb{E}(\hat{\boldsymbol{\theta}} \mid X) = \boldsymbol{\theta}^\star$$

$$\text{Var}(\hat{\boldsymbol{\theta}} \mid X) = (X^\top X)^{-1} \sigma^2$$

Proof : The same as in the case of fixed design with the conditional expectation

Rem: We cannot compute the $\mathbb{E}(\hat{\boldsymbol{\theta}})$ nor $\text{Var}(\hat{\boldsymbol{\theta}})$ because the matrix X has full rank is now random !

Rem: One solution is to rely on asymptotic convergence

Asymptotics of $\hat{\boldsymbol{\theta}}$

Under model II, whenever the covariance matrix $\text{cov}(X)$ has full rank, we have

$$\sqrt{n}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*) \xrightarrow{d} \mathcal{N}(0, \sigma^2 S^{-1})$$

with $S = \mathbb{E}[\mathbf{x}\mathbf{x}^\top]$

Outline of the proof : It could happen that $\hat{\boldsymbol{\theta}}$ is not uniquely defined, so we put

$$\hat{\boldsymbol{\theta}} = (X^\top X)^+ X^\top Y$$

where A^+ is the generalized inverse of A

- ▶ With high probability, we have that $X^\top X$ is invertible because $\frac{X^\top X}{n} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \mathbf{x}_i^\top$ goes to S

Asymptotics

Outline of the proof :

- ▶ As a consequence, in the asymptotics we can replace $(X^\top X)^+$ by $(X^\top X)^{-1}$ (that we shall admit)

Then we use that

$$\sqrt{n}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^\star) = \left(\frac{X^\top X}{n} \right)^{-1} \left(\frac{X^\top \boldsymbol{\epsilon}}{\sqrt{n}} \right)$$

- ▶ The term on the right $\frac{X^\top \boldsymbol{\epsilon}}{\sqrt{n}}$ converges to $\mathcal{N}(0, \mathbb{E}[\mathbf{x}\mathbf{x}^\top]\sigma^2)$ in distribution
- ▶ The term on the left $\left(\frac{X^\top X}{n} \right)^{-1}$ goes to S^{-1} in probability

Asymptotics

- In the random design model, since closed formulas for the bias and variance of $\boldsymbol{\theta}$ are lacking; Asymptotics is used to validate the procedure and to build-up the variance estimator

By the previous Proposition, the **variance** to estimate is

$$\sigma^2 S^{-1}$$

a natural “Plug-in” estimator is

$$\hat{\sigma}^2 \hat{S}_n^+$$

with $\hat{\sigma}^2 = \frac{1}{n - \text{rank}(X)} \|\mathbf{y} - X\hat{\boldsymbol{\theta}}\|_2^2$

Rem: It coincides with the estimator in the case of fixed design

Variance estimation

Noise level is conditionally unbiased : Under model Π , whenever the matrix $X = (\mathbf{x}_1, \dots, \mathbf{x}_n)^\top$ has full rank, we have

$$\mathbb{E}(\hat{\sigma}^2 \mid X) = \sigma^2$$

Exercise: Write the proof

Convergence of the variance estimator : Under model Π , if the covariance matrix $\text{cov}(X)$ has full rank, we have

$$\hat{\sigma}^2 \hat{S}_n^+ \rightarrow \sigma^2 S^{-1}$$

in probability

Qualitative variables

A variable is qualitative, when its state space is discrete (non-necessarily numeric)

Exemple : colors, gender, cities, etc.

Classically : “One-hot encoder” consists in representing a qualitative variable with several dummy variables (valued in $\{0, 1\}$)

If each x_i is valued in a_1, \dots, a_K , we define the following K explanatory variables :
 $\forall k \in \llbracket 1, K \rrbracket$, $\mathbb{1}_{a_k} \in \mathbb{R}^n$ is given by

$$\forall i \in \llbracket 1, n \rrbracket, \quad (\mathbb{1}_{a_k})_i = \begin{cases} 1, & \text{if } x_i = a_k \\ 0, & \text{else} \end{cases}$$

Examples

Binary case : M/F, yes/no, I like it/I don't.

Client	Gender
1	H
2	F
3	H
4	F
5	F

→

$$\begin{pmatrix} F & H \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{pmatrix}$$

General case : colors, cities, etc.

Client	Colors
1	Blue
2	Blanc
3	Red
4	Red
5	Blue

→

$$\begin{pmatrix} \text{Blue} & \text{Blanc} & \text{Red} \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

Somme difficulties

Correlations : $\sum_{k=1}^K \mathbb{1}_{a_k} = \mathbf{1}_n$! We can drop-off one modality
(*e.g.*, `drop_first=True` dans `get_dummies` de pandas)

Without intercept, with all modalities : $X = [\mathbb{1}_{a_1}, \dots, \mathbb{1}_{a_K}]$. If $x_{n+1} = a_k$ then
 $\hat{y}_{n+1} = \hat{\boldsymbol{\theta}}_k$

With intercept, with one less modality : $X = [\mathbf{1}_n, \mathbb{1}_{a_2}, \dots, \mathbb{1}_{a_K}]$, dropping-off the first modality

If $x_{n+1} = a_k$ then $\hat{y}_{n+1} = \begin{cases} \hat{\boldsymbol{\theta}}_0, & \text{if } k = 1 \\ \hat{\boldsymbol{\theta}}_0 + \hat{\boldsymbol{\theta}}_k, & \text{else} \end{cases}$

Rem: might give null column in Cross-Validation (if a modality is not present in a CV-fold)

Rem: penalization might help (*e.g.*, Lasso, Ridge)

What if $n < p$?

Many of the things presented before need to be adapted

For instance : if $\text{rank}(X) = n$, then $H = \text{Id}_n$ and $\hat{\mathbf{y}} = X\hat{\boldsymbol{\theta}} = \mathbf{y}$!

The vector space generated by the columns $[\mathbf{x}_0, \dots, \mathbf{x}_p]$ is \mathbb{R}^n , making the observed signal and predicted signal are **identical**

Rem: typical kind of problem in large dimension (when p is large)

Possible solution : variable selection, *cf.* Lasso and greedy methods (coming soon)

Web sites and books

- ▶ Python Packages for OLS :
`statsmodels`
`sklearn.linear_model.LinearRegression`
- ▶ McKinney (2012) about python for statistics
- ▶ Lejeune (2010) about the Linear Model
- ▶ Delyon (2015) Advanced course on regression
<https://perso.univ-rennes1.fr/bernard.delyon/regression.pdf>

SD-TSIA204 - Statistics : linear models

Confidence interval estimation and Hypothesis tests

Ekhiñe Irurozki
Télécom Paris

Confidence Intervals

- ▶ Confidence intervals provide a range of plausible values for a population parameter.
- ▶ They quantify the uncertainty associated with point estimates.
- ▶ A typical confidence interval is of the form : $\hat{\theta} \pm \text{margin of error}$.
- ▶ Margin of error depends on the desired confidence level (e.g., 95% confidence) and the sample data.
- ▶ The confidence level represents the probability that the interval contains the true parameter.
- ▶ Common confidence levels include 90%, 95%, and 99%.
- ▶ The formula for a confidence interval depends on the statistical distribution used.

Confidence interval

Context : regard an estimator $\hat{g}(y_1, \dots, y_n)$ for the value g . We would like to have an interval \hat{I} around \hat{g} which contains g with high probability.

We construct $\hat{I} = [\underline{C}, \overline{C}]$ based on the observations (y_1, \dots, y_n) : confidence interval is a random variable

$$\mathbb{P}(\hat{I} \text{ contains } g) = \mathbb{P}(\underline{C} \leq g \text{ and } \overline{C} \geq g) = 95\%$$

Confidence interval of level $1 - \alpha$

A confidence interval of **level** $1 - \alpha$ for a value g is a function of the sample

$$\hat{I} : (y_1, \dots, y_n) \mapsto \hat{I} = [\underline{C}(y_1, \dots, y_n), \overline{C}(y_1, \dots, y_n)]$$

such that

$$\mathbb{P} \left[g \in \hat{I}(y_1, \dots, y_n) \right] \geq 1 - \alpha$$

or

$$\mathbb{P} \left[g \notin \hat{I}(y_1, \dots, y_n) \right] \leq \alpha$$

Rem: usual choices $\alpha = 5\%, 1\%, 0.1\%$, etc. Defined often by the consideration data complexity / number of observations.

Rem: In the following we will denote confidence interval by CI.

Example : survey

Election survey with two candidates : A and B . Choice of the i th respondent follows Bernoulli distribution having parameter p , with $y_i = 1$ if he votes for A and 0 otherwise.

Aim : estimate p and give a CI

Sample of size n : a reasonable estimator is then $n = 1000$

$$\hat{p} = \frac{1}{n} \sum_{i=1}^n y_i = \bar{y}_n.$$

The goal is to establish an oracle : is there a clear winner in this survey ?

What is the confidence interval for p ?

► is this estimator likely or not ?

Method 1 for CI : concentration inequalities

- Search for an interval $\hat{I} = [\hat{p} - \delta, \hat{p} + \delta]$ such that $\mathbb{P}(p \in \hat{I}) \geq 0.95 \Leftrightarrow$ search for δ such that $\mathbb{P}[|\hat{p} - p| > \delta] \leq 0.05$
- Constituent : **Tchebyschev** inequality

$$\forall \delta > 0, \quad \mathbb{P}(|X - \mathbb{E}(X)| > \delta) \leq \frac{\text{Var}(X)}{\delta^2}$$

For $X = \hat{p} = \frac{1}{n} \sum_{i=1}^n Y_i$ we know that $\mathbb{E}(\hat{p}) = p$ and $\text{Var}(\hat{p}) = \frac{p(1-p)}{n}$:

$$\forall p \in (0, 1), \forall \delta > 0, \quad \mathbb{P}(|\hat{p} - p| > \delta) \leq \frac{p(1-p)}{n\delta^2} \leq \frac{1}{4n\delta^2}$$

Application : for a CI of 95%, find δ such that

$\frac{1}{4n\delta^2} = 0.05$, *i.e.* $\delta = (0.2n)^{-1/2}$. For $n = 1000$, $\hat{p} = 55\%$:

$$\delta = 0.07 ; \quad \hat{I} = [0.48, 0.62]$$

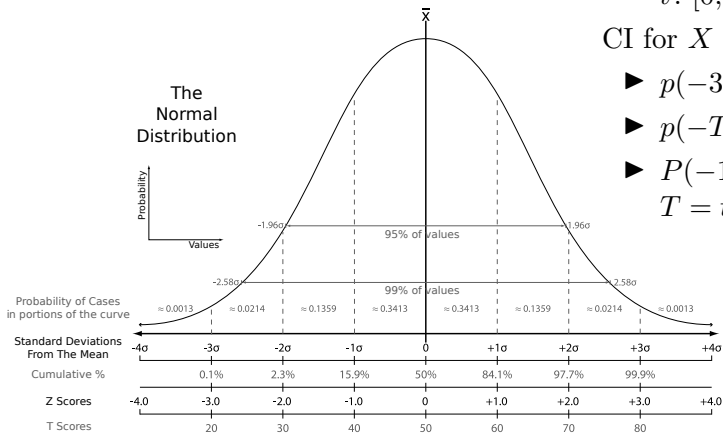
Intervals - Gaussian case

► cumulative distribution (cdf) :
 $P(X \leq x)$

► quantile ($\text{ppf}_\alpha, t_\alpha, z_\alpha$) :
 $t: [0, 1] \rightarrow \mathbb{R}, x \text{ s.t. } p(X \leq x) = \alpha$

CI for $X \sim N(0, 1)$ are easy

- $p(-3 \leq X \leq 3) = 1 - 2 * \text{cdf}(-3)$
- $p(-T \leq X \leq T) = 1 - \alpha, T = t_{1-\alpha/2}$
- $P(-1.96 \leq X \leq 1.96) = 0.95$ or
 $T = t_{(1-.05)/2}$



The quantile function

The quantile function is a fundamental concept in probability and statistics. It is also referred to as the *quantile function* or *inverse cumulative distribution function* or *Percent Point Function*.

Definition : The quantile function of a RV X is a function that maps a probability p to the value x such that $P(X \leq x) = p$.

Notation :

- ▶ The quantile function of a distribution is denoted as t_p .
- ▶ Mathematically, $t_p = x$ iff $P(X \leq x) = p$.
- ▶ The quantile function is useful for finding critical values, confidence intervals, and performing hypothesis tests.

Usage : `norm.ppf(q, loc=0, scale=1)` in `scipy.stats`

Convergence in law

Is the weakest mode of convergence. It defines a relationship not between the RVs themselves but between their cumulative distribution functions.

Convergence in Law : A sequence of RVs $(X_n)_{n \in \mathbb{N}^*}$ converges in law to X if

$$\lim_{n \rightarrow \infty} F_{X_n}(x) = F_X(x) \quad \text{for all } x \text{ where } F_X \text{ is continuous.}$$

This convergence is denoted as : $X_n \xrightarrow{L} X$.

Central Limit Theorem (CLT)

If $(X_n)_{n \in \mathbb{N}^*}$ is a sequence of independent and identically distributed (i.i.d.) RVs with the same mean μ and the same standard deviation $\sigma > 0$, then, by defining $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$, we have :

$$\frac{\bar{X} - \mu}{\sigma/\sqrt{n}} \xrightarrow{L} N(0, 1)$$

In other words, the standardized sum of i.i.d. RVs with finite variance converges in law to the standard normal distribution $N(0, 1)$.

In practice, this theorem is very useful because it allows us to approximate that, for a sufficiently large n , the sum of i.i.d. RVs approximately follows a normal distribution.

Case When n is Sufficiently Large

By "sufficiently large" we generally mean $n \geq 30$.

- ▶ The probability distribution of \bar{X} depends on the distribution of X itself.
- ▶ The CLT asserts that the sequence of RV $U_n = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}}$, converges in law to $N(0, 1)$.
- ▶ In practice, this means that for a sufficiently large n , the RV \bar{X} approximately follows the normal distribution $N(\mu, \sigma^2/n)$ even if the parent distribution is not normal.

Confidence interval of level 95% for μ when we know \bar{X} , n and σ

Setting : Let $X_i \sim P_{\mu, \sigma}$ where σ is known. Goal : give a CI for μ . Rem: for $U \sim N(0, 1)$, we have $\alpha = .05$, $t_{1-\alpha/2} \approx 1.96$ and thus

$P(-1.96 \leq U \leq 1.96) = 0.95$. Applying this result to the variable $U_n = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}}$, we obtain an approximate CI for μ at the 95% confidence level :

$$P\left(-1.96 \leq \frac{\bar{X} - \mu}{\sigma/\sqrt{n}} \leq 1.96\right) = 0.95.$$

Reordering

$$P\left(\bar{X} - 1.96 \frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{X} + 1.96 \frac{\sigma}{\sqrt{n}}\right) = 0.95.$$

In summary, when σ is known and n is sufficiently large, we can construct a CI for μ at the 95% confidence level :

$$\text{CI}_{0.95}(\mu) = \left[\bar{X} - 1.96 \frac{\sigma}{\sqrt{n}}; \bar{X} + 1.96 \frac{\sigma}{\sqrt{n}}\right]$$

What if the population variance is unknown ?

Method 2 for CI : Asymptotic confidence intervals

The survey example : $y_i \in \{0, 1\}$, $n = 1000$,

$$\hat{p} = n^{-1} \sum_{i=1}^n y_i = 0.55$$

We assume that n is sufficiently large, such that

$$\sqrt{n} \left(\frac{\hat{p} - p}{\hat{\sigma}} \right) \sim \mathcal{N}(0, 1)$$

$$\hat{\sigma}^2 = n^{-1} \sum_{i=1}^n (y_i - \hat{p})^2 = \hat{p} - \hat{p}^2$$

We know the quantiles of the normal distribution (numerically)

$$t_{1-0.05/2} = \text{norm.ppf}(1 - 0.05/2) \simeq 1.96$$

Following the CLT and approximation of the Gaussian quantiles

$$\mathbb{P} \left[-1.96 < \sqrt{n} \frac{0.55 - p}{\hat{\sigma}} < 1.96 \right] \approx 0.95$$

new CI : $\hat{I} = [0.52, 0.58]$: better ! (**more optimistic**)

The Student's t-distribution

- ▶ the quantile function is `t.ppf(q, df, loc=0, scale=1)` in `scipy.stats`
- ▶ It is similar in shape to the standard normal distribution but has heavier tails.
- ▶ The t-distribution is used when the population standard deviation is unknown and sample sizes are small.
- ▶ It depends on a single parameter called degrees of freedom (ν), which determines the shape of the distribution. For $\nu = 1$ t_ν becomes the standard Cauchy distribution, whereas for $\nu \rightarrow \infty$ it becomes the standard normal distribution.

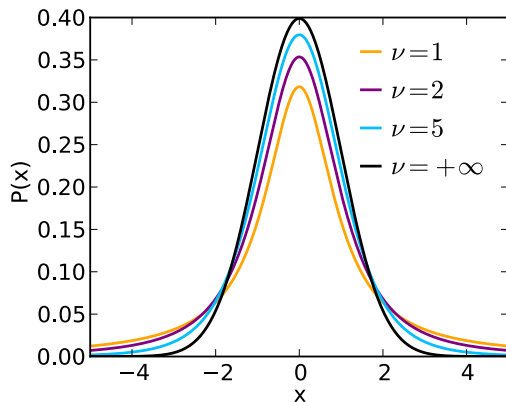


Figure – Probability density function of the Student's t-distribution with different degrees of freedom (Wikipedia).

The Chi-Squared Distribution (χ^2)

Definition : The chi-squared (χ^2) distribution is a continuous probability distribution that arises in various statistical applications.

Parameters : The χ^2 distribution depends on a single parameter, which is the degrees of freedom (ν). It is denoted as χ_ν^2 .

Probability Density Function (PDF) : The probability density function of the χ^2 distribution is given by :

$$f(x; \nu) = \frac{1}{2^{\frac{\nu}{2}} \Gamma(\frac{\nu}{2})} x^{\frac{\nu}{2}-1} e^{-\frac{x}{2}}, \quad x > 0$$

where $\Gamma(\cdot)$ represents the gamma function.

Chi-Squared Distribution with n Degrees of Freedom

Let Y_1, Y_2, \dots, Y_n be independent RVs, each following the standard normal distribution $N(0, 1)$. Then, the RV

$$Z = Y_1^2 + Y_2^2 + \dots + Y_n^2 \sim \chi_n^2 \tag{1}$$

follows the chi-squared distribution with n degrees of freedom.

Relating distributions

Definition Let U be a RV following the standard normal distribution $N(0, 1)$, and let Z be a RV, independent of U , following a chi-squared (χ_ν^2) distribution with ν degrees of freedom (where $\nu \in \mathbb{N}^*$). The t-Student RV, denoted as T , is defined as :

$$T = \frac{U}{\sqrt{Z/\nu}} \quad (2)$$

- ▶ T follows the t-Student distribution with ν degrees of freedom.
- ▶ It arises in statistical inference, particularly when dealing with small sample sizes and unknown population standard deviation.

Application

Let $X \sim N(\mu, \sigma^2)$ and $S^2 = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^2$ be the empirical variance of the sample, then the RV

$$T = \frac{\bar{X} - \mu}{S/\sqrt{n-1}}$$

follows the Student's t-distribution, $T \sim T_{n-1}$.

Proof sketch : Consider the RV $U = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}}$. We know that $U \sim N(0, 1)$.

Furthermore, we have that $\frac{nS^2}{\sigma^2}$ follows the χ_{n-1}^2 distribution with $\nu = n - 1$ degrees of freedom.

$$T = \frac{\frac{\bar{X} - \mu}{\sigma/\sqrt{n}}}{\sqrt{\frac{nS^2}{\sigma^2(n-1)}}} = \frac{\bar{X} - \mu}{S/\sqrt{n-1}}.$$

CI for the Mean - unknown variance

For example, if $1 - \alpha = 0.95$ and $n = 10$, then $t_{1-\alpha/2} = 2.262$.

- We can easily isolate μ by rearranging the equation :

$$\begin{aligned} -t_{1-\alpha/2} \leq T \leq t_{1-\alpha/2} &\Leftrightarrow -t_{1-\alpha/2} \leq \frac{\bar{X} - \mu}{S/\sqrt{n-1}} \leq t_{1-\alpha/2} \\ &\Leftrightarrow \bar{X} - t_{1-\alpha/2} \frac{S}{\sqrt{n-1}} \leq \mu \leq \bar{X} + t_{1-\alpha/2} \frac{S}{\sqrt{n-1}}. \end{aligned}$$

- Therefore, we obtain a random CI for μ at the confidence level $1 - \alpha$:

$$\text{CI}_{1-\alpha}(\mu) = \left[\bar{X} - t_{1-\alpha/2} \frac{S}{\sqrt{n-1}}, \bar{X} + t_{1-\alpha/2} \frac{S}{\sqrt{n-1}} \right]$$

CI (method 2) for the regression coefficients (I)

Proposition

$$\text{If } \epsilon \sim \mathcal{N}(0, \sigma^2 \text{Id}_n), \text{ then } T_j = \frac{\hat{\theta}_j - \theta_j^*}{\hat{\sigma} \sqrt{(X^\top X)^{-1}_{j,j}}} \sim \mathcal{T}_{n-\text{rang}(X)}$$

where $\mathcal{T}_{n-\text{rang}(X)}$ is a Student- t distribution with $n - \text{rang}(X)$ degrees of freedom.

Its density, quantiles, etc..., can be computed numerically and are accessible in any software.

proof Recall that $(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*) \sim N(0, \sigma^2 (X^T X)^{-1})$ and that $(n - p - 1) \hat{\sigma}^2 / \sigma^2 \sim \chi^2_{n-p-1}$ for X full rank and $\hat{\sigma}^2 := (n - p - 1)^{-1} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$ an unbiased estimator for σ^2 . It follows that

$$\frac{\frac{\hat{\theta}_i - \theta_i}{\sqrt{\sigma^2 (X^T X)^{-1}_{jj}}}}{\sqrt{\frac{(n-p-1) \hat{\sigma}^2}{(n-p-1) \sigma^2}}} = \frac{\hat{\theta}_i - \theta_i}{\sqrt{\hat{\sigma}^2 (X^T X)^{-1}_{jj}}} \sim T_{n-p-1}$$

CI for the regression coefficients (II)

Under the Gaussian assumption, since

$$T_j = \frac{\hat{\theta}_j - \theta_j^*}{\hat{\sigma} \sqrt{(X^\top X)_{j,j}^{-1}}} \sim \mathcal{T}_{n-p-1}$$

and noting $t_{1-\alpha/2}$ a quantile of order $1 - \alpha/2$ of the distribution \mathcal{T}_{n-p-1} ,

$$\left[\hat{\theta}_j - t_{1-\alpha/2} \hat{\sigma} \sqrt{(X^\top X)_{j,j}^{-1}}, \hat{\theta}_j + t_{1-\alpha/2} \hat{\sigma} \sqrt{(X^\top X)_{j,j}^{-1}} \right]$$

for the quantity θ_j^* .

Rem: $\mathbb{P}(|T_j| < t_{1-\alpha/2}) = 1 - \alpha$ since the Student- t distribution is symmetric.

CI for the predicted values

Now we would like to construct a CI for the predicted value at a single (new) given point $x = (1, x_1, \dots, x_p)^\top \in \mathbb{R}^{p+1}$.

The predicted value at x (under the true model) is defined as

$$y^* = x^\top \boldsymbol{\theta}^*.$$

Under the Gaussian assumption, with the same notation, the following confidence interval is of level $1 - \alpha$

$$\left[x^\top \hat{\boldsymbol{\theta}} - t_{1-\alpha/2} \hat{\sigma} \sqrt{x^\top (X^\top X)^{-1} x}, x^\top \hat{\boldsymbol{\theta}} + t_{1-\alpha/2} \hat{\sigma} \sqrt{x^\top (X^\top X)^{-1} x} \right]$$

for the quantity y^* .

CI for the new values (aka, Prediction interval)

The CI from above is for the regression hyperplane, i.e. it is reflecting uncertainty of the fitted values.

How to build a CI for a new value at a single (new) given point $x = (1, x_1, \dots, x_p)^\top \in \mathbb{R}^{p+1}$?

A new predicted value at x (under the true model) is defined as

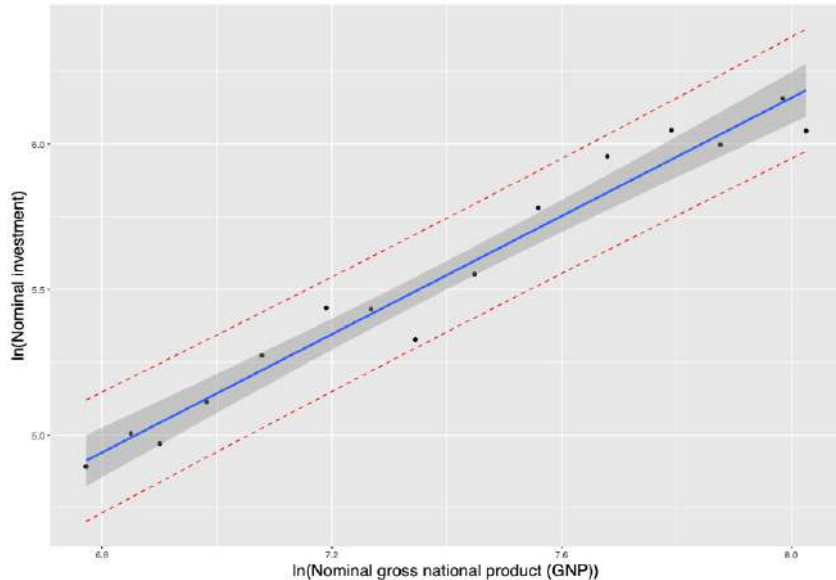
$$y = y^* + \epsilon.$$

where $\epsilon \sim \mathcal{N}(0, \sigma^2)$.

One can show that, in this case, and with the same notation, the following confidence interval is of level α

$$\left[x^\top \hat{\boldsymbol{\theta}} - t_{1-\alpha/2} \hat{\sigma} \sqrt{1 + x^\top (X^\top X)^{-1} x}, x^\top \hat{\boldsymbol{\theta}} + t_{1-\alpha/2} \hat{\sigma} \sqrt{1 + x^\top (X^\top X)^{-1} x} \right]$$

Example : Investment data (II).



Hypothesis testing : General principle

Context

- ▶ We observe X_1, \dots, X_n from a common distribution \mathcal{P}
- ▶ We are interested in $\theta \in \Theta$, a parameter of \mathcal{P}

The goal is to decide whether an assumption on θ is likely (or not)

$$\mathcal{H}_0 = \{\theta \in \Theta_0\}$$

against some alternative

$$\mathcal{H}_1 = \{\theta \in \Theta_1\}$$

Call \mathcal{H}_0 the null hypothesis, \mathcal{H}_1 the alternative

Determine a test statistic $T(X_1, \dots, X_n)$ and a region R such that if

$$T(X_1, \dots, X_n) \in R \Rightarrow \text{we reject } \mathcal{H}_0$$

In other words the observed data discriminates between \mathcal{H}_0 and \mathcal{H}_1

General principle : Hypothesis Testing for "Heads or Tails"

Scenario : You are given a fair coin, and you want to test whether it's indeed fair or biased towards heads.

Hypotheses :

- ▶ Null Hypothesis (\mathcal{H}_0) : The coin is fair, and the probability of getting heads ($P(\text{Heads})$) is 0.5.
- ▶ Alternative Hypothesis (\mathcal{H}_1) : The coin is biased towards heads, and $P(\text{Heads}) > 0.5$.

Test Statistic : You decide to flip the coin 100 times and record the number of heads (X).

Statistical Test : Using a significance level of $\alpha = 0.05$, perform a one-sided hypothesis test to determine if there is enough evidence to reject the null hypothesis in favor of the alternative hypothesis based on the observed number of heads.

Conclusion : You may "reject" the null hypothesis and conclude that the coin is biased towards heads or not reject it.

Do we reject or do we accept ?

In most practical situations, \mathcal{H}_0 is simple, i.e.,
$$\Theta_0 = \{\theta_0\}$$

and $\Theta_1 = \Theta \setminus \Theta_0$ is large

(\mathcal{H}_0 is often an hypothesis on which we care particularly, e.g., something acknowledged to be true, easy to formulate)

We only reject \mathcal{H}_0 : If \mathcal{H}_0 is not rejected we cannot conclude \mathcal{H}_0 is true because \mathcal{H}_1 is too general

e.g. $\{p \in [0, 0.5[\cup]0.5, 1]\}$ can not be rejected !

2 types of error

	\mathcal{H}_0	\mathcal{H}_1
\mathcal{H}_0 is not rejected	Correct (True positive)	Wrong (False negative)
\mathcal{H}_0 is rejected	Wrong (False positive)	Correct (True negative)

- Type I : probability of a wrong reject

$$P(T(X_1, \dots, X_n) \in R \mid \mathcal{H}_0)$$

- Type II : probability of wrong non-reject

$$P(T(X_1, \dots, X_n) \notin R \mid \mathcal{H}_1)$$

Significance level and power

Significance level α if $\limsup_{n \rightarrow +\infty} P(T(X_1, \dots, X_n) \in R \mid \mathcal{H}_0) \leq \alpha$

We speak of 95%-test when α is 0.05%

Consistency : A test statistics (given by $T(X_1, \dots, X_n)$ and a region R) is said to be α -consistent if the significant level is α and if the power goes to one, i.e.,

$$\limsup_{n \rightarrow +\infty} P(T(X_1, \dots, X_n) \in R \mid \mathcal{H}_0) \leq \alpha$$

$$\lim_{n \rightarrow \infty} P(T(X_1, \dots, X_n) \in R \mid \mathcal{H}_1) = 1$$

Test statistic and reject region

Goal : to build a α -consistent test

- (1) Define the test statistic $T(X_1, \dots, X_n)$ and the level α you wish
- (2) Do some maths to determine a reject region R that achieves a significance level α
- (3) Prove the consistency
- (4) Rule decision : reject whenever $T_n(X_1, \dots, X_n) \in R$

Famous tests

- ▶ Test of the equality of the mean for 1 sample
- ▶ Test of the equality of the means between 2 samples
- ▶ Chi-square test for the variance
- ▶ Chi-square test of independence
- ▶ Regression coefficient non-effects test

Conformity Test for the Mean of a Normal RV with Known Variance

Let $X_i \sim N(\mu, \sigma^2)$ and $\bar{X} = \sum_{i=1}^n X_i$. Is $\mu = \mu_0$? We will proceed in 4 steps :

Step 1 : Formulate Hypotheses Let's start with the null hypothesis $\mathcal{H}_0 : \mu = \mu_0$, where the alternative hypothesis is $H_1 : \mu \neq \mu_0$. We assume \mathcal{H}_0 is true.

Step 2 : Distribution of X Under \mathcal{H}_0 As a result of \mathcal{H}_0 , X follows a normal distribution $N(\mu_0, \sigma^2)$, and consequently :

$$U = \frac{\bar{X} - \mu_0}{\sigma/\sqrt{n}} \sim N(0, 1).$$

Step 3 : Define the reject region Let's choose a significance level α , which we consider negligible. This leads to an interval $[-t_{1-\alpha/2}, t_{1-\alpha/2}]$ within which the variable U (our decision variable) has a probability of $(1 - \alpha)$ of falling if the null hypothesis is true. Consequently, if \mathcal{H}_0 is true, we have $P(|U| > t_{1-\alpha/2}) = \alpha$. Neglecting the probability α means considering it very unlikely to find U outside the interval $[-t_{1-\alpha/2}, t_{1-\alpha/2}]$ if the null hypothesis is true. The reject region is thus $R =] - \infty, -t_{1-\alpha/2}[\cup] t_{1-\alpha/2}, \infty [$

Interpreting the Test Results

Step 4 : Interpretation of Results Check whether $u \in R$:

- ▶ If $u \in R$ we prefer to reject the hypothesis \mathcal{H}_0 . However, it's important to acknowledge that by doing so, we are accepting the risk α of making a Type I error, meaning we might reject \mathcal{H}_0 incorrectly.
- ▶ If $u \notin R$ it does not imply that \mathcal{H}_0 is true. Rather, it indicates that the collected data is not in contradiction with the hypothesis. In other words, we are unable to conclude in favor or against the hypothesis. In practical applications, this is often less problematic than it may seem because the focus is on avoiding the incorrect rejection of \mathcal{H}_0 , while maintaining the status quo corresponds to retaining the hypothesis.

Step 5(*) : Calculating the p-value Calculate the probability of observing a test statistic as extreme as $|u|$ in both tails of the distribution. The resulting p-value represents the likelihood of observing such an extreme result under the null hypothesis.

Hypothesis Testing Example : Gaussian Mean (Two-Sided Test)

Scenario : Suppose a manufacturer produces light bulbs, and the claimed mean lifespan of these bulbs is 1000 hours with a known standard deviation of 50 hours. To test the manufacturer's claim, a random sample of 36 light bulbs is selected and tested. The sample has a mean lifespan of 990 hours. We want to determine if there is enough evidence to reject the manufacturer's claim at a 5% significance level.

Hypotheses :

- ▶ Null Hypothesis (\mathcal{H}_0) : The mean lifespan of the bulbs produced by the manufacturer is equal to 1000 hours, i.e., $\mu = 1000$ hours.
- ▶ Alternative Hypothesis (\mathcal{H}_1) : The mean lifespan of the bulbs is not equal to 1000 hours, i.e., $\mu \neq 1000$ hours (Two-Sided Test).

Step 1 : Formulate Hypotheses

- ▶ $\mathcal{H}_0 : \mu = 1000$ hours
- ▶ $\mathcal{H}_1 : \mu \neq 1000$ hours (Two-Sided Test)

Hypothesis Testing Example : Gaussian Mean (Two-Sided Test, Continued)

Step 2 : Distribution of X Under \mathcal{H}_0

Calculate the test statistic (z) using the sample data, population mean (μ), and standard deviation (σ).

$$U = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}} = \frac{990 - 1000}{50/\sqrt{36}} = -1.2$$

Step 3 : Define reject region Determine the two-tailed critical values at the 5% level of significance :

$$t_{1-\alpha/2} = -1.96 \text{ and } 1.96 \text{ (from the standard normal distribution)}$$

$$R =] - \infty, -t_{1-\alpha/2}[\cup] t_{1-\alpha/2}, \infty[=] - \infty, -1.96[\cup] 1.96, \infty[$$

Step 4 : Interpretation of Results Since $u \notin R$ we don't reject \mathcal{H}_0 .

Step 5 : p-value, type-I and type-II errors

One-Sided vs. Two-Sided Tests

One-Sided Test :

- ▶ Used to detect an effect in one specific direction (greater than or less than).
- ▶ Has a single critical region in one tail of the distribution.
- ▶ Hypotheses :
 - ▶ \mathcal{H}_0 : No effect or no difference ($\mu = \mu_0$).
 - ▶ \mathcal{H}_1 : Effect or difference in a specific direction ($\mu > \mu_0$ or $\mu < \mu_0$).

Two-Sided Test :

- ▶ Used to detect an effect in either direction (greater than or less than).
- ▶ Has two critical regions in both tails of the distribution.
- ▶ Hypotheses :
 - ▶ \mathcal{H}_0 : No effect or no difference ($\mu = \mu_0$).
 - ▶ \mathcal{H}_1 : Effect or difference in either direction ($\mu \neq \mu_0$).

Example :

- ▶ One-Sided Test : Testing if a new drug increases blood pressure ($\mathcal{H}_0 : \mu \leq \mu_0$ vs. $\mathcal{H}_1 : \mu > \mu_0$).
- ▶ Two-Sided Test : Testing if a scale is accurate ($\mathcal{H}_0 : \mu = \mu_0$ vs. $\mathcal{H}_1 : \mu \neq \mu_0$).

The p-value

Quantifies the strength of evidence against the null hypothesis (\mathcal{H}_0). The p-value represents the probability of obtaining a result as extreme as, or more extreme than, the one observed, assuming that \mathcal{H}_0 is true.

- ▶ Calculate the test statistic u based on the sample data and \mathcal{H}_0 assumptions.
- ▶ Determine the direction of the test (two-tailed, left-tailed, or right-tailed) based on the alternative hypothesis (\mathcal{H}_1).
- ▶ For a two-tailed test, calculate the probability of observing a test statistic as extreme as $|u|$ in both tails of the distribution.
- ▶ For a one-tailed test (left-tailed or right-tailed), calculate the probability of observing a test statistic as extreme as u in the specified tail.

Interpreting the p-value :

- ▶ If the p-value is less than the chosen significance level α , it suggests strong evidence against the null hypothesis, and we may reject \mathcal{H}_0 .
- ▶ If the p-value is greater than or equal to α , it implies that the observed data is consistent with \mathcal{H}_0 , and we do not have sufficient evidence to reject it.

Test of no-effect : Gaussian case

Gaussian Model

$$y_i = \theta_0^* + \sum_{k=1}^p \theta_k^* x_{i,k} + \varepsilon_i$$

$$x_i^\top = (1, x_{i,1}, \dots, x_{i,p}) \in \mathbb{R}^{p+1} \text{ (deterministic)}$$

$$\varepsilon_i \stackrel{i.i.d}{\sim} \mathcal{N}(0, \sigma^2), \text{ for } i = 1, \dots, n$$

Rem: Let $X = (x_1, \dots, x_n)^\top \in \mathbb{R}^{n \times (p+1)}$ of full rank, and $\hat{\sigma}^2 = \|\mathbf{y} - X\hat{\boldsymbol{\theta}}\|_2^2 / (n - (p+1))$, then

$$\hat{T}_j = \frac{\hat{\theta}_j - \theta_j^*}{\hat{\sigma} \sqrt{(X^\top X)^{-1}_{j,j}}} \sim \mathcal{T}_{n-(p+1)}$$

Rem: $\theta_j^* = 0$ then column X_j has no effect on Y

Test of no-effect : Gaussian case

Goal : Develop a test of significance level α to check whether $\theta_j^* = 0$

Null hypothesis, $\mathcal{H}_0 : \theta_j^* = 0$, equivalently, $\Theta_0 = \{\theta \in \mathbb{R}^p : \theta_j = 0\}$

Under \mathcal{H}_0 , we know the value of \hat{T}_j :

$$T_j := \frac{\hat{\theta}_j}{\hat{\sigma} \sqrt{(X^\top X)^{-1}_{j,j}}} \sim \mathcal{T}_{n-(p+1)}$$

Choosing $R = [-t_{1-\alpha/2}, t_{1-\alpha/2}]^c$ with $t_{1-\alpha/2}$ the $1 - \alpha/2$ -quantile of $\mathcal{T}_{n-(p+1)}$, we decide to reject \mathcal{H}_0 whenever

$$|\hat{T}_j| > t_{1-\alpha/2}$$

Link between IC and test

Reminder (Gaussian model) :

$$IC_\alpha := \left[\hat{\theta}_j - t_{1-\alpha/2} \hat{\sigma} \sqrt{(X^\top X)_{j,j}^{-1}}, \hat{\theta}_j + t_{1-\alpha/2} \hat{\sigma} \sqrt{(X^\top X)_{j,j}^{-1}} \right]$$

is a CI at level α for θ_j^* . Stating “ $0 \in IC_\alpha$ ” means

$$|\hat{\theta}_j| \leq t_{1-\alpha/2} \hat{\sigma} \sqrt{(X^\top X)_{j,j}^{-1}} \quad \Leftrightarrow \quad \frac{|\hat{\theta}_j|}{\hat{\sigma} \sqrt{(X^\top X)_{j,j}^{-1}}} \leq t_{1-\alpha/2}$$

It is equivalent to accepting the hypothesis $\theta_j^* = 0$ at level α . The smallest α such that $0 \in IC_\alpha$ is called the **p-value**.

Rem: Taking α close to zero IC_α covers the full space, hence one can find (by continuity) an α achieving equality in the aforementioned equations.

“Diabetes” data set

patient	age	sex	bmi	bp	Serum measurements						Resp
	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	y
1	59	2	32.1	101	157	93	38	4	4.9	87	151
2	48	1	21.6	87	183	103	70	3	3.9	69	75
...
...
441	36	1	30.0	95	201	125	42	5	5.1	85	220
442	36	1	19.6	71	250	133	97	3	4.6	92	57

$n = 442$ patients having diabetes, $p = 10$ variables “baseline” body mass index (bmi), average blood pressure (bp), etc have been measured.

Goal : predict disease progression one year in advance after the “baseline” measurement.

- ▶ Each variable of the data set from *sklearn* has been previously standardized.
- ▶ We apply an “expensive” version of the **forward variable selection** method

“Diabetes” data set

- We define a vector of covariates with intercept $\tilde{X} = (\mathbf{1}, \mathbf{x}_1, \dots, \mathbf{x}_{10})$.

Step 0

- for each variable \tilde{X}_k , $k = 1, \dots, 11$, we consider the model

$$\mathbf{y} \simeq \theta_k \mathbf{x}_k$$

- we test whether its regression coefficient equals zero, *i.e.*

$$H_0 : \theta_k = 0$$

using the statistic $\frac{\hat{\theta}_k}{\hat{s}_k}$ with \hat{s}_k being the estimated standard deviation.

- we compare all of the p -values, and keep the one possessing the smallest p -value. We save the residuals in the vector V_0 .

“Diabetes” data set

Step ℓ We have selected ℓ variable(s) : $\tilde{X}^{(\ell)} \in \mathbb{R}^\ell$. Those not selected are noted $\tilde{X}^{(-\ell)} \in \mathbb{R}^{p-\ell}$. We possess the vector of residuals $V_{\ell-1}$ calculated on the previous step.

- for each variable \mathbf{x}_k in $\tilde{X}^{(-\ell)}$, we consider the model

$$V_{\ell-1} \simeq \theta_k \mathbf{x}_k$$

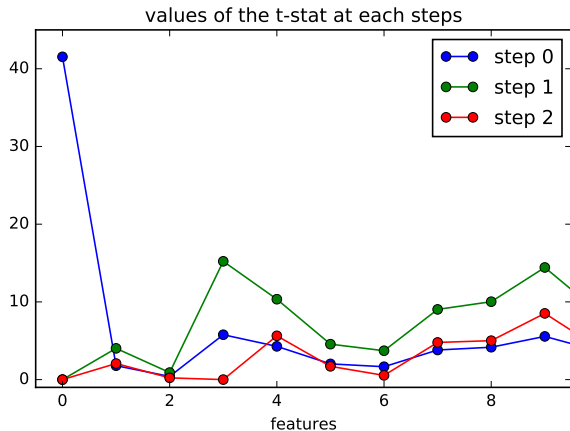
- we test if its regression coefficient equal zero, *i.e.*

$$H_0 : \theta_k = 0$$

using the test statistic $\frac{\hat{\theta}_k}{\hat{s}_k}$ with \hat{s}_k being the estimated standard deviation.

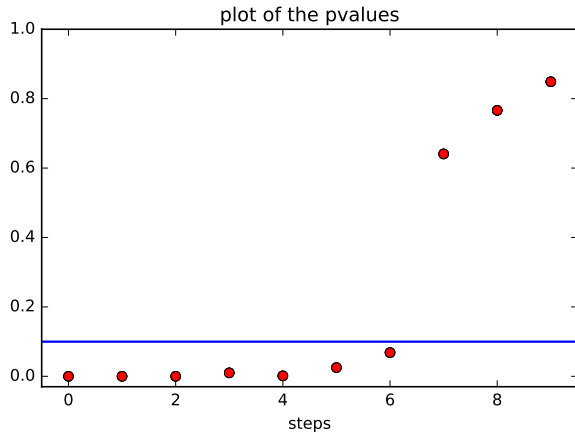
- we compare all of the p -values, and keep the one possessing the smallest p -value. We save the residuals in the vector V_ℓ .

Values of the test statistics at each step



- The test statistic of the selected variable is 0 on the following steps.
- The intercept is the first selected variable, then x_3 , etc

Values of the test statistics at each step



► Sequence of the selected variables with the test size 0.1 :

[0, 3, 9, 5, 4, 2, 7]

ROC curve, Medical context

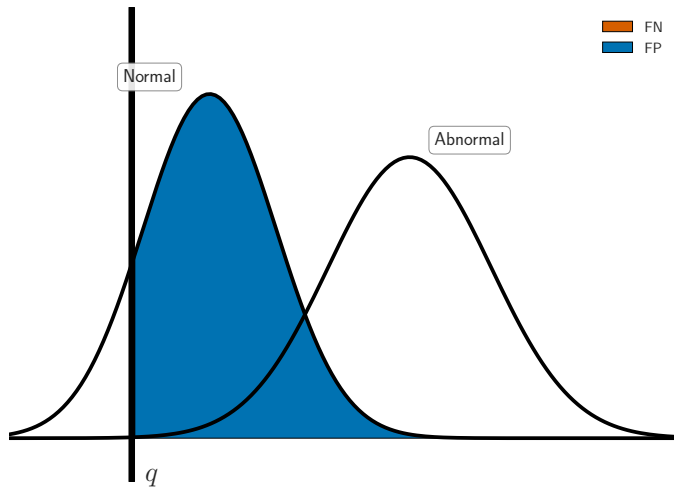
- ▶ A group of patients $i = 1, \dots, n$ is followed for disease screening.
- ▶ For each individual, the test relies on a random variable $X_i \in \mathbb{R}$ and a threshold $q \in \mathbb{R}$

as soon as $X_i > q$ the test is **positive**
o.w. the test is **negative**

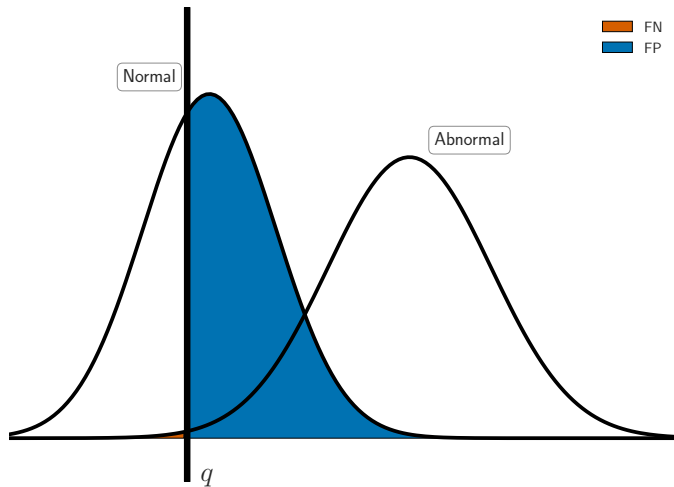
Set of possible configurations

	Normal H_0	Sick H_1
negative	true negative	false negative
positive	false positive	true positive

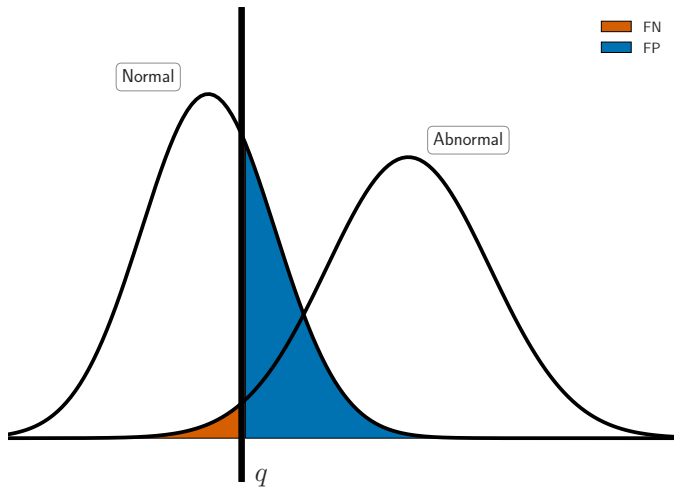
False positive vs. false negative



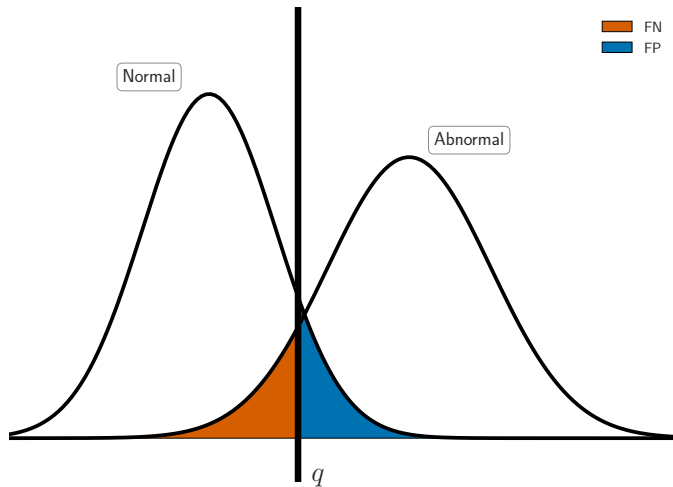
False positive vs. false negative



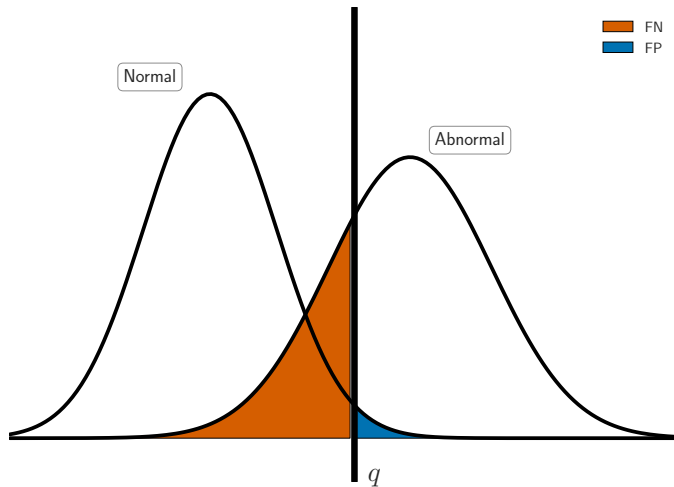
False positive vs. false negative



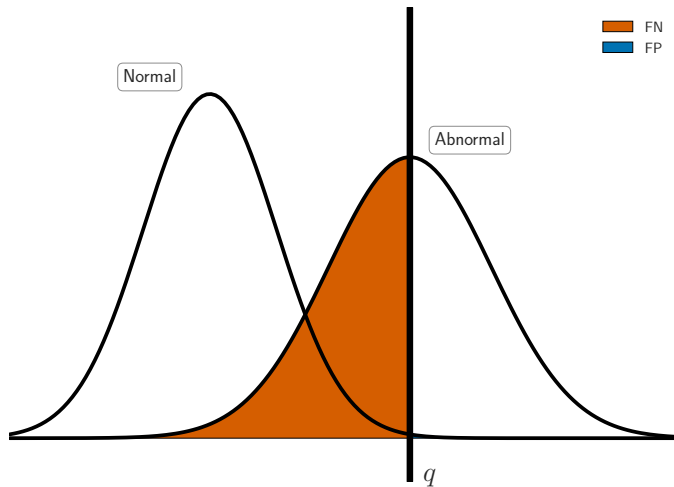
False positive vs. false negative



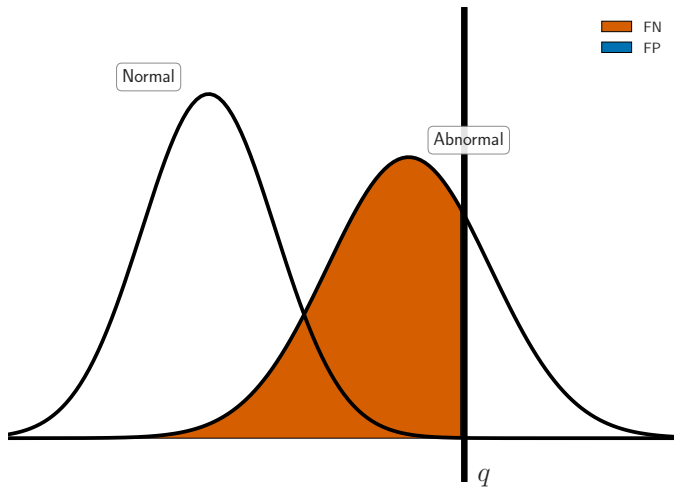
False positive vs. false negative



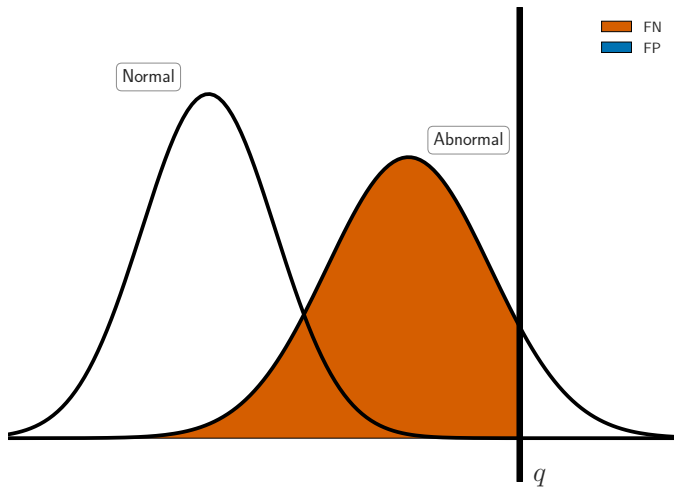
False positive vs. false negative



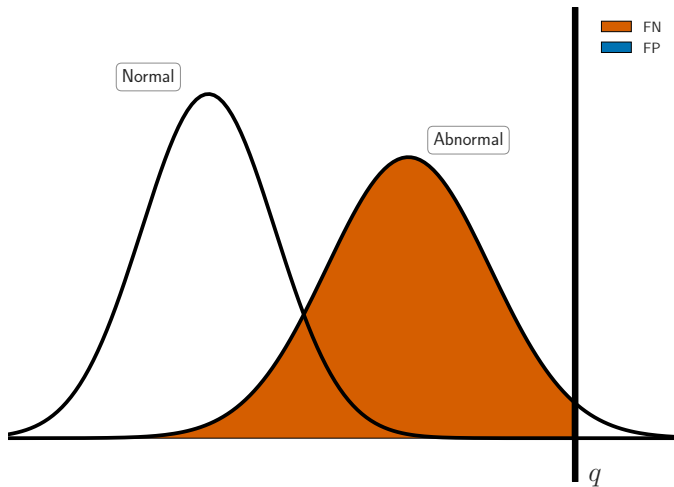
False positive vs. false negative



False positive vs. false negative



False positive vs. false negative



Sensitivity - Specificity

- ▶ Assumption : Normal individuals have the same c.d.f. F
- ▶ Assumption : Sick individual have the same c.d.f G

Definition

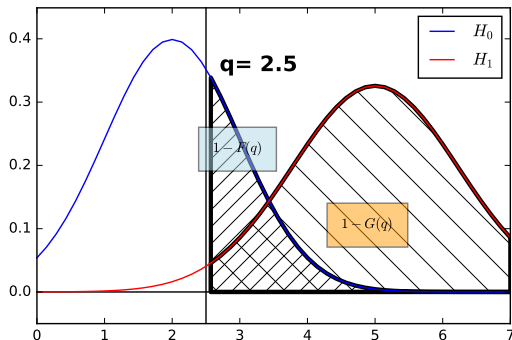
- ▶ Sensitivity : $sen(q) = 1 - G(q)$ (1- type 2nd error)
- ▶ Specificity : $spe(q) = F(q)$ (1- type 1st error)

ROC curve

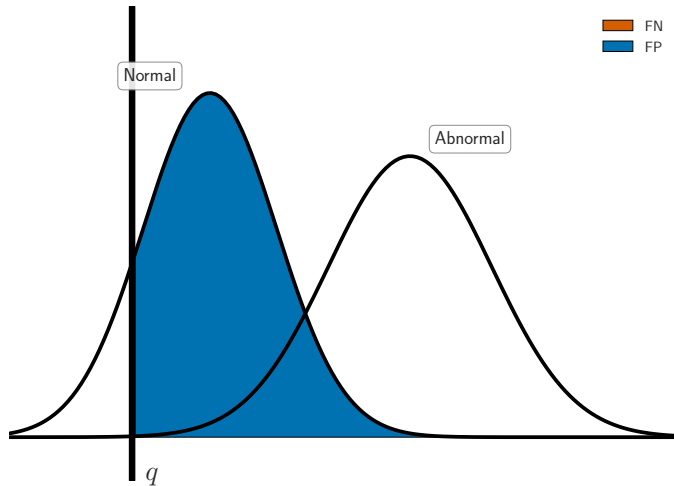
Definition The ROC curve is the curve described by $(1 - spe(q), sen(q))$, when $q \in \mathbb{R}$. Hence, it is the function $[0, 1] \rightarrow [0, 1]$

$$ROC(t) = 1 - G(F^{-}(1 - t))$$

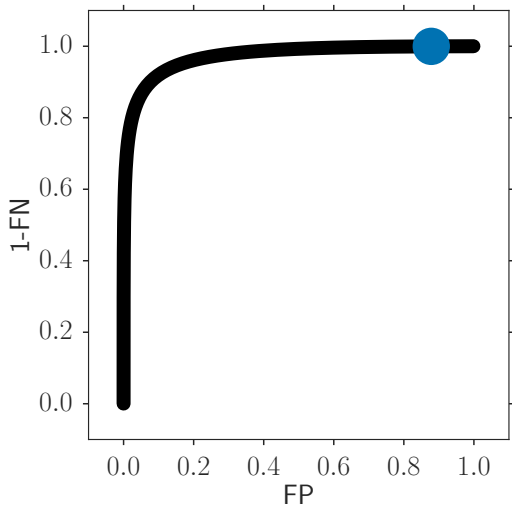
where $F^{-}(1 - t) = \inf\{x \in \mathbb{R} : F(x) \geq 1 - t\}$.



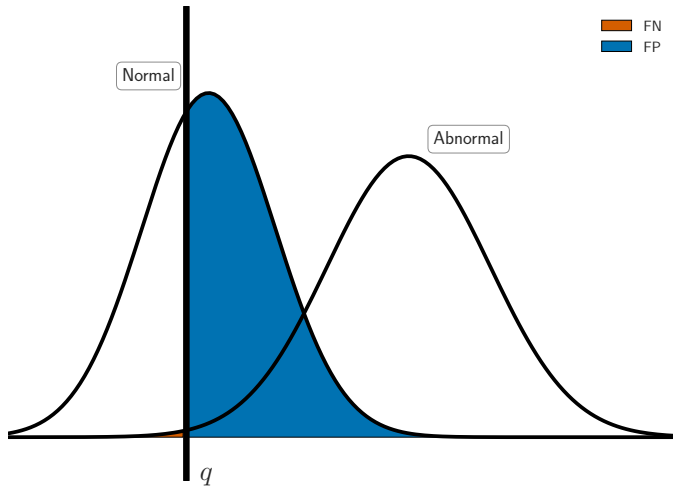
ROC Curve



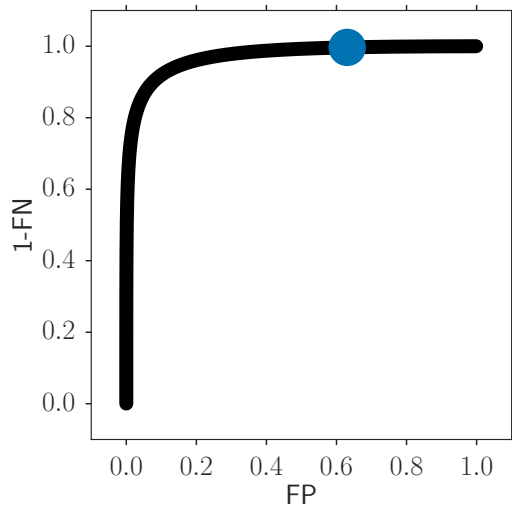
ROC Curve



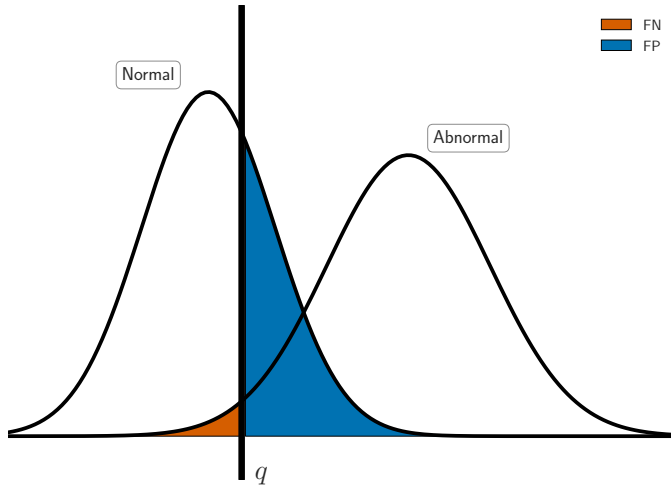
ROC Curve



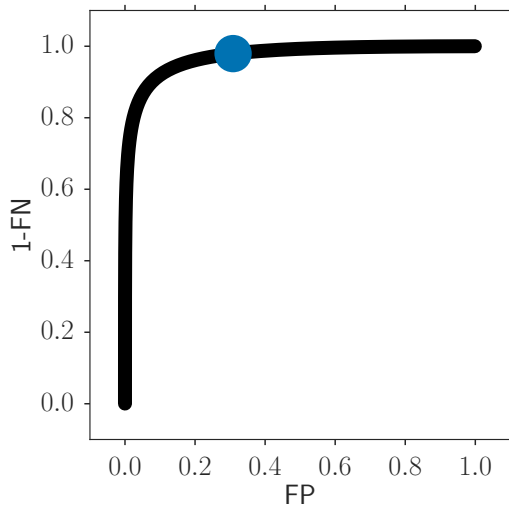
ROC Curve



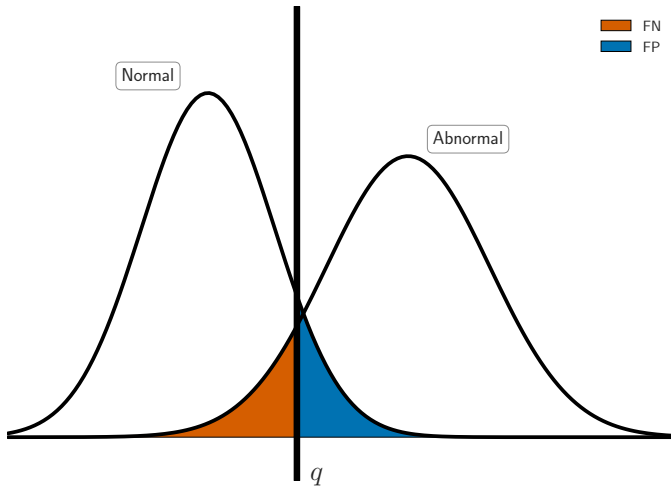
ROC Curve



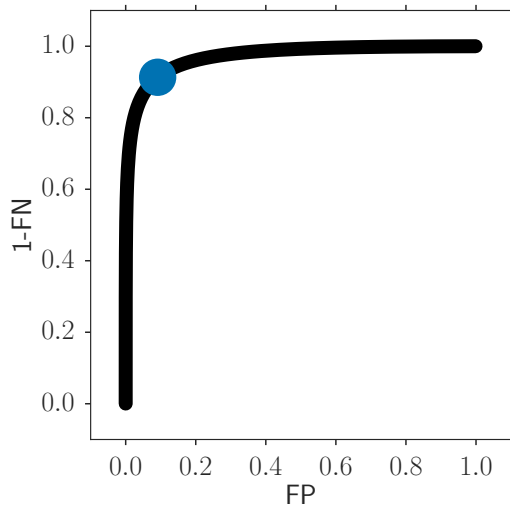
ROC Curve



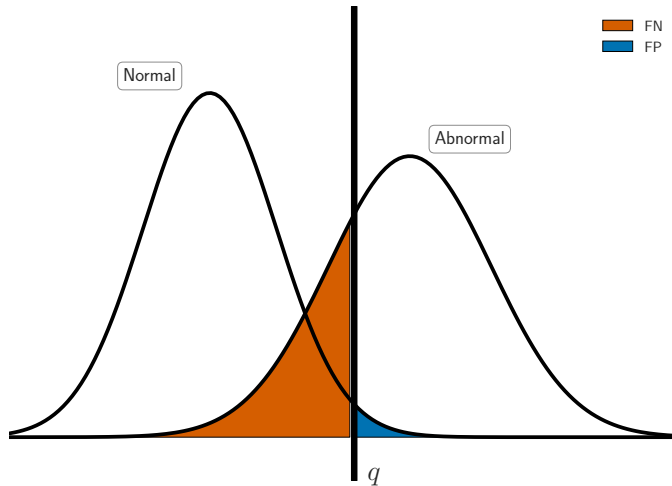
ROC Curve



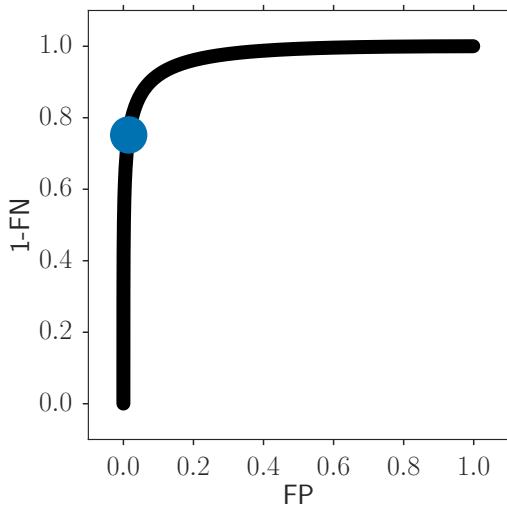
ROC Curve



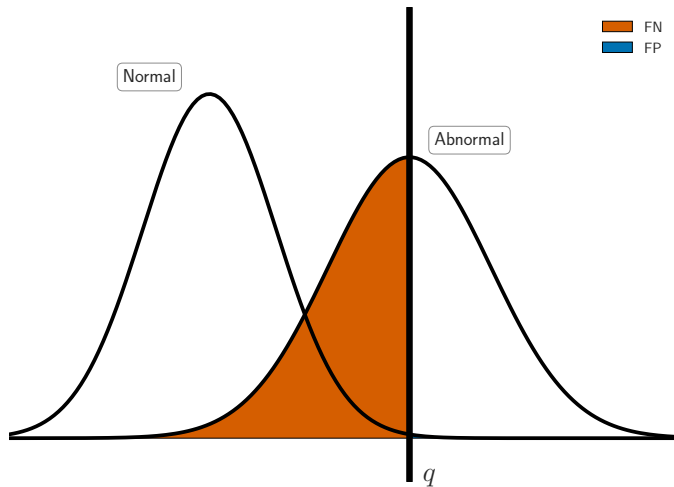
ROC Curve



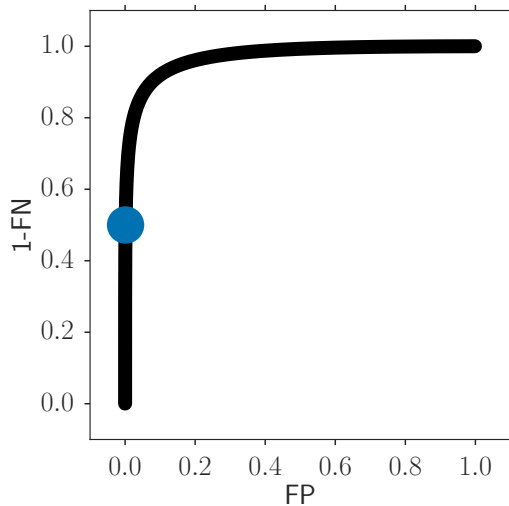
ROC Curve



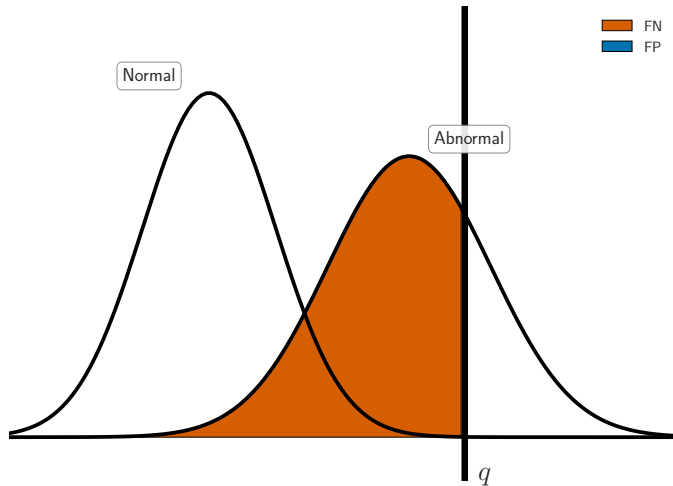
ROC Curve



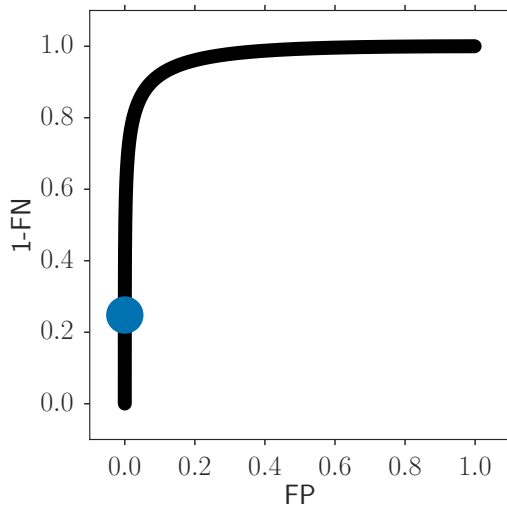
ROC Curve



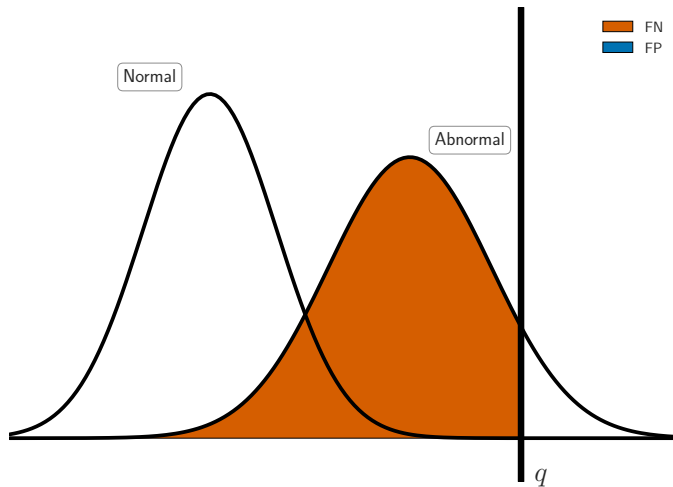
ROC Curve



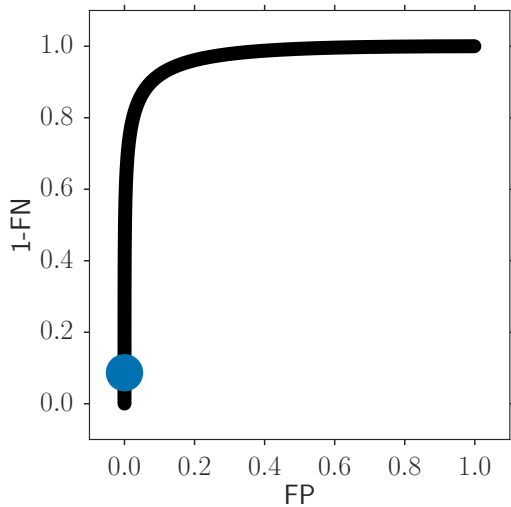
ROC Curve



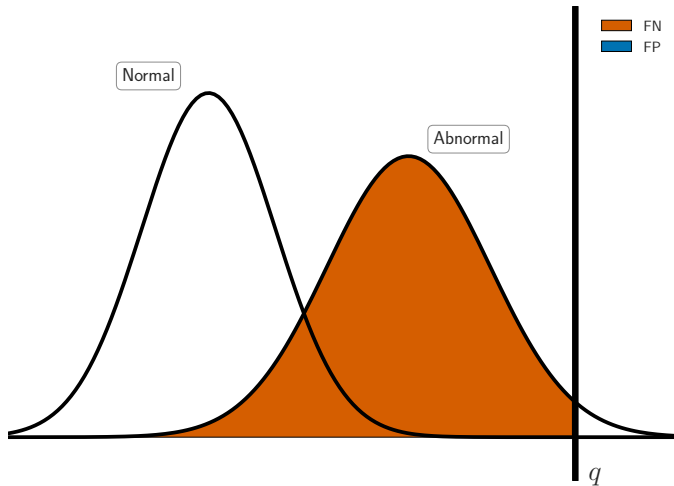
ROC Curve



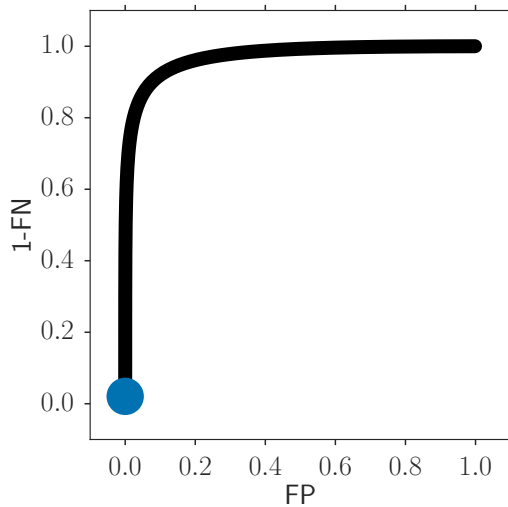
ROC Curve



ROC Curve

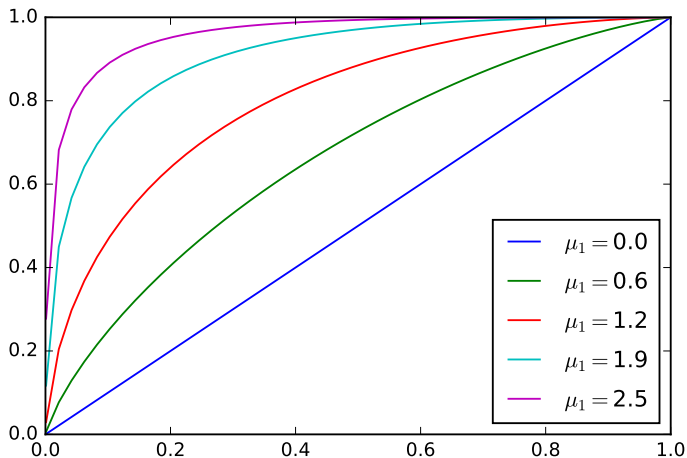


ROC Curve



ROC curves for bi-normal case

- ▶ F and G are Gaussian with parameter μ_0, σ_0 and μ_1, σ_1 , respectively.
- ▶ Here $\mu_0 = 0$, $\sigma_0 = \sigma_1 = 1$, and μ_1 varies



Estimation–application

ROC curve estimation

- ▶ Maximum likelihood
- ▶ Non-parametric
- ▶ Bayesian with latent variables
- ▶ Estimation of the area under the ROC curve (AUC)

Application

- ▶ To compare different statistic tests
- ▶ To compare different (supervised) learning algorithm
- ▶ To compare variable selection methods (*e.g.* Lasso, OMP, etc.)

nb : ROC = Receiver Operating Characteristic

SD-TSIA204 - Statistics: linear models

Ridge

Ekhiñe Irurozki
Télécom Paris

Problems of OLS and Motivation for Ridge Regression

We saw in the first session that $\hat{\theta} = (X^T X)^{-1} X^T Y$ is only well-defined if $(X^T X)^{-1}$ exists.

Collinearity: When two columns of the design matrix $X \in \mathbb{R}^{n \times p}$ are (almost) linearly dependent, $X^T X$ is ill-conditioned.

Supercollinearity: When $n < p$, $\text{rank}(X) = \min(n, p) = n$.

Two Approaches:

1. The first uses the Moore-Penrose inverse of the matrix.
2. The second is ridge regression, an ad-hoc fix for inversion problems with an interpretation in terms of penalization of the coefficients.

Method 1, based on the Moore-Penrose inverse

$X^T X \theta = X^T Y$. The matrix $X^T X$ is of rank n , while θ is a vector of length p . If $p > n$, the vector θ cannot be uniquely determined from this system of equations.

Rem Let U be the n -dimensional space spanned by the columns of X , and let V be the $p - n$ -dimensional space, the orthogonal complement of U , i.e., $V = U^\perp$. Then, $Xv = 0_p$ for all $v \in V$, making V the non-trivial null space of X , $Ker(X)$.

Consequently, as $X^T X v = X^T 0_p = 0_n$, the solution of the normal equations is:

$$\hat{\theta} = (X^T X)^+ X^T Y + v \quad \text{for all } v \in V,$$

The Moore-Penrose inverse of the matrix A is defined as follows for a square symmetric matrix:

$$A^+ = \sum_{j=1}^p \frac{1}{s_j} v_j v_j^\top \mathbb{I}\{s_j \neq 0\}, \quad \text{where } \nu_j \neq 0 \text{ for } j = 1, 2, \dots, p.$$

Method 2, Ridge

The ridge regression estimator can be seen as an ad-hoc fix for the singularity of $X^T X$

$$\hat{\theta}_\lambda = (X^T X + \lambda I_{pp})^{-1} X^T Y,$$

for $\lambda > 0$.

Exercise Show that $(X^T X + \lambda I_{pp})$ is invertible

Ridge : penalized definition

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_2^2}_{\text{regularization}} \right)$$

- Note that the *Ridge* estimator is **unique** for any fixed $\lambda > 0$

- We recover the limiting cases:

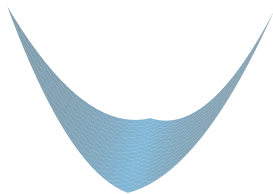
$$\lim_{\lambda \rightarrow 0} \hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = \hat{\boldsymbol{\theta}}^{\text{OLS}} \text{ (solution with smallest } \|\cdot\|_2 \text{ norm)}$$

$$\lim_{\lambda \rightarrow +\infty} \hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = \mathbf{0} \in \mathbb{R}^p$$

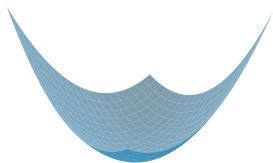
- First order conditions:

$$\nabla f(\boldsymbol{\theta}) = X^{\top}(X\boldsymbol{\theta} - \mathbf{y}) + \lambda\boldsymbol{\theta} = \mathbf{0} \Leftrightarrow (X^{\top}X + \lambda\text{Id}_p)\boldsymbol{\theta} = X^{\top}\mathbf{y}$$

Motivation



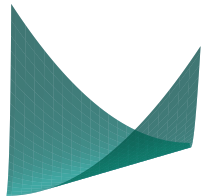
OLS



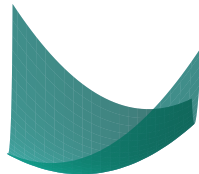
Ridge

x and y -axis are the OLS coefficients θ_0 and θ_1 , z axis is the RSS
Regularize: simplify the problem when ill-conditioned

Motivation



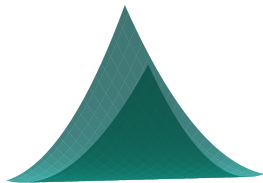
OLS



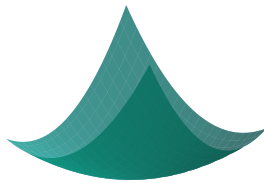
Ridge

x and y -axis are the OLS coefficients θ_0 and θ_1 , z axis is the RSS
Regularize: simplify the problem when ill-conditioned

Motivation



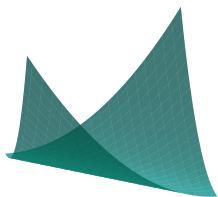
OLS



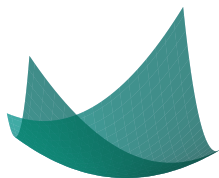
Ridge

x and y -axis are the OLS coefficients θ_0 and θ_1 , z axis is the RSS
Regularize: simplify the problem when ill-conditioned

Motivation



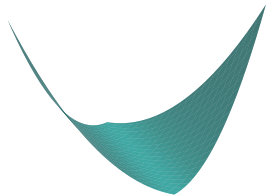
OLS



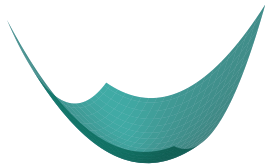
Ridge

x and y -axis are the OLS coefficients θ_0 and θ_1 , z axis is the RSS
Regularize: simplify the problem when ill-conditioned

Motivation



OLS



Ridge

x and y -axis are the OLS coefficients θ_0 and θ_1 , z axis is the RSS

Regularize: simplify the problem when ill-conditioned

Constraint interpretation

A “Lagrangian” formulation is as follows:

$$\arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_2^2}_{\text{regularization}} \right)$$

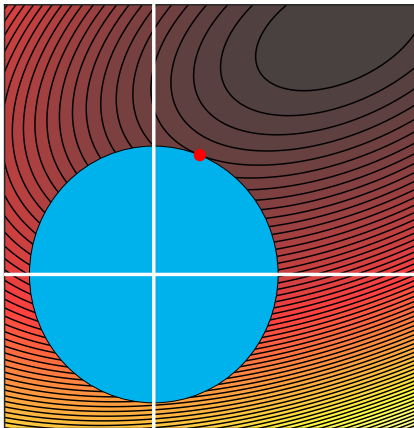
has for a certain $T > 0$ the same solution as:

$$\begin{cases} \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 \\ \text{s.t. } \|\boldsymbol{\theta}\|_2^2 \leq T \end{cases}$$

Rem the link $T \leftrightarrow \lambda$ is not explicit!

- ▶ If $T \rightarrow 0$ we recover the null vector: $\mathbf{0} \in \mathbb{R}^p$
- ▶ If $T \rightarrow \infty$ we recover $\hat{\boldsymbol{\theta}}^{\text{OLS}}$ (un-constrained)

Level lines and constraint set



Optimization under ℓ_2 constraints:

$$\begin{cases} \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2, \\ \text{s.t. } \|\boldsymbol{\theta}\|_2^2 \leq T \end{cases}$$

Associated prediction

From the *Ridge* coefficient:

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = (\lambda \text{Id}_p + X^{\top} X)^{-1} X^{\top} \mathbf{y}$$

the associated prediction is given by:

$$\hat{\mathbf{y}}_{\lambda} = X \hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = X (\lambda \text{Id}_p + X^{\top} X)^{-1} X^{\top} \mathbf{y} = H_{\lambda} \mathbf{y}$$

Rem the estimator $\hat{\mathbf{y}}_{\lambda}$ is linear w.r.t. \mathbf{y}

Ridge shrinks the singular values

Note $X = UDV^T = \sum_{i=1}^{\text{rg}(X)} s_i \mathbf{u}_i \mathbf{v}_i^T$, (SVD)

Proposition The ridge penalty shrinks the singular values. To see this, show that $\hat{\theta} = V(D^T D)^{-1} D^T U^T Y$ and $\hat{\theta}_\lambda = V(D^T D + \lambda I)^{-1} D^T U^T Y$

The matrix $H_\lambda := X(\lambda \text{Id}_p + X^T X)^{-1} X^T = \sum_{j=1}^{\text{rg}(X)} \frac{s_j^2}{s_j^2 + \lambda} \mathbf{u}_j \mathbf{u}_j^T$ and

$H_+ := X(X^T X)^+ X^T$ are the equivalent of the hat matrix for both methods respectively

Exercise Show that H^+ is an orthogonal projector and H_λ is not.

Exercise Show that ridge fit $\hat{\mathbf{y}}_\lambda$ is not orthogonal to the associated ridge residuals $\hat{\mathbf{e}}_\lambda$, defined as $\hat{\mathbf{e}}_\lambda = Y - X\hat{\boldsymbol{\theta}}_\lambda$.

Remarks

Reminder: normalizing the p features the same way is necessary if you want the penalty to be similar for all features:

- ▶ center the observation and the features \Rightarrow no coefficient for the constants (hence no constraint on it)
- ▶ not centering features \Rightarrow do not put constraint on the constant feature (bias/intercept)

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \|\mathbf{y} - X\boldsymbol{\theta} - \theta_0 \mathbf{1}_n\|_2^2 + \lambda \sum_{j=1}^p \theta_j^2$$

Rem for cross validation one can use $\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2n}$ rather than $\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2}$ as the data fitting part

General form of the bias

Under the fixed-design model, $\mathbf{y} = X\boldsymbol{\theta}^* + \boldsymbol{\varepsilon}$ with $\mathbb{E}(\boldsymbol{\varepsilon}) = 0$:

$$\begin{aligned}\mathbb{E}(\hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}}) &= \mathbb{E}[(\lambda \text{Id}_p + X^{\top} X)^{-1} X^{\top} \mathbf{y}] \\ &= \mathbb{E}[(\lambda \text{Id}_p + X^{\top} X)^{-1} X^{\top} X \boldsymbol{\theta}^* + (\lambda \text{Id}_p + X^{\top} X)^{-1} X^{\top} \boldsymbol{\varepsilon}] \\ &= (\lambda \text{Id}_p + X^{\top} X)^{-1} X^{\top} X \boldsymbol{\theta}^* \\ &= \sum_{i=1}^{\text{rg}(X)} \frac{s_i^2}{s_i^2 + \lambda} \mathbf{v}_i \mathbf{v}_i^{\top} \boldsymbol{\theta}^*\end{aligned}$$

Rem one recovers $\mathbb{E}(\hat{\boldsymbol{\theta}}^{\text{OLS}}) \rightarrow \sum_{i=1}^{\text{rg}(X)} \mathbf{v}_i \mathbf{v}_i^{\top} \boldsymbol{\theta}^*$ when $\lambda \rightarrow 0$

Exercise Show that the bias for an orthonormal X is $(1 + \lambda)^{-1} \boldsymbol{\theta} - \boldsymbol{\theta}$

Variance in the general case

Under the assumption $\mathbb{E}(\boldsymbol{\varepsilon}) = 0$, and with a homoscedastic model: $\mathbb{E}(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\top) = \sigma^2 \text{Id}_n$

Variance / Covariance

$$V_\lambda^{\text{rdg}} = \mathbb{E} \left((\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}} - \mathbb{E}(\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}}))(\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}} - \mathbb{E}(\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}}))^\top \right)$$

Explicit computation:

$$\begin{aligned} V_\lambda^{\text{rdg}} &= \mathbb{E}((\lambda \text{Id}_p + X^\top X)^{-1} X^\top \boldsymbol{\varepsilon} \boldsymbol{\varepsilon}^\top X (\lambda \text{Id}_p + X^\top X)^{-1}) \\ &= (\lambda \text{Id}_p + X^\top X)^{-1} X^\top \mathbb{E}(\boldsymbol{\varepsilon} \boldsymbol{\varepsilon}^\top) X (\lambda \text{Id}_p + X^\top X)^{-1} \\ &= \sum_{i=1}^{\text{rg}(X)} \frac{s_i^2 \sigma^2}{(s_i^2 + \lambda)^2} \mathbf{v}_i \mathbf{v}_i^\top \end{aligned}$$

Rem one recovers $V^{\text{OLS}} = \sum_{i=1}^{\text{rg}(X)} \frac{\sigma^2}{s_i^2} \mathbf{v}_i \mathbf{v}_i^\top$ when $\lambda \rightarrow 0$

Rem one find a null variance when $\lambda \rightarrow \infty$

Exercise Show that the variance of when X is orthogonal is $\sigma^2(1 + \lambda)^{-2} \text{Id}_p$

Prediction risk

Homoscedastic assumption: $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 \text{Id}_n$

Quadratic prediction risk $\mathbb{E}\|X\boldsymbol{\theta}^\star - X\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}}\|^2$ under the homoscedastic assumption:

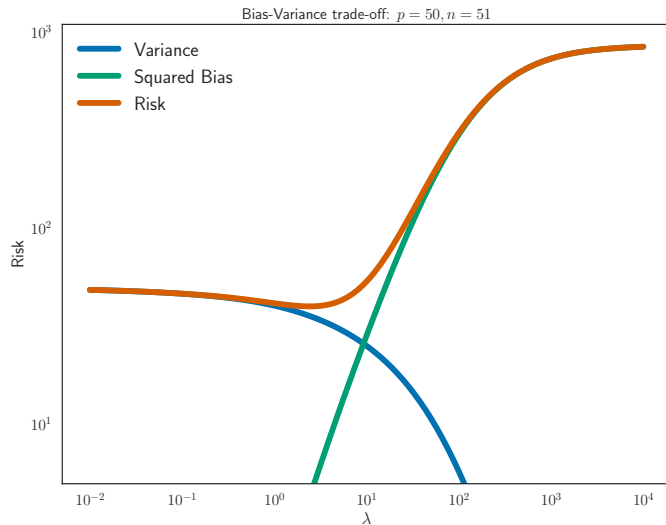
$$R_{\text{pred}}(\boldsymbol{\theta}^\star, \hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}}) = \mathbb{E}\left[(\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}} - \boldsymbol{\theta}^\star)^\top (X^\top X)(\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}} - \boldsymbol{\theta}^\star)\right]$$

Explicit computation (begins as for OLS):

$$\begin{aligned} R_{\text{pred}}(\boldsymbol{\theta}^\star, \hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}}) &= \mathbb{E}\left[(\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}} - \boldsymbol{\theta}^\star)^\top (X^\top X)(\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}} - \boldsymbol{\theta}^\star)\right] \\ &= \mathbb{E}\left[(X(X^\top X + \lambda \text{Id}_p)^{-1} X^\top \boldsymbol{\varepsilon})^\top (X(X^\top X + \lambda \text{Id}_p)^{-1} X^\top \boldsymbol{\varepsilon})\right] \\ &\quad + \lambda^2 \boldsymbol{\theta}^{\star\top} (X^\top X + \lambda \text{Id}_p)^{-2} \boldsymbol{\theta}^\star \\ &= \sum_{i=1}^{\text{rg}(X)} \frac{s_i^4 \sigma^2}{(s_i^2 + \lambda)^2} + n^2 \lambda^2 \boldsymbol{\theta}^{\star\top} (X^\top X + \lambda \text{Id}_p)^{-2} \boldsymbol{\theta}^\star \end{aligned}$$

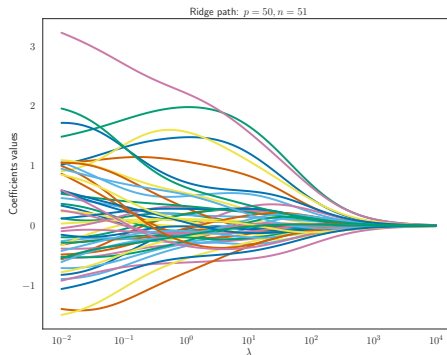
Rem $\lim_{\lambda \rightarrow 0} R_{\text{pred}}(\boldsymbol{\theta}^\star, \hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}}) = \text{rg}(X)\sigma^2$, $\lim_{\lambda \rightarrow \infty} R_{\text{pred}}(\boldsymbol{\theta}^\star, \hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}}) = \|X\boldsymbol{\theta}^\star\|_2^2$

Bias / Variance: simulated example



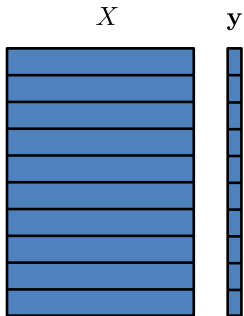
Choosing λ

```
n_features = 50; n_samples = 50
X = np.random.randn(n_samples, n_features)
theta_true = np.zeros([n_features, ])
theta_true[0:5] = 2.
y_true = np.dot(X, theta_true)
y = y_true + 1. * np.random.rand(n_samples,)
```



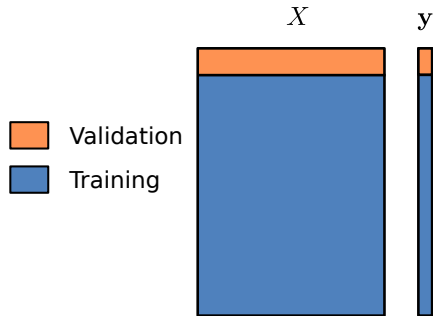
K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, y) into K blocks (sample-wise):



K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, \mathbf{y}) into K blocks (sample-wise):

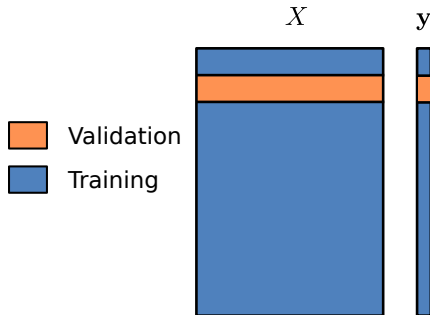


$$k = 1$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, y) into K blocks (sample-wise):

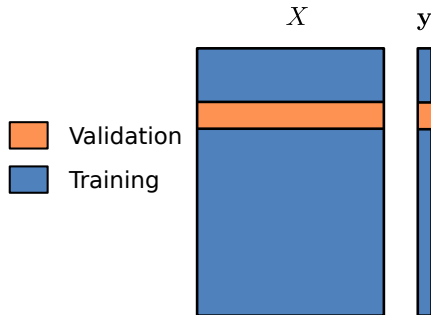


$$k = 2$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, \mathbf{y}) into K blocks (sample-wise):

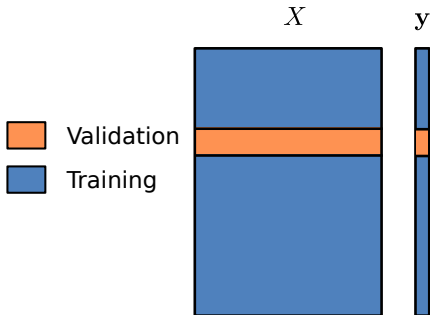


$$k = 3$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, \mathbf{y}) into K blocks (sample-wise):

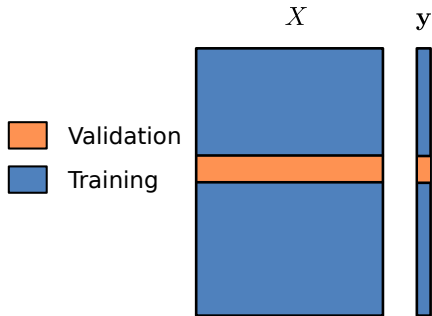


$$k = 4$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, y) into K blocks (sample-wise):

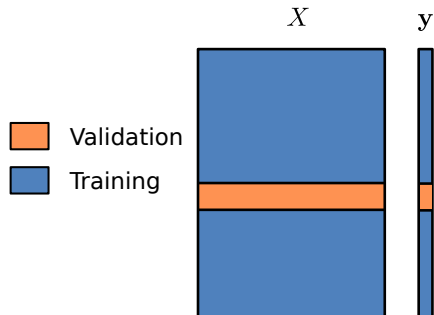


$$k = 5$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, y) into K blocks (sample-wise):

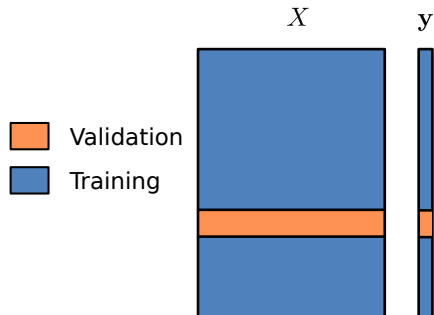


$$k = 6$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, y) into K blocks (sample-wise):

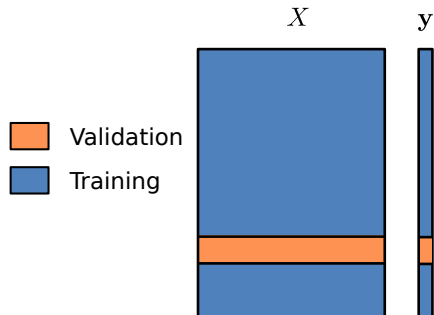


$$k = 7$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, y) into K blocks (sample-wise):

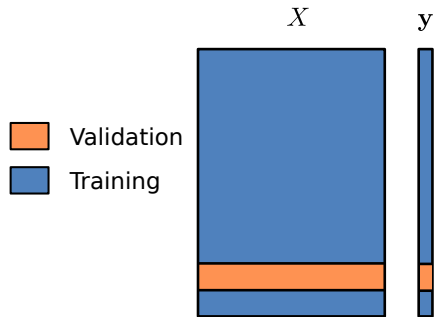


$$k = 8$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, y) into K blocks (sample-wise):

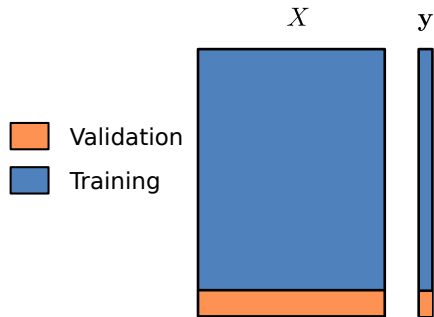


$$k = 9$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, y) into K blocks (sample-wise):

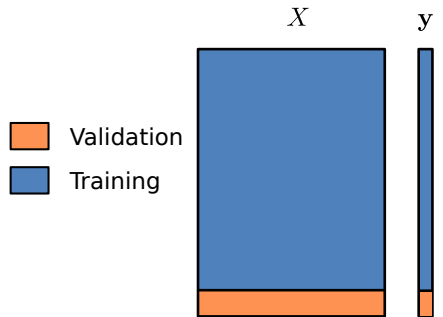


$$k = 10$$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

K -fold Cross-Validation ($K = 10$)

- ▶ Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- ▶ Divide (X, y) into K blocks (sample-wise):



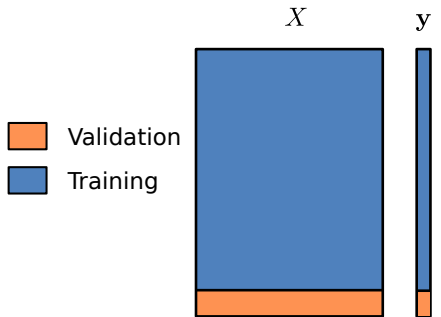
$k = 10$

1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

Parameter choice: averaging the previous errors over k gives $\widehat{\text{Error}}_1, \dots, \widehat{\text{Error}}_r$.
Then choose $i^* \in \llbracket 1, r \rrbracket$ achieving the smallest one

K -fold Cross-Validation ($K = 10$)

- Choose a grid of r λ 's to test: $\lambda_1, \dots, \lambda_r$
- Divide (X, y) into K blocks (sample-wise):



- $k = 10$
1. Compute with the training part the estimators for $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
 2. Evaluate the (prediction) error $\text{Error}_1^k, \dots, \text{Error}_r^k$ over the validation part,

Parameter choice: averaging the previous errors over k gives $\widehat{\text{Error}}_1, \dots, \widehat{\text{Error}}_r$. Then choose $i^* \in \llbracket 1, r \rrbracket$ achieving the smallest one

Re-calibration: compute $\hat{\theta}^{\lambda_{i^*}}$ over the whole sample

CV in practice

Extreme cases of CV

- ▶ $K = 1$ impossible, needs $K = 2$
- ▶ $K = n$, “*leave-one-out*” strategy (cf. **Jackknife**): as many blocks as observations
Rem $K = n$ (often) computationally efficient but unstable

Practical advice:

- ▶ “randomise the sample” : having samples in random order avoid artifacts block (each fold needs to be representative of the whole sample!)
- ▶ standard choices: $K = 5, 10$

Alternatives: random partition validation/test, time series variants, etc.

http://scikit-learn.org/stable/modules/cross_validation.html

CV variants sklearn

Crucial points: the structures train/test artificially created should represent faithfully the underlying learning problem

Classical alternatives:

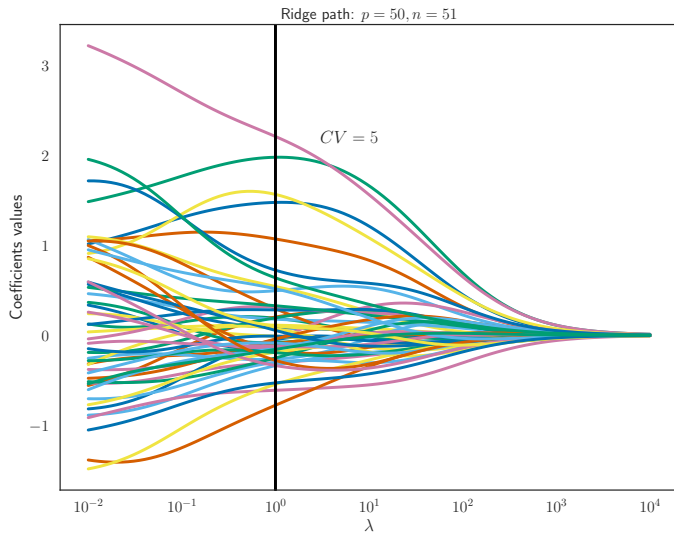
- ▶ random partitioning in train/test sets (`cf.train_test_split`)
- ▶ Time series variant: `TimeSeriesSplit` (never predict the past with future information)
- ▶ For classification tasks with unbalanced classes `StratifiedKFold`

Rem averaging estimators (with weights reflecting their performance) is also relevant for prediction

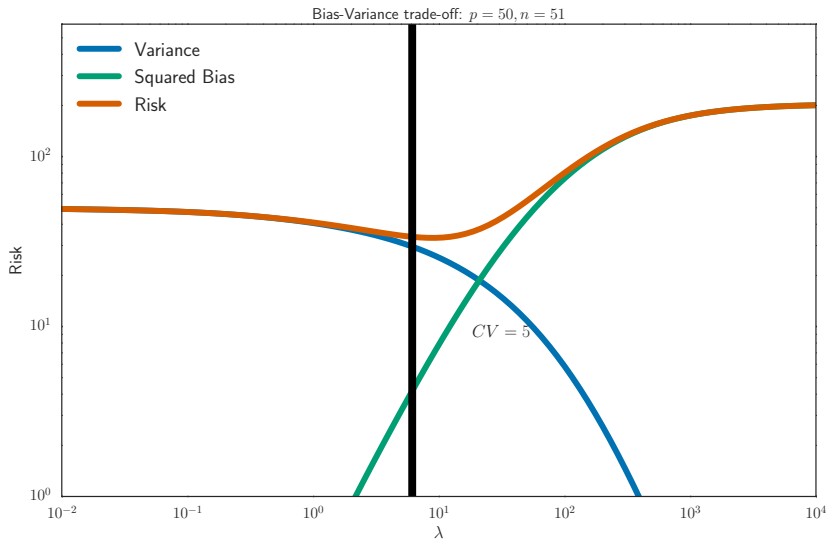
More details:

http://scikit-learn.org/stable/modules/cross_validation.html

Choosing λ : example with $CV = 5$ (I)



Choosing λ : example with $CV = 5$ (II)



Algorithms to compute the *Ridge* estimator

- ▶ 'svd': most stable method, useful for computing many λ 's cause the SVD price is paid only once
- ▶ 'cholesky' : matrix decomposition leading to a close form solution
`scipy.linalg.solve`
- ▶ 'sparse_cg': conjugate gradient descent, useful also for sparse cases and high dimension (set `tol/max_iter` to a small value)
- ▶ stochastic gradient descent approaches : if n is huge

cf. the code of `Ridge`, `ridge_path`, `RidgeCV` in the module `linear_model` of `sklearn`

Rem it is rare to compute the *Ridge* estimator only for one single λ

Rem crucial issue of computing SVD for huge matrices...

SD-TSIA204 - Statistics: linear models SVMs and kernels

Ekhiñe Irurozki

Télécom Paris

Slides by Florence d'Alché-Buc

Table of contents

1. Introduction
2. Linear SVM
3. Nonlinear SVM and Kernels
4. Support Vector Regression
5. Conclusion et References
6. References

Introduction

Linear SVM

Nonlinear SVM and Kernels

Support Vector Regression

Conclusion et References

References

Statistical learning: a methodology

- The main problems to be solved :
 - **Representation problem**: determine in which representation space the data will be encoded and determine which family of mathematical functions will be used
 - **Optimization problem** : formulate the learning problem as an optimization problem (with statistical criteria), develop an optimization algorithm
 - **Model Selection**: determine the complexity of the model or any other hyperparameter prior to test
 - **Evaluation problem**: fix evaluation metrics, provide a performance estimate on test set

Machine Learning: two tasks

Let $\mathcal{S}_n = \{(x_i, y_i)\}_{i=1}^n$, a i.i.d. sample drawn from μ a joint probability distribution defined on (X, Y) : X takes its values in \mathbb{R}^d and Y is real-valued.

- **Learning:** get $h_n = \mathcal{A}(\mathcal{S}_n, \mathcal{H}, \ell, \lambda, \Omega)$ with
 - \mathcal{S}_n : training data
 - \mathcal{H} : class of functions
 - λ : some hyperparameter
 - ℓ : Local loss function
 - Ω : regularizing function
 - \mathcal{A} : learning algorithm
- **Prediction:** given x , and compute $h_n(x)$

Today's lesson (1) - starting point

Let $\mathcal{S}_n = \{(x_i, y_i)\}_{i=1}^n$, a i.i.d. sample drawn from μ a joint probability distribution defined on (X, Y) : X takes its values in \mathbb{R}^d and Y is real-valued.

- **Learning:** get $h_n = \mathcal{A}(\mathcal{S}_n, \mathcal{H}, \ell, \lambda, \Omega)$ with
 - \mathcal{S}_n : training data
 - \mathcal{H} : class of functions $h(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x} + b)$
 - λ : some hyperparameter
 - ℓ : Local loss function **HINGE loss / Margin loss**
 - Ω : regularizing function **ℓ_2 norm**
 - \mathcal{A} : learning algorithm
- **Prediction:** given x , and compute $h_n(x)$

Introduction

Linear SVM

Nonlinear SVM and Kernels

Support Vector Regression

Conclusion et References

References

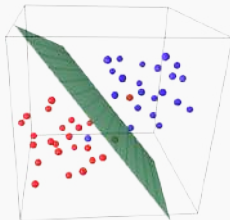
Linear Separator

Definition

Soit $\mathbf{x} \in \mathbb{R}^p$

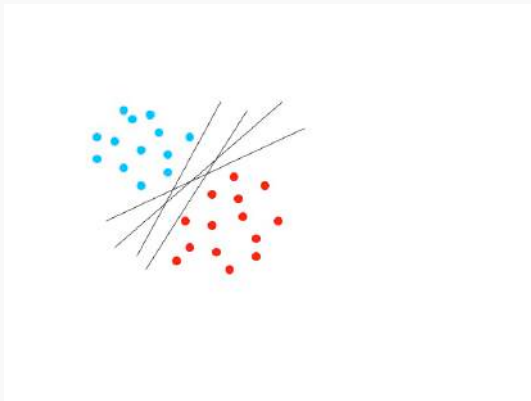
$$h(\mathbf{x}) = \text{sign}(\mathbf{w}^* \phi(\mathbf{x}) + b)$$

Equation $\mathbf{w}^T \mathbf{x} + b = 0$ defines an hyperplane in the euclidean space \mathbb{R}^p



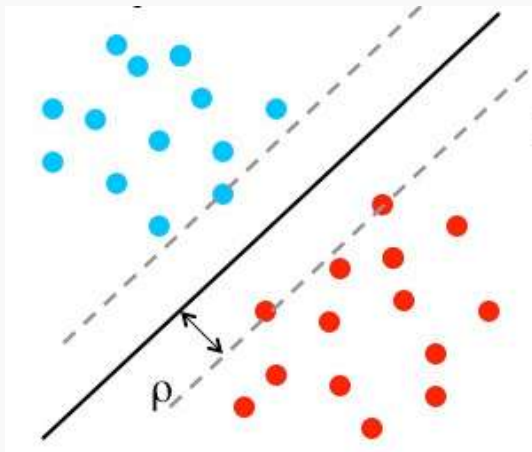
Example in 3D

Data linearly separable



What to choose ?

Margin criterion



Geometrical margin

- To separate data, let us consider a triplet of hyperplanes:
 - $H: \mathbf{w}^T \mathbf{x} + b = 0$, $H_1: \mathbf{w}^T \mathbf{x} + b = 1$, $H_{-1}: \mathbf{w}^T \mathbf{x} + b = -1$
- We call *geometrical margin*, $\rho(\mathbf{w})$ the smallest distance between the data and Hyperplane H thus, here half of the distance between H_1 and H_{-1}
- A simple calculation gives : $\rho(\mathbf{w}) = \frac{1}{\|\mathbf{w}\|}$.

New objective function to optimize

How to find w and b ?

- Maximize the margin $\rho(\mathbf{w})$ while separating the data using H_1 and H_{-1}
- Classify the blue data ($y_i = 1$) : $\mathbf{w}^T \mathbf{x}_i + b \geq 1$
- Classify the red data ($y_i = -1$) : $\mathbf{w}^T \mathbf{x}_i + b \leq -1$

Optimization in the primal

$$\begin{aligned} & \underset{\mathbf{w}, b}{\text{minimize}} && \frac{1}{2} \|\mathbf{w}\|^2 \\ & \text{under constraints} && y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, n. \end{aligned}$$

Référence

Boser, B. E.; Guyon, I. M.; Vapnik, V. N. (1992). "A training algorithm for optimal margin classifiers". Proceedings of the fifth annual workshop on Computational learning theory - COLT '92. p. 144.

Programming under inequality constraints

Problem of the following kind:

$$\min_{\theta} f(\theta)$$

$$\text{s.c. } g(\theta) \leq 0$$

- Here: $g(\theta)$: linear (affine) constraints
 - f is strictly convex
1. Lagrangian: $J(\theta, \lambda) = f(\theta) + \lambda g(\theta)$, where $\lambda \geq 0$ is a Lagrangian coefficient
 2. The problem benefits from the saddle point theorem: there is a unique solution to $\min_{\theta} \max_{\lambda} J(\theta, \lambda)$.
 3. We can start by solving the problem in λ or in θ , the two solutions are linked.

Programming under inequality constraints

$$\begin{array}{ll}\text{minimize}_{\mathbf{w}, b} & \frac{1}{2} \|\mathbf{w}\|^2 \\ \text{s. t.} & 1 - y_i(\mathbf{w}^T \mathbf{x}_i + b) \leq 0, \quad i = 1, \dots, n.\end{array}$$

Lagrangian

$$\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2} \|\mathbf{w}\|^2 + \sum_i \alpha_i (1 - y_i(\mathbf{w}^T \mathbf{x}_i + b))$$
$$\forall i, \alpha_i \geq 0$$

At the extremum, we have:

$$\nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}) = \mathbf{w} - \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i = 0$$

$$\nabla_b \mathcal{L}(b) = - \sum_{i=1}^n \alpha_i y_i = 0$$

$$\forall i, \alpha_i \geq 0$$

$$\forall i, \alpha_i [1 - y_i(\mathbf{w}^T \mathbf{x}_i + b)] = 0$$

Obtaining the α_i 's : solution the dual

$$\mathcal{L}(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i^T \mathbf{x}_j)$$

- Maximize \mathcal{L} under the constraints $\alpha_i \geq 0$ et $\sum_i \alpha_i y_i = 0, \forall i = 1, \dots, n$
- Call for a quadratic solver

Optimal Margin Hyperplan (linear SVM)

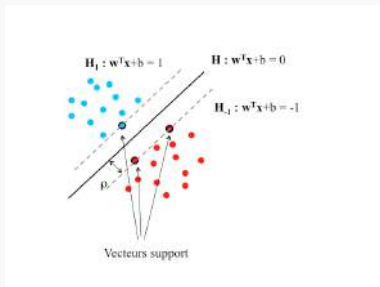
Assume the Lagrangian coefficients α_i have been found :

Linear SVM equation

$$f(\mathbf{x}) = \text{signe}\left(\sum_{i=1}^n \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b\right)$$

To classify a novel \mathbf{x} , this classifier makes all the support data vote with an importance weight equal to $\alpha_i \mathbf{x}_i^T \mathbf{x}$ that measures how much \mathbf{x} is close to the support data.

Support Vectors



Training data \mathbf{x}_i such that $\alpha_i \neq 0$ belong to either H_1 or H_{-1} . Only those data, called **support vectors**, are taking into account in $\mathbf{w} = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i$
NB : b is obtained by choosing one (or all) support data such that $(\alpha_i \neq 0)$

Realistic case: linear SVM in the case of nonlinearly separable data

For each training data, introduce a slack variable ξ_i :

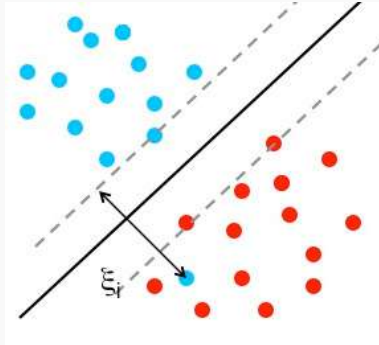
New problem in the primal space

$$\min_{\mathbf{w}, b, \xi} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i$$

such that: $y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i \quad i = 1, \dots, n.$

$$\xi_i \geq 0 \quad i = 1, \dots, n.$$

Realistic case: linear SVM in the case of nonlinearly separable data



Notion of soft margin

Lagrangian

Let us introduce, Lagrangian coefficients: $\alpha_i, \xi_i, \mu_i, i = 1, \dots, n$

$$\mathcal{L}(\mathbf{w}, b, \xi, \alpha, \mu) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_i \xi_i + \sum_i \alpha_i (1 - \xi_i - y_i (\mathbf{w}^T \mathbf{x}_i + b)) - \sum_i \xi_i \mu_i$$

Write the first optimality conditions ...

$$\nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}) = \mathbf{w} - \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i = 0$$

$$\nabla_b \mathcal{L}(b) = - \sum_{i=1}^n \alpha_i y_i = 0$$

$$\forall i = 1, \dots, n, \frac{\partial \mathcal{L}(\xi)}{\partial \xi_i} = C - \alpha_i - \mu_i = 0$$

Realistic case: linear SVM in the case of nonlinearly separable data

Dual problem

$$\max_{\alpha} \quad \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

under the constraints $0 \leq \alpha_i \leq C \quad i = 1, \dots, n.$

$$\sum_i \alpha_i y_i = 0$$

Karush-Kuhn-Tucker Conditions (KKT)

Let α^* be the solution of the dual problem:

$$\forall i, [y_i f_{w^*, b^*}(x_i) - 1 + \xi_i^*] \leq 0 \quad (1)$$

$$\forall i, \alpha_i^* \geq 0 \quad (2)$$

$$\forall i, \alpha_i^* [y_i f_{w^*, b^*}(x_i) - 1 + \xi_i^*] = 0 \quad (3)$$

$$\forall i, \mu_i^* \geq 0 \quad (4)$$

$$\forall i, \mu_i^* \xi_i^* = 0 \quad (5)$$

$$\forall i, \alpha_i^* + \mu_i^* = C \quad (6)$$

$$\forall i, \xi_i^* \geq 0 \quad (7)$$

$$\mathbf{w}^* = \sum_i \alpha_i^* y_i \mathbf{x}_i \quad (8)$$

$$\sum_i \alpha_i^* y_i = 0 \quad (9)$$

$$(10)$$

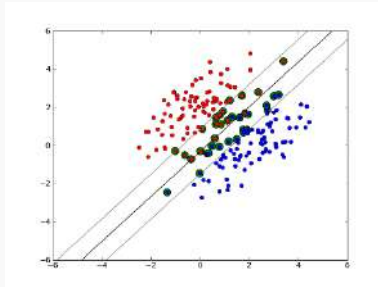
What are the support vectors?

A training datapoint x_i is called a *support vector* if $\alpha_i^* \neq 0$.

- if $\alpha_i^* = 0$ (x_i is not a support vector), then $\mu_i^* = C > 0$ and thus, $\xi_i^* = 0$: x_i is well classified
- if $0 < \alpha_i^* < C$ (x_i is a support vector) then $\mu_i^* > 0$ and thus, $\xi_i^* = 0$
: x_i is such that: $y_i f(x_i) = 1$
- if $\alpha_i^* = C$ (x_i is a support vector), then $\mu_i^* = 0$, $\xi_i^* = 1 - y_i f_{w^*, b^*}(x_i)$

NB : we compute b^* by using i such that $0 < \alpha_i^* < C$

Realistic case: linear SVM with soft margin



- A training data is a support vector if either it lies on the hyperplanes H_1, H_{-1} or between H_1 and H_{-1}
- C is a hyperparameter that controls the compromise between the model complexity and the training classification error

SVM as a penalized regression problem

Optimization in the primal space

$$\min_{\mathbf{w}, b} \sum_{i=1}^n (1 - y_i(\mathbf{w}^T \mathbf{x}_i + b))_+ + \lambda \frac{1}{2} \|\mathbf{w}\|^2$$

With: $(z)_+ = \max(0, z)$

$f(\mathbf{x}) = \text{signe}(h(\mathbf{x}))$

Loss function: $L(\mathbf{x}, y, h(\mathbf{x})) = (1 - yh(\mathbf{x}))_+$

$yh(\mathbf{x})$ is called the classifier margin

Optimization: subgradient or proximal approach

Introduction

Linear SVM

Nonlinear SVM and Kernels

Kernels

Support Vector Regression

Conclusion et References

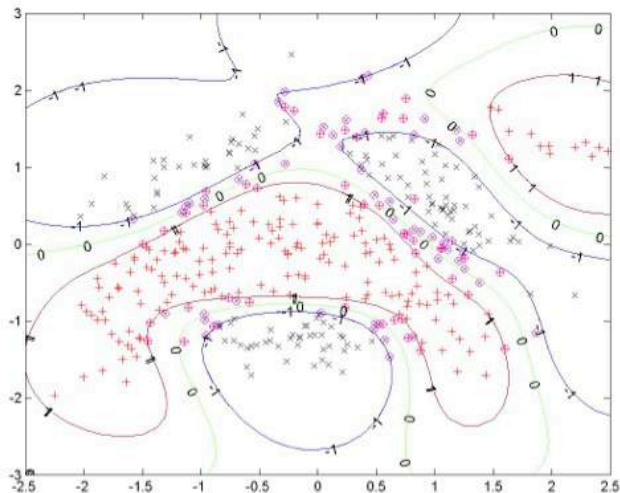
References

Today's lesson (2) - starting point

Let $\mathcal{S}_n = \{(x_i, y_i)\}_{i=1}^n$, a i.i.d. sample drawn from μ a joint probability distribution defined on (X, Y) : X takes its values in \mathbb{R}^d and Y is real-valued.

- **Learning:** get $h_n = \mathcal{A}(\mathcal{S}_n, \mathcal{H}, \ell, \lambda, \Omega)$ with
 - \mathcal{S}_n : training data
 - \mathcal{H} : class of functions $h(\mathbf{x}) = \text{sign}(\langle \mathbf{w}, \phi(\mathbf{x}) \rangle_{\mathcal{F}} + b)$
 - λ : some hyperparameter
 - ℓ : Local loss function **HINGE loss / Margin loss**
 - Ω : regularizing function **ℓ_2 norm**
 - \mathcal{A} : learning algorithm
- **Prediction:** given x , and compute $h_n(x)$

Support Vector Machine : nonlinear frontiers in 2D space



Remark 1

Finding the Optimal Margin Hyperplane does involve only inner products between training data

$$\begin{aligned} \max_{\alpha} \quad & \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{sous les contraintes} \quad & 0 \leq \alpha_i \leq C \quad i = 1, \dots, n. \\ & \sum_i \alpha_i y_i = 0 \quad i = 1, \dots, n. \end{aligned}$$

Let us use a feature map

If data are transformed according a nonlinear feature map $\phi : \mathcal{X} \rightarrow \mathcal{F}$, and if we know how to compute all the inner products $\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$, then we are able to learn a nonlinear decision frontier.

$$\max_{\alpha} \quad \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j < \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) >$$

sous les contraintes $0 \leq \alpha_i \leq C \quad i = 1, \dots, n.$

$$\sum_i \alpha_i y_i = 0.$$

To classify a new datapoint \mathbf{x} , we only need to be able to calculate $\phi(\mathbf{x})^T \phi(\mathbf{x}_i)$.

Nonlinear SVM with a feature map

Choose $\phi : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^p$ a nonlinear mapping.

Linear SVM equation

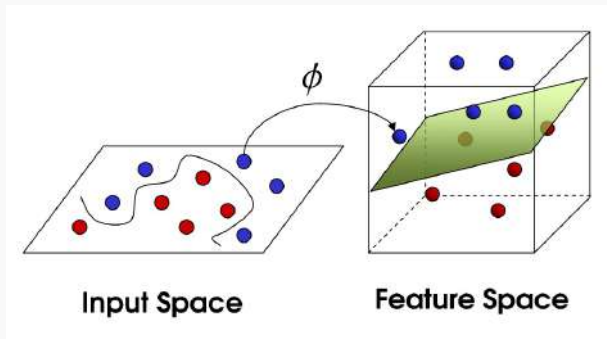
$$f(\mathbf{x}) = \text{sign}\left(\sum_{i=1}^n \alpha_i y_i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle + b\right)$$

If we substitute $\mathbf{x}_i^T \mathbf{x}_j$ by the image of a function $k : k(\mathbf{x}_i, \mathbf{x}_j)$ such that there exists a feature space \mathcal{F} and feature map $\phi : \mathcal{X} \rightarrow \mathcal{F}$ et $\forall(\mathbf{x}, \mathbf{x}') \in \mathcal{X}, k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T \phi(\mathbf{x}')$, then We are able to apply the same learning algorithm (Optimal Margin Hyperplane) and we get

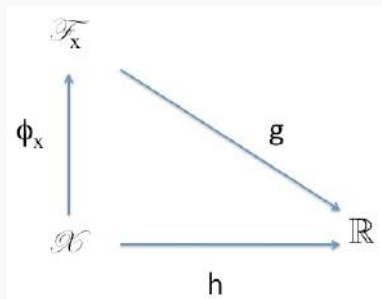
$$f(\mathbf{x}) = \text{signe}(\sum_{i=1}^n \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b)$$

Such functions do exist and they are called PDS kernels: positive definite symmetric kernels.

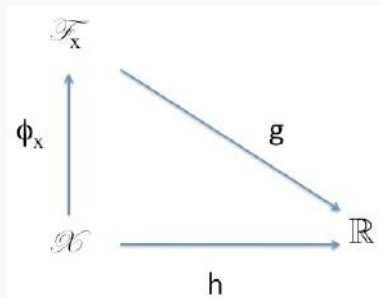
Kernel trick and feature space 1/2



Kernel trick and feature space 2/2



Kernel trick and feature space 2/2



$$h(\mathbf{x}) = \sum_{i=1}^n \beta_i \phi(x)^T \phi(x_i) = \sum_{i=1}^n \beta_i k(x, x_i),$$

with $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ a positive definite symmetric kernel.

Introduction

Linear SVM

Nonlinear SVM and Kernels

Kernels

Support Vector Regression

Conclusion et References

References

Definition

Let \mathcal{X} be a non empty set. Let $k: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$, be a symmetric function. Function k is called a Positive Definite Symmetric kernel if and only if for any finite set of size m , $\{\mathbf{x}_1, \dots, \mathbf{x}_m\} \subset \mathcal{X}$, and any column vector $\mathbf{c} \in \mathbb{R}^m$,

$$\mathbf{c}^T K \mathbf{c} = \sum_{i,j=1}^m c_i c_j k(\mathbf{x}_i, \mathbf{x}_j) \geq 0$$

NB: any finite Gram matrix built from k and a finite number of elements of \mathcal{X} is semi-definite positive

Moore-Aronzajn Theorem (1950)

Let K be PDS kernel. Then, there exists a Hilbert Space called *Feature Space* and a function called a *feature map* $\phi : \mathcal{X} \rightarrow \mathcal{F}$, such that $\forall (x, x') \in \mathcal{X}^2, \langle \phi(x), \phi(x') \rangle_{\mathcal{F}} = k(x, x')$.

Moreover, there exists a unique feature space $\phi(x) = k(\cdot, x) \in \mathcal{F}$ that satisfies the reproducing property, i.e.:

$$\forall f \in \mathcal{F}, \forall x \in \mathcal{X}, \langle k(\cdot, x), f \rangle_{\mathcal{F}} = f(x)$$

We refer to this kernel as the canonical one.

Please also notice that:

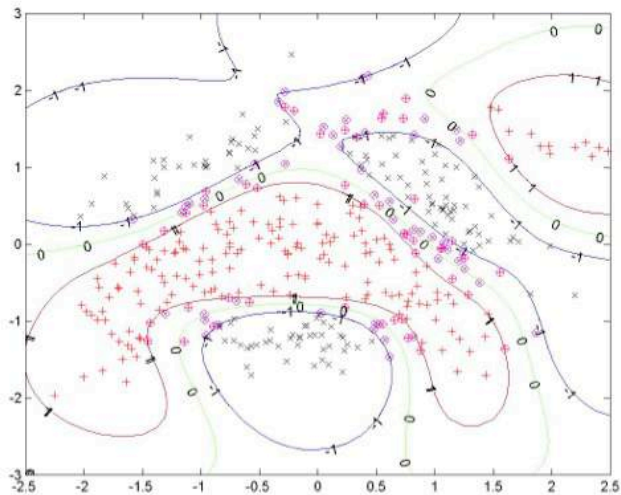
$$\forall (x, x') \in \mathcal{X}^2, \langle k(\cdot, x), k(\cdot, x') \rangle_{\mathcal{F}} = k(x, x')$$

Kernel between vectors

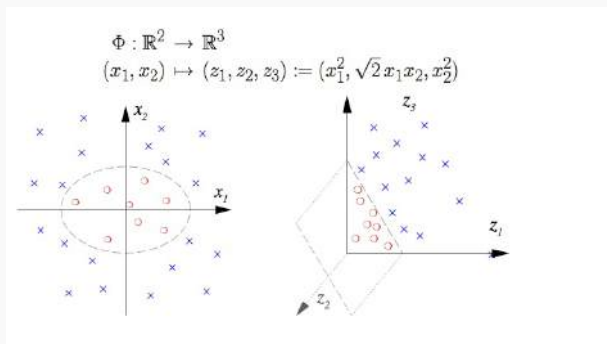
$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^p$

- Trivial linear kernel : $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{x}'$
- Polynomial kernel : $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + c)^d$
- Gaussian kernel : $k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2)$

Support Vector Machine



Example: polynomial kernel



Example : polynomial kernel

Kernel trick

We notice that $\phi(\mathbf{x}_1)^T \phi(\mathbf{x}')$ can be computed without working in \mathbb{R}^3

We can define directly $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T \phi(\mathbf{x}') = (\mathbf{x}^T \mathbf{x}')^2$

Closure properties of kernels

closure property	feature space representation
a) $K_1(x, y) + K_2(x, y)$	$\Phi(x) = (\Phi_1(x), \Phi_2(x))^T$
b) $\alpha K_1(x, y)$ for $\alpha > 0$	$\Phi(x) = \sqrt{\alpha} \Phi_1(x)$
c) $K_1(x, y) K_2(x, y)$	$\Phi(x)_{ij} = \Phi_1(x)_i \Phi_2(x)_j$ (tensor product)
d) $f(x)f(y)$ for any f	$\Phi(x) = f(x)$
e) $x^T A y$ for $A \succeq 0$ (i.e. psd)	$\Phi(x) = L^T x$ for $A = LL^T$ (Cholesky)

From those properties, we conclude that a polynomial of kernels is still a kernel.
the pointwise limit of kernels is also a kernel.

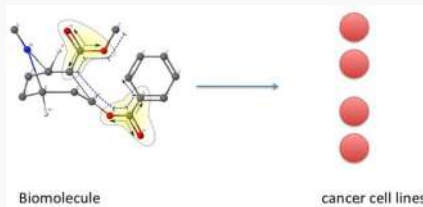
Much more interesting: kernels for complex objects

Kernels for

- **Complex (unstructured) objects:** texts, images, documents, signal, biological objects (gene, mRNA, protein, ...), functions, histograms
- **Structured objects:** sequences, trees, graphs, any composite objects

This made the success of kernels in computational biology, information retrieval (categorization for instance), but also in unexpected areas such as software metrics

Example: predict the property of a molecule



- **Inputs** : molecule (drug candidate)
- **Output** : activity on a cancer line (or several cancer lines)

A regression problem from structured data.

Kernel for labeled graphs

For a given length L , let us first enumerate all the paths of length $\ell \leq L$ in the training dataset (data are molecule = labeled graphs). Let m be the size of this (huge) set. For a graph, define $\phi(G) = (\phi_1(G), \dots, \phi_m(G), \dots, \phi_L(G))^T$ where $\phi_m(T)$ is 1 if the m^{th} path appears in the labeled graph G , and 0 otherwise.

Definition 1:

$$k_L(G, G') = \langle \phi(G), \phi(G') \rangle$$

Tanimoto kernel

$$k_L^t(G, G') = \frac{k_L(G, G')}{k_L(G, G) + k_L(G', G') - k_L(G, G')}$$

idea: k_m^t calculates the ratio between the number of elements of the intersection of the two sets of paths (G and G' are seen as bags of paths) and the number of elements of the union of the two sets.

Reference: Ralaivola et al. 2005, Su et al. 2011

Definition:

Suppose that $x \in \mathcal{X}$ is a **composite structure** and x_1, \dots, x_D are its "parts" according a relation R such that $(R(x, x_1, x_2, \dots, x_D))$ is true, with $x_d \in \mathcal{X}_d$ for each $1 \leq d \leq D$, D being a positive integer. k_d be a PDS kernel on a set $\mathcal{X} \times \mathcal{X}$, for all (x, x') , we define:

$$k_{conv}(x, x') = \sum_{(x_1, \dots, x_d) \in R^{-1}(x), (x'_1, \dots, x'_d) \in R^{-1}(x')} \prod_{d=1}^D k_d(x_d, x'_d)$$

$R^{-1}(x) =$ all decompositions (x_1, \dots, x_D) such that $(R(x, x_1, x_2, \dots, x_D))$. k_{conv} is a PDS kernel as well. Intuitive kernel, used as a building principle for a lot of other kernels. Next, we will see two examples.

Combine the advantages of graphical models and discriminative methods

Let $x \in \mathbb{R}^p$ be the input vector of a classifier.

- Learn a generative model $p_\theta(x)$ from unlabeled data x_1, \dots, x_n
- Define the Fisher vector as : $\mathbf{u}_\theta(x) = \nabla_\theta \log p_\theta(x)$
- Estimate the Fisher Information matrix of p_θ :
$$F_\theta = \mathbb{E}_{x \sim p_\theta} [\mathbf{u}_\theta(x) \mathbf{u}_\theta(x)^T]$$
- **Definition:** $k_{Fisher}(x, x') = \mathbf{u}_\theta(x)^T F_\theta^{-1} \mathbf{u}_\theta(x')$

Applications

Classification of secondary structure of proteins, topic modeling in documents, image classification and object recognition, audio signal classification ... Ref: Haussler, 1998. Perronnin et al. 2013.

- Use closure properties to build new kernels from existing ones
- Kernels can be defined for various objects:
 - **Structured objects:** (sets), graphs, trees, sequences, ...
 - Unstructured data with underlying structure: texts, images, documents, signal, biological objects
- **Kernel learning:**
 - Hyperparameter learning: see Chapelle et al. 2002
 - Multiple Kernel Learning: given k_1, \dots, k_m , learn a convex combination $\sum_i \beta_i k_i$ of kernels (see SimpleMKL Rakotomamonjy et al. 2008, unifying view in Kloft et al. 2010)

Introduction

Linear SVM

Nonlinear SVM and Kernels

Support Vector Regression

Conclusion et References

References

Probabilistic and Statistical Framework 1/2

- Let X be a random vector $\mathcal{X} = \mathbb{R}^p$
- and Y be a continuous random variable $\mathcal{Y} = \mathbb{R}$
- Let \mathbb{P} be the joint probability law of (X, Y)
- Let $\mathcal{S}_n = \{(x_1, y_1), \dots, (x_n, y_n)\}$, i.i.d. sample from \mathbb{P} .

Probabilistic and Statistical Framework 2/2

- Let $h : \mathbb{R}^p \rightarrow \mathbb{R} \in \mathcal{H}$, \mathcal{H} : some family of functions
- Let $\ell : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ be a local loss function
- Empirical risk : $R_n(h) = \frac{1}{n} \sum_i \ell(y_i, h(x_i))$, Regularizing term: $\Omega(h)$ which measures *complexity* de h .
- We search for : $\hat{h} = \arg \min_{h \in \mathcal{H}} R_n(h) + \lambda \Omega(h)$

Theorem (Minimal Risk under Squared Error Loss (MSE))

When ℓ is the squared loss: $\ell(y, h(x)) = (y - h(x))^2$, the best solution for the regression problem is the so-called regression function $h^*(x) = \mathbb{E}[Y|x]$. h^* is the function that provides the minimal risk.

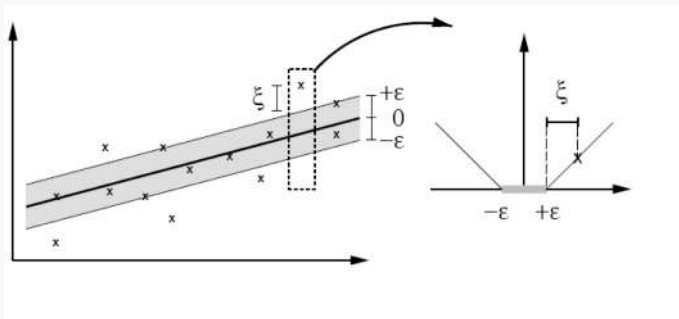
Proof:

Let h a predictive model.

Show that $R(h) = \mathbb{E}[(\mathbb{E}[Y|X] - h(X))^2] + R(h^*)$. Then, $R(h) \geq R(h^*)$ for any predictive model h , and therefore, $\min R(h) = R(h^*)$.

Support Vector Regression

- Extend the idea of maximal soft margin to regression
- Impose an ϵ -tube : ϵ -insensitive loss $|y' - y|_\epsilon = \max(0, |y' - y| - \epsilon)$



Support Vector Regression

SVR in the primal space

Given C and ϵ

$$\min_{w,b,\xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i (\xi_i + \xi_i^*)$$

s.c.

$$\forall i = 1, \dots, n, y_i - f(x_i) \leq \epsilon + \xi_i$$

$$\forall i = 1, \dots, n, f(x_i) - y_i \leq \epsilon + \xi_i^*$$

$$\forall i = 1, \xi_i \geq 0, \xi_i^* \geq 0$$

$$\text{with } f(x) = \langle w, \phi(x) \rangle + b$$

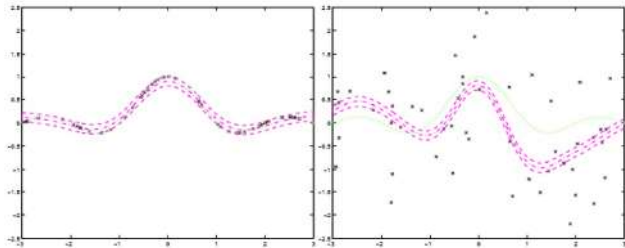
General case : ϕ is a feature map associated with a positive definite kernel k .

$$\begin{aligned} \min_{\alpha, \alpha^*} \quad & \sum_{i,j} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)k(x_i, x_j) + \epsilon \sum_i (\alpha_i + \alpha_i^*) - \sum_i y_i (\alpha_i - \alpha_i^*) \\ \text{s.c.} \quad & \sum_i (\alpha_i - \alpha_i^*) = 0 \text{ and } 0 \leq \alpha_i \leq C \text{ and } 0 \leq \alpha_i^* \leq C \\ & w = \sum_{i=1}^n (\alpha_i - \alpha_i^*) \phi(x_i) \end{aligned}$$

Solution

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha_i^*)k(x_i, x) + b$$

Support Vector Regression: example in 1D



Identical machine parameters ($\varepsilon = 0.2$), but different amounts of noise in the data.

B. Schölkopf, Candès, February 2002

Introduction

Linear SVM

Nonlinear SVM and Kernels

Support Vector Regression

Conclusion et References

References

Supervised Classification by Support Vector Machine

Advantages

- A unique minimum, convex programming with linear constraints: exact solution !
- Some properties of the Gaussian kernel: kernel machine with a Gaussian kernel = universal approximator
- Flexibility: the kernel is chosen to be adapted to the nature of data, a systematic way to deal with **complex data**
- Multi-class: M classifiers - One-versus-all
- Structural Risk driven : algorithm inspired from Vapnik and Chervonenkis 's works
- Can be composed with a preprocessing (first neural network layers) - kernel learning
- Kernels and their approximations are used to shed light on deep neural networks

Drawbacks

- Kernel choice
- Solver expensive in time and memory - does not scale !
- In order to scale up kernel methods: go through approximations of Gram matrix or Random Fourier Features (approximation spectral approximation of the kernel function itself)

Kernel trick (working with canonical feature map: $\phi(x) = k(\cdot, x)$ and associated Hilbert Space (called reproducing Hilbert space))

- apply to many other algorithms !
- Easiest: Ridge Regression
- PCA \rightarrow Kernel PCA
- CCA \rightarrow Kernel CCA
- Kalman Filtering \rightarrow Kernel Kalman Filter

Introduction

Linear SVM

Nonlinear SVM and Kernels

Support Vector Regression

Conclusion et References

References

References

- Article really cool (a bit of maths, preparation to M2) : **A tutorial review of RKHS methods in Machine Learning, Hofman , Schoelkopf, Smola, 2005**
(https://www.researchgate.net/publication/228827159_A_Tutorial_Review_of_RKHS_Methods_in_Machine_Learning)
- FOundations of Machine Learning, chapter about kernels, Morhi et al., MIT press (2012).
- BOSER, Bernhard E., Isabelle M. GUYON, and Vladimir N. VAPNIK, 1992. A training algorithm for optimal margin classifiers. In: COLT '92: Proceedings of the Fifth Annual Workshop on Computational Learning Theory. New York, NY, USA: ACM Press, pp. 144-152.
- CORTES, Corinna, and Vladimir VAPNIK, 1995. Support-vector networks. Machine Learning, 20(3), 273-297.

SD-TSIA204 : PCA and LASSO

Ekhine Irurozki

Télécom Paris, IP Paris

Lasso : Reminding the model

$$\mathbf{y} = X\boldsymbol{\theta}^{\star} + \boldsymbol{\varepsilon} \in \mathbb{R}^n$$

$$X = [\mathbf{x}_1, \dots, \mathbf{x}_p] = \begin{pmatrix} x_{1,1} & \dots & x_{1,p} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \dots & x_{n,p} \end{pmatrix} \in \mathbb{R}^{n \times p}, \boldsymbol{\theta}^{\star} \in \mathbb{R}^p$$

Motivation

In the presence of super-collinearity the OLS estimators can not be given.

Estimators $\hat{\theta}$ with many zero coefficients are useful :

- ▶ for interpretation
- ▶ for computational efficiency if p is huge

Underlying idea : **variable selection**

Rem: also useful if θ^* has few non-zero coefficients

Variable selection overview

- ▶ Screening : remove the \mathbf{x}_j 's whose correlation with \mathbf{y} is weak
 - pros : fast (+++), *i.e.*, one pass over data, intuitive (+++)
 - cons : neglect variables interactions \mathbf{x}_j , weak theory (- - -)
- ▶ Greedy methods aka stagewise / stepwise
 - pros : fast (++), intuitive (++)
 - cons : propagates wrong selection forward ; weak theory (-)
- ▶ Sparsity enforcing penalized methods (e.g., Lasso)
 - pros : better theory for convex cases (++)
 - cons : can be still slow (-)

The ℓ_0 pseudo-norm

The support of $\boldsymbol{\theta} \in \mathbb{R}^p$ is the set of indexes of non-zero coordinates :

$$\text{supp}(\boldsymbol{\theta}) = \{j \in \llbracket 1, p \rrbracket, \theta_j \neq 0\}$$

The ℓ_0 pseudo-norm of a $\boldsymbol{\theta} \in \mathbb{R}^p$ is the number of non-zero coordinates :

$$\|\boldsymbol{\theta}\|_0 = \text{card}\{j \in \llbracket 1, p \rrbracket, \theta_j \neq 0\}$$

Rem: $\|\cdot\|_0$ is not a norm, $\forall t \in \mathbb{R}^*, \|t\boldsymbol{\theta}\|_0 = \|\boldsymbol{\theta}\|_0$

Rem: $\|\cdot\|_0$ it is not even convex, $\boldsymbol{\theta}_1 = (1, 0, 1, \dots, 0)$ $\boldsymbol{\theta}_2 = (0, 1, 1, \dots, 0)$ and
 $3 = \|\frac{\boldsymbol{\theta}_1 + \boldsymbol{\theta}_2}{2}\|_0 \geq \frac{\|\boldsymbol{\theta}_1\|_0 + \|\boldsymbol{\theta}_2\|_0}{2} = 2$

Regularization with the ℓ_0 penalty

First try to get a sparsity enforcing penalty : use ℓ_0 as a penalty (or regularization)

$$\hat{\boldsymbol{\theta}}_{\lambda} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_0}_{\text{regularization}} \right)$$

Combinatorial problem !!!

Exact solution : require considering all sub-models, *i.e.*, computing OLS for all possible support ; meaning one might need 2^p least squares computation !

Example :

$p = 10$ possible : $\approx 10^3$ least squares

$p = 30$ impossible : $\approx 10^{10}$ least squares

Rem: problem “NP-hard”, can be solved for small problems by mixed integer programming.

Regularization with the ℓ_1 penalty : Lasso

Lasso : *Least Absolute Shrinkage and Selection Operator* Tibshirani (1996)

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_1}_{\text{regularization}} \right)$$

or $\|\boldsymbol{\theta}\|_1 = \sum_{j=1}^p |\theta_j|$ (sum of absolute values of the coefficients)

► We recover the limiting cases :

$$\begin{aligned} \lim_{\lambda \rightarrow 0} \hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} &= \hat{\boldsymbol{\theta}}^{\text{OLS}} \\ \lim_{\lambda \rightarrow +\infty} \hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} &= \mathbf{0} \in \mathbb{R}^p \end{aligned}$$

Constraint point of view

The following problem :

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\frac{1}{2} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_1}_{\text{regularization}} \right)$$

shares the same solutions as the constrained formulation :

$$\begin{cases} \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 \\ \text{s.t. } \|\boldsymbol{\theta}\|_1 \leq T \end{cases}$$

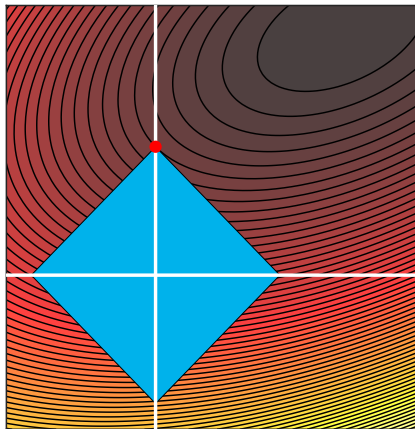
for some $T > 0$.

Rem: unfortunately the link $T \leftrightarrow \lambda$ is not explicit

- ▶ If $T \rightarrow 0$ one recovers the null vector : $0 \in \mathbb{R}^p$
- ▶ If $T \rightarrow \infty$ one recovers $\hat{\boldsymbol{\theta}}^{\text{OLS}}$ (unconstrained)

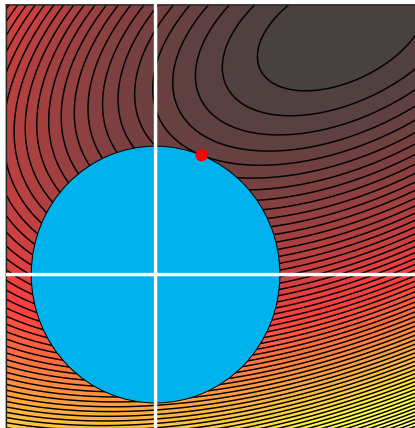
Interpretation : Optimization under ℓ_1 constraint, sparse solution

$$\begin{aligned} \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} & \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 \\ \text{s.t. } & \|\boldsymbol{\theta}\|_1 \leq T \end{aligned}$$



Interpretation : Optimization under ℓ_2 constraint, non-sparse solution

$$\begin{aligned} \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} & \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 \\ \text{s.t. } & \|\boldsymbol{\theta}\|_2 \leq T \end{aligned}$$



Existence and uniqueness

Exercise : the Lasso estimator is not always **unique** for a fixed λ (consider cases with two equal columns in X). However, the prediction is unique. Show these points.

Analytical solution

Non-smooth problem

In general, there is no explicit solution

- ▶ Quadratic programming with constraints
- ▶ Iterative ridge
- ▶ Proximal gradient method

Sub-gradients / sub-differential

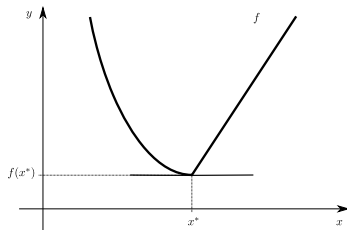
For a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $u \in \mathbb{R}^n$ is a sub-gradient of f at x^* , if for any $x \in \mathbb{R}^n$,

$$f(x) \geq f(x^*) + \langle u, x - x^* \rangle$$

The sub-differential is the set of all sub-gradients,

$$\partial f(x^*) = \{u \in \mathbb{R}^n : \forall x \in \mathbb{R}^n, f(x) \geq f(x^*) + \langle u, x - x^* \rangle\}.$$

Rem: if the sub-gradient is unique, one recovers the standard gradient



Sub-gradients / sub-differential

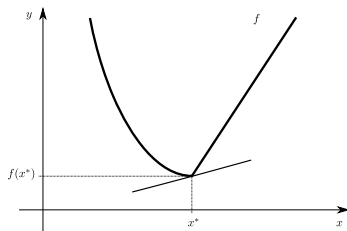
For a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $u \in \mathbb{R}^n$ is a sub-gradient of f at x^* , if for any $x \in \mathbb{R}^n$,

$$f(x) \geq f(x^*) + \langle u, x - x^* \rangle$$

The sub-differential is the set of all sub-gradients,

$$\partial f(x^*) = \{u \in \mathbb{R}^n : \forall x \in \mathbb{R}^n, f(x) \geq f(x^*) + \langle u, x - x^* \rangle\}.$$

Rem: if the sub-gradient is unique, one recovers the standard gradient



Sub-gradients / sub-differential

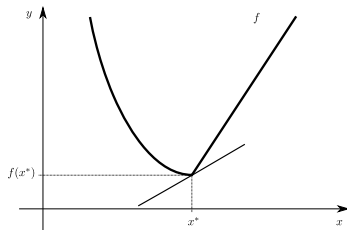
For a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $u \in \mathbb{R}^n$ is a sub-gradient of f at x^* , if for any $x \in \mathbb{R}^n$,

$$f(x) \geq f(x^*) + \langle u, x - x^* \rangle$$

The sub-differential is the set of all sub-gradients,

$$\partial f(x^*) = \{u \in \mathbb{R}^n : \forall x \in \mathbb{R}^n, f(x) \geq f(x^*) + \langle u, x - x^* \rangle\}.$$

Rem: if the sub-gradient is unique, one recovers the standard gradient



Sub-gradients / sub-differential

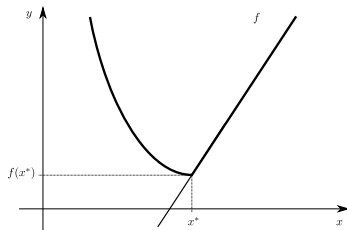
For a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $u \in \mathbb{R}^n$ is a sub-gradient of f at x^* , if for any $x \in \mathbb{R}^n$,

$$f(x) \geq f(x^*) + \langle u, x - x^* \rangle$$

The sub-differential is the set of all sub-gradients,

$$\partial f(x^*) = \{u \in \mathbb{R}^n : \forall x \in \mathbb{R}^n, f(x) \geq f(x^*) + \langle u, x - x^* \rangle\}.$$

Rem: if the sub-gradient is unique, one recovers the standard gradient



Fermat's Rule : optimality of x^*

A point x^* is a minimum of a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ if and only if $0 \in \partial f(x^*)$

Proof : use the sub-gradient definition :

- ▶ 0 is a sub-gradient of f at x^* if and only if $\forall x \in \mathbb{R}^n, f(x) \geq f(x^*) + \langle 0, x - x^* \rangle$

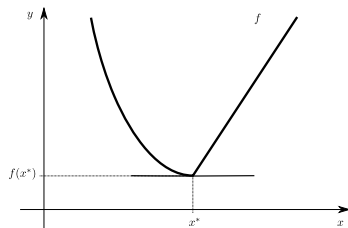
Fermat's Rule : optimality of x^*

A point x^* is a minimum of a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ if and only if $0 \in \partial f(x^*)$

Proof : use the sub-gradient definition :

- 0 is a sub-gradient of f at x^* if and only if $\forall x \in \mathbb{R}^n, f(x) \geq f(x^*) + \langle 0, x - x^* \rangle$

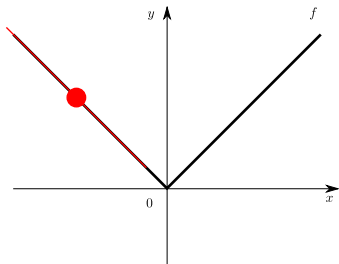
Rem:Visually, it corresponds to a horizontal tangent



Absolute value sub-differential

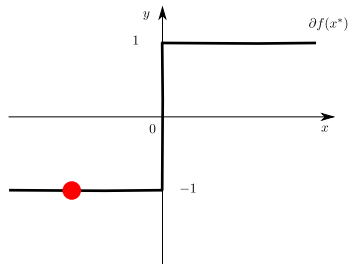
Function (abs) :

$$f : \begin{cases} \mathbb{R} & \rightarrow \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



Sub-differential (sign)

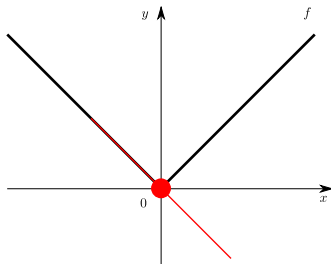
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

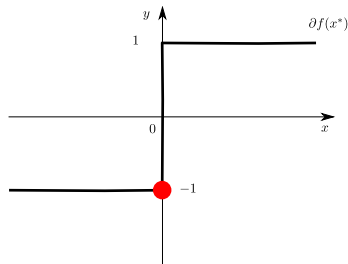
Function (abs) :

$$f : \begin{cases} \mathbb{R} & \rightarrow \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



Sub-differential (sign)

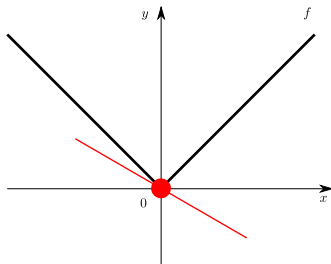
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

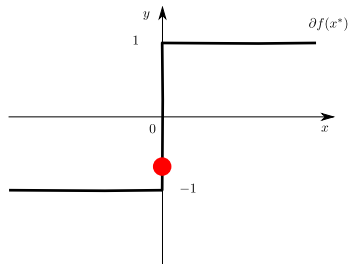
Function (abs) :

$$f : \begin{cases} \mathbb{R} & \rightarrow \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



Sub-differential (sign)

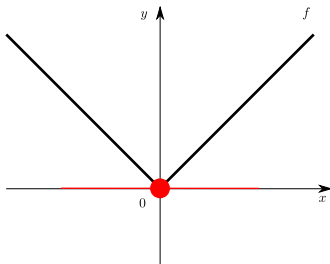
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

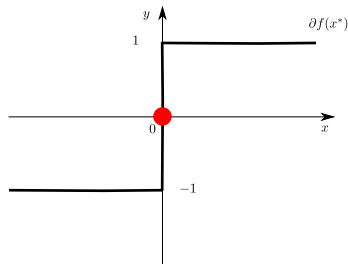
Function (abs) :

$$f : \begin{cases} \mathbb{R} & \rightarrow \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



Sub-differential (sign)

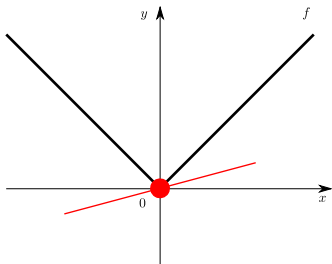
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

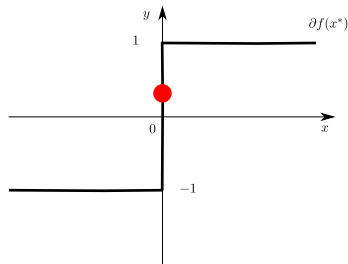
Function (abs) :

$$f : \begin{cases} \mathbb{R} & \rightarrow \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



Sub-differential (sign)

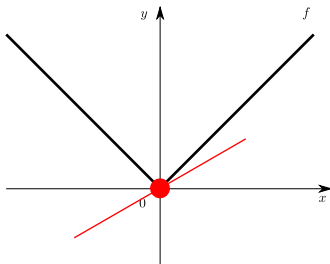
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

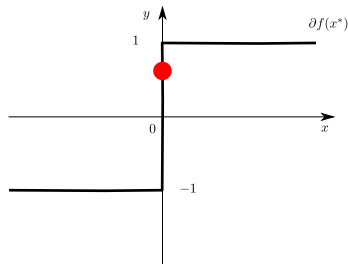
Function (abs) :

$$f : \begin{cases} \mathbb{R} & \rightarrow \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



Sub-differential (sign)

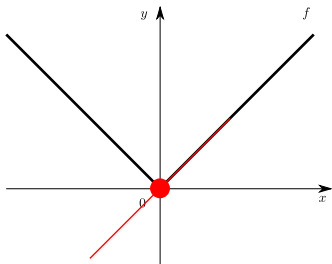
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

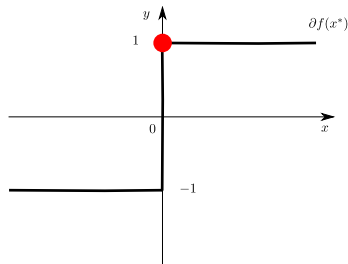
Function (abs) :

$$f : \begin{cases} \mathbb{R} & \rightarrow \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



Sub-differential (sign)

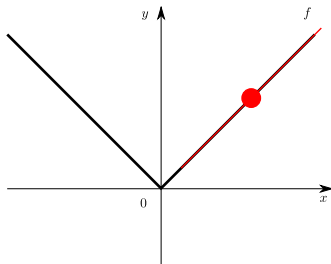
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

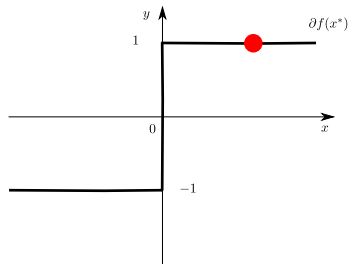
Function (abs) :

$$f : \begin{cases} \mathbb{R} & \rightarrow \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



Sub-differential (sign)

$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Fermat's rule for the Lasso

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_1}_{\text{regularization}} \right)$$

Necessary and sufficient optimality (Fermat) :

$$\forall j \in [p], \mathbf{x}_j^{\top} \left(\frac{\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}}}{\lambda} \right) \in \begin{cases} \{\text{sign}(\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}})_j\} & \text{if } (\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}})_j \neq 0, \\ [-1, 1] & \text{if } (\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}})_j = 0. \end{cases}$$

Rem: If $\lambda > \lambda_{\max} := \max_{j \in \llbracket 1, p \rrbracket} |\langle \mathbf{x}_j, \mathbf{y} \rangle|$, then $\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} = \mathbf{0}$

Iterative algorithm for Lasso (Sub-gradient descent)

Lasso analysis

Theory : more involved for the Lasso than for least squares / Ridge

Recent reference : Bühlmann and van de Geer (2011)

In a nutshell : add bias to the standard least squares to perform variance reduction

Combining Lasso and Ridge (ℓ_1/ℓ_2 regularization) : Elastic-net

The Elastic-Net, introduced by **Zou and Hastie (2005)** is the (unique) solution of

$$\hat{\boldsymbol{\theta}}_{\lambda} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left[\frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 + \lambda \left(\gamma \|\boldsymbol{\theta}\|_1 + (1 - \gamma) \frac{\|\boldsymbol{\theta}\|_2^2}{2} \right) \right]$$

Motivation : help selecting all relevant but correlated variable (not only one as for the Lasso)

Rem: two parameters needed, one for global regularization, one trading-off Ridge vs. Lasso

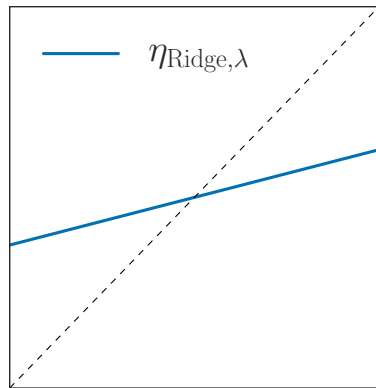
Rem: the solution is unique and the size of the Elastic-Net support is smaller than $\min(n, p)$

Comparing regularizers in 1D : Ridge

Solve :

$$\eta_{\lambda}(z) = \arg \min_{x \in \mathbb{R}} x \mapsto \frac{1}{2}(z - x)^2 + \frac{\lambda}{2}x^2$$

$$\eta_{\lambda}(z) = \frac{z}{1 + \lambda}$$



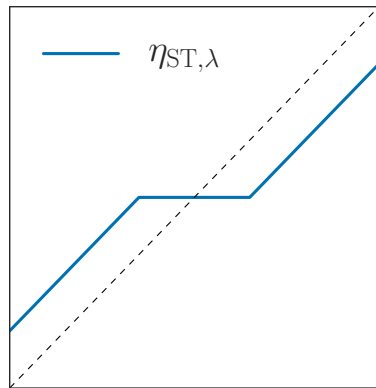
ℓ_2 shrinkage : Ridge

Comparing regularizers in 1D : Lasso

Solve :

$$\eta_{\lambda}(z) = \arg \min_{x \in \mathbb{R}} x \mapsto \frac{1}{2}(z - x)^2 + \lambda|x|$$

$$\eta_{\lambda}(z) = \text{sign}(z)(|z| - \lambda)_+$$



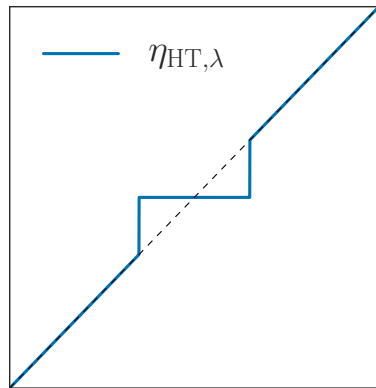
ℓ_1 shrinkage : soft thresholding

Comparing regularizers in 1D : ℓ_0

Solve :

$$\eta_\lambda(z) = \arg \min_{x \in \mathbb{R}} x \mapsto \frac{1}{2}(z - x)^2 + \lambda \mathbb{1}_{x \neq 0}$$

$$\eta_\lambda(z) = z \mathbb{1}_{|z| \geq \sqrt{2\lambda}}$$

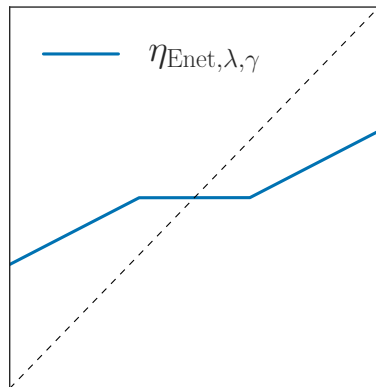


ℓ_0 shrinkage : hard thresholding

Comparing regularizers in 1D : Elastic-Net

Solve :

$$\eta_{\lambda}(z) = \arg \min_{x \in \mathbb{R}} x \mapsto \frac{1}{2}(z-x)^2 + \lambda(\gamma|x| + (1-\gamma)\frac{x^2}{2})$$



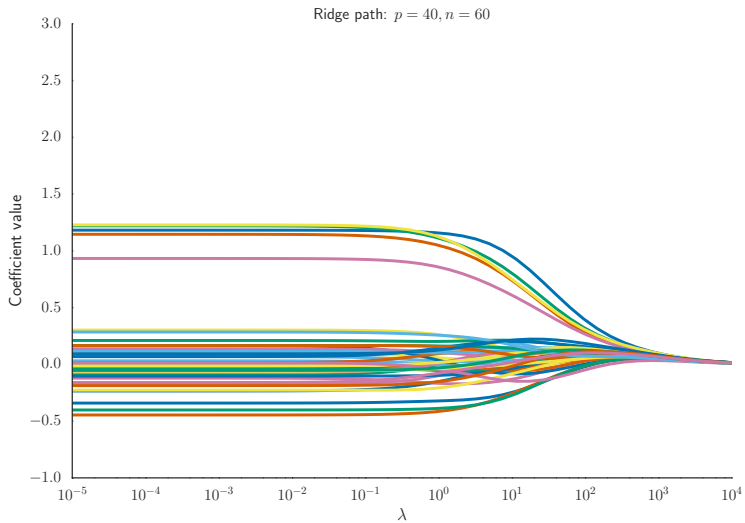
ℓ_1/ℓ_2

Numerical example on simulated data

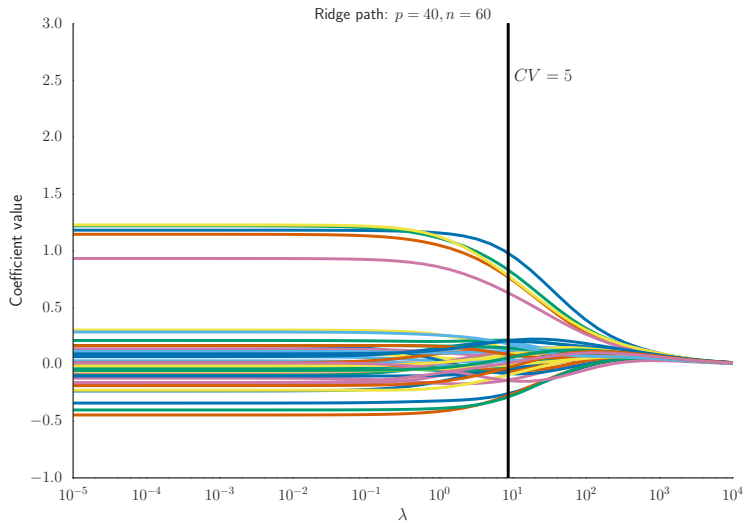
- ▶ $\theta^* = (1, 1, 1, 1, 1, 0, \dots, 0) \in \mathbb{R}^p$ (5 non-zero coefficients)
- ▶ $X \in \mathbb{R}^{n \times p}$ has columns drawn according to a Gaussian distribution
- ▶ $y = X\theta^* + \varepsilon \in \mathbb{R}^n$ with $\varepsilon \sim \mathcal{N}(0, \sigma^2 \text{Id}_n)$
- ▶ We use a grid of 50 λ values

For this example : $n = 60, p = 40, \sigma = 1$

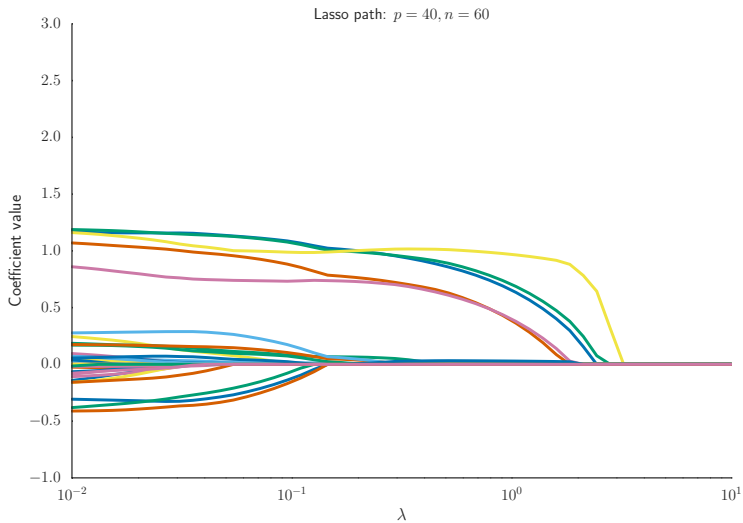
Lasso vs Ridge



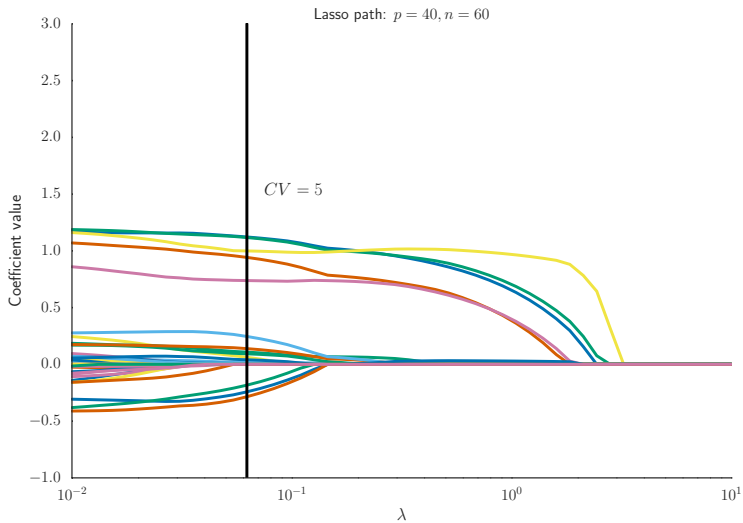
Lasso vs Ridge



Lasso vs Ridge



Lasso vs Ridge



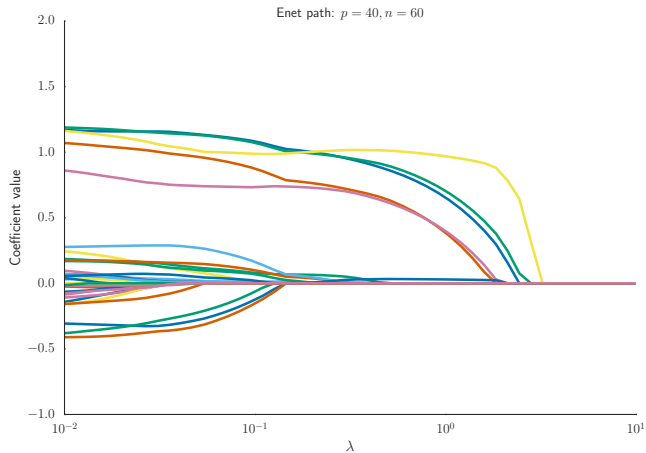
Lasso properties

- ▶ Solutions is not necessarily unique
- ▶ The analytic form does not necessarily exist
- ▶ Numerical aspect : the Lasso is a **convex** problem
- ▶ Variable selection / sparse solutions : $\hat{\theta}_{\lambda}^{\text{Lasso}}$ has potentially many zeroed coefficients. The λ parameter controls the sparsity level : if λ is large, solutions are very sparse.

Example : We got 17 non-zero coefficients for LassoCV in the previous simulated example

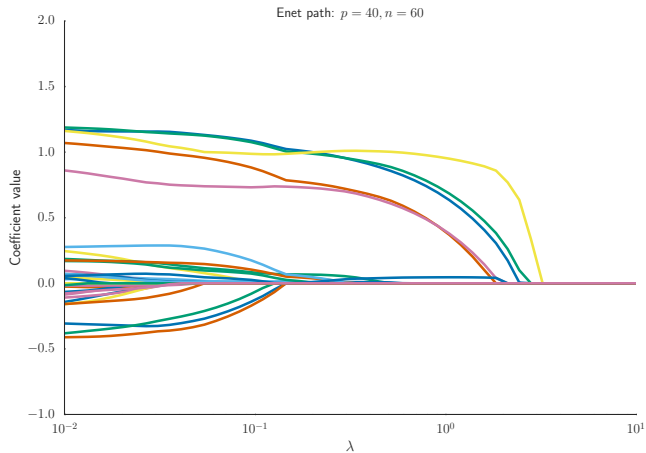
Rem: RidgeCV has no zero coefficients

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



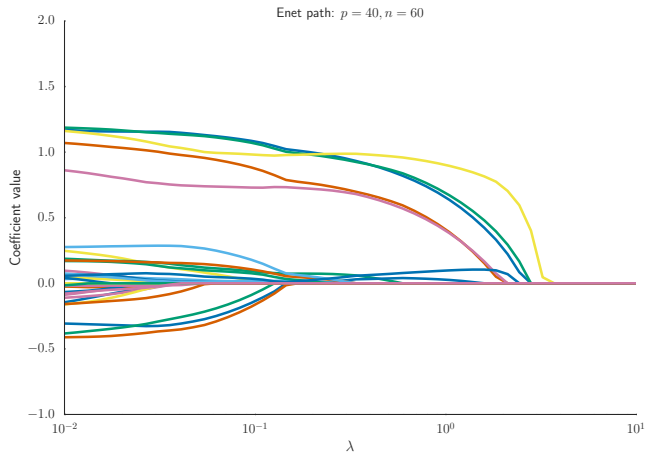
$\gamma = 1.00$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



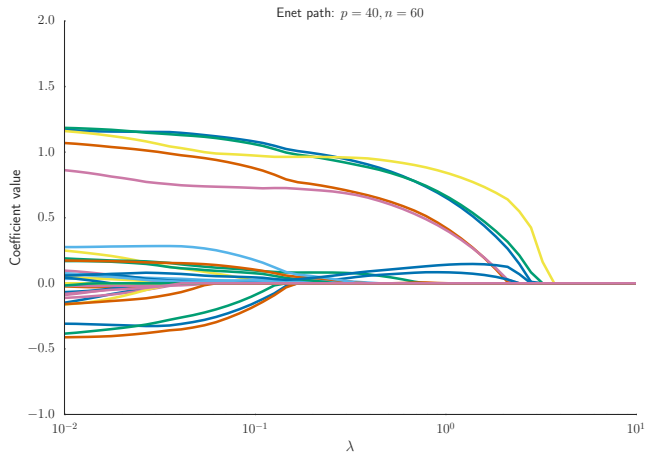
$\gamma = 0.99$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



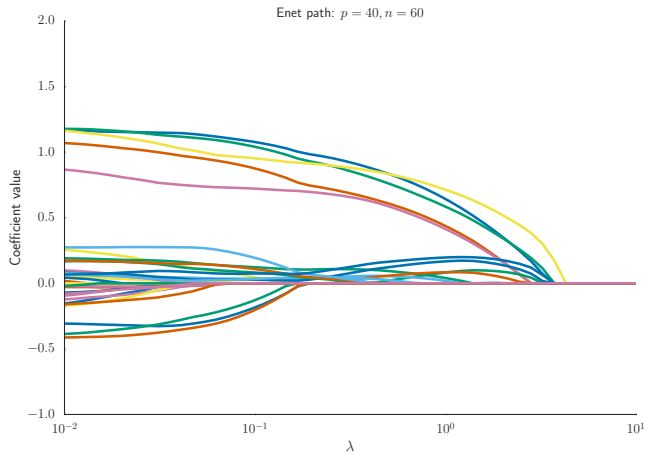
$\gamma = 0.95$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



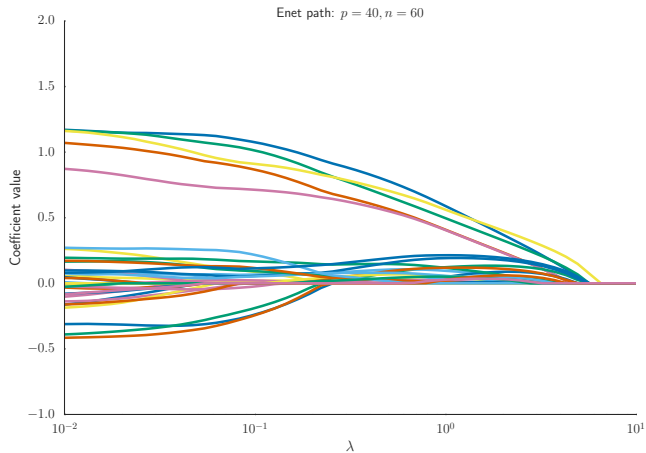
$\gamma = 0.90$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



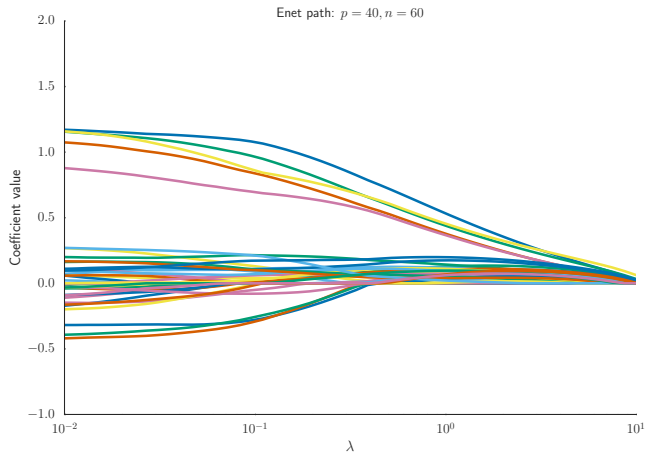
$\gamma = 0.75$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



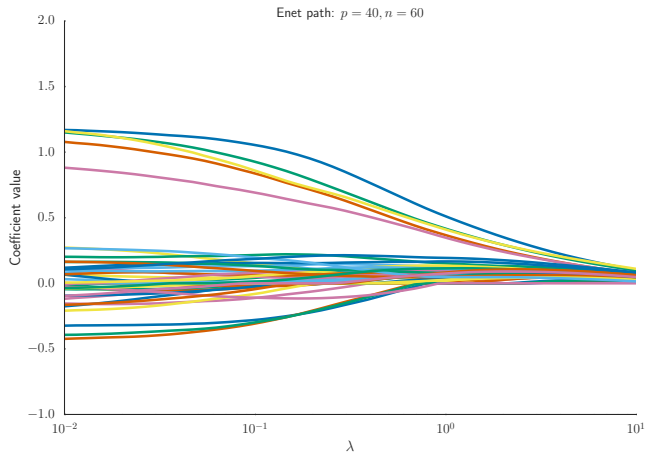
$\gamma = 0.50$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



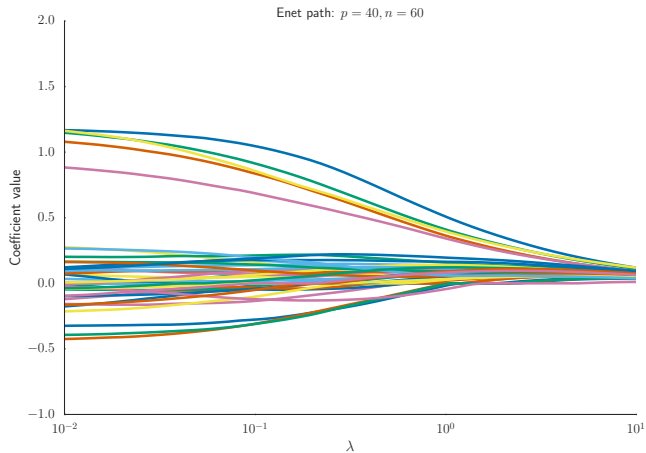
$\gamma = 0.25$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



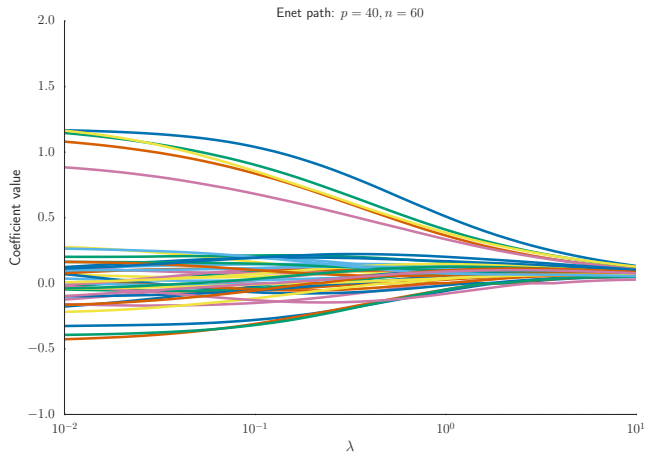
$\gamma = 0.1$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



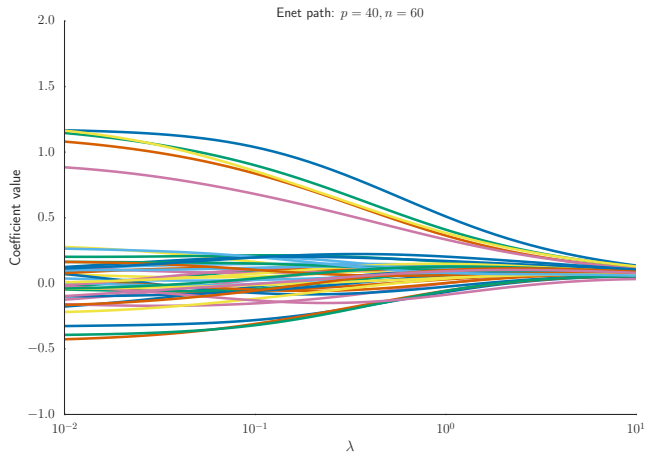
$\gamma = 0.05$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



$\gamma = 0.01$

Elastic-Net : $\gamma\|\boldsymbol{\theta}\|_1 + (1 - \gamma)\|\boldsymbol{\theta}\|_2^2/2$



$\gamma = 0.00$

The Lasso bias

The Lasso is biased : it shrinks large coefficients towards 0

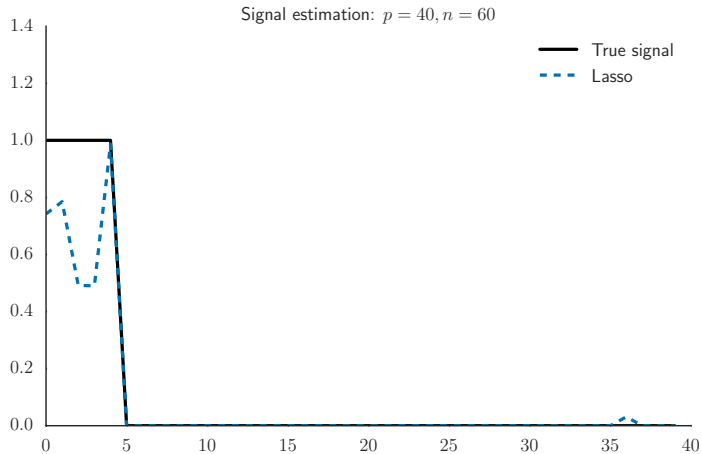


Illustration over the previous example

The Lasso bias

The Lasso is biased : it shrinks large coefficients towards 0

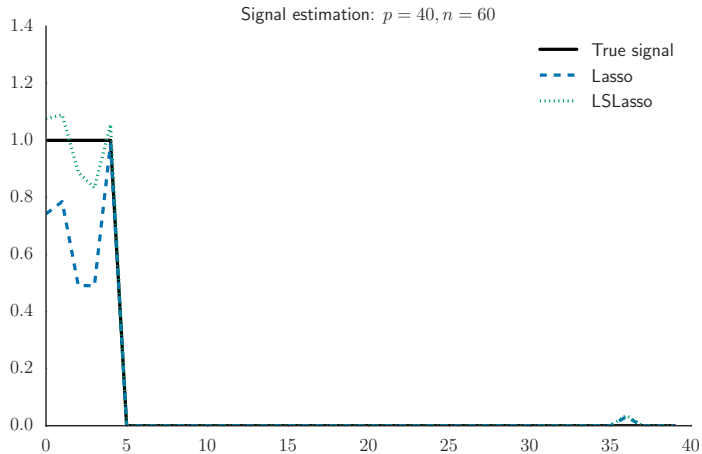


Illustration over the previous example

The Lasso bias : a simple remedy

How to rescale shrunk coefficients ?

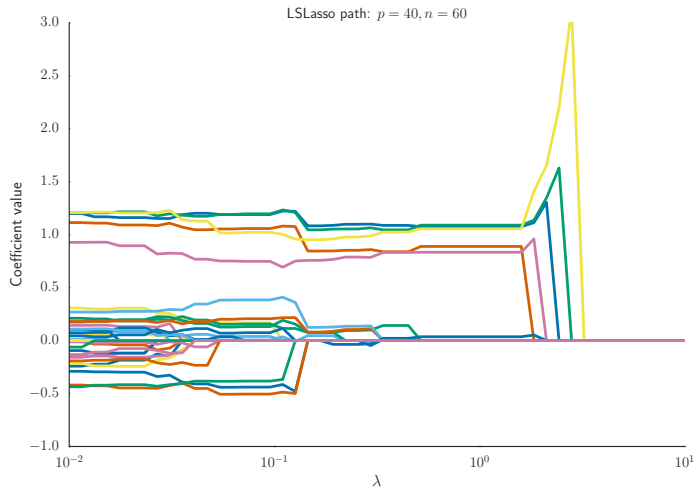
LSLasso (Least Square Lasso)

1. Lasso : compute $\hat{\theta}_{\lambda}^{\text{Lasso}}$
 2. Perform least squares over selected variables : $\text{supp}(\hat{\theta}_{\lambda}^{\text{Lasso}})$
- $$\hat{\theta}_{\lambda}^{\text{LSLasso}} = \arg \min_{\substack{\theta \in \mathbb{R}^p \\ \text{supp}(\theta) = \text{supp}(\hat{\theta}_{\lambda}^{\text{Lasso}})}} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\theta\|_2^2$$

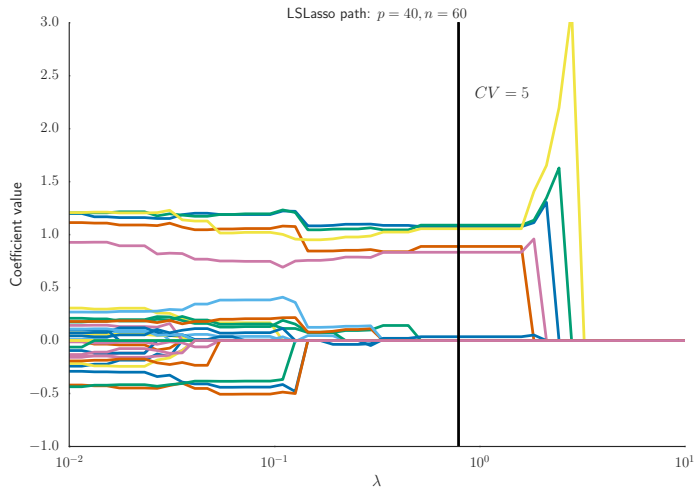
Rem: perform CV for the double step procedure ; choosing λ by LassoCV and then performing OLS keeps too many variables

Rem: LSLasso is not coded in standard packages

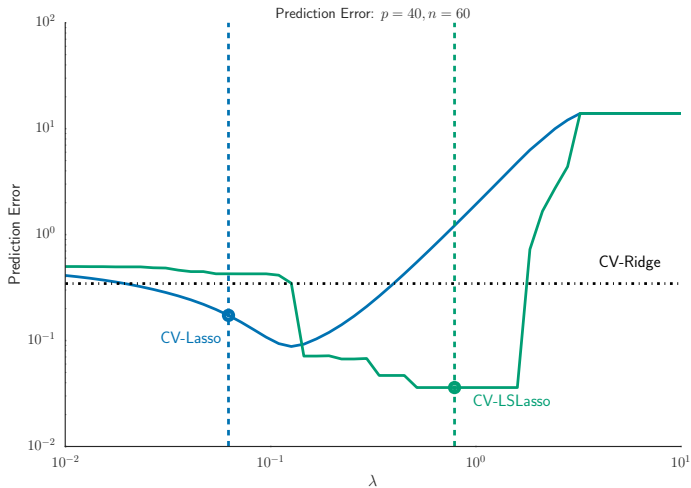
De-biasing



De-biasing



Prediction : Lasso vs. LSLasso



LSLasso evaluation

Pros

- ▶ the “true” large coefficients are less shrunk
- ▶ CV recovers less “parasite” variables (improve interpretability)
e.g., in the previous example the LSLassoCV recovers exactly the 5 “true” non zero variables, up to a single false positive

LSLasso : especially useful for estimation

Cons

- ▶ the difference in term of prediction is not always striking
- ▶ requires (slightly) more computation : needs to compute as many OLS as λ 's

Principal components analysis, PCA

What is it ?

- ▶ PCA is an unsupervised learning technique : the goal is to find a lower dimensional representation of the data that keeps as much of the variance of the original data. Can be used as a preprocessing for Clustering
- ▶ We use it here as a preprocessing for the OLS (aka PCA before OLS, aka PCRegression, ...)

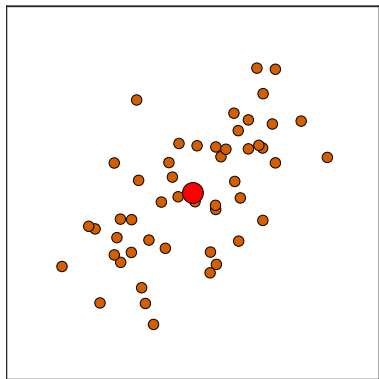
Goal : Reduce the dimensionality while keeping the variance in the data

High level idea : remove

- ▶ Super-collinearity
- ▶ Close to 0 variance features

Graphical representation (not to be confused with OLS)

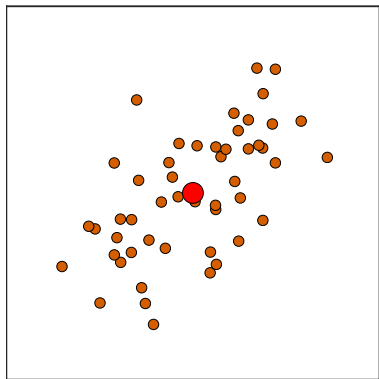
Main axis : variance maximization



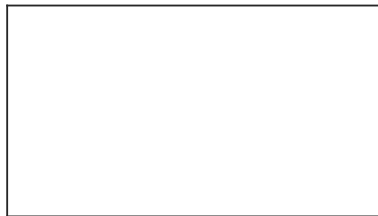
Data and mean



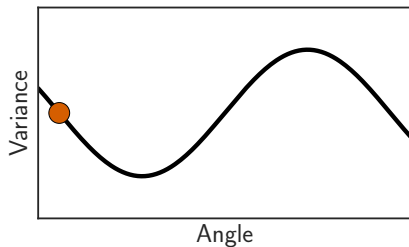
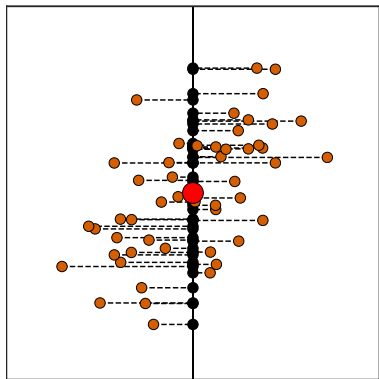
Main axis : variance maximization



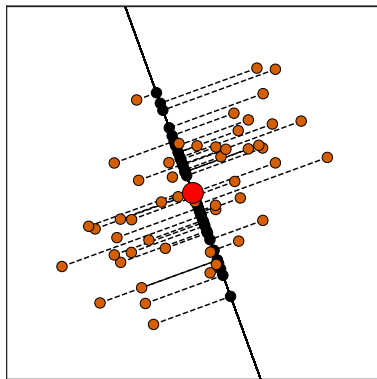
Data and mean



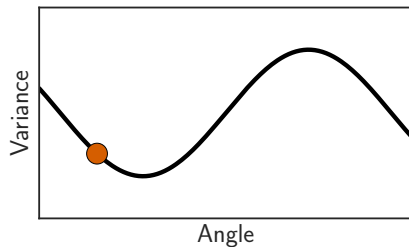
Main axis : variance maximization



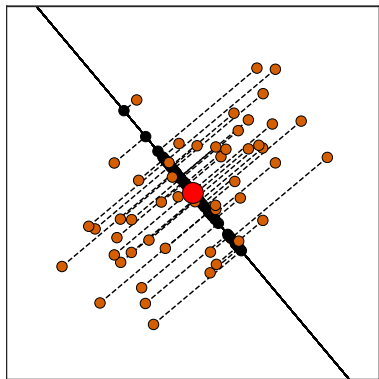
Main axis : variance maximization



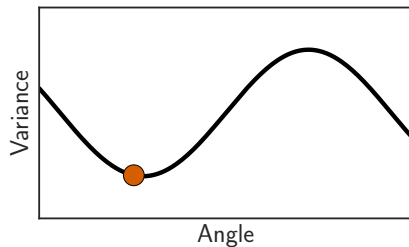
Data, mean and projection



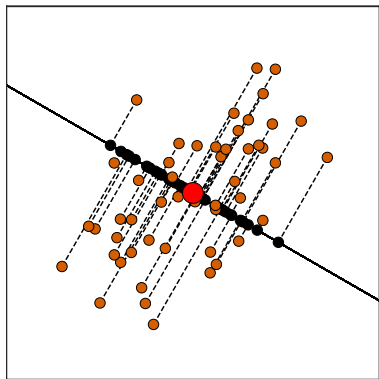
Main axis : variance maximization



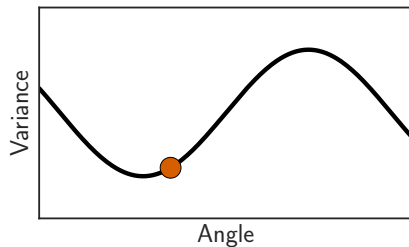
Data, mean and projection



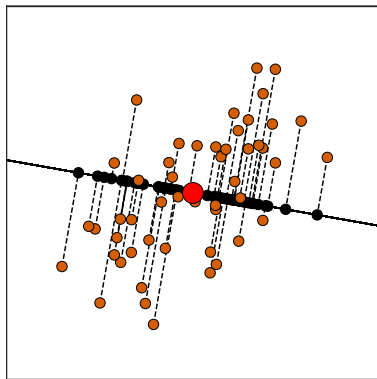
Main axis : variance maximization



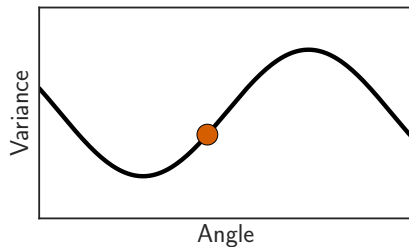
Data, mean and projection



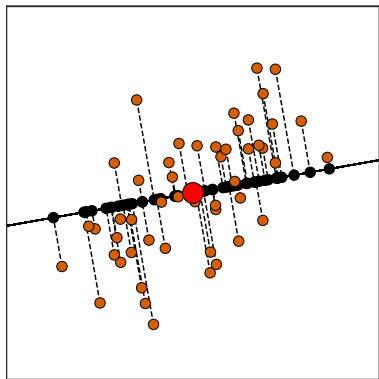
Main axis : variance maximization



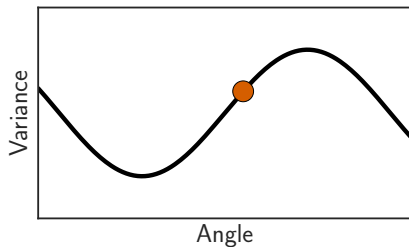
Data, mean and projection



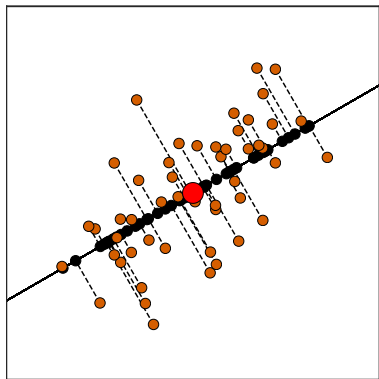
Main axis : variance maximization



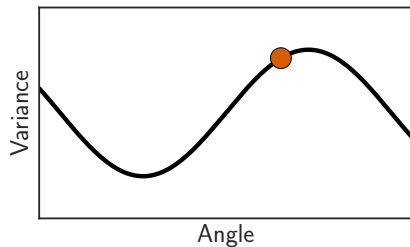
Data, mean and projection



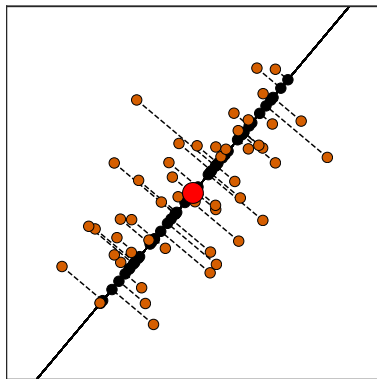
Main axis : variance maximization



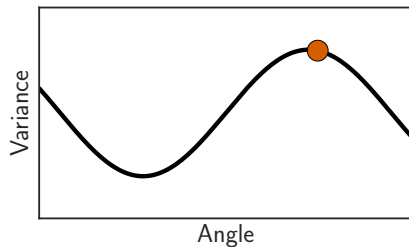
Data, mean and projection



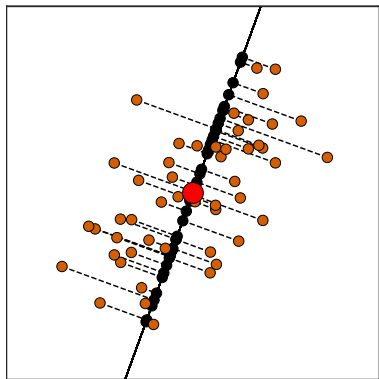
Main axis : variance maximization



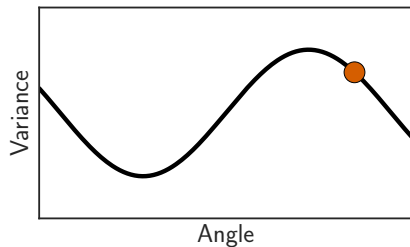
Data, mean and projection



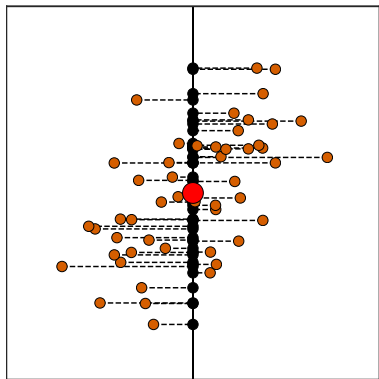
Main axis : variance maximization



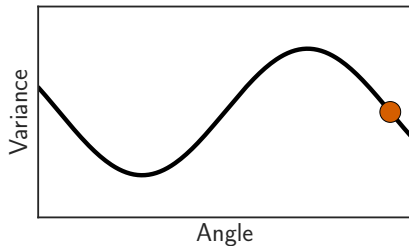
Data, mean and projection



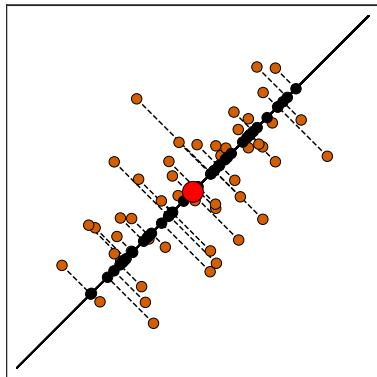
Main axis : variance maximization



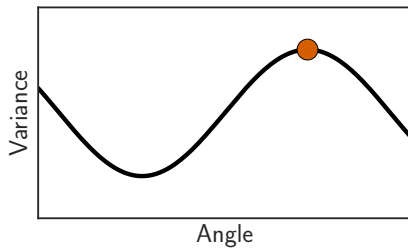
Data, mean and projection



Main axis : variance maximization

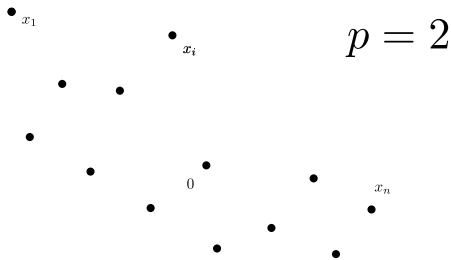


Principal direction (main axis)



Variance of the distances along direction v

We observe n points x_1, \dots, x_n , i.e., $X = [x_1, \dots, x_n]^T \in \mathbb{R}^{n \times p}$, n observations (rows), p features (columns)

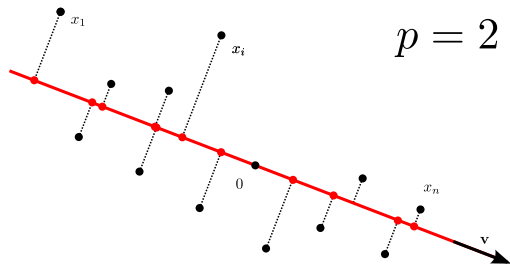


Rem: we have to center and scale the dataset : the points have a zero average
 $X \leftarrow [x_1 - \bar{x}_n, \dots, x_n - \bar{x}_n]^T = X - \mathbf{1}_n \bar{x}_n^T$
and variance 1.

Rem: The distance from x_i to the origin is $x_i^T v$, and the variances are $\sum_{i=1}^n (x_i^T v_1)^2$

Variance of the distances along direction v

We observe n points x_1, \dots, x_n , i.e., $X = [x_1, \dots, x_n]^T \in \mathbb{R}^{n \times p}$, n observations (rows), p features (columns)

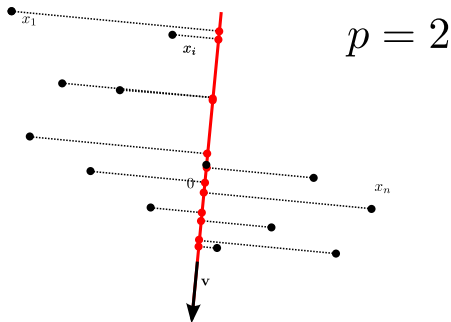


Rem: we have to center and scale the dataset : the points have a zero average
 $X \leftarrow [x_1 - \bar{x}_n, \dots, x_n - \bar{x}_n]^T = X - \mathbf{1}_n \bar{x}_n^T$
and variance 1.

Rem: The distance from x_i to the origin is $x_i^T v$, and the variances are $\sum_{i=1}^n (x_i^T v)^2$

Variance of the distances along direction v

We observe n points x_1, \dots, x_n , i.e., $X = [x_1, \dots, x_n]^T \in \mathbb{R}^{n \times p}$, n observations (rows), p features (columns)



Rem: we have to center and scale the dataset : the points have a zero average
 $X \leftarrow [x_1 - \bar{x}_n, \dots, x_n - \bar{x}_n]^T = X - \mathbf{1}_n \bar{x}_n^T$
and variance 1.

Rem: The distance from x_i to the origin is $x_i^T v$, and the variances are $\sum_{i=1}^n (x_i^T v_1)^2$

Connection between PCA and variance (sketch), first step

Goal : find the direction v_1 that maximizes the variance of the data

- ▶ The data is centered and standardized
- ▶ Direction $v_1 \in \mathbb{R}^p$ is a linear combination of the original dimensions of X and $\|v\| = 1$
- ▶ The distance from the origin to the projection of x_i onto v_1 is $x_i^\top v_1$
- ▶ The variance along v_i of the projections is $\sum_{i=1}^n (x_i^\top v_1)^2 = \|Xv_1\|^2 = v_1^\top X^\top X v_1$
- ▶ Gram matrix : $G = (n-1)^{-1} X^\top X$, a symmetric covariance matrix
- ▶ We rewrite the variance $\sum_{i=1}^n (x_i^\top v_1)^2 \propto v_1^\top G v_1$
- ▶ Optimization problem : the direction v_1 that maximizes the variance of the data is

$$v_1 = \arg \max_{v \in \mathbb{R}^p, \|v\|=1} \sum_{i=1}^n (x_i^\top v)^2 = \arg \max_{v \in \mathbb{R}^p, \|v\|=1} v^\top G v$$

Connection between PCA and variance, first step

By the method of Lagrange multipliers the solution of $\mathbf{v}_1 = \arg \max_{\mathbf{v} \in \mathbb{R}^p, \|\mathbf{v}\|=1} \mathbf{v}^\top G \mathbf{v}$ is

$$G\mathbf{v}_1 = \lambda_1 \mathbf{v}_1$$

- ▶ λ_1, \mathbf{v}_1 are the eigenvalue/vector
- ▶ λ_1 is also the variance
- ▶ \mathbf{v}_1 is the eigenvector associated to the largest eigenvalue

To summarize, we have found that if we wish to find a 1-dimensional subspace with which to approximate the data, we should choose \mathbf{v} to be the principal eigenvector of G .

Then, to represent $x^{(i)}$ in this basis, we need only compute the corresponding scalar :

$$\mathbf{v}_1^\top x^{(i)} \in \mathbb{R}.$$

Further components

In the following “iterations”, find \mathbf{v}_2 , a direction $\perp \mathbf{v}_1$ that maximizes the variance.

Let λ_i, \mathbf{v}_i the i -th largest eigenvalue and its associated eigenvector. Then $\mathbf{v}_i \perp \mathbf{v}_{i-1}$ for $i > 1$ (since G is symmetric p.s.d.) and maximizes the variance

If we wish to project our data into a k -dimensional subspace ($k < d$), we should choose $\mathbf{v}_1, \dots, \mathbf{v}_k$ to be the top k eigenvectors of G . The \mathbf{v}_i 's now form a new, orthogonal basis for the data.

Then, to represent $x^{(i)}$ in this basis, we need only compute the corresponding vector

$$\begin{bmatrix} \mathbf{v}_1^T x^{(i)} \\ \mathbf{v}_2^T x^{(i)} \\ \vdots \\ \mathbf{v}_k^T x^{(i)} \end{bmatrix} \in \mathbb{R}^k.$$

Lower dimensional representation of X

- ▶ The axes (of direction) $\mathbf{v}_1, \dots, \mathbf{v}_p \in \mathbb{R}^p$ are called **principal components**
- ▶ The new variables $\mathbf{c}_j = X\mathbf{v}_j, j = 1, \dots, p$ are called scores

New representation (order k) :

- ▶ The matrix XV_k (with $V_k = [\mathbf{v}_1, \dots, \mathbf{v}_k]$) is the matrix representing the data in the base of the first k eigenvectors

Reconstruction in the original space (debruiteur) :

- ▶ "Perfect" reconstruction for $\mathbf{x} \in \mathbb{R}^p$: $\mathbf{x} = \sum_{j=1}^p (\mathbf{x}^\top \mathbf{v}_j) \mathbf{v}_j$
- ▶ Reconstruction with loss of information : $\hat{\mathbf{x}} = \sum_{j=1}^k (\mathbf{x}^\top \mathbf{v}_j) \mathbf{v}_j$

PCA before OLS

Algorithme : PCA before OLS

Entrées : $X \in \mathbb{R}^{n \times p}$, itérations K

$V_k \leftarrow k$ -th eigenvectors assoc to the k largest eigenvalues

$Z = XV_k$ is the new (projected) dataset

OLS in Z

When does it work ?

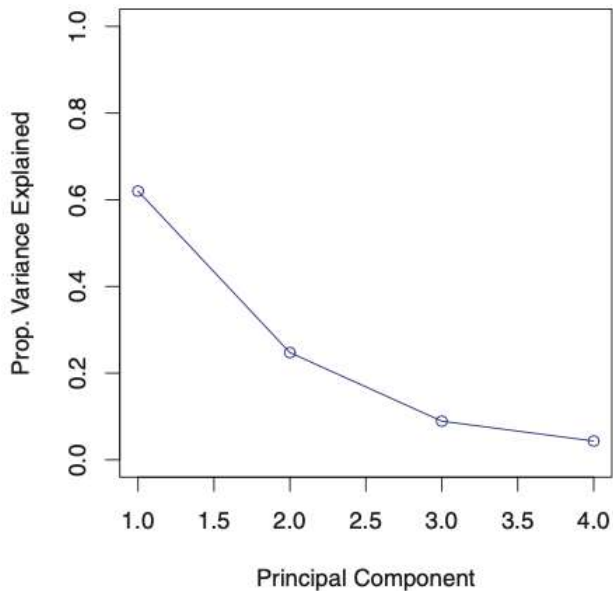
For practical reasons, we usually prefer to use the SVD of X than the eigen-decomposition of $X^T X$

Exercise: Show that the i -th singular value of X , σ_i , and the i -th eigenvalue of $X^T X$, λ_i , are related as follows $\lambda_i = (n - 1)^{-1} \sigma_i^2$

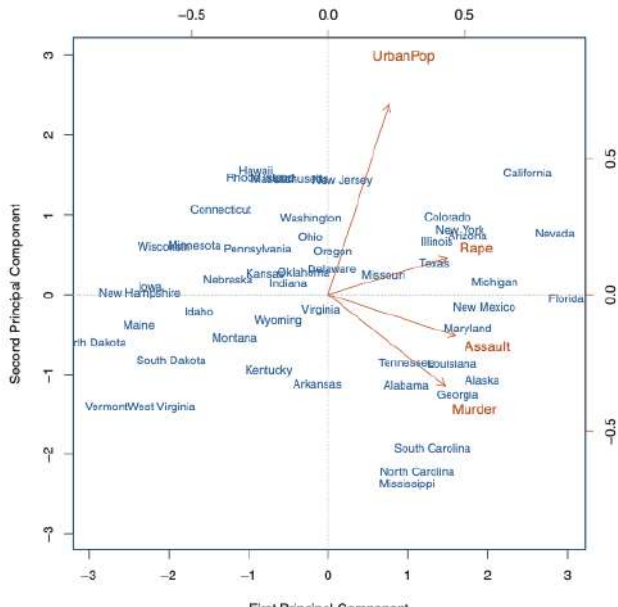
Understanding the projection/direction, dataset USArrests

		Murder	Assault	UrbanPop	Rape
0	Alabama	13.2	236	58	21.2
1	Alaska	10.0	263	48	44.5
2	Arizona	8.1	294	80	31.0
3	Arkansas	8.8	190	50	19.5
4	California	9.0	276	91	40.6
...					

Percentage of variance explained



Principal components



Conclusions

- ▶ PCA is an unsupervised technique
- ▶ Dimensionality reduction (more than a feature subset selection method)
- ▶ When the target y is correlated with the variance directions then its useful
- ▶ Interpretation of the proportion of variance explained
- ▶ Projection to low dimensions
- ▶ No interpretability on lower dimensions

Lecture notes on ordinary least squares¹

¹This document is a first version. Please let me know if you find typos or mistakes (irurozki@telecom-paris.fr). The author is grateful to Joseph Salmon (<http://josephsalmon.eu/>) for some help on the writing of this course and for sharing some materials.

Contents

1	Definition of ordinary least-squares and first properties	7
1.1	Definition	7
1.2	Existence and uniqueness	7
1.3	To centre the data or not to centre the data	9
1.4	The determination coefficient	9
2	Statistical model	13
2.1	The fixed-design model	13
2.1.1	Bias, variance and risk	13
2.1.2	Best linear unbiased estimator (BLUE)	14
2.1.3	Noise estimation	15
2.2	The Gaussian model	15
2.2.1	Estimating the Error Variance	16
2.2.2	A concentration inequality	17
2.3	The random design model	18
3	Confidence intervals and hypothesis testing	21
3.1	Confidence intervals	21
3.1.1	Gaussian model	21
3.1.2	Nongaussian case	22
3.2	Hypothesis testing	23
3.2.1	Definitions	23
3.2.2	Test of no effect	24
3.3	Forward variable selection	25
4	Ridge regularization	29
4.1	PCA before OLS	29
4.2	Definition of the Ridge estimator	30
4.3	Bias and variance	31
4.4	Choice of the regularization parameter	31
5	The LASSO	33
5.1	Definition	33
5.2	Theoretical properties	33
5.3	Computation	36
5.4	Extensions	37
A	Elementary results from linear algebra	39

B Singular value decomposition and principal component analysis	41
B.1 Matrix decomposition	41
B.2 Principal component analysis	42
C Concentration inequalities	43
D Optimization of convex functions	45

Notations

- $\langle \cdot, \cdot \rangle$ is the usual inner product in \mathbb{R}^d . $\|\cdot\|$ is the Euclidean norm. The elements forming the canonical basis of \mathbb{R}^d are denoted by e_0, \dots, e_{d-1} . Additionally, the ℓ_q -norm of $x \in \mathbb{R}^d$ is denoted by $\|x\|_q^q = \sum_{k=1}^d x_k^q$.
- If $A \in \mathbb{R}^{n \times d}$ is a matrix, $A^T \in \mathbb{R}^{d \times n}$ is the transpose matrix, $\ker(A) = \{u \in \mathbb{R}^d : Au = 0\}$.
- For any set of vectors (u_1, \dots, u_d) in \mathbb{R}^n , $\text{span}(u_1, \dots, u_d) = \{\sum_{k=1}^d \alpha_k u_k : (\alpha_1, \dots, \alpha_d) \in \mathbb{R}^d\}$. When A is a matrix $\text{span}(A)$ stands for the linear subspace generated by its columns.
- When A is a square invertible matrix, the inverse is denoted by A^{-1} . The Moore–Penrose inverse is denoted by A^+ . The trace of A is given by $\text{tr}(A)$.
- The identity matrix in $\mathbb{R}^{d \times d}$ is I_d . The vector $1_n \in \mathbb{R}^n$ contains n ones.
- For any sequence z_1, z_2, \dots , the empirical mean over the n first elements is denoted by $\bar{z}^n = \sum_{i=1}^n z_i / n$.
- When two random variables X and Y have the same distribution we write $X \sim Y$.
- When X_n is a sequence of random variables that converges in distribution (resp. in probability) to X , we write $X_n \rightsquigarrow X$ (resp. $X_n \xrightarrow{p} X$).

Chapter 1

Definition of ordinary least-squares and first properties

1.1 Definition

The general goal of regression analysis is to learn some relationship between a variable to predict $y \in \mathbb{R}$ and some covariates $x = (x_1, \dots, x_p)^T \in \mathbb{R}^p$, with $p \geq 1$. This is done by *learning a link function* that maps the input x to the output y . Linear regression is interested in modeling y using a linear link function of x , i.e., the variable y is modeled by $\theta_0 + \theta_1 x_1 + \dots + \theta_p x_p$ where $(\theta_0, \dots, \theta_p)$ are the parameters of the linear link function. To learn the values of these parameters, $(\theta_0, \dots, \theta_p)$, we observe $n \geq 1$ pairs (x_i, y_i) that are supposed to come from the same generating mechanism. In what follows we introduce the ordinary least squares (OLS) approach which basically consists in minimizing the sum of squares of the distance between the observed values y_i and the predicted values at x_i under the linear model.

We focus on a regression problem with $n \geq 1$ observations and $p \geq 1$ covariates. For notational convenience, for $i = 1, \dots, n$, we consider $y_i \in \mathbb{R}$ and $x_i = (x_{i,0}, \dots, x_{i,p})^T \in \mathbb{R}^{p+1}$ with $x_{i,0} = 1$. This is only to include the intercept in the same way as the other coefficients. The OLS estimator is any coefficient vector $\hat{\theta}_n = (\hat{\theta}_{n,0}, \dots, \hat{\theta}_{n,p})^T \in \mathbb{R}^{p+1}$ such that

$$\hat{\theta}_n \in \operatorname{argmin}_{\theta \in \mathbb{R}^{p+1}} \sum_{i=1}^n (y_i - x_i^T \theta)^2. \quad (1.1)$$

It is useful to introduce the notations

$$X = \begin{pmatrix} x_1^T \\ \vdots \\ x_n^T \end{pmatrix} = \begin{pmatrix} x_{1,0} & \dots & x_{1,p} \\ \vdots & & \vdots \\ x_{n,0} & \dots & x_{n,p} \end{pmatrix} \in \mathbb{R}^{n \times (p+1)}, \quad Y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}.$$

The matrix X which contains the covariates is called the *design matrix*. With the previous notation, (1.1) becomes

$$\hat{\theta}_n \in \operatorname{argmin}_{\theta \in \mathbb{R}^{p+1}} \|Y - X\theta\|^2,$$

where $\|\cdot\|$ stands for the Euclidean norm.

1.2 Existence and uniqueness

With the above formulation, the OLS has a nice geometric interpretation : $\hat{Y} = X\hat{\theta}_n$ is the closest point to Y in the linear subspace $\operatorname{span}(X) \subset \mathbb{R}^n$ (where $\operatorname{span}(A)$ stands for the linear subspace generated by the

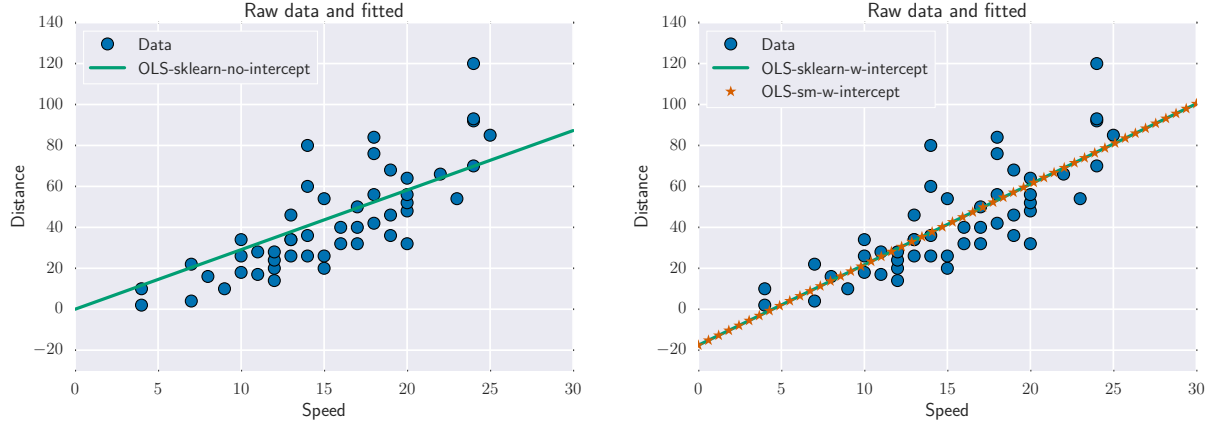


Figure 1.1: The dataset is the cars dataset from the R software. We use `sklearn` to compute OLS. The graph on the left represents the OLS line without intercept and the graph on the right is the OLS line computed with intercept.

columns of A). Using the Hilbert projection theorem (\mathbb{R}^n is a Hilbert space, $\text{span}(X)$ is a (closed) linear subspace of \mathbb{R}^n), \hat{Y} is unique and is characterized by the fact that the vector $Y - \hat{Y}$ is orthogonal to $\text{span}(X)$. This property is equivalent to the so-called normal equation:

$$X^\top(Y - \hat{Y}) = 0.$$

Since $\hat{Y} = X\hat{\theta}_n$, we obtain that the vector $\hat{\theta}_n$ must verify

$$X^\top X \hat{\theta}_n = X^\top Y. \quad (1.2)$$

Note that in contrast with \hat{Y} (which is always unique), the vector $\hat{\theta}_n$ is not uniquely defined without further assumptions on the data. For instance, take $u \in \ker(X)$ then $\hat{\theta}_n + u$ verifies (1.2) as well as $\hat{\theta}_n$. The uniqueness of the OLS is actually determined by the kernel of X which is related to the invertibility of the so called Gram matrix introduced below (see Exercise 1).

Definition 1. The matrix $\hat{G}_n = X^\top X/n$ is called the Gram matrix. Denote by $\hat{H}_{n,X} \in \mathbb{R}^{n \times n}$ the orthogonal projector on $\text{span}(X)$.

When the Gram matrix is invertible, the OLS is uniquely defined. When it is not the case, (1.1) has an infinite number of solutions.

Proposition 1. The OLS estimator always exists and the associated prediction is given by $\hat{Y} = \hat{H}_{n,X}Y$. It is either

- (i) uniquely defined. This happens if and only if the Gram matrix is invertible, which is equivalent to $\ker(X) = \ker(X^\top X) = \{0\}$. In this case, the OLS has the following expression:

$$\hat{\theta}_n = (X^\top X)^{-1} X^\top Y.$$

- (ii) or not unique, with an infinite number of solutions. This happens if and only if $\ker(X) \neq \{0\}$. In this case, the set of solutions writes $\hat{\theta}_n + \ker(X)$ where $\hat{\theta}_n$ is a particular solution.

¹Recall that P is the orthogonal projector on E , a subspace of \mathbb{R}^n , if and only if $P^2 = P$, $P^\top = P$ and $\ker(P) = E^\perp$.

Proof. The existence has already been shown using the Hilbert projection theorem. The linear system (1.2) has therefore a unique solution or an infinite number of solutions depending on whether the Gram matrix is invertible or not. Hence it remains to show that $\ker(X) = \ker(X^T X)$ which follows easily from the identity $\|Xu\|^2 = u^T X^T X u$. \square

When the OLS is not unique, the solution traditionally considered is

$$\hat{\theta}_n = (X^T X)^+ X^T Y,$$

where $(X^T X)^+$ denotes the Moore–Penrose inverse of $X^T X$, which always exists. For a demi-definite positive symmetric matrix with eigenvectors u_i and corresponding eigenvalues $\lambda_i \geq 0$, the Moore–Penrose inverse is given by $\sum_i \lambda_i^{-1} u_i u_i^T 1_{\{\lambda_i > 0\}}$.

Corollary 1. *The set of solution of OLS (1.1) is given by $\{(X^T X)^+ X^T Y + u : u \in \ker(X)\}$.*

Proof. Let $u \in \ker(X)$. Verify that $(X^T X)^+ X^T Y + u$ is a solution (see exercise 7). Then assuming that v is a solution, note that $v - (X^T X)^+ X^T Y$ belongs to $\ker(X)$. \square

1.3 To centre the data or not to centre the data

We now state the equivalence between this 2 procedures : doing OLS, with the intercept, on (Y, X) (as defined before) and doing OLS, without the intercept, on the centred variables. The later estimation procedure consists in the following. Let $X = (1_n, \tilde{X})$, $Y_c = Y - 1_n(1_n^T Y)/n$ and $\tilde{X}_c = \tilde{X} - 1_n(1_n^T \tilde{X})/n$. Hence the quantities Y_c and \tilde{X}_c are just centred version of Y and \tilde{X} , respectively. Define

$$\hat{\theta}_{n,c} = \operatorname{argmin}_{\theta \in \mathbb{R}^p} \|Y_c - \tilde{X}_c \theta\|.$$

Proposition 2. *It holds that*

$$\min_{\theta \in \mathbb{R}^p} \|Y_c - \tilde{X}_c \theta\| = \min_{\theta \in \mathbb{R}^{p+1}} \|Y - X \theta\|.$$

and, assuming that X has full rank, we have the following relationship between the traditional OLS and the OLS based on centred data,

$$(\hat{\theta}_{n,1}, \dots, \hat{\theta}_{n,p}) = \hat{\theta}_{n,c}^T.$$

Consequently, the 2 methods gives the same predictor.

Proof. See exercise 9. \square

1.4 The determination coefficient

To avoid trivial cases, we suppose in the following that $\sum_{i=1}^n (y_i - \bar{y}^n)^2 > 0$, i.e., that the sequence y_i is not constant. The determination coefficient, denoted by R^2 , is defined as the quotient between the explained sum of squares and the total sum of squares. It is given by

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y}^n)^2}{\sum_{i=1}^n (y_i - \bar{y}^n)^2} = \frac{\|\hat{Y} - \bar{y}^n 1_n\|^2}{\|Y - \bar{y}^n 1_n\|^2}.$$

Because of the orthogonality between $\hat{Y} - Y$ and \hat{Y} and between $\hat{Y} - Y$ and $\bar{y}^n 1_n$, we have that

$$R^2 = 1 - \frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{\sum_{i=1}^n (y_i - \bar{y}^n)^2} = 1 - \frac{\|\hat{Y} - Y\|^2}{\|Y - \bar{y}^n 1_n\|^2}. \quad (1.3)$$

The last expression involves a new quantity, called the residual sum of squares, which is small as soon as the OLS procedure went well, i.e., as soon as the predicted values are close to the observed values. Hence the closer to 1 the R^2 the better. The following statement justifies the use of the R^2 as a score supporting the quality of the OLS estimation :

- $R^2 = 1$ if and only if $Y = \hat{Y}$.
- $R^2 = 0$ if and only if $\hat{Y} = \hat{H}_{1_n} Y$ implying that $\hat{\theta}_n = (\bar{y}^n, 0, \dots, 0)$ is one OLS estimator.

Exercises

Exercise 1. Show that $\ker(X^T X) = \ker(X)$ and that $\text{span}(X^T) = \text{span}(X^T X)$ (for the latter, one might first note that $\ker(X) = \text{span}(X^T)^\perp$). Deduce that the normal equations always have at least one solution.

Exercise 2. Give $\hat{\theta}_n \in \mathbb{R}$ and $\hat{Y} \in \mathbb{R}^n$ in the case where $X = 1_n$ and $Y \in \mathbb{R}^n$.

Exercise 3. Show that any invertible transformation on the covariate, i.e. X is replaced by XA with A invertible, does not change the prediction \hat{Y} .

Exercise 4. Show that $\sum_{i=1}^n \hat{\epsilon}_i = 0$, where $\hat{\epsilon} = Y - \hat{Y} = (I - \hat{H}_{n,X})Y$.

Exercise 5. Aim is to express the uniqueness condition of the OLS in terms of the empirical covariance matrix $\hat{\Sigma}_n = n^{-1} \sum_{i=1}^n (x_i - \bar{x}^n)(x_i - \bar{x}^n)^T$.

(a) Show that $\ker(X) = \ker(X^T X)$.

(b) Prove that $X^T X = \sum_{i=1}^n x_i x_i^T$.

(c) Verify that $\ker(X) = \{0\}$ if and only if the empirical covariance matrix $\hat{\Sigma}_n$ is invertible (hint : one might work on the condition that $\hat{\Sigma}_n$ is non-invertible, i.e., there exists $u \in \mathbb{R}^d \setminus \{0\}$ such that $\hat{\Sigma}_n u = 0$).

Exercise 6. Aim is to obtain the formula $\hat{H}_{n,X} = X(X^T X)^+ X^T$.

(a) Verify that for any non-negative symmetric matrix $A \in \mathbb{R}^{p \times p}$, show that $A^+ A = A^+$.

(b) Show that $X(X^T X)^+ X^T$ is idempotent and symmetric (making it an orthogonal projector).

(c) Using that $X(X^T X)^+ X^T$ writes as UU^T for some matrix U that we shall specify, obtain that $\ker(\hat{H}_{n,X}) = \ker(X^T)$.

(d) Conclude showing that $\text{span}(\hat{H}_{n,X}) = \text{span}(X)$.

Exercise 7. Show that $\hat{\theta}_n = (X^T X)^+ X^T Y$ is a solution of the OLS problem.

Exercise 8. Show [\(1.3\)](#).

Exercise 9. Aim is to prove [Proposition 2](#).

(a) Start by obtaining that the inequality \geq holds true.

(b) Then show that for any sequence (z_i) , and for all $z \in \mathbb{R}$, it holds that $\|Z - z1_n\| \geq \|Z - \bar{z}^n 1_n\|$, where $Z = (z_1, \dots, z_n)$ and $\bar{z}^n = n^{-1} \sum_{i=1}^n z_i$.

(c) Find \hat{a}_n such that, for any $\theta_0 \in \mathbb{R}$ and $\tilde{\theta} \in \mathbb{R}^p$, $\|Y - \theta_0 1_n - \tilde{X} \tilde{\theta}\| \geq \|Y - \hat{a}_n(\tilde{\theta}) 1_n - \tilde{X} \tilde{\theta}\|$ where $\tilde{X} \in \mathbb{R}^{n \times p}$ is the same as X without the first column.

(d) Conclude that $\min_{\theta \in \mathbb{R}^p} \|Y_c - \tilde{X}_c \theta\| = \min_{\theta \in \mathbb{R}^p, \theta_0 \in \mathbb{R}} \|Y - X(\theta_0, \theta)^T\|$

(e) Use the Lebesgue projection theorem to conclude that whenever $\ker(X) = \{0\}$, $(\hat{\theta}_{n,1}, \dots, \hat{\theta}_{n,p}) = \hat{\theta}_{n,c}^T$.

Exercise 10 (on-line ols and cross-validation). *The goal of this exercise is to show that the OLS estimator $\hat{\theta}_n$ associated with design matrix $X_{(n)} \in \mathbb{R}^{n \times (p+1)}$ and output $\mathbf{y}_{(n)} \in \mathbb{R}^n$ can be easily updated when a new pair of observation $(\mathbf{x}_{n+1}^T, y_{n+1}) \in \mathbb{R}^{(p+1)} \times \mathbb{R}$ is given. We apply the result to cross validation procedure in the end.*

To clarify the notation:

$$X_{(n+1)} = \begin{pmatrix} X_{(n)} \\ \mathbf{x}_{n+1}^T \end{pmatrix} \in \mathbb{R}^{(n+1) \times (p+1)}, \quad \text{and} \quad \mathbf{y}_{(n+1)} = \begin{pmatrix} \mathbf{y}_{(n)} \\ y_{n+1} \end{pmatrix} \in \mathbb{R}^{n+1}$$

We assume from now on that $X_{(n)}$ and $X_{(n+1)}$ are full column rank (i.e., the columns of each matrix are independent vectors).

NB : Some of the questions require some computation (in particular obtaining (1.4) and (1.6)). Even if you could not prove it, it can be use later.

(a) Let A, B, C, D be matrices with respective sizes (d, d) , (d, k) , (k, k) , (k, d) . Show that if A and C are invertible, then

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1}. \quad (1.4)$$

(b) Obtain that

$$(X_{(n+1)}^T X_{(n+1)})^{-1} = (X_{(n)}^T X_{(n)})^{-1} - \frac{\zeta_{n+1} \zeta_{n+1}^T}{1 + b_{n+1}} \quad (1.5)$$

where $\zeta_{n+1} = (X_{(n)}^T X_{(n)})^{-1} \mathbf{x}_{n+1}$ and $b_{n+1} = \mathbf{x}_{n+1}^T (X_{(n)}^T X_{(n)})^{-1} \mathbf{x}_{n+1}$.

(c) Express $X_{(n+1)}^T \mathbf{y}_{(n+1)}$ with respect to $X_{(n)}^T \mathbf{y}_{(n)}$ and $y_{n+1} \mathbf{x}_{n+1}$.

(d) Show that the OLS estimator $\hat{\theta}_{n+1}$ associated with design matrix $X_{(n+1)}$ and output $\mathbf{y}_{(n+1)}$ can be obtained as follows:

$$\hat{\theta}_{n+1} = \hat{\theta}_n + \frac{u_{n+1}}{1 + b_{n+1}} \zeta_{n+1}, \quad (1.6)$$

where $u_{n+1} = y_{n+1} - \mathbf{x}_{n+1}^T \hat{\theta}_n$.

(e) Keeping in memory $(X_{(n)}^T X_{(n)})^{-1}$ and $\hat{\theta}_n$, explain how to update $\hat{\theta}_{n+1}$ using a minimal number of operations of the kind : matrix $(p+1, p+1)$ times vector $(p+1, 1)$. How many such operation are needed?

(f) Using Equation (1.5) above, show that

$$1 + b_{n+1} = \frac{1}{1 - h_{n+1}}$$

where $h_{n+1} = \mathbf{x}_{n+1}^T (X_{(n+1)}^T X_{(n+1)})^{-1} \mathbf{x}_{n+1}$.

(g) The prediction of y_{n+1} given by the model is $\hat{y}_{n+1} := \mathbf{x}_{n+1}^T \hat{\theta}_{n+1}$. With the following formula

$$\hat{y}_{n+1} = \mathbf{x}_{n+1}^T \hat{\theta}_n + \frac{u_{n+1} b_{n+1}}{1 + b_{n+1}}.$$

prove that

$$y_{n+1} - \hat{y}_{n+1} = u_{n+1}(1 - h_{n+1}).$$

(h) Given some data (\mathbf{y}, X) , leave-one-out cross-validation consists in computing the risk

$$R_{cv} = \frac{1}{n} \sum_{i=1}^n (y_i - \mathbf{x}_i^T \hat{\boldsymbol{\theta}}_{(-i)})^2$$

where $\hat{\boldsymbol{\theta}}_{(-i)}$ is the OLS estimator based on $(\mathbf{y}_{(-i)}, X_{(-i)})$, i.e., the data (\mathbf{y}, X) without the i -th line. Applying what have been done so far, show that

$$R_{cv} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 / (1 - \hat{h}_i)^2,$$

with $\hat{h}_i = \mathbf{x}_i^T (X^T X)^{-1} \mathbf{x}_i$ and $\hat{y}_i = \mathbf{x}_i^T \hat{\boldsymbol{\theta}}_n$, $\hat{\boldsymbol{\theta}}_n$ being the OLS estimator of (\mathbf{y}, X) .

Chapter 2

Statistical model

In the previous section, we have defined the OLS estimator based on the observed data without any assumption on the generating process associated to the data. When assuming that the observations are independent realizations of some random variables, we can rely on probability theory to further study the behaviour of the OLS. In the following we describe different probabilistic models : fixed design model, random design model and the Gaussian noise model.

2.1 The fixed-design model

The fixed design model takes the form:

$$Y_i = x_i^T \theta^* + \epsilon_i, \quad \text{for all } i = 1, \dots, n,$$

where (x_i) is a sequence of deterministic points in \mathbb{R}^{p+1} and (ϵ_i) is a sequence of random variables in \mathbb{R} such that

$$\mathbb{E}[\epsilon] = 0, \quad \text{var}(\epsilon) = \sigma^2 I_n, \quad \text{with } \epsilon = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix}.$$

For instance, (ϵ_i) can be an identically distributed and independent sequence of centred random variables with variance σ^2 . The level of noise σ of course reflects the difficulty of the problem.

The fixed-design model is appropriate when the sequence (x_i) is chosen by the analyst, e.g., in a physics laboratory experiment, one can fix some variables such as the temperature, or in a clinical survey one can give to patients a determined quantity of some serum. In contrast, the random design (see Section 2.3) model is appropriate when the covariates are unpredictable as for instance the wind speed observed in the nature or the age of some individuals in a survey.

Based on this model, we can derive some statistical properties that we present in the following. These properties are concerned with different types of error related to the estimation of θ^* by $\hat{\theta}_n$ and will be obtained under the assumption that the dimension of $\text{span}(X)$ equals $p+1$, implying that $\ker(X) = \{0\}$ and that $\hat{\theta}_n$ is unique. We therefore implicitly assume that $n \geq p+1$. We can now state a useful decomposition: provided that $\ker(X) = \{0\}$, it holds that

$$\hat{\theta}_n - \theta^* = (X^T X)^{-1} X^T \epsilon. \tag{2.1}$$

2.1.1 Bias, variance and risk

The bias, the variance and the risk are important quantities because they are measures of the estimation quality. For instance, an estimator is accurate when the bias is 0 and the variance is small. The following notion of bias is related to the whole statistical model (for all θ^* , not for a particular one).

Definition 2. An estimator $\theta(X, Y)$ is said to be unbiased if for all (X, ϵ, θ^*) used to generate Y according to the model, it holds that $\mathbb{E}[\theta(X, Y)] = \theta^*$.

The risk measures the average error associated to an estimation procedure. Different notions of risk can be defined: the quadratic risk is defined on the regression coefficients β , the prediction risk takes care of the prediction error, i.e., the error when predicting y . Formal definitions are given below.

Definition 3. The quadratic risk associated to $\hat{\theta}_n$ estimating θ^* is

$$R_{quad}(\hat{\theta}_n, \theta^*) = \mathbb{E}[\|\hat{\theta}_n - \theta^*\|^2].$$

The prediction risk is

$$R_{pred}(\hat{\theta}_n, \theta^*) = \mathbb{E}[\|Y^* - \hat{Y}\|^2]/n,$$

where Y^* is the prediction we would make if we knew the true regression vector, i.e., $Y^* = X\theta^*$.

Proposition 3. When $\ker(X) = \{0\}$, the following holds:

- (i) the OLS estimator is unbiased i.e., it holds that $\mathbb{E}[\hat{\theta}_n] = \theta^*$.
- (ii) Its variance is given by $\text{var}(\hat{\theta}_n) = (X^T X)^{-1} \sigma^2$.
- (iii) $R_{pred}(\hat{\theta}_n, \theta^*) = (p+1)\sigma^2/n$.
- (iv) $R_{quad}(\hat{\theta}_n, \theta^*) = \text{tr}((X^T X)^{-1})\sigma^2$.

Hence whenever the smallest eigenvalue of \hat{G}_n is larger than b (independently of n), the quadratic risk of the OLS decreases with the rate $1/n$, which is the classical estimation rate in statistics, e.g., empirical average estimating the expectation.

2.1.2 Best linear unbiased estimator (BLUE)

This section is dedicated to the so called Gauss-Markov theorem which asserts that the OLS is BLUE.

We introduce the following partial order (reflexivity, anti-symmetry and transitivity) on the set of symmetric matrices. Let $V_1 \in \mathbb{R}^{d \times d}$ and $V_2 \in \mathbb{R}^{d \times d}$ be two symmetric matrices. We write $V_1 \leq V_2$ whenever $u^T V_1 u \leq u^T V_2 u$ for every $u \in \mathbb{R}^d$. This partial order is particularly useful to compare the covariance matrices of estimators. Indeed if $\hat{\beta}_1$ and $\hat{\beta}_2$ are estimators with respective covariance V_1 and V_2 . Then, $V_1 \leq V_2$ if and only if any linear combination of $\hat{\beta}_1$ has a smaller variance than the same linear combination of $\hat{\beta}_2$.

Definition 4. An estimator is said to be linear if, for any dataset (Y, X) , it writes as AY , where $A \in \mathbb{R}^{(p+1) \times n}$ depends only on X .

Proposition 4 (Gauss-Markov). Under the fixed design model, among all the unbiased linear estimators AY , $\hat{\theta}_n$ is the one with minimal variance, i.e.,

$$\text{cov}(\hat{\theta}_n) \leq \text{cov}(AY),$$

with equality if and only if $A = (X^T X)^{-1} X^T$.

Proof. First note that AY is unbiased if and only if $(A - (X^T X)^{-1} X^T) X \theta^* = 0$ for all θ^* , equivalently, $BX = 0$ with $B = (A - (X^T X)^{-1} X^T)$. Consequently, using that $E[\epsilon \epsilon^T] = \sigma^2 I_n$, $\text{cov}(BY, \hat{\theta}_n) = 0$. Then, just write

$$\begin{aligned} \text{cov}(AY) &= \text{cov}(BY + \hat{\theta}_n) \\ &= \text{cov}(BY) + \text{cov}(\hat{\theta}_n) \\ &= \sigma^2 B B^T + \text{cov}(\hat{\theta}_n) \geq \text{cov}(\hat{\theta}_n). \end{aligned}$$

The previous inequality is an equality if and only if $B = 0$. □

2.1.3 Noise estimation

Providing only an estimate $\hat{\theta}_n$ of θ^* is often not enough as it does not give any clue on the accuracy of the estimation. When possible, one should also furnish an estimation of the error σ^2 . If one knew the residuals (ϵ_i) , one would take the empirical variance of $\epsilon_1, \dots, \epsilon_n$, but this is not possible. Alternatively, one can take

$$\tilde{\sigma}_n^2 = n^{-1} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2.$$

Because of the first normal equations expressed in (1.2), we have $\sum_{i=1}^n (Y_i - \hat{Y}_i) = 0$. Consequently, $\tilde{\sigma}_n^2$ is simply the empirical variance estimate of the residual vector $Y_i - \hat{Y}_i$. Noting that $\tilde{\sigma}_n^2 = n^{-1} \|(I_n - \hat{H}_{n,X})\epsilon\|^2$ one can compute the expectation:

$$\mathbb{E}[\tilde{\sigma}_n^2] = \sigma^2(n - p - 1)/n.$$

The unbiased version (which should be used in practice) is then

$$\hat{\sigma}_n^2 = \tilde{\sigma}_n^2 \left(\frac{n}{n - p - 1} \right),$$

where from now on we assume that $n > p + 1$. In the case when $n = p + 1$ and X has rank $p + 1$, we obtain that $Y_i = \hat{Y}_i$ for all $i = 1, \dots, n$.

2.2 The Gaussian model

Here we introduce the Gaussian model as a submodel of the fixed design model where the distribution of the noise sequence (ϵ_i) is supposed to be Gaussian with mean 0 and variance σ^2 . The Gaussian model can then be formulated as follows:

$$Y_i \stackrel{i.i.d.}{\sim} \mathcal{N}(x_i^T \theta^*, \sigma^2), \quad \text{for all } i = 1, \dots, n,$$

where (x_i) is non-random sequence of vector in \mathbb{R}^{p+1} . We keep assuming that $\ker(X) = \{0\}$ in the following.

The Student's t-distribution with p degrees of freedom is defined as the distribution of the random variable $X/\sqrt{Z/p}$, where X (resp. Z) has standard normal distribution (resp. chi-square distribution with p degrees of freedom).

Proposition 5. *Under the Gaussian model, if $\ker(X) = \{0\}$ and $n > p + 1$, it holds that*

- $\hat{\theta}_n$ and $\hat{\sigma}_n^2$ are independent,
- $n^{1/2}(\hat{\theta}_n - \theta^*) \sim \mathcal{N}(0, n\sigma^2(X^T X)^{-1})$,
- $(n - p - 1)(\hat{\sigma}_n^2/\sigma^2) \sim \chi_{n-p-1}^2$,
- if $\hat{s}_{n,k}^2$ is the k -th term in the diagonal of \hat{G}_n^{-1} , then

$$(n^{1/2}/\hat{s}_{n,k}\hat{\sigma}_n)(\hat{\theta}_{n,k} - \theta_k^*) \sim \mathcal{T}_{n-p-1},$$

where \mathcal{T}_{n-p-1} is the Student's t-distribution with $n - p - 1$ degrees of freedom.

Proof. For the first point, remark that $X^T \epsilon$ and $(I - \hat{H}_{n,X})\epsilon$ are two independent Gaussian vectors:

$$\text{cov}(X^T \epsilon, (I - \hat{H}_{n,X})\epsilon) = \mathbb{E}[X^T \epsilon \epsilon^T (I - \hat{H}_{n,X})] = 0.$$

Then writing

$$(n-p-1)\hat{\sigma}^2 = \|Y - \hat{Y}\|^2 = \|(I - \hat{H}_{n,X})Y\|^2 = \|(I - \hat{H}_{n,X})\epsilon\|^2$$

$$\hat{\theta}_n - \theta^* = (X^T X)^{-1} X^T \epsilon,$$

we see that $\hat{\theta}_n$ and $\hat{\sigma}^2$ are measurable transformations of two independent Gaussian vector. They then are independent. We can use for instance the following characterisation of independence, say for random variables ξ_1 and ξ_2 : for any f_1 and f_2 positive and measurable, $\mathbb{E}[f_1(\xi_1)f_2(\xi_2)] = \mathbb{E}[f_1(\xi_1)]\mathbb{E}[f_2(\xi_2)]$.

For the second point, as ϵ is Gaussian, one just has to compute the variance.

For the third point, let $V \in \mathbb{R}^{n \times n}$ be an orthogonal matrix such that $V = (V_1, V_2)$ where V_1 is a basis of $\text{span}(X)$, and note that $V_1^T(I - \hat{H}_{n,X}) = 0$ and $V_2^T(I - \hat{H}_{n,X}) = V_2^T$. As the norm is invariant by orthogonal transformation, one has

$$(n-p-1)\hat{\sigma}^2 = \|(I - \hat{H}_{n,X})\epsilon\|^2 = \|V^T(I - \hat{H}_{n,X})\epsilon\|^2 = \|V_2^T\epsilon\|^2.$$

Consequently,

$$(n-p-1)(\hat{\sigma}^2/\sigma^2) = \sum_{i=1}^{n-p-1} \tilde{\epsilon}_i^2,$$

with $\tilde{\epsilon} = V_2^T\epsilon/\sigma$. It remains to show that $\tilde{\epsilon}$ is a Gaussian vector with covariance I_{n-p-1} .

For the fourth point, use the second point to obtain that

$$(n^{1/2}/\hat{s}_{n,k}\sigma)(\hat{\theta}_{n,k} - \theta_k^*) \sim \mathcal{N}(0, 1).$$

Then $(n^{1/2}/\hat{s}_{n,k}\hat{\sigma}_n)(\hat{\theta}_{n,k} - \theta_k^*)$ writes as the quotient of two independent random variables: a Gaussian and the square root of a chi-square. This is a Student's t-distribution with $n-p-1$ degrees of freedom. \square

A direct application of the previous proposition gives us the following equality, which is informative on the estimation error, for any $k = 0, \dots, p$,

$$\mathbb{P}(|\hat{\theta}_{n,k} - \theta_k^*| \geq t) = 2S_{T_{n-p-1}}(tn^{1/2}/\hat{s}_{n,k}\hat{\sigma}_n),$$

where $S_{T_{n-p-1}}$ is the survival function of the distribution T_{n-p-1} .

2.2.1 Estimating the Error Variance

We are now ready to establish the result for the residual variance. Recall that we define the residuals as $\hat{\epsilon}_i = Y_i - \hat{Y}_i$

Theorem 1. *The unbiased estimator for the variance of the residuals , i.e., $\mathbb{E}[\hat{\sigma}^2] = \sigma^2$, is given by*

$$\hat{\sigma}^2 := \frac{1}{n-p-1} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2,$$

Proof. We first note that

$$\begin{aligned} (n-p-1)\hat{\sigma}^2 &= (Y - \hat{Y})^\top (Y - \hat{Y}) \\ &= (Y - HY)^\top (Y - HY) \\ &= Y^\top (I - H)^\top (I - H)Y \\ &= Y^\top (I - H)Y \\ &= (X\theta^* + \varepsilon)^\top (I - H)(X\theta^* + \varepsilon) \\ &= \theta^{*\top} X^\top (I - H)X\theta^* + 2\varepsilon^\top (I - H)X\theta^* + \varepsilon^\top (I - H)\varepsilon \quad (\text{Lemma ??}) \\ &= \varepsilon^\top (I - H)\varepsilon. \end{aligned} \tag{2.2}$$

Now we can apply Cochran's lemma. This shows that

$$\frac{1}{\sigma^2}(n-p-1)\hat{\sigma}^2 = \frac{1}{\sigma^2}\varepsilon^\top(I-H)\varepsilon \sim \chi_{n-p-1}^2.$$

Since the expectation of a χ_k^2 distribution equals k , we find

$$\frac{1}{\sigma^2}(n-p-1)\mathbb{E}(\hat{\sigma}^2) = n-p-1$$

and thus

$$\mathbb{E}(\hat{\sigma}^2) = \sigma^2.$$

□

There are two remarkable corollaries to these results.

Proposition 6. *The random vector of coefficients $\hat{\boldsymbol{\theta}}$ and the random number $\hat{\sigma}^2$ are independent of each other.*

Proposition 7. *The quantity*

$$T := \frac{\hat{\theta}_i - \theta_i}{\sqrt{\hat{\sigma}^2(X^T X)^{-1}_{ii}}}$$

follows a T-student distribution of $n-p-1$ degrees of freedom (provided X is full rank).

2.2.2 A concentration inequality

We now provide an additional guarantee for the OLS estimator under the Gaussian model. It consists of a concentration inequality : an upper bound on the probability that the estimation error exceeds any given $t > 0$. The upper bound unsurprisingly depends on p , n , t , and the smallest eigenvalue of \hat{G}_n .

Proposition 8. *Suppose that the Gaussian model is valid. Denote by $\hat{\lambda}_n$ the smallest eigenvalue of \hat{G}_n and suppose that $\hat{\lambda}_n > 0$ for all $n \geq 1$. Then, for any $k \in \{0, \dots, p\}$, $n \geq 1$ and $\delta > 0$, it holds with probability $1 - \delta$,*

$$\left| \hat{\boldsymbol{\theta}}_{n,k} - \boldsymbol{\theta}_k^* \right| \leq \sqrt{\frac{2\sigma^2 \hat{s}_{n,k}^2 \log(2/\delta)}{n}}.$$

where $\hat{s}_{n,k}^2 = e_k^T \hat{G}_n^{-1} e_k$. Moreover,

$$\max_{k=0,\dots,p} \left| \hat{\boldsymbol{\theta}}_{n,k} - \boldsymbol{\theta}_k^* \right| \leq \sqrt{\frac{2\sigma^2 \log(2(p+1)/\delta)}{n\hat{\lambda}_n}}.$$

Proof. Let $\tilde{X}_i = (X^T X)^{-1} X_i$. Apply Lemma 4 of Appendix C to the sequence $\sum_{i=1}^n (u^T \tilde{X}_i) \epsilon_i$ to obtain that

$$\mathbb{P} \left(\left| \sum_{i=1}^n (\tilde{X}_i^T u) \epsilon_i \right| > t \right) \leq 2 \exp \left(-t^2 / (2\sigma^2 \sum_{i=1}^n (u^T \tilde{X}_i)^2) \right),$$

Choosing $u = e_k$, we have $\sum_{i=1}^n (u^T \tilde{X}_i)^2 = n^{-1} \hat{s}_{n,k}^2$, and using (2.1), we obtain that

$$\mathbb{P} \left(\left| \hat{\boldsymbol{\theta}}_{n,k} - \boldsymbol{\theta}_k^* \right| > t \right) \leq 2 \exp(-t^2 n / (2\sigma^2 \hat{s}_{n,k}^2)).$$

Choose t appropriately to obtain the first inequality. The second inequality follows from $\hat{s}_{n,k}^2 \leq \hat{\lambda}_n^{-1} = \max_{\|u\|=1} |u^T \hat{G}_n^{-1} u|$ and the union bound:

$$\begin{aligned} \mathbb{P} \left(\max_{k=0,\dots,p} |\hat{\theta}_{n,k} - \theta_k^*| > t \right) &= \mathbb{P} \left(\bigcup_{k=0,\dots,p} \{ |\hat{\theta}_{n,k} - \theta_k^*| > t \} \right) \\ &\leq \sum_{k=0,\dots,p} \mathbb{P} \left(|\hat{\theta}_{n,k} - \theta_k^*| > t \right). \end{aligned}$$

□

Remark 1. The first inequality of Proposition 8 is important as it shows that each coordinate might not behave similarly depending on the associated diagonal element of \hat{G}_n . For instance, for the intercept, the bound just becomes $\sqrt{2\sigma^2 \log(2/\delta)/n}$. The quantity $\hat{s}_{n,k}$ will play an important role in practice when building confidence intervals (see section 3).

Remark 2. Proposition 8 suggests that the value of the smallest eigenvalue $\hat{\lambda}_n$ of \hat{G}_n plays a certain role on the accuracy of the estimation. The smaller $\hat{\lambda}_n$ the worst the estimation accuracy.

2.3 The random design model

In the random design model, we suppose that (Y_i, X_i) is a sequence of independent and identically distributed random vectors defined on the probability space $(\Omega, \mathcal{A}, \mathbb{P})$. The aim is to estimate the best linear approximation of Y_1 made up with X_1 in terms of L_2 -risk, i.e., to find θ that minimizes $\mathbb{E}[(Y_1 - X_1^T \theta)^2]$. Such a minimizer can be characterized with the help of the normal equation. Recall that $X_1 \in \mathbb{R}^{p+1}$ and $X_{1,0} = 1$ almost surely.

Proposition 9. Suppose that for all $k = 0, \dots, p$, $\mathbb{E}[X_{1,k}^2] < \infty$ and $\mathbb{E}[Y_1^2] < \infty$, then

$$\inf_{\theta} \mathbb{E}[(Y_1 - X_1^T \theta)^2] = \mathbb{E}[(Y_1 - X_1^T \theta^*)^2],$$

if and only if

$$\mathbb{E}[X_1 X_1^T] \theta^* = \mathbb{E}[X_1 Y_1].$$

Proof. Note that the minimization problem of interest is equivalent to

$$\inf_{Z_1 \in \mathcal{F}} \mathbb{E}[(Y_1 - Z_1)^2],$$

where \mathcal{F} is the linear subspace of the Hilbert space $L_2(\Omega, \mathcal{A}, \mathbb{P})$ generated by $X_{1,0}, \dots, X_{1,p}$. As \mathcal{F} is a closed linear subspace (because it has a finite dimension), the minimizer is unique and characterized by the normal equations. □

The previous proposition does not imply that θ^* is unique. In fact we are facing a similar situation as in Proposition 1: either θ^* is unique, which is equivalent to $\mathbb{E}[X_1 X_1^T]$ is invertible, or θ^* is not uniquely defined. Note that θ^* is not unique whenever one variable is a combination of the others. In this case one might consider any of the solution, e.g., $\theta^* = \mathbb{E}[X_1 X_1^T]^+ \mathbb{E}[X_1 Y_1]$. Some asymptotic properties are available. They will be useful to run some statistical tests. We consider the following definition, valid for any $n \geq 1$,

$$\hat{\theta}_n = (X^T X)^+ X^T Y.$$

Proposition 10. Suppose that $\mathbb{E}[X_1 X_1^T]$ and $\mathbb{E}[Y_1^2]$ exist and that $\mathbb{E}[X_1 X_1^T]$ is invertible. Then

$$n^{1/2}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^*) \rightsquigarrow \mathcal{N}(0, \sigma^2 G^{-1}),$$

where $\sigma^2 = \text{var}(Y_1 - X_1^T \boldsymbol{\theta}^*)$ and $G = \mathbb{E}[X_1 X_1^T]$. Moreover

$$\hat{\sigma}_n^2 \rightarrow \sigma^2, \text{ in probability.}$$

In particular, $(n^{1/2}/\hat{\sigma}_{n,k}\hat{\sigma}_n)(\hat{\boldsymbol{\theta}}_{n,k} - \boldsymbol{\theta}_k^*) \rightsquigarrow \mathcal{N}(0, 1)$.

Proof. Note that

$$n^{1/2}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^*) = n^{1/2}(X^T X)^+ X^T \epsilon + n^{1/2}((X^T X)^+ (X^T X) - I_{p+1}) \boldsymbol{\theta}^*.$$

It suffices to show that the term in the right converges to 0 in probability and that the term in the left converges in distribution to the stated limit. The first point is a consequence of the continuity of the determinant. The second point is a consequence of Slutsky's theorem using the fact that the Moore-Penrose inverse is a continuous operation. For more details, see Exercise [11](#).

The convergence of $\hat{\sigma}_n^2$ is obtained by the decomposition

$$\begin{aligned} \hat{\sigma}_n^2 &= (n - p + 1)^{-1} \|(I - \hat{H}_{n,X})\epsilon\|_2^2 \\ &= (n - p + 1)^{-1} (\|\epsilon\|^2 - \epsilon^T X (X^T X)^+ X^T \epsilon). \end{aligned}$$

Invoking the law of large number, we only need to show that the term on the right goes to 0 in probability. We have

$$\epsilon^T X (X^T X)^+ X^T \epsilon = \left(n^{-1/2} \sum_{i=1}^n X_i \epsilon_i \right)^T \hat{G}_n^+ \left(n^{-1/2} \sum_{i=1}^n X_i \epsilon_i \right)$$

Because $\hat{G}_n^+ \rightarrow G^{-1}$ and $n^{-1/2} \sum_{i=1}^n X_i \epsilon_i \rightsquigarrow \mathcal{N}(0, G)$, we get that

$$\epsilon^T X (X^T X)^+ X^T \epsilon \rightsquigarrow \|\mathcal{N}(0, \sigma^2 I_{p+1})\|^2 = \sigma^2 \chi_{p+1}^2.$$

When divided by $(n - p + 1)$ the previous term goes to 0. \square

Remark 3. A more general regression problem can be formulated without specifying a linear link : the regression function f^* is any measurable function that minimizes the risk

$$R(f) = \mathbb{E}[(Y_1 - f(X_1))^2].$$

When $\mathbb{E}[Y_1^2] < \infty$, the minimizer is unique and coincides, in $L^2(\Omega, \mathcal{A}, \mathbb{P})$, with the conditional expectation of Y given X_1 : $f^*(X_1) = \mathbb{E}[Y_1 | X_1]$, almost surely.

Exercises

Exercise 11 (Asymptotics for the OLS in Random design). Let $(X_1, Y_1), (X_2, Y_2), \dots$ be an i.i.d. sequence of random vectors. Each pair (X_i, Y_i) is valued in $\mathbb{R}^p \times \mathbb{R}$. Denote by $X_i^T = (X_i^{(1)}, \dots, X_i^{(p)})$. Suppose that for all $(k, l) \in \{1, \dots, p\}^2$, $\mathbb{E}[|X_1^{(k)} X_1^{(l)}|] < \infty$ and $G = \mathbb{E}[X_1 X_1^T]$ is invertible. The goal is to show that

$$n^{1/2}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^*) \rightsquigarrow \mathcal{N}(0, \sigma^2 G^{-1}),$$

where $\boldsymbol{\theta}^*$ is defined in Proposition [9](#). Recall that for each $n \in \mathbb{N}_+^*$, the OLS is given by

$$\hat{\boldsymbol{\theta}}_n = \hat{G}_n^+ \left(n^{-1} \sum_{i=1}^n X_i Y_i \right), \quad (2.3)$$

with $\hat{G}_n = n^{-1} \sum_{i=1}^n X_i X_i^T$.

1. Let $\mathcal{S}_p(\mathbb{R})$ be the space of symmetric matrices with real coefficients. Let $T : \mathcal{S}_p(\mathbb{R}) \rightarrow \mathcal{S}_p(\mathbb{R})$ be such that $T(A) = A^+$. Show the continuity of T at each point A such that $\det(A) \neq 0$. We recall that for any A such that $\det(A) \neq 0$, $A^{-1} = (\det(A))^{-1} \text{Com}(A)^T$ where $\text{Com} : \mathcal{S}_p(\mathbb{R}) \rightarrow \mathcal{S}_p(\mathbb{R})$ is continuous (it is called the comatrix).
2. What is the limit of \hat{G}_n^+ ? In which sense?
3. Show that $\hat{\theta}_n - \theta^* = \hat{G}_n^+ \hat{\mu}_n + (\hat{G}_n^+ \hat{G}_n - I_p) \theta^*$ with $\hat{\mu}_n = n^{-1} \sum_{i=1}^n X_i(Y_i - X_i^T \theta^*)$.
4. Obtain that $\sqrt{n} \hat{G}_n^+ \hat{\mu}_n \xrightarrow{\mathcal{L}} \mathcal{N}(0, \Sigma)$ where Σ needs to be determined.
5. Prove that $\sqrt{n}(\hat{G}_n^+ \hat{G}_n - I) \beta_0 \xrightarrow{\mathbb{P}} 0$. One can consider the event $\det(G_n) \neq 0$.
6. Show that $\sqrt{n}(\hat{\theta}_n - \theta^*) \xrightarrow{\mathcal{L}} \mathcal{N}(0, \Sigma)$.
7. Let $k \in \{1, \dots, p\}$, find $\hat{s}_{n,k}$, depending only on σ^2 and \hat{G}_n , such that $\sqrt{n}(\frac{\hat{\theta}_{n,k} - \theta_k^*}{\hat{s}_{n,k}}) \xrightarrow{\mathcal{L}} \mathcal{N}(0, 1)$.
8. Deduce a $(1 - \alpha)$ -confidence interval for θ_k^* . Verify it has level $1 - \alpha$.

Chapter 3

Confidence intervals and hypothesis testing

3.1 Confidence intervals

From a practical perspective, building confidence intervals is often an inevitable step as it permits to evaluate the quality of the estimation. The construction of confidence intervals follows the estimation step. Intuitively, a confidence interval is simply a region (based on the observed data) in which the parameter of interest is most likely to lie. The accuracy/quality of the estimation is then naturally measured by the size of the underlying confidence interval. As we shall see, the construction of a confidence interval is based on the estimation of the variance.

We consider a regression model with n observed data points (Y, X) and we focus on the task of building confidence intervals for the k -th coordinate θ_k^* of the regression vector (where $k \in \{0, \dots, p\}$).

Definition 5. A confidence interval of level $1 - \alpha$ is an interval $\hat{I}_n(Y, X) \subset \mathbb{R}$ satisfying, for all $n \geq 1$,

$$\mathbb{P}(\theta_k^* \in \hat{I}_n(Y, X)) \geq 1 - \alpha.$$

3.1.1 Gaussian model

Confidence intervals for the regression coefficients

Confidence intervals can be obtained easily when the assumption on the model allows to know the distribution of the quantity $\hat{\theta}_{n,k} - \theta_k^*$. This is the case for instance in the popular Gaussian model in virtue of Proposition 5. Recall that, when it exists,

$$\hat{s}_{n,k}^2 = e_k^T \hat{G}_n^{-1} e_k.$$

Proposition 11. In the Gaussian model, if $\ker(X) = \{0\}$ and $n > p + 1$,

$$\hat{\theta}_{n,k} + \left[- \left(\frac{\hat{s}_{n,k} \hat{\sigma}_n}{n^{1/2}} \right) Q_{n-p-1}(1 - \alpha/2), \left(\frac{\hat{s}_{n,k} \hat{\sigma}_n}{n^{1/2}} \right) Q_{n-p-1}(1 - \alpha/2) \right],$$

where Q_{n-p-1} is the quantile function of the distribution \mathcal{T}_{n-p-1} , is a confidence interval of level $1 - \alpha$.

Confidence intervals for the predicted values

We are now interested in building confidence intervals for the predicted value under the true model at a single given point $x = (1, x_1, \dots, x_p) \in \mathbb{R}^p$. The predicted value at x under the true model is defined as

$y^* = x^T \theta^*$. In the Gaussian model, using preservation properties of the Student's distribution, we find the following confidence interval $\text{CI}(x)$ of level $1 - \alpha$. With probability equal to $1 - \alpha$,

$$y^* \in \text{CI}(x),$$

where

$$\text{CI}(x) = x^T \hat{\theta}_n \pm Q_{n-p-1}(1 - \alpha/2) \hat{\sigma} \sqrt{x^T (X^T X)^{-1} x},$$

and $\hat{\sigma}_n^2 = \sum_{i=1}^n (Y_i - x_i^T \hat{\theta}_n)^2 / (n - p - 1)$ (it has been introduced in Chapter 2). A related question is to build a confidence interval on the value of y (not y^*) under the true model. This can be done in a similar manner as before but one needs to pay a particular attention to the additive noise in the model. Indeed, we have that $y = y^* + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma^2)$. It follows that

$$y \in \text{PI}(x),$$

with

$$\text{PI}(x) = x^T \hat{\theta}_n \pm Q_{n-p-1}(1 - \alpha/2) \hat{\sigma} \sqrt{1 + x^T (X^T X)^{-1} x}.$$

For more details on the derivation of those confidence intervals, see Exercise 12.

3.1.2 Nongaussian case

When the noise distribution is not Gaussian, the previous confidence interval has no reason to be valid. In this case, there are basically two techniques permitting the construction of confidence intervals:

- Concentration inequalities. This usually produces pessimistic (too large) confidence interval.
- Asymptotics. This only produces asymptotically valid confidence interval (often too small).

We start by deriving 2 confidence intervals based, respectively, on two concentration inequalities : the Markov and the Hoeffding inequalities.

Proposition 12. *In the fixed design model, suppose that (for clarity) $X^T X = nI_n$ and that (ϵ_i) is an identically distributed sequence of centered random variables with variance σ^2 , then for each $k \in \{0, 1, \dots, p\}$, the interval*

$$\hat{\theta}_{n,k} + \left[-\sqrt{\sigma^2/(n\alpha)}, \sqrt{\sigma^2/(n\alpha)} \right],$$

is a confidence interval of level $1 - \alpha$. If moreover, $|\epsilon_i| \leq c$ for all $i = 1, \dots, n$, then for each $k \in \{0, 1, \dots, p\}$, the interval

$$\hat{\theta}_{n,k} + \left[-\sqrt{2c \log(2/\alpha)c/n}, \sqrt{2c \log(2/\alpha)c/n} \right],$$

is a confidence interval of level $1 - \alpha$.

Proof. We have using (2.1), $\hat{\theta}_n - \theta^* = X^T \epsilon / n$. Applying the Markov inequality

$$\begin{aligned} \mathbb{P}(|\hat{\theta}_{n,k} - \theta_k^*| \geq t) &\leq t^{-2} \mathbb{E}[(\hat{\theta}_{n,k} - \theta_k^*)^2] \\ &\leq \sigma^2 \sum_{i=1}^n X_{i,k}^2 / (t^2 n^2) \\ &= \sigma^2 / (t^2 n), \end{aligned}$$

leading to the first confidence interval.

Applying Hoeffding inequality with the sequence (ϵ_i) , one has

$$\mathbb{P}(|\hat{\theta}_{n,k} - \theta_k^*| \geq t) \leq 2 \exp \left(-2(nt)^2 / \sum_{i=1}^n (b_i - a_i)^2 \right),$$

where $a_i \leq \epsilon_i \leq b_i$. Choosing $a_i = -c$, $b_i = c$, we get that

$$\mathbb{P}(|\hat{\theta}_{n,k} - \theta_k^*| \geq t) \leq 2 \exp(-t^2 n / 2c),$$

leading to the second confidence interval. \square

Note that the first confidence interval based on Markov inequality is very pessimistic (i.e., very large) compared to the second one, based on Hoeffding's inequality. This is because $\log(1/\alpha) \ll 1/\alpha$ when $\alpha \rightarrow 0$.

Proposition 13. *In the random design model, suppose that $\mathbb{E}[X_{1,k}^2] < \infty$ and $\mathbb{E}[Y_1^2] < \infty$, then*

$$\hat{\theta}_{n,k} + \left[- \left(\frac{\hat{s}_{n,k} \hat{\sigma}_n}{n^{1/2}} \right) \Phi^-(1 - \alpha/2), \left(\frac{\hat{s}_{n,k} \hat{\sigma}_n}{n^{1/2}} \right) \Phi^-(1 - \alpha/2) \right],$$

where Φ^- is the quantile function of the distribution $\mathcal{N}(0, 1)$, is, asymptotically, a confidence interval of level $1 - \alpha$, i.e.,

$$\liminf_{n \rightarrow \infty} \mathbb{P}(\theta_k^* \in \hat{I}_n(\alpha)) \geq 1 - \alpha.$$

Proof. That $X_n \rightsquigarrow \mathcal{N}(0, 1)$ means that $P(X_n \in [-\Phi^-(1 - \alpha/2), \Phi^-(1 - \alpha/2)]) \rightarrow \Phi(\Phi^-(1 - \alpha/2)) - \Phi(\Phi^-(\alpha/2)) = 1 - \alpha$ where Φ is the cumulative distribution function of $\mathcal{N}(0, 1)$. \square

3.2 Hypothesis testing

We start by recalling some definitions and some vocabulary related to statistical testing. Then we consider no effect tests on the covariates of a regression. These tests play an important role in practice as they might quantify the importance of each covariate in the regression. As an application, we consider the forward variable selection method in Section 3.3.

3.2.1 Definitions

Statistical testing aims at answering whether or not an hypothesis \mathcal{H}_0 is likely. It is usually performed by constructing a test statistic \hat{T}_n and deciding to reject, or not, whenever \hat{T}_n is in \mathcal{R} , or not. The region \mathcal{R} is called the reject region. As soon as \hat{T}_n and \mathcal{R} are specified, the process is quite simple:

Reject whenever $\hat{T}_n \in \mathcal{R}$

Do not reject whenever $\hat{T}_n \notin \mathcal{R}$.

The terminology “not to reject” rather than “to accept” comes from the fact that \mathcal{H}_0 is often too much thin and unlikely to be “accepted”, e.g., a simple hypothesis $\theta_1^* = 3.14159$. There are basically 2 kinds of error that we wish to control:

Type-1: to reject whereas \mathcal{H}_0 is true

Type-2: not to reject whereas \mathcal{H}_0 is not true.

The proportion of Type-1 errors is called the level of the test. One minus the proportion of Type-2 errors is called the power of the test. The consistency imposes that, for any level $1 - \alpha$, asymptotically, the level is smaller than α while the power is one. To achieve consistency, it is natural to let the reject region depend on α .

Definition 6. A statistical test $(\hat{T}_n, \mathcal{R}_\alpha)$ is said to be (asymptotically) consistent whenever for all level $1 - \alpha \in (0, 1)$

$$\limsup_{n \rightarrow \infty} P_{\mathcal{H}_0}(\hat{T}_n \in \mathcal{R}_\alpha) \leq \alpha$$

$$\lim_{n \rightarrow \infty} P_{\mathcal{H}_1}(\hat{T}_n \in \mathcal{R}_\alpha) = 1.$$

Remark 4. In practice, a standard choice is $\alpha = 0.05$. Of course when the sample size is too small one cannot be too demanding and larger values of α might be more reasonable.

3.2.2 Test of no effect

In a linear regression model, a covariate has no effect if and only if its associated regression coefficient is null. A test of no effect of a covariate, say the k -th, then consists in testing the nullity of its regression coefficient θ_k^* :

$$\mathcal{H}_0 : \theta_k^* = 0.$$

Proposition 14. Under the random design model, if $\mathbb{E}[X_1 X_1^T]$ and $\mathbb{E}[Y_1^2]$ exist and $\mathbb{E}[X_1 X_1^T]$ is invertible, the statistic and reject region, respectively given by

$$\hat{T}_{n,k} = \left(\frac{n^{1/2}}{\hat{s}_{n,k} \hat{\sigma}_n} \right) |\hat{\theta}_{n,k}|,$$

$$\mathcal{R}_\alpha = (\Phi^-(1 - \alpha/2), +\infty),$$

produce a consistent test.

Proof. For the level, it is very similar to confidence interval. For the power, suppose that $\theta_k^* \neq 0$. Let $Z_n = (n^{1/2}/\hat{s}_{n,k} \hat{\sigma}_n)(\hat{\theta}_{n,k} - \theta_k^*)$ and $q = \Phi^-(1 - \alpha/2)$. Then $\hat{T}_{n,k} \in \mathcal{R}_\alpha$ if and only if

$$Z_n + (n^{1/2}/\hat{s}_{n,k} \hat{\sigma}_n) \theta_k^* < -q \quad \text{or} \quad Z_n + (n^{1/2}/\hat{s}_{n,k} \hat{\sigma}_n) \theta_k^* > q.$$

If θ_k^* is positive (resp. negative) one can show that the event on the right (resp. left) has probability going to 1. We consider only the case $\theta_k^* > 0$. It has been shown in the proof of Proposition 10 that $\hat{s}_{n,k} \hat{\sigma}_n$ converges in probability to a finite value. We can work on the event that $\hat{s}_{n,k} \hat{\sigma}_n < M$. Let $K > 0$. For n large enough $q - (n^{1/2}/\hat{s}_{n,k} \hat{\sigma}_n) \theta_k^* < -K$. Hence

$$P(Z_n + (n^{1/2}/\hat{s}_{n,k} \hat{\sigma}_n) \theta_k^* > q) \geq P(Z_n > -K).$$

Hence

$$\liminf_{n \rightarrow \infty} P(Z_n + (n^{1/2}/\hat{s}_{n,k} \hat{\sigma}_n) \theta_k^* > q) \geq 1 - \Phi(-K).$$

But K is arbitrary and the result follows. \square

Remark 5. In practice, the statistic $\hat{T}_{n,k}$ is scale invariant: if D is a positive diagonal matrix, then the statistic $\hat{T}_{n,k}$ constructed from the sample X is the same as the statistic $\hat{T}_{n,k}$ constructed from the sample XD .

Remark 6. In the Gaussian case, the test statistic and the reject region are given by

$$\hat{T}_{n,k} = \left(\frac{n^{1/2}}{\hat{s}_{n,k} \hat{\sigma}_n} \right) |\hat{\theta}_{n,k}|,$$

$$\mathcal{R}_\alpha = (Q_{n-p-1}(1 - \alpha/2), \infty).$$

Such a test has a level exactly equal to $1 - \alpha$. To derive that the power goes to 1, one can assume that for all $n \geq 1$, $\hat{s}_{n,k} \hat{\sigma}_n$ is bounded.

patient	age	sex	bmi	bp	Serum measurements						output
	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	y
1	59	2	32.1	101	157	93	38	4	4.9	87	151
2	48	1	21.6	87	183	103	70	3	3.9	69	75
...
...
441	36	1	30.0	95	201	125	42	5	5.1	85	220
442	36	1	19.6	71	250	133	97	3	4.6	92	57

Table 3.1: The dataset is composed of $n = 442$ patients, $p = 10$ variables “baseline” body mass index, bmi), average blood pressure (bp), etc... The output is a score corresponding to the disease evolution. Each covariate has been standardized [Efron et al. \(2004\)](#).

Remark 7 (test and confidence intervals). *Making no effect tests consists in rejecting whenever 0 (or more generally any tested values) is not lying inside the confidence interval. For instance, in the random design model, to reject is equivalent to*

$$\frac{n^{1/2}}{\hat{s}_{n,k}\hat{\sigma}_n}|\hat{\theta}_{n,k}| \in (\Phi^-(1 - \alpha/2), +\infty),$$

which is equivalent to

$$0 \notin \hat{\theta}_{n,k} + \left[-\left(\frac{\hat{s}_{n,k}\hat{\sigma}_n}{n^{1/2}} \right) \Phi^-(1 - \alpha/2), \left(\frac{\hat{s}_{n,k}\hat{\sigma}_n}{n^{1/2}} \right) \Phi^-(1 - \alpha/2) \right].$$

3.3 Forward variable selection

The method of forward selection is a stepwise procedure that aims at selecting the most “important” variables. The method starts with no covariate and add a new one at each step. This kind of methods is sometimes referred to as *greedy methods*. The criterion used to select the best covariate follows from the test statistic for the test of no effect: $n^{1/2}|\hat{\theta}_{n,k}|/(\hat{s}_{n,k}\hat{\sigma}_n)$. Intuitively, the larger the statistic, the more important the effect of the k -th variable.

More formally, let $X = (1_n, \tilde{X}_1, \dots, \tilde{X}_p)$. Each (non-constant) covariate \tilde{X}_k is competing against the others via 1-dimensional regression submodels $Y \simeq \theta_0 + X_k\theta_k$. For any $Y \in \mathbb{R}^n$ and $\tilde{X}_k \in \mathbb{R}^n$, define the OLS

$$\hat{\theta}_n(Y, \tilde{X}_k) = \operatorname{argmin}_{(\theta_0, \theta_1) \in \mathbb{R}^2} \|Y - \theta_0 1_n - \theta_1 \tilde{X}_k\|^2.$$

Within each submodel, the Gram matrix and the noise level estimate are given by

$$\begin{aligned} \hat{G}_n(\tilde{X}_k) &= n^{-1}(1_n, \tilde{X}_k)^T(1_n, \tilde{X}_k), \\ \hat{\sigma}_n(Y, \tilde{X}_k)^2 &= (n - 2)^{-1}\|Y - (1_n, \tilde{X}_k)\hat{\theta}_n(Y, \tilde{X}_k)\|^2. \end{aligned}$$

Another quantity of interest is $\hat{s}_n(\tilde{X}_k)^2 = e_1^T \hat{G}_n(\tilde{X}_k)^{-1} e_1$. The criterion used to compare the importance of each variable is the test statistic of the test of no effect, computed within each submodel:

$$\hat{T}_n(Y, \tilde{X}_k) = \frac{\hat{\theta}_n(Y, \tilde{X}_k)}{\hat{s}_n(\tilde{X}_k)\hat{\sigma}_n(Y, \tilde{X}_k)}.$$

For each covariate, such a quantity is compared and the largest value is selected. This criterion has an interpretation in terms of p -values. When the test is described by $(\hat{T}_n(Y, \tilde{X}_k), \mathcal{R}_\alpha)$, the p -value is the smallest value of α for which we still reject. For instance, in the *random design model*,

$$\inf\{\alpha \in [0, 1] : \hat{T}_n(Y, \tilde{X}_k) > \Phi^-(1 - \alpha/2)\} = 2(1 - \Phi(\hat{T}_n(Y, \tilde{X}_k))).$$

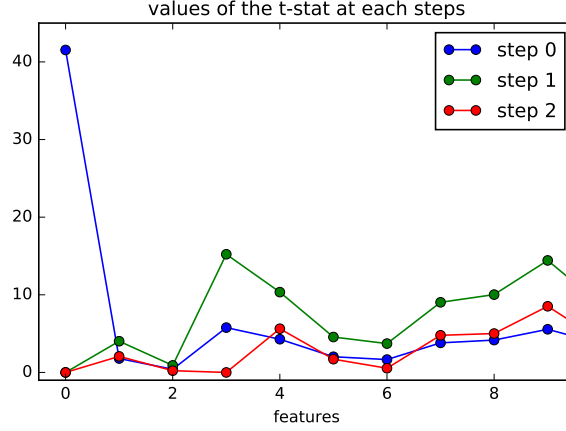


Figure 3.1: The statistics of each selected variable is 0 in the next step. The intercept is the first selected variable, then X_3 , etc...

Hence taking the largest $\hat{T}_n(Y, \tilde{X}_k)$ is equivalent to take the smallest p -value for the underlying test of no effect. A stopping rule can be based on the p -value: stop as soon as none of the p -value is smaller than 0.05. As soon as one variable, say \tilde{X}_k , is selected, one needs to account for the predictive information it has brought in the modeling of Y . This is to prevent from selecting 2 identical covariates. This is done by replacing the output Y by the residual $Y - (1_n, \tilde{X}_k)\hat{\theta}_n(Y, \tilde{X}_k)$.

Algorithm 1 (forward variable selection).

Inputs: (Y, X) a threshold p_{stop} . Start with $r = Y$, $\mathcal{S} = \emptyset \subset \mathcal{A} = \{0, \dots, p\}$.

- (i) For each $k \in \mathcal{A} \setminus \mathcal{S}$, compute $\hat{T}_n(r, \tilde{X}_k)$.
- (ii) Stop if no p -values are smaller than p_{stop} .
 Else compute $k^* \in \arg\max \hat{T}_n(r, \tilde{X}_k)$.
 And update $\mathcal{S} = \mathcal{S} \cup \{k^*\}$ and $r = r - (1_n, \tilde{X}_{k^*})\hat{\theta}_n(Y, \tilde{X}_{k^*})$.

Figure 3.3 illustrates the procedure described by Algorithm 1 applied to the “diabetes” dataset of sklearn presented in Table 3.1

Remark 8. Different stopping rules might be considered. For instance, in Zhang (2009), the authors recommend to consider the residuals sum of squares and to stop as soon as $\|r\|^2 < \epsilon$.

Exercises

Exercise 12 (explicit formulas when $p = 1$ for prediction intervals). Let us consider the following fixed-design one-dimensional ($p = 1$) linear regression model:

$$Y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma) \quad i.i.d., \quad i = 1, \dots, n.$$

Being a particular but simply interpretable case it facilitates intuitive understanding and enables easy two-dimensional visualization. Let $\bar{x}^n = n^{-1} \sum_{i=1}^n x_i$ and $\bar{Y}^n = n^{-1} \sum_{i=1}^n Y_i$. We further assume that x_i is not constant, i.e., that $\sum_{i=1}^n (x_i - \bar{x}^n)^2 \neq 0$.

1. Show that the OLS estimators $\hat{\beta}_0$ and $\hat{\beta}_1$ are

$$\hat{\beta}_0 = \bar{Y}^n - \hat{\beta}_1 \bar{x}^n \quad \text{and} \quad \hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x}^n)(Y_i - \bar{Y}^n)}{\sum_{i=1}^n (x_i - \bar{x}^n)^2}$$

2. Show that

$$e_0^T (X^T X)^{-1} e_0 = \left(\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \right) \quad \text{and} \quad e_1^T (X^T X)^{-1} e_1 = \frac{1}{\sum_{i=1}^n (x_i - \bar{x})^2},$$

3. Give the distribution of $\mathbb{V}[\hat{\beta}_0]^{-1/2}(\hat{\beta}_0 - \beta_0)$ and $\mathbb{V}[\hat{\beta}_1]^{-1/2}(\hat{\beta}_1 - \beta_1)$

$$\mathbb{V}[\hat{\beta}_0] = \hat{\sigma}^2 \left(\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \right) \quad \text{and} \quad \mathbb{V}[\hat{\beta}_1] = \frac{\hat{\sigma}^2}{\sum_{i=1}^n (x_i - \bar{x})^2},$$

$$\text{where } \hat{\sigma}^2 = \frac{1}{n-2} \sum_{i=1}^n (Y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i))^2.$$

4. Give the reject region for the test $\mathcal{H}_0 : \beta_j = 0$.

5. For a new pair (Y, x) observed from the Gaussian model above, the value $\hat{\beta}_0 + \hat{\beta}_1 x$ is called the point prediction. Show that

$$\frac{(\hat{\beta}_0 + \hat{\beta}_1 x) - (\beta_0 + \beta_1 x)}{\hat{\sigma} \sqrt{\frac{1}{n} + \frac{(x - \bar{x})^2}{\sum_{i=1}^n (x_i - \bar{x})^2}}} \sim t(n-2) \quad \text{and} \quad \frac{Y - (\hat{\beta}_0 + \hat{\beta}_1 x)}{\hat{\sigma} \sqrt{1 + \frac{1}{n} + \frac{(x - \bar{x})^2}{\sum_{i=1}^n (x_i - \bar{x})^2}}} \sim t(n-2).$$

6. Build confidence intervals for $(\beta_0 + \beta_1 x)$ and Y . Note that these intervals correspond, respectively, to CI and PI given in section [3.1.1](#). The last one is often called prediction interval.

Chapter 4

Ridge regularization

In this chapter we use the singular-value decomposition (SVD), a matrix decomposition presented in Appendix B. The SVD of X shall provide useful expression for quantities related to the OLS estimate. The SVD is also important to understand principal component analysis (PCA), a method that compresses the data without losing too much information, also presented in Appendix B.

The ridge estimator is introduced to overcome the issues caused by poorly conditioned Gram matrix \hat{G}_n , i.e., when some of the eigenvalues are too small. As indicated by the singular-value decomposition of $X = \sum_{k=1}^r s_i u_i v_i^T$, where r stands for the rank of X and s_i (resp. u_i and v_i) are the singular-values (resp. singular-vector) of X , we have that $\hat{\theta}_n = \sum_{k=1}^r s_i v_i u_i^T y$. Consequently, the estimate is numerically unstable as soon as some of the s_i are close to 0. As we can see looking at the variance of the OLS or at Proposition 8, the smallest eigenvalues of \hat{G}_n have a bad influence on the statistical behaviour of the OLS. The ridge estimator is a solution to control these bad effects due to poor conditioning. Before going through the definition of the ridge estimator, we discuss another method which consists in doing PCA before running OLS.

For ease of notation (to avoid working with \tilde{X}_c as before), in the rest of the chapter, we consider the formulation of OLS without intercept with centered variable $Y \in \mathbb{R}^n$ and $X \in \mathbb{R}^{n \times p}$. As established in Proposition 2, this is equivalent to include an intercept in the OLS with non-centered variables.

4.1 PCA before OLS

Many practitioners are familiar with the method of combining PCA and OLS. In addition to visualize and explore the centered covariates X , aim is to reduce the number of covariates to avoid inverting a possibly too large matrix $X^T X$. Be careful that after running PCA, due to its definition (see Definition 8 in Appendix B), the prediction must be operated with respect to the centered covariate X and centered output Y . Hence the intercept is no longer necessary as explained in Proposition 2. The algorithm is as follows.

Algorithm 2 (PCA before OLS).

Inputs: (Y, X) , centered variables, and an integer k (the number of components to keep). **output:** prediction of Y at $x \in \mathbb{R}^p$.

-
- (i) Do a PCA on X and keep the k first components U_1, \dots, U_k .
 - (ii) Let $P_k = \sum_{i=1}^k U_i U_i^T$. Compute $\hat{\theta}_{n,k}$, the OLS associated with $(Y, X P_k)$.
 - (iii) Return the prediction $x^T P_k \hat{\theta}_{n,k}$.
-

Trying to legitimize the approach, one can write

$$\|Y - X \hat{\theta}_n\| \leq \|Y - X P_k \hat{\theta}_{n,k}\| \leq \|Y - X \hat{\theta}_n\| + \|X(\hat{\theta}_n - \hat{\theta}_{n,k})\| + \|(X P_k - X) \hat{\theta}_{n,k}\|$$

in which, by Proposition 27, the last term should be small. However, the second term in the right hand side might be large. The lack of guarantee for this approach is due to the fact that the PCA used from the beginning is independent of the output Y . Doing such a process might result in some loss in accuracy.

4.2 Definition of the Ridge estimator

For centered variables (Y, X) , the ridge estimator is defined as a solution of the following minimization problem

$$\|Y - X\theta\|^2 + n\lambda\|\theta\|^2, \quad (4.1)$$

where $\lambda > 0$, called the regularization parameter, is fixed by the analyst. Before dealing with the choice of λ , we describe some properties of the ridge estimate. First of all, let us briefly state some simple remarks:

- Intuitively, when $\lambda \rightarrow 0$, we obtain the OLS. When $\lambda \rightarrow +\infty$, we estimate 0.
- Doing ridge is adding a regularization term to the square loss of OLS, aiming to penalize for large coefficients in θ . Other norms might be used such as $\sum_{k=1}^p |\theta_k|$ (see the next chapter about the LASSO).
- As the expression in (4.1) is a Lagrangian with constraint $\|\theta\|^2 \leq c$ the Ridge is an OLS under constraints. The link between c and λ is not explicit.
- To make the ridge estimate scale invariant, one might replace X by $XD^{-1/2}$ where D is the diagonal matrix with entries $e_k^T X^T X e_k$. Actually, this normalization permits to justify having 1 single parameter λ to control the influence of the penalty. The ridge estimate is classically defined without intercept (to prevent from penalizing the intercept). Hence one needs to first center Y and X so that the intercept of the OLS is automatically 0.

Proposition 15. *The minimizer of (4.1) exists and is unique. It is given by*

$$\hat{\theta}_n^{(rdg)} = (X^T X + n\lambda I_p)^{-1} X^T Y.$$

Proof. Let f denote the objective function of (4.1). Considering the behaviour of f at the limit of the domain, there exists A such that whenever $\|\theta\| > A$, $f(\theta) > f(0)$. But the set $\|\theta\| \leq A$ is compact and so a minimum exists and is achieved. Note that for any θ ,

$$\begin{aligned} f(\theta) - f(0) &= -2 \langle Y, X\theta \rangle + \|X\theta\|^2 + n\lambda\|\theta\|^2 \\ &= -2 \langle Y, X\theta \rangle + \|A\theta\|^2, \end{aligned}$$

with $A = ((X^T X) + n\lambda I_p)^{1/2}$ a positive matrix. For uniqueness, note that, for any u and v , we have

$$\|tu + (1-t)v\|^2 = t\|u\|^2 + (1-t)\|v\|^2 - t(1-t)\|u-v\|^2. \quad (4.2)$$

Then suppose that θ_1 and θ_2 are two distinct minimizers with $f^* = f(\theta_1) = f(\theta_2)$. We have, from (4.2),

$$\begin{aligned} &f(t\theta_1 + (1-t)\theta_2) \\ &= tf(\theta_1) + (1-t)f(\theta_2) - t(1-t)\|A(\theta_1 - \theta_2)\|^2 < f^*. \end{aligned}$$

Hence $\hat{\theta}_n^{(rdg)}$ is unique. The first order equation is

$$((X^T X) + n\lambda I_p)\theta = X^T Y.$$

□

4.3 Bias and variance

We have seen that, similarly to the OLS, the ridge estimator is the solution of a linear system of equations. In the ridge system of equations the matrix that was previously $X^T X$ in the OLS is now replaced by $X^T X + n\lambda I_p$. As λ is chosen by the user, it allows us to control the smallest eigenvalue of the underlying Gram matrix. Such a change of course influence the bias and the variance of the estimate. To express these quantities, we consider the fixed design model.

Proposition 16. *In the fixed-design model :*

- (i) *The bias of the ridge is $\mathbb{E}[\hat{\theta}_n^{(rdg)}] - \theta^* = -\lambda n(X^T X + n\lambda I_p)^{-1} \theta^*$*
- (ii) *The variance of the ridge estimator expresses as $\text{var}(\hat{\theta}_n^{(rdg)}) = \sigma^2(X^T X + n\lambda I_p)^{-1} X^T X (X^T X + n\lambda I_p)^{-1}$.*
- (iii) *We have that $\text{var}(\hat{\theta}_n^{(rdg)}) < \text{var}(\hat{\theta}_n)$, where $\hat{\theta}_n$ is the OLS solution.*

Proof. For the last point, we use the SVD of X to write that

$$\text{var}(\hat{\theta}_n^{(rdg)}) = \sigma^2 \sum_{k=1}^p \frac{s_k^2}{(s_k^2 + n\lambda)^2} u_k u_k^T.$$

In terms of eigenvalues $\hat{\lambda}_k$ associated to \hat{G}_n , we have

$$\text{var}(\hat{\theta}_n^{(rdg)}) = \sigma^2 \sum_{k=1}^p \frac{\lambda_k}{(\lambda_k + n\lambda)^2} u_k u_k^T.$$

Doing the same for $\hat{\theta}_n$ and using that $\lambda_k/(\lambda_k + \lambda)^2 < 1/\lambda_k$, we obtain the result. \square

4.4 Choice of the regularization parameter

As we have seen before, ridge regression reduces the variance of the OLS but introduces some bias. Actually this is the parameter λ that decides whether we reduce the bias, $\lambda \rightarrow 0$, or the variance, $\lambda \rightarrow \infty$. As it cannot be accomplished simultaneously, we are facing a trade-off commonly known under the name of bias-variance trade-off. In the next few lines, we promote the use of cross validation to select the parameter λ . This technique of cross validation works in more general context and is of common use as soon as one needs to choose a parameter to run a method. Examples include the choice of the bandwidth in kernel smoothing methods, the choice of the scale parameter in RKHS as well as the choice of the cut-off parameter in Huber regression.

Divide the data (Y, X) according to the lines into K -folds of (approximately) equal size $\lfloor K/n \rfloor$. Let $(Y_{(k)}, X_{(k)})$ (resp. $(Y_{-(k)}, X_{-(k)})$) denote the observation in the k -th fold (resp. all the observation outside the k -th fold). Proceed as follows:

- (i) Compute $\hat{\theta}_{n,k}^{(rdg)}$ based on each sample $(Y_{-(k)}, X_{-(k)})$.
- (ii) Compute the (unnormalized) prediction error over each fold $Y_{(k)} - X_{(k)} \hat{\theta}_{n,k}^{(rdg)}$. The risk is given by

$$\hat{R}(\lambda) = \sum_{k=1}^K \|Y_{(k)} - X_{(k)} \hat{\theta}_{n,k}^{(rdg)}\|^2.$$

The quantity $\hat{R}(\lambda)$ reflects the prediction risk associated to λ . It is then natural to minimize \hat{R} over $\lambda \in (0, \infty)$. In practice, this is usually done by taking a finite grid.

Remark 9. *A computational advantage of using the SVD is that even if considering many values of λ the SVD could be done once for each fold.*

Exercises

Exercise 13. Recall the SVD of $X = VSU^T = \sum_{k=1}^r s_k u_k v_k^T$ where $r = \text{rank}(X)$.

1. Show that $\hat{\theta}_n = \sum_{k=1}^r s_k^{-1} u_k v_k^T y = X^+ Y$ and its variance is $\text{var}(\hat{\theta}_n) = \sigma^2 \sum_{k=1}^r s_k^{-2} u_k u_k^T$.
2. Show that

$$(X^T X + n\lambda I_p)^{-1} X^T = X^T (X X^T + n\lambda I_n)^{-1}$$

(hint : one might prefer to use the complete SVD rather than its reduced form)

3. If $n \ll p$, give an efficient method that would compute the Ridge estimator and would cost less than the formula of Proposition [15](#). Compare the number of operations required.

Chapter 5

The LASSO

The LASSO (least absolute shrinkage and selection operator), introduced in Tibshirani (1996) is a regression technique that consists in minimizing the usual least-squares loss with an ℓ_1 -norm regularization. As another regularization method, it is similar to the Ridge method, presented in the previous chapter, which uses the ℓ_2 -norm to regularize. In contrast with the Ridge, some of the LASSO coefficients are usually equal to 0, meaning that the corresponding variables are no longer included in the predictive model. The LASSO thus achieves in the mean time estimation and variable selection.

5.1 Definition

As for the Ridge estimator, the LASSO is usually defined with centered variables so that we can skip estimating the intercept. We consider the following framework: $X \in \mathbb{R}^{n \times p}$ denote the covariates vector is such that $1_n^T X = 0$ and $Y \in \mathbb{R}^n$ is the output and satisfies $1_n^T Y = 0$. In other words, X and Y are supposed to have (empirical) mean 0. The LASSO estimate is defined by

$$\hat{\theta}_{\text{LASSO}} \in \operatorname{argmin}_{\theta \in \mathbb{R}^p} \left\{ \frac{1}{2} \|Y - X\theta\|_2^2 + \lambda \|\theta\|_1 \right\}, \quad (5.1)$$

where $\|\cdot\|_q$ stands for the ℓ_q -norm. Some remarks are of order:

- Note that the solution of (5.1) can be recovered by working with non centered variables X and Y and adding an intercept.
- As for the Ridge, it is standard to let the LASSO estimate be scale invariant. This is usually done by an additional standardization step that consists in using $XD^{-1/2}$ in place of X , where D is the diagonal matrix with entries $e_k^T X^T X e_k$, $k = 1, \dots, p$ (e_k being the k -th element of the canonical basis of the space \mathbb{R}^p).
- The LASSO is not unique. Some conditions for uniqueness are given and discussed in Tibshirani (2013).
- In contrast with the Ridge approach, the ℓ_1 -penalty of the LASSO objective function allows to shrink to 0 the coefficients in $\hat{\theta}_{\text{LASSO}}$ associated to the variables that are useless to predict Y .

5.2 Theoretical properties

From a theoretical perspective, the LASSO takes advantage of *sparse* regression models. A regression model is sparse whenever many of the coefficients of the parameter vector θ are equal to zero, i.e., many of the covariates are useless to predict Y . We consider the Gaussian regression model

$$Y = X\theta^* + \epsilon, \quad \text{with } \epsilon \sim \mathcal{N}(0, \sigma^2 I_n), \quad (5.2)$$

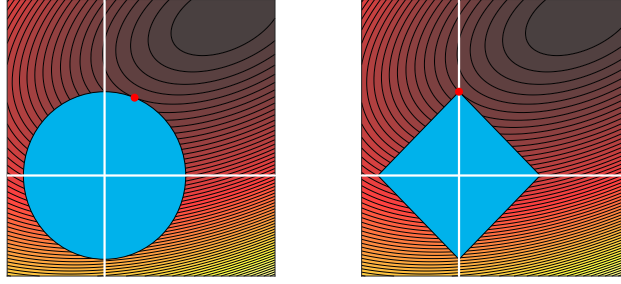


Figure 5.1: Graphical representation of the Ridge and LASSO penalties with the level set of the quadratic loss.

and we define the *active set* $S^* \subset \{1, \dots, p\}$ as

$$S^* = \{j = 1, \dots, p : \theta_j^* \neq 0\}.$$

The number of elements in S^* , that we denote by s , quantifies the level of sparsity associated to the regression model. We will see that the generalization bounds for the LASSO improve whenever s becomes small. We follow the approach presented in [Hastie et al. \(2015\)](#), in which the theoretical analysis of the LASSO is carried out using the *restricted eigenvalue condition*. This condition is basically dealing with the smallest eigenvalue of the matrix $X^T X$. It is called “restricted” because it is concerned only with particular eigenvectors that are “away” from the *not active* directions. More precisely, it only considers vectors living in certain cone that leaves away the direction S^{*c} . The collection of cones of interest are now defined. For $\alpha > 0$ and $S \subset \{1, \dots, p\}$, we set

$$C(\alpha, S) = \{u \in \mathbb{R}^p : \|u_{S^c}\|_1 \leq \alpha \|u_S\|_1\}.$$

The *restricted eigenvalue condition* (RE) for (γ, α, S) is satisfied whenever

$$n^{-1} \|Xu\|_2^2 \geq \gamma \|u\|_2^2, \quad \forall u \in C(\alpha, S). \quad (5.3)$$

The following lemma is crucial to understand the role played by the cone $C(3, S^*)$ in the analysis of the LASSO.

Lemma 1. *Whenever $\lambda \geq 2\|X^T \epsilon\|_\infty$, then*

$$0 \leq \|X\hat{u}\|_2^2 \leq \lambda(3\|\hat{u}\|_1 - \|\hat{u}_{S^{*c}}\|_1). \quad (5.4)$$

In particular, $(\hat{\theta}_{\text{LASSO}} - \theta^) \in C(3, S^*)$.*

Proof. Define

$$G(u) = \|Y - X(\theta^* + u)\|^2/2 + \lambda\|\theta^* + u\|_1 = \|\epsilon - Xu\|^2/2 + \lambda\|\theta^* + u\|_1.$$

Let $\hat{u} = \hat{\theta}_{\text{LASSO}} - \theta^*$. Because $G(\hat{\theta}_{\text{LASSO}}) \leq G(0)$, we have

$$\|X\hat{u}\|_2^2/2 \leq \langle \epsilon, X\hat{u} \rangle + \lambda(\|\theta^*\|_1 - \|\theta^* + \hat{u}\|_1).$$

From the triangle inequality, $\|(\theta^* - (-\hat{u}))_{S^*}\|_1 \geq \|\theta_{S^*}^*\|_1 - \|\hat{u}_{S^*}\|_1 \geq \|\theta_{S^*}^*\|_1 - \|\hat{u}_{S^*}\|_1$, implying that

$$\begin{aligned} \|\theta^*\|_1 - \|\theta^* + \hat{u}\|_1 &= \|\theta^*\|_1 - \|(\theta^* + \hat{u})_{S^*}\|_1 - \|(\theta^* + \hat{u})_{S^{*c}}\|_1 \\ &\leq \|\theta^*\|_1 - \|\theta_{S^*}^*\|_1 + \|\hat{u}_{S^*}\|_1 - \|(\theta^* + \hat{u})_{S^{*c}}\|_1 \\ &= \|\hat{u}_{S^*}\|_1 - \|\hat{u}_{S^{*c}}\|_1. \end{aligned}$$

From Holder inequality, we get $\langle \epsilon, X\hat{u} \rangle \leq \|X^T \epsilon\|_\infty \|\hat{u}\|_1$, which leads to

$$\|X\hat{u}\|_2^2/2 \leq \|X^T \epsilon\|_\infty \|\hat{u}\|_1 + \lambda(\|\hat{u}_{S^*}\|_1 - \|\hat{u}_{S^{*c}}\|_1).$$

Consequently, because $2\|X^T \epsilon\|_\infty \leq \lambda$, we obtain that

$$0 \leq \|X\hat{u}\|_2^2/2 \leq \lambda(\|\hat{u}\|_1/2 + \|\hat{u}_{S^*}\|_1 - \|\hat{u}_{S^{*c}}\|_1),$$

and the conclusion follows. \square

Now we can state the main result dealing with the analysis of the LASSO error.

Theorem 2. *Under the Gaussian model (5.2), assume that $\forall k \in \{1, \dots, p\}$, $(X^T X)_{k,k} \leq n$ and that RE for $(\gamma, 3, S^*)$ is satisfied. Then provided that $\lambda = 2\sqrt{2n\sigma^2 \log(2p)}$, we have with probability $1 - \delta$ that*

$$\|X(\hat{\theta}_{LASSO} - \theta^*)\|_2^2 \leq \frac{64s\sigma^2 \log(2p)}{\gamma}.$$

In addition, we have with probability $1 - \delta$ that

$$\|\hat{\theta}_{LASSO} - \theta^*\|_2 \leq \frac{6}{\gamma} \sqrt{\frac{2\sigma^2 s \log(2p)}{n}}.$$

Proof. Let $\hat{u} = \hat{\theta}_{LASSO} - \theta^*$. Suppose for now that $\lambda \geq 2\|X^T \epsilon\|_\infty$ (this will be shown to hold with probability $1 - \delta$ at the end of the proof). We have, from (5.4) and Jensen inequality, that

$$\|X\hat{u}\|_2^2 \leq 3\lambda\|\hat{u}_{S^*}\|_1 \leq 3\lambda\sqrt{s}\|\hat{u}_{S^*}\|_2 \leq 3\lambda\sqrt{s}\|\hat{u}\|_2. \quad (5.5)$$

By Lemma 1 and the RE condition, it holds that

$$\|\hat{u}\|_2^2 \leq (\gamma n)^{-1} \|X\hat{u}\|_2^2.$$

Injecting this in (5.5), we obtain the first and second statement. It remains to show that with probability $1 - \delta$, $\lambda \geq 2\|X^T \epsilon\|_\infty$. This follows from the use of Lemma 4, a Gaussian concentration result stated in Appendix C. Let $k \in \{1, \dots, p\}$, note that $(X^T \epsilon)_k$ is distributed as $\mathcal{N}(0, (X^T X)_{k,k} \sigma^2)$. Applying Lemma 4 gives that

$$\mathbb{P}(|(X^T \epsilon)_k| > t) \leq 2 \exp(-t^2/(2(X^T X)_{k,k} \sigma^2)).$$

From the union bound, it follows that

$$\mathbb{P}(\|X^T \epsilon\|_\infty > t) \leq (2p) \exp(-t^2/(2n\sigma^2)),$$

or equivalently, that with probability $1 - \delta$,

$$\|X^T \epsilon\|_\infty \leq \sqrt{2n\sigma^2 \log(2p/\delta)} = \lambda/2.$$

\square

Some other (asymptotic) properties of the LASSO are derived in Knight and Fu (2000). The authors assume the following

$$n^{-1} X^T X \rightarrow C, \text{ a positive definite matrix,} \quad (5.6)$$

and

$$(Y_i - \theta_0^* - X_i^T \theta^*)_i \text{ is an iid sequence with mean 0 and variance } \sigma^2. \quad (5.7)$$

Theorem 3 (Knight and Fu (2000)). *Suppose that (5.6) and (5.7) hold and that $\lambda/n \rightarrow 0$, then $\hat{\theta}_{LASSO} \rightarrow \theta^*$, in probability. If moreover $\lambda/\sqrt{n} \rightarrow \lambda_0 \geq 0$, then $\sqrt{n}(\hat{\theta}_{LASSO} - \theta^*)$ converges weakly.*

5.3 Computation

In contrast with the OLS or the Ridge, we have no closed formula for the LASSO solutions. This is due to the lack of smoothness of the ℓ_1 -norm. In particular, the traditional first order conditions are derived using subgradients (rather than gradients). In Appendix [D](#) some basic definitions and properties are given concerning subgradients and subdifferentials.

The following remark is helpful to characterize the set of LASSO solutions : if $f : \mathbb{R}^p \rightarrow \mathbb{R}$ is a convex function, the point x^* is a minimum if and only if $0 \in \partial f(x^*)$, where ∂f is the subdifferential of f (see Appendix [D](#) for details). This is often refereed to as the Fermat's rule.

Proposition 17. *Denote by X_k the k -th column of X . The LASSO solution satisfies*

$$\forall k = \{1, \dots, p\}, \quad \langle X_k, Y - X\hat{\theta}_{LASSO} \rangle \in \begin{cases} \{sign(\hat{\theta}_{LASSO,k})\} & \text{if } \hat{\theta}_{LASSO,k} \neq 0 \\ [-1, 1] & \text{if } \hat{\theta}_{LASSO,k} = 0 \end{cases}$$

Actually, the previous set of equations has no explicit solutions. The LASSO problem becomes much simpler when we fix all coordinates except one, and try to minimize with respect to this coordinate. For this reason, the LASSO is usually computed using a coordinate descent, i.e., by iteratively solving the first order conditions (involving subgradients) for each coordinate. For any $\lambda \geq 0$, define the function $\eta_\lambda : \mathbb{R} \rightarrow \mathbb{R}$ given by

$$\eta_\lambda(z) = \begin{cases} z + \lambda & \text{if } z < -\lambda \\ 0 & \text{if } z \in [-\lambda, \lambda] \\ z - \lambda & \text{if } z > \lambda \end{cases} \quad (5.8)$$

The function η_λ intervenes in solving least-squares with an absolute penalty (see Exercise [14](#)). Note that $\eta_\lambda(z) = \text{sign}(z)(|z| - \lambda)_+$. As shown in the following development, the function η is useful to update the LASSO in the coordinate descent algorithm. To avoid trivial cases, we suppose in the following that $\|X_k\|^2 \neq 0$. In practice, one just need to remove the constant variables. Let

$$z_k = Y_k - \sum_{j \neq k} X_j \theta_j.$$

Minimizing [\(5.1\)](#) with respect to the k -th coordinates is the same as minimizing

$$\begin{aligned} & \left\{ \frac{1}{2} (\|z_k - X_k \theta_k\|_2^2 - \|z_k\|_2^2) + \lambda |\theta_k| \right\} \\ &= \left\{ \frac{1}{2} (-2 \langle z_k, X_k \rangle \theta_k + \theta_k^2 \|X_k\|_2^2) + \lambda |\theta_k| \right\}, \end{aligned}$$

which is the same as minimizing

$$\left\{ \frac{1}{2} \left(\left\langle z_k, \frac{X_k}{\|X_k\|_2^2} \right\rangle - \theta_k \right)^2 + \frac{\lambda}{\|X_k\|_2^2} |\theta_k| \right\}.$$

Consequently, the update is given by

$$\hat{\theta}_k = \eta_{\lambda/\|X_k\|_2^2} \left(\left\langle z_k, \frac{X_k}{\|X_k\|_2^2} \right\rangle \right).$$

As it is standard, one can start with a Ridge solution and then update each coordinates with the previous formula.

5.4 Extensions

Among the extension of the LASSO, we have the LSLASSO (Least-Square LASSO) which consists in (i) running the LASSO to find the support and (ii) applying OLS on the non-zero coefficients. Another extension of LASSO is Elastic Net, introduced in [Zou and Hastie \(2005\)](#), which computes the regression coefficient

$$\hat{\boldsymbol{\theta}}_{\text{E-NET}} \in \operatorname{argmin}_{\boldsymbol{\theta} \in \mathbb{R}^p} \left\{ \frac{1}{2} \|Y - X\boldsymbol{\theta}\|_2^2 + \lambda \{ \alpha \|\boldsymbol{\theta}\|_1 + \frac{1}{2} (1 - \alpha) \|\boldsymbol{\theta}\|_2^2 \} \right\}. \quad (5.9)$$

The previous estimate is unique. Finally, a notable extension is adaptive LASSO, introduced in [Zou \(2006\)](#). The adaptive LASSO is an iterative strategy that attributes weights to each coefficient $|\boldsymbol{\theta}_k|$ in the penalty term. The weights might take the form $1/\hat{\boldsymbol{\theta}}_{OLS,k}^{1/2}$ penalizing mostly the small OLS coefficients. The resulting estimate recover the support with probability going to 1.

Exercises

Exercise 14. Show that $\eta_\lambda(z) = \operatorname{argmin}_{x \in \mathbb{R}} \{(z - x)^2/2 + \lambda|x|\}$ where η_λ is defined in [\(5.8\)](#).

Exercise 15. Following the approach given for the LASSO, find the update for the Elastic Net defined in [\(5.9\)](#).

Appendix A

Elementary results from linear algebra

The vector space \mathbb{R}^d is endowed with the usual inner product

$$\forall (u, v) \in \mathbb{R}^d \times \mathbb{R}^d, \quad \langle u, v \rangle = u^T v = \sum_{k=1}^d u_k v_k,$$

where u^T stands for the transpose of u . If $\langle u, v \rangle = 0$ we say that u and v are orthogonal and we write $u \perp v$. If E is a set of vector in \mathbb{R}^d , we define its orthogonal complement as

$$E^\perp = \{u \in \mathbb{R}^d : x^T u = 0, \quad \forall x \in E\}.$$

Proposition 18. *If E is a linear subspace of \mathbb{R}^d , then $(E^\perp)^\perp = E$.*

For any matrix $A \in \mathbb{R}^{p \times d}$, define

$$\begin{aligned} \text{span}(A) &= \{Ax : x \in \mathbb{R}^d\}, \\ \ker(A) &= \{x \in \mathbb{R}^d : Ax = 0\}. \end{aligned}$$

The set $\text{span}(A)$ is called the image of the matrix A . It is the linear space generated by the columns of A . The set $\ker(A)$ is called the kernel of A . Both sets are linked by the following property.

Proposition 19. *Let $A \in \mathbb{R}^{p \times d}$. Then $\ker(A) = \text{span}(A^T)^\perp$.*

Proposition 20. *Let $A \in \mathbb{R}^{p \times d}$. Then $\ker(A) = \{0\}$ if and only if $\text{span}(A^T) = \mathbb{R}^d$. Consequently, if $p < d$ then $\ker(A) \neq \{0\}$.*

Let $A \in \mathbb{R}^{p \times d}$, $b \in \mathbb{R}^d$. Let S be the set of solutions of the linear system $Ax = b$.

Proposition 21. *We have only three possible configurations:*

1. S contains only one element,
2. $S = \emptyset$,
3. the number of elements in S is infinite.

Note that S is empty if and only if $b \notin \text{span}(A)$.

Proposition 22. *Suppose that $b \in \text{span}(A)$ and let $x_0 \in S$, then*

$$S = x_0 + \ker(A).$$

We now recall a classical result called the spectral decomposition of symmetric matrices or the eigen decomposition of symmetric matrices.

Proposition 23. *Let $A \in \mathbb{R}^{d \times d}$ be a symmetric matrix. Then there exist $\lambda_1 \geq \dots \geq \lambda_d$, called eigenvalues, and an orthonormal matrix $U \in \mathbb{R}^{d \times d}$ (i.e., $U^T U = I_d$) of eigenvectors, such that $A = U D U^T$, where $D = \text{diag}(\lambda_1, \dots, \lambda_d)$.*

Definition 7. *A linear transformation $P \in \mathbb{R}^{d \times d}$ is called orthogonal projector if $P^2 = P$ and $P^T = P$.*

The next proposition says that an orthogonal projector is characterized by its span and, therefore, by its kernel from Proposition [18](#) and [19](#).

Proposition 24. *The eigenvalues of an orthogonal projector are either 1 or 0. Hence any orthogonal projector can be written as $U U^T$ where $U \in \mathbb{R}^{p \times r}$ forms a basis of $\text{span}(P)$.*

Proposition 25. *The trace of an orthogonal projector is equal to the dimension of its span.*

Appendix B

Singular value decomposition and principal component analysis

Before we present the method of principal component analysis (PCA), it is appropriate to recall some matrix decomposition results and more particularly the singular value decomposition (SVD).

B.1 Matrix decomposition

The usual eigen decomposition of symmetric matrices can be extended to arbitrary matrices (even not squared matrix). The price to pay is that the left and right eigenvectors are different. This is called the SVD.

Proposition 26. *Let $X \in \mathbb{R}^{n \times p}$. Then there exist two orthogonal matrices : $U \in \mathbb{R}^{p \times p}$ and $V \in \mathbb{R}^{n \times n}$ of singular vectors; and $s_1 \geq \dots \geq s_{\min(n,p)} \geq 0$, called singular values, such that*

$$X = VSU^T,$$

where $S \in \mathbb{R}^{n \times p}$ contains 0 everywhere except on the diagonal formed by $(s_1, \dots, s_{\min(n,p)})$.

Proof. Without loss of generality, we suppose that $p \leq n$. Otherwise we apply the result to the X^T . Applying Proposition 23 to $X^T X$, there exists $U \in \mathbb{R}^{p \times p}$ such that $U^T (X^T X) U$ is diagonal with r positive coefficients. Hence $U_1^T (X^T X) U_1 = D \in \mathbb{R}^{r \times r}$ and $XU_2 = 0$. Take $V_1^T = D^{-1/2} U_1^T X^T$ (an orthogonal set of r vectors : $V_1^T V_1 = I_r$) to find that $V_1^T X U_1 = D^{1/2}$. Consequently, $V_1^T X (U_1, U_2) = (D^{1/2}, 0)$. Remarking that v orthogonal to V_1 means that $v^T X U_1 = 0$ implying that $v^T X (U_1, U_2) = 0$ leading to $v^T X = 0$. Now taking V_2 such that $V = (V_1, V_2) \in \mathbb{R}^{n \times p}$ is orthogonal, we obtain the claimed decomposition with $S^2 = \text{diag}(d_1, \dots, d_p)$. \square

We have the following reduced SVD formula, if $r \geq 1$ stands for the dimension of $\text{span}(X)$,

$$X = \tilde{V}_r \tilde{S}_r \tilde{U}_r^T,$$

where $\tilde{U}_r = (U_1, \dots, U_r)$, $\tilde{V}_r = (V_1, \dots, V_r)$, and $\tilde{S}_r \in \mathbb{R}^{r \times r}$ contains only the positive singular-values.

An attractive property of the SVD is that it defines subspaces on which one can project the data X without loosing too much.

Proposition 27. *Let $X \in \mathbb{R}^{n \times p}$. For any projector $P \in \mathbb{R}^{p \times p}$ with rank smaller than k , it holds that*

$$\|X - XP_k\|_F \leq \|X - XP\|_F,$$

where $P_k = \sum_{i \leq k} U_i U_i^T$.

Proof. Suppose that $1 \leq k < r$. By Pythagorean identity, $\|X - XP\|_F^2 = \|X\|_F^2 - \|XP\|_F^2$. Hence one just has to show that $\|XP_k\|_F^2 \geq \|XP\|_F^2$. Considering the reduced SVD $X = U_r S_r V_r^T$, we have

$$\begin{aligned}\|XP\|_F^2 &= \text{tr}((PU_r)S^2(PU_r)^T) \\ &= \text{tr}\left(\sum_{i \leq r} s_i^2 W_i W_i^T\right) \\ &= \sum_{i \leq r} s_i^2 \|W_i\|_2^2,\end{aligned}$$

with $W_i = PU_i$ and the constraints that $\|W_i\|_2^2 \leq 1$ and $\sum_{i \leq r} \|W_i\|_2^2 \leq k$. Note that this corresponds to the optimization problem

$$\max_{m_1, \dots, m_{r'}} \sum_{i \leq r'} s_i^2 m_i \quad \text{u.c. } m_i \in (0, k_i), \sum_{i \leq r'} m_i \leq k,$$

in which we suppose that $s_1 < \dots < s_{r'}$ with $r' \leq r$ and $k_i \geq 1$ stands for the multiplicity. We derive the maximum. Note first that necessarily $\sum_{i \leq r'} m_i = k$. Then if i is the first index such that $0 < m_i < k_i$, the function cannot achieve its maximum. Then we get that the maximizer is achieved when m_i is either 0 or 1. Clearly the maximum is $\sum_{i \leq k} s_i^2$ which is achieved when $P = \sum_{i \leq k} U_i U_i^T$. \square

B.2 Principal component analysis

Definition 8. Let $X \in \mathbb{R}^{n \times p}$ and define $X_c = X - \frac{1}{n} X X^T X$. The PCA of X of degree k is given by the k first elements of the SVD of X_c , i.e., the singular values (s_1, \dots, s_k) , the principal components U_1, \dots, U_k and the principal axes V_1, \dots, V_k .

Introduce the estimated covariance matrix

$$\hat{\Sigma}_n = n^{-1} X_c^T X_c.$$

Proposition 28. The principal components $U = U_1, \dots, U_k$ forms a set of orthonormal vectors along which the empirical variance is maximal, i.e.,

$$\sum_{i \leq k} U_i^T \hat{\Sigma}_n U_i \geq \sum_{i \leq k} \tilde{U}_i^T \hat{\Sigma}_n \tilde{U}_i,$$

for any $(\tilde{U}_1, \dots, \tilde{U}_k)$ orthonormal vectors. The principal components U can be obtained by an eigendecomposition of $\hat{\Sigma}_n$.

Proof. Take \tilde{U} and U as define in the statement. Define $\tilde{P} = \tilde{U} \tilde{U}^T$ and $P = U U^T$, the associated projectors of rank k . Write

$$\sum_{i \leq k} U_i^T \hat{\Sigma}_n U_i = \text{tr}(\hat{\Sigma}_n P) = n^{-1} \text{tr}(X_c^T X_c P) = n^{-1} \|X_c P\|_F^2.$$

Using Proposition [27](#) and the Pythagorean identity, we get that $\|X_c P\|^2 \geq \|X_c \tilde{P}\|^2$. \square

Remark 10. As the PCA of X depends on the scale of each covariate, one may prefer in practice to rescale the matrix X before running the PCA algorithm. This can be done by taking $XD^{-1/2}$ rather than X , with D equal to the diagonal matrix whose elements are $e_k^T \hat{\Sigma}_n e_k$, $k = 1, \dots, n$. Then each covariate of XD has the same empirical variance.

Appendix C

Concentration inequalities

To derive concentration inequalities for the errors of the estimators, we use the notion of sub-Gaussianity as defined for instance in (Boucheron et al., 2013, Section 2.3). Recall that the moment generating function of a Gaussian random variable W with mean μ and variance σ^2 is equal to $\lambda \mapsto E[\exp(\lambda W)] = \exp(\mu\lambda + \lambda^2\sigma^2/2)$.

Definition 9. A centered random variable Y is sub-Gaussian with variance factor $\tau^2 > 0$, notation $Y \in \mathcal{G}(\tau^2)$, if $\log E[\exp(\lambda Y)] \leq \lambda^2\tau^2/2$ for all $\lambda \in \mathbb{R}$.

If $Y \in \mathcal{G}(\tau^2)$, then necessarily $\text{var}(Y) \leq \tau^2$ (Boucheron et al., 2013, Exercise 2.16). Centered, bounded random variables taking values in an interval $[a, b]$ are sub-Gaussian with variance factor at most $(b - a)^2/4$ (Boucheron et al., 2013, Lemma 2.2). Chernoff's inequality provides exponential bounds on the tails of sub-Gaussian random variables.

Lemma 2 (Chernoff). If $Y \in \mathcal{G}(\tau^2)$, then $P(Y > t) \leq \exp(-t^2/(2\tau^2))$.

Proof. For any $\lambda \in \mathbb{R}$, we have $1_{Y>t} \leq \exp(\lambda(Y - t))$. Hence it holds that $P(Y > t) \leq E[\exp(\lambda(Y - t))] \leq \exp(\lambda^2\tau^2/2 - \lambda t)$. Minimizing the previous bound in λ gives $\lambda = t/\tau^2$, from which we deduce that the stated bound. \square

Finally, the sum of independent sub-Gaussian variables is again sub-Gaussian. This is the statement of the following Lemma, which proof is left as an exercise.

Lemma 3. If $(Y_i)_{i \geq 1}$ is a sequence of independent random variables such that, for all $i \geq 1$, $Y_i \in \mathcal{G}(\tau_i^2)$, then $\sum_{i=1}^n Y_i \in \mathcal{G}(\sum_{i=1}^n \tau_i^2)$.

To conclude we state a concentration inequality for the sum of independent sub-Gaussian random variables. This is just a consequence of the two previous Lemma.

Lemma 4. If $(Y_i)_{i \geq 1}$ is a sequence of independent random variables such that, for all $i \geq 1$, $Y_i \in \mathcal{G}(\tau_i^2)$, then, for all $n \geq 1$, and $t \geq 0$,

$$P\left(\left|\sum_{i=1}^n Y_i\right| > t\right) \leq 2 \exp\left(-t^2/(2 \sum_{i=1}^n \tau_i^2)\right)$$

Appendix D

Optimization of convex functions

We recall here some definitions and basic properties dealing with the minimization of convex functions.

Definition 10 (convex function). *A function $f : \mathbb{R}^p \rightarrow \mathbb{R}$ is said to be convex if*

$$\forall (x, y) \in \mathbb{R}^p \times \mathbb{R}^p, \forall \alpha \in [0, 1], \quad f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y).$$

A function $f : \mathbb{R}^p \rightarrow \mathbb{R}$ is said to be strictly convex if

$$\forall x \neq y, \forall \alpha \in (0, 1), \quad f(\alpha x + (1 - \alpha)y) < \alpha f(x) + (1 - \alpha)f(y).$$

Definition 11. *Let $f : \mathbb{R}^p \rightarrow \mathbb{R}$ be a convex function. A subgradient of f at x is any vector $u \in \mathbb{R}^p$ satisfying*

$$\forall y \in \mathbb{R}^p, \quad f(y) - f(x) \geq u^T(y - x).$$

The subdifferential of f at x , noted $\partial f(x)$, is the set of all subgradients of f at x .

By simply using the definition of the subdifferential, we obtain the following characterization of minimum points (often refereed to as the Fermat's rule). This is useful in deriving the LASSO first-order conditions.

Proposition 29. *if $f : \mathbb{R}^p \rightarrow \mathbb{R}$ is a convex function, the point x^* is a minimum if and only if $0 \in \partial f(x^*)$.*

For differentiable (convex) function, the notion of subgradient coincides with the notion of gradient. This is stated in the following.

Definition 12 (differentiable function). *A function $f : \mathbb{R}^p \rightarrow \mathbb{R}$ is said to be differentiable at x_0 if there exists a vector $u \in \mathbb{R}^p$ such that*

$$\lim_{h \rightarrow 0} \frac{|f(x_0 + h) - f(x_0) - u^T h|}{\|h\|} \text{ exists}$$

As a consequence of the previous definition (when taking $h = te_k, t \rightarrow 0$) the partial derivatives (gradient) exists for differentiable functions. The gradient of a differentiable function f at $x \in \mathbb{R}^p$ is denoted by $\nabla f(x)$.

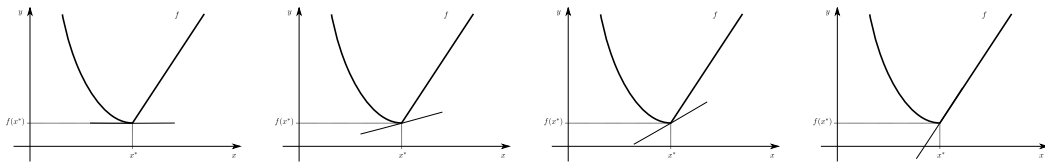


Figure D.1: Draws of subgradients.

Proposition 30 (differentiable function and convexity). *A differentiable function $f : \mathbb{R}^p \rightarrow \mathbb{R}$ is convex if and only if*

$$\forall (x, y) \in \mathbb{R}^p \times \mathbb{R}^p, \quad f(y) - f(x) \geq \nabla f(x)^T (y - x).$$

Moreover, for any $x \in \mathbb{R}^p$, the gradient $\nabla f(x)$ is the only vector satisfying the previous equation for any y . Consequently, for differentiable and convex functions $\partial f(x) = \{\nabla f(x)\}$.

Bibliography

- Boucheron, S., G. Lugosi, and P. Massart (2013). *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press.
- Efron, B., T. Hastie, I. Johnstone, R. Tibshirani, et al. (2004). Least angle regression. *The Annals of statistics* 32(2), 407–499.
- Hastie, T., R. Tibshirani, and M. Wainwright (2015). *Statistical learning with sparsity: the lasso and generalizations*. CRC press.
- Knight, K. and W. Fu (2000). Asymptotics for lasso-type estimators. *Annals of statistics*, 1356–1378.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 267–288.
- Tibshirani, R. J. (2013). The lasso problem and uniqueness. *Electronic Journal of Statistics* 7, 1456–1490.
- Zhang, T. (2009). Adaptive forward-backward greedy algorithm for sparse learning with linear models. In *Advances in Neural Information Processing Systems*, pp. 1921–1928.
- Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association* 101(476), 1418–1429.
- Zou, H. and T. Hastie (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67(2), 301–320.