Advanced Machine Learning

Lecture 2: Robust Regression

Nora Ouzir : nora.ouzir@centralesupelec.fr
Lucca Guardiola : lucca.guardiola@centralesupelec.fr

Oct. - Nov. 2020

CentraleSupélec

Content

Today's Lecture

1. Reminders
    1. Linear Regression
    2. Regularized regression

2. Robust regression
    1. M-estimation
    2. The IRLS algorithm

© Parts of these slides are borrowed from E. Chouzenoux

Today’s course

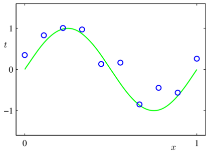
1. Reminders
    1. Linear Regression
    2. Regularized regression
2. Robust regression
    1. M-estimation
    2. The IRLS algorithm

Regression: A supervised approach

- Let $\boldsymbol{X} = (x_1, ..., x_n)$ and $\boldsymbol{Y} = (y_1, ..., y_n)$ be a set of $\boldsymbol{n}$ input/ouput training samples.
- Estimate/learn a prediction function $y = f(x)$
- $\boldsymbol{Y}$ known : supervised

Regression

- $y \in \mathbb{R}$ is a continuous variable
- Predict a numerical value



Applications

Stock price prediction, weather forecast, ...

Today’s course

1. Reminders
   1. Linear Regression
   2. Regularized regression

2. Robust regression
   1. M-estimation
   2. The IRLS algorithm

Linear Regression: Motivations

- Simple approach (essential to understand more sophisticated ones)
- Interpretable description of the relations inputs/outputs
- Can outperform nonlinear models, in the case of few training data/high noise/sparse data
- Extended applicability when combined with basis-function methods (see Lab)

Linear Regression: Applications

- Future product sales based on purchase history/client behaviour
- Economic growth of a country
- Housing market prediction: in a few months, at what price?
- Number of goals a player will score in the coming matches based on previous performance
- Hours of study needed to pass a test

Linear regression: Problem Formulation

Using the training set, learn the linear function $\boldsymbol{f}_\beta$ (parametrized by $\boldsymbol{\beta}$) predicting a real value $\boldsymbol{y} \in \mathbb{R}$ from an observation $\boldsymbol{x}_i \in \mathbb{R}^d$:

$$y_i \approx f_\beta(\boldsymbol{x}_i), \quad \forall i \in \{1, .., n\}$$

Fitting model

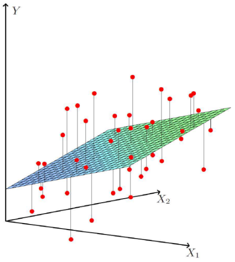
$$f_\beta(\boldsymbol{x}_i) = \beta_0 1 + \beta_1 \boldsymbol{x}_{i,1} + ... + \beta_d \boldsymbol{x}_{i,d} = \boldsymbol{x'}_i{}^T \boldsymbol{\beta} = [\boldsymbol{X}\beta]_i$$

- $\boldsymbol{X} \in \mathbb{R}^{n \times d+1}$ with $\boldsymbol{x'}_i = [1, \boldsymbol{x}_{i,1}, ..., \boldsymbol{x}_{i,d}]$ its ith line
- $\boldsymbol{\beta} = [\beta_1, ..., \beta_d]$ defines a hyperplan in $\mathbb{R}^d$
- $\boldsymbol{\beta}_0$ can be viewed as a bias that shifts the function $\boldsymbol{f}$ perpendicularly to the hyperplan

Least Squares

Find β minimizing the sum of squared residuals $\mathbf{e} = \boldsymbol{X}\beta - \boldsymbol{y}$

$$L(\beta) = \frac{1}{2}\sum_{i=1}^{n}[y_i - f(\boldsymbol{x}_i)]^2 = \frac{1}{2}\|\boldsymbol{X}\beta - \boldsymbol{y}\|^2 = \frac{1}{2}\|\mathbf{e}\|^2$$



Solution $\rightarrow$ equate the gradient to zero

Least Squares Solution

Solve $\min_{\boldsymbol{\beta}} L(\boldsymbol{\beta})$ where $L : \mathbb{R}^{d+1} \to \mathbb{R}$ is convex

$$\hat{\beta} \text{ minimizer of } L \iff \nabla L(\hat{\beta}) = 0$$

where

$$[\nabla L(\boldsymbol{\beta})]_j = \frac{\partial L(\beta)}{\partial \beta_j}, \ \forall j \in 1, .., d$$

How do we compute the solution?

- $L(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{X}\beta - \boldsymbol{y}\|^2 = \frac{1}{2}\boldsymbol{y}^T\boldsymbol{y} - \beta^T\boldsymbol{X}^T\boldsymbol{y} + \frac{1}{2}\beta^T\boldsymbol{X}^T\boldsymbol{X}\beta$
- $\nabla L(\boldsymbol{\beta}) = -\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{X}^T\beta$

If $\boldsymbol{X}$ is full column rank then $\boldsymbol{X}^T\boldsymbol{X}$ is positive definite, the solution is unique and

$$\hat{\beta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

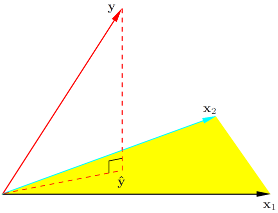
White board

Interpretation of the LS solution

We obtain the fitted values for the training inputs

$$y = \boldsymbol{X}\beta = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y} = \boldsymbol{H}\boldsymbol{y}$$

- $\boldsymbol{H}$ is called the *hat matrix* and computes the orthogonal projection of $\boldsymbol{y}$ onto the vectorial subspace spanned by the columns of $\boldsymbol{X}$.



Statistical properties

For uncorrelated observations $\boldsymbol{y}_i$ with variance σ^2, and deterministic $\mathbf{x}_i$

$$\mathrm{Var}(\hat{\beta}) = (\mathbf{X}^\top \mathbf{X})^{-1}\sigma^2$$

Unbiased estimator:

$$\hat{\sigma}^2 = \frac{1}{n-(d+1)} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Inference properties

Assume that $Y = \beta_0 + \sum_{j=1}^{d} X_j \beta_j + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Then $\hat{\beta}$ and $\hat{\sigma}$ are independant and

- $\hat{\beta} \sim \mathcal{N}(\beta, (\mathbf{X}^\top \mathbf{X})^{-1}\sigma^2)$
- $(n-(d+1))\hat{\sigma}^2 \sim \sigma^2 \chi^2_{n-(d+1)}$

Today’s course

1. Reminders
   1. Linear Regression
   2. Regularized regression
2. Robust regression
   1. M-estimation
   2. The IRLS algorithm

High dimensional linear regression

Some problems arise with least squares regression when d is large

- Accuracy: The hyperplan fits the data well but predicts/generalizes badly. (low bias / large variance)
- Interpretation: Identify a small subset of features important/relevant for predicting the data.

High dimensional linear regression

Some problems arise with least squares regression when d is large

- Accuracy: The hyperplan fits the data well but predicts/generalizes badly. (low bias / large variance)
- Interpretation: Identify a small subset of features important/relevant for predicting the data.

How can we tackle the above issues ?

Use an additional regularization term $R(\beta)$

$$L(\beta) = \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda R(\beta)$$

Common regularization types

Ridge regression

Reduces the magnitudes of the coefficients in β

$$R(\beta) = \frac{1}{2}\|\beta\|^2$$ Explicit solution!

Shrinkage

Sparsifies the coefficients in β : most elements are zero

$$R(\beta) = \|\beta\|_1$$ Optimization method needed!

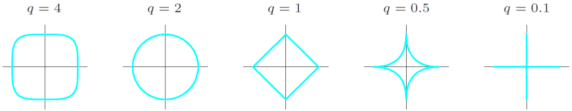
Subset selection

Sparsifies the coefficients in β : most elements are zero

$$R(\beta) = \|\beta\|_0$$ Optimization method needed!

White board

Penalty functions



Contour plots for $\sum_j |\beta_j|^q$

When the columns of $\mathbf{X}$ are orthonormal, the estimators can be deduced from the LS estimator $\hat{\beta}$ according to:

- Ridge : $\hat{\beta}_j/(1+\lambda)$ weight decay
- Lasso : $\text{sign}(\hat{\beta}_j)(|\hat{\beta}_j| - \lambda)_+$ soft tresholding
- Best subset : $\hat{\beta}_j \cdot \delta\left(\hat{\beta}_j^2 \geq 2\lambda\right)$ hard tresholding

Proximal Gradient

- Gradient step

$$\bar{\beta}_k = \beta_k - \theta \nabla L(\beta_k)$$

- Proximal gradient step

$$\beta_{k+1} = \underset{\beta}{argmin} \frac{1}{2} \|\bar{\beta}_k - \beta\|^2 + \lambda \theta R(\beta)$$

- Guarantied convergence to a local minimum when $\theta \in]0, \frac{2}{\|X^T X\|}[$
- $F(\beta_k)$ will decrease monotonicallly with k
- Global minimum with ℓ_1 norm
- Local minimum with the non-convex ℓ_0-norm

Solutions for the proximal step

- $R(\beta) = \|\beta\|_1$

$$\beta_{k+1} = sign(\bar{\beta}_k) \times max(|\bar{\beta}_k| - \lambda\theta, 0)$$

- $R(\beta) = \|\beta\|_0$

$$\beta_{k+1} = \bar{\beta}_k \times \delta(\bar{\beta}_k^2 \leq 2\lambda\theta)$$

Element-wise operations!

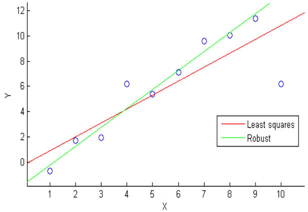
Today's course

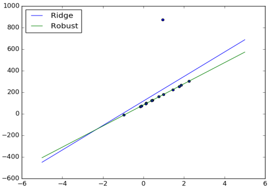
1. Reminders
    1. Linear Regression
    2. Regularized regression

2. Robust regression
    1. M-estimation
    2. The IRLS algorithm

Least Squares shortcomings

What happens in the presence of outliers ?

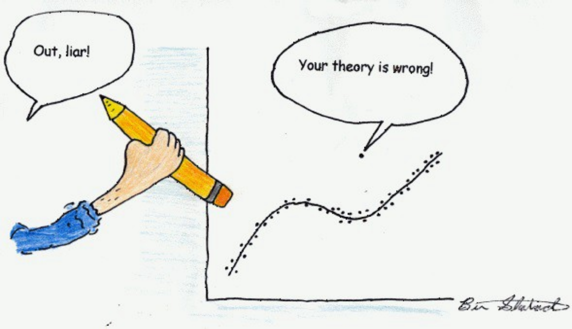




Different problems can be adressed

- Outliers in the dataset
- Mismodelling of the data set

Robust Statistics



Goal

Design estimation methods insensitive to outliers and possibly high leverage points

Today's course

1. Reminders
    1. Linear Regression
    2. Regularized regression

2. Robust regression
    1. M-estimation
    2. The IRLS algorithm

Robust Statistics: M-estimation

Maximum Likelihood-type estimates

One-Parameter Case: $X_1, ..., X_n$ i.i.d. $\sim f(x;\theta), \theta \in \Theta$

- Likelihood function: $L(x_1, ..., x_n; \theta) = \prod_{i=1}^{n} f(x_i; \theta)$
- Minimize the negative log-likelihood:

$$\min_{\theta} \sum_{i=1}^{n} \rho(x_i; \theta) \text{ where } \rho(x; \theta) = -\log f(x; \theta).$$

- Solve the likelihood equations:

$$\sum_{i=1}^{n} \psi(x_i; \theta) = 0 \text{ where } \psi(x_i; \theta) = \frac{\partial \rho(x_i; \theta)}{\partial \theta}$$

Robust regression: Objective function approach

We can achieve robustness using M-estimation

$$L(\beta) = \sum_{i=1}^{n} \rho(y_i - \mathbf{x}_i'^\top \beta)$$

with ρ a potential function satisfying:

- $\rho(e) \geq 0$ and $\rho(0) = 0$
- $\rho(e) = \rho(-e)$
- $\rho(e) \geq \rho(e')$ for $|e| \geq |e'|$

Robust regression: Objective function approach

We can achieve robustness using M-estimation

$$L(\beta) = \sum_{i=1}^{n} \rho(y_i - \mathbf{x}_i'^\top \beta)$$

with ρ a potential function satisfying:

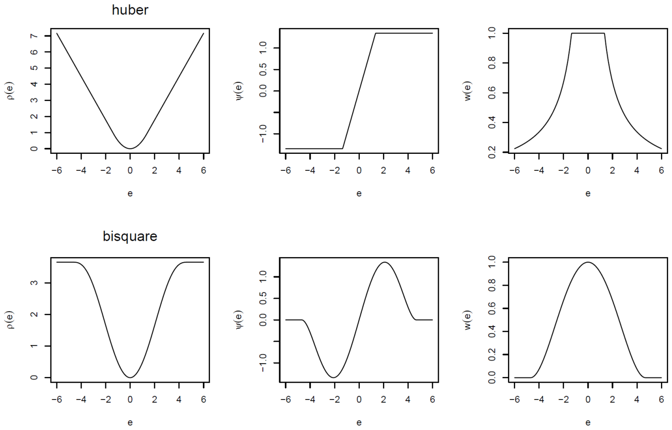
- $\rho(e) \geq 0$ and $\rho(0) = 0$
- $\rho(e) = \rho(-e)$
- $\rho(e) \geq \rho(e')$ for $|e| \geq |e'|$

Minimizer satisfies

$$\dot{\rho}(y_i - \mathbf{x}_i'^\top \hat{\beta})\mathbf{x}_i' = 0, \quad i = 1, \ldots, n$$

$\rightarrow$ IRLS algorithm

Examples of functions ρ



Today's course

1. Reminders
    1. Linear Regression
    2. Regularized regression

2. Robust regression
    1. M-estimation
    2. The IRLS algorithm

IRLS : Iteratively Reweighted Least Squares

Key Idea: Iteratively down-weight outliers using information about the errors, *i.e.*,
The largest errors at step $k-1$ ($\mathbf{e}^k = \boldsymbol{X}\hat{\beta}^k - \mathbf{y}$) are assigned low weights $\boldsymbol{W}^k$ at the current step k

- The weights are computed using a weight function (based on ρ)

$$\omega(e_i) = \frac{\dot{\rho}(e_i)}{e_i}, \ \forall i = 1, ..., n$$

- Recall that the minimizer $\hat{\beta}$ satisfies

$$\dot{\rho}(y_i - \mathbf{x}_i'^{\top}\hat{\beta})\mathbf{x}_i' = 0, \quad i = 1, \dots, n \qquad (1)$$

$$\omega(e_i)e_i\frac{\partial e_i}{\partial \beta_j} = 0, \quad j = 1, \dots, d \qquad (2)$$

- (2) corresponds to solving:
  min $\sum_{i=1}^{n} \omega(e_i^{k-1})e_i^2$ (or min $\|\boldsymbol{W}_{k-1}^{\frac{1}{2}}(\boldsymbol{X}\hat{\beta}_k - \mathbf{y})\|^2$)

IRLS algorithm: Majorization-Minimization approach

Let ρ be defined as

$$(\forall x \in \mathbb{R}) \qquad \rho(x) = \phi(|x|)$$

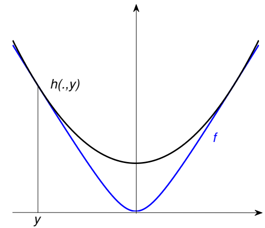
where

(i) ϕ is differentiable on $]0, +\infty[$,

(ii) $\phi(\sqrt{\cdot})$ is concave on $]0, +\infty[$,

(iii) $(\forall x \in [0, +\infty[) \quad \dot{\phi}(x) \geq 0$,

(iv) $\lim_{\substack{x \to 0 \\ x > 0}} \left(\omega(x) := \frac{\dot{\phi}(x)}{x} \right) \in \mathbb{R}$.



Then, for all $y \in \mathbb{R}$,

$$(\forall x \in \mathbb{R}) \quad \rho(x) \leq \rho(y) + \dot{\rho}(y)(x - y) + \frac{1}{2}\omega(|y|)(x - y)^2.$$

Examples of functions ρ

	$\rho(x)$	$\omega(x)$ (exercise)
Convex	$\lvert x\rvert - \delta \log(\lvert x\rvert/\delta + 1)$	
	$\begin{cases} x^2 & \text{if } \lvert x\rvert < \delta \\ 2\delta\lvert x\rvert - \delta^2 & \text{otherwise} \end{cases}$	
	$\log(\cosh(x))$	
	$(1 + x^2/\delta^2)^{\kappa/2} - 1$	
Nonconvex	$1 - \exp(-x^2/(2\delta^2))$	
	$x^2/(2\delta^2 + x^2)$	
	$\begin{cases} 1 - (1 - x^2/(6\delta^2))^3 & \text{if } \lvert x\rvert \leq \sqrt{6}\delta \\ 1 & \text{otherwise} \end{cases}$	
	$\tanh(x^2/(2\delta^2))$	
	$\log(1 + x^2/\delta^2)$	

$$(\lambda, \delta) \in]0, +\infty[^2,\ \kappa \in [1, 2]$$

White board

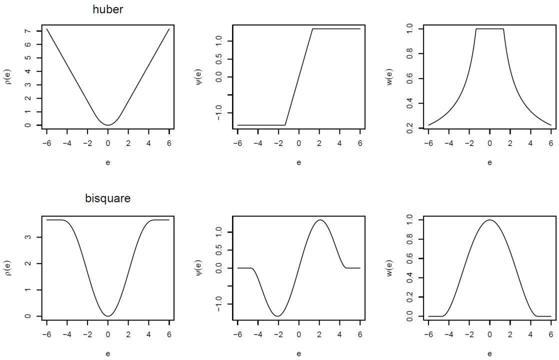
IRLS algorithm

The IRLS weight matrix is

$$\mathbf{W}_k = \mathrm{Diag}(\omega(\mathbf{y} - \mathbf{X}\beta_k))$$

and

$$(\forall k \in \mathcal{N}) \quad \beta_{k+1} = (\mathbf{X}^\top \mathbf{W}_k \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{W}_k \mathbf{y}.$$



IRLS algorithm: Summary

Algorithm

- Initialize $\boldsymbol{\beta}_0$ (e.g., LS estimates)
- Then iterate between the following steps:

For $k = 1, ..., k_{max}$

1. Compute $\boldsymbol{e}^{k-1} = \boldsymbol{Y} - \boldsymbol{X}^T \boldsymbol{\beta}_{k-1}$
2. Compute $\boldsymbol{W}_k = diag[\omega(e_i^{k-1})]$
3. Update the estimate $\boldsymbol{\beta}_k = (\mathbf{X}^\top \mathbf{W}_k \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{W}_k \mathbf{y}$