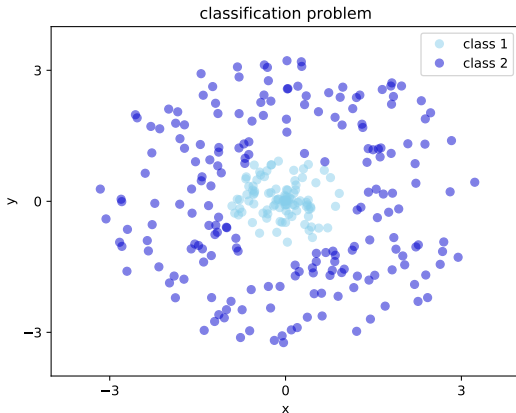


Fondamentaux théoriques du machine learning



Overview of lecture 4

Ridge regression

- Summary of OLS

- Ridge regression estimator

- Cross validation

- Numerical resolution of OLS and Ridge regression

Feature maps

Classification

- Problem statement

- Convexification of the risk and calibration

- Logistic regression

- Maximum likelihood

Ridge regression

- Summary of OLS

- Ridge regression estimator

- Cross validation

- Numerical resolution of OLS and Ridge regression

Feature maps

Classification

- Problem statement

- Convexification of the risk and calibration

- Logistic regression

- Maximum likelihood

Why study OLS

- ▶ Illustrates the bias-variance decomposition
- ▶ Can be extended to non-linear features (see section II)

Summary

- ▶ If X is injective, then we have a closed form solution :

$$\hat{\theta} = (X^T X)^{-1} X^T y \quad (1)$$

- ▶ Risk decomposition

$$E[R_X(\theta)] - R_X(\theta^*) = \|E[\theta] - \theta^*\|_{\hat{\Sigma}}^2 + E\left[\|\theta - E[\theta]\|_{\hat{\Sigma}}^2\right]$$

- ▶ Ecess risk :

$$E[R_X(\hat{\theta})] - R_X(\theta^*) = \frac{\sigma^2 d}{n} \quad (2)$$

Expected value of empirical risk

Proposition

The expected value of the empirical risk of $\hat{\theta}$ writes :

$$E[R_n(\hat{\theta})] = \frac{n-d}{n} \sigma^2 \quad (3)$$

Expected value of empirical risk

Proposition

The expected value of the empirical risk of $\hat{\theta}$ writes :

$$E[R_n(\hat{\theta})] = \frac{n-d}{n} \sigma^2 \quad (4)$$

Exercise 1 : Variance estimation. Could we use this to estimate the variance ?

Expected value of empirical risk

Proposition

$$E[R_n(\hat{\theta})] = \frac{n-d}{n} \sigma^2 \quad (5)$$

Two consequences :

- ▶ In expectation, the amount of overfitting is $\frac{2\sigma^2}{n}$.
- ▶ We can have an **unbiased estimator** of the variance σ^2 with :

$$\frac{\|Y - X\hat{\theta}\|_2^2}{n-d} \quad (6)$$

Issues in high dimension

The problem can become **ill-conditioned**.

When d is large (for instance when $\frac{d}{n}$ is close to 1), then

- ▶ the amount of excess risk is not way smaller than σ^2 .
- ▶ if $d = n$ and $X^T X$ is invertible, we can fit the training data exactly, which is bad for generalization.

If $d > n$, $X^T X$ is not invertible, we do not have a closed form solution anymore, we can have a subspace of solutions.

Remark : in low d as well, the problem can be ill-conditioned (for instance if X has colinear columns).

Regularization

To avoid these problems, a solution is to perform **regularization** of the objective function.

Regularizing the problem is an approach to enforce the unicity of the solution at the cost of introducing a **bias** in the estimator. The unicity is guaranteed by the **strong convexity** of the new loss function.

Ridge regression estimator

$$\hat{\theta}_{\lambda} = \arg \min_{\theta \in \mathbb{R}^d} \left(\frac{1}{n} \|Y - X\theta\|_2^2 + \lambda \|\theta\|_2^2 \right) \quad (7)$$

with $\lambda > 0$.

Ridge regression estimator

Proposition

The Ridge regression estimator is unique even if $X^T X$ is not invertible and is given by

$$\hat{\theta}_\lambda = \frac{1}{n}(\hat{\Sigma} + \lambda I_d)^{-1} X^T Y$$

with

$$\hat{\Sigma} = \frac{1}{n} X^T X \in \mathbb{R}^{d,d} \tag{8}$$

Proof

Step 1 : Prove that the loss $R_n(\theta)$ is strongly convex.

Proof

Step 1 : Prove that the loss $R_n(\theta)$ is strongly convex.

- ▶ $x \mapsto \|\theta\|^2$ is 2-convex on \mathbb{R}^d
 - ▶ $\theta \mapsto \theta_i$ is linear
 - ▶ $u \mapsto u^2$ is 2-convex on \mathbb{R}
- ▶ $R_n(\theta)$ is a sum of a convex function and $\theta \mapsto \lambda \|\theta\|_2^2$.

Proof

Step 2 : as $R_n(\theta)$ is strongly convex, there exists a unique minimizer obtained by cancellation of the gradient.
Compute the gradient of $R_n(\theta)$.

Proof

Step 2 : as $R_n(\theta)$ is strongly convex, there exists a unique minimizer obtained by cancellation of the gradient.

Compute the gradient of $R_n(\theta)$.

$$\nabla_{\theta} R_n(\theta) = \frac{2}{n}(X^T X \theta - X^T y) + 2\lambda \theta$$

Proof

The equation of the cancellation of the gradient is

$$\left(\frac{2}{n}n\hat{\Sigma} + 2\lambda I_d\right)\theta_\lambda = \frac{2}{n}X^T y$$

which we can write

$$n(\hat{\Sigma} + \lambda I_d)\theta_\lambda = X^T y$$

Proof

The equation of the cancellation of the gradient is

$$\left(\frac{2}{n}n\hat{\Sigma} + 2\lambda I_d\right)\theta_\lambda = \frac{2}{n}X^T y$$

which we can write

$$n(\hat{\Sigma} + \lambda I_d)\theta_\lambda = X^T y$$

$\hat{\Sigma} + 2\lambda I_d$ is a symmetric matrix with all eigenvalues $\geq 2\lambda$. Thus, it is invertible. Also, $\forall a \in \mathbb{R}^*$ and $A \in GL_d \mathbb{R}$, $(aA)^{-1} = \frac{1}{a}A^{-1}$, which concludes the proof.

Statistical analysis of ridge regression

Proposition

Under the linear model assumption, with fixed design setting, the ridge regression estimator has the following excess risk

$$E[R(\hat{\theta}_\lambda) - R^*] = \lambda^2 \theta^{*T} (\hat{\Sigma} + \lambda I_d)^{-2} \hat{\Sigma} \theta^* + \frac{\sigma^2}{n} \text{tr}[\hat{\Sigma}^2 (\hat{\Sigma} + \lambda I_d)^{-2}] \quad (9)$$

Choice of λ

Is it possible that the excess risk is smaller with ridge regression than OLS?

Proposition

With the choice

$$\lambda^* = \frac{\sigma \sqrt{\text{tr}(\hat{\Sigma})}}{\|\theta^*\|_2 \sqrt{n}} \quad (10)$$

then

$$E[R(\hat{\theta}_\lambda) - R^*] \leq \frac{\sigma \sqrt{\text{tr}(\hat{\Sigma})} \|\theta^*\|_2}{\sqrt{n}} \quad (11)$$

Choice of λ

$$E[R(\hat{\theta}_\lambda) - R^*] \leq \frac{\sigma \sqrt{\text{tr}(\hat{\Sigma})} \|\theta^*\|_2}{\sqrt{n}} \quad (12)$$

- ▶ dimension-free bound
- ▶ $\frac{1}{n}$ (OLS) vs $\frac{1}{\sqrt{n}}$ (ridge), with different constants, dimension-free in ridge.

Hyperparameter

- ▶ In practical situations, the quantities involved in the computation of λ^* in 10 are typically unknown. However this equation show that there may exist a λ with a good prediction performance, which can be found by cross validation in practice.
- ▶ λ is an example of **hyperparameter**.

Hyperparameter

scikit

- ▶ cross validation
- ▶ grid search

Neural networks

With neural networks, it seems that it is possible to have $d \gg n$ but no overfitting (simplicity bias). Why?

Numerical resolution

- ▶ closed-form OLS and ridge estimator require matrix inversions.
- ▶ $\mathcal{O}(d^3)$ operation. This is prohibitive in large dimensions (e.g. $\geq 10^5$).
- ▶ **iterative algorithms** are preferred :
 - ▶ Gradient descent (GD)
 - ▶ Stochastic gradient descent (SGD)

Gradient descent

$$\theta \leftarrow \theta - \gamma \nabla_{\theta} f \quad (13)$$

γ is a parameter called the learning rate.

- ▶ We will study gradient algorithms later in the course
- ▶ In many cases, it is possible to compute explicit convergence rates.

Ridge regression

- Summary of OLS

- Ridge regression estimator

- Cross validation

- Numerical resolution of OLS and Ridge regression

Feature maps

Classification

- Problem statement

- Convexification of the risk and calibration

- Logistic regression

- Maximum likelihood

Feature maps

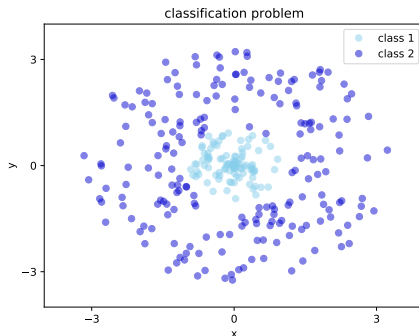
Often, we do not work with the $x_i \in \mathcal{X}$, but with **representations** $\phi(x_i)$, with $\phi : \mathcal{X} \rightarrow \mathbb{R}^d$. Possible motivations :

- ▶ \mathcal{X} need not be a vector space.
- ▶ $\phi(x)$ can provide more useful **features** for the considered problem (classification, regression).
- ▶ The prediction function is then allowed to depend **non-linearly** on x .

Feature map

Exercise 2: Finding a feature map

What feature map could be used to be able to linearly separate these data ?



Application to OLS and ridge

Instead of

$$X = \begin{pmatrix} x_1^T \\ \dots \\ x_i^T \\ \dots \\ x_n^T \end{pmatrix} = \begin{pmatrix} x_{11}, \dots, x_{1j}, \dots, x_{1d} \\ \dots \\ x_{i1}, \dots, x_{ij}, \dots, x_{id} \\ \dots \\ x_{n1}, \dots, x_{nj}, \dots, x_{nd} \end{pmatrix}$$

The design matrix is

$$\phi = \begin{pmatrix} \phi(x_1)^T \\ \dots \\ \phi(x_i)^T \\ \dots \\ \phi(x_n)^T \end{pmatrix}$$

Application to OLS and ridge

The statistical results are maintained, as a function of d , the dimension of $\phi(x)$.

Linear estimator

We often encounter estimators of the form

$$f(x) = h(\langle \phi(x), \theta \rangle) = h(\phi(x)^T \theta) \quad (14)$$

- ▶ They are often called "linear models"
- ▶ Being linear in θ is not the same as being linear in x .

Linear estimator

We often encounter estimators of the form

$$f(x) = h(\langle \phi(x), \theta \rangle) = h(\phi(x)^T \theta) \quad (15)$$

- ▶ regression : $h = Id$
- ▶ classification : $h = \text{sign}$.

Linear estimator

Interpretation of a linear model as a vote, in the case of classification.

$$f(x) = h(\langle \phi(x), \theta \rangle) = h(\phi(x)^T \theta) \quad (16)$$

Kernel methods

The topic of feature maps is very rich and important in machine learning

- ▶ **kernel methods** : ϕ is **chosen**. Many famous choices are available (gaussian kernels, polynomial kernels, etc).
- ▶ **neural networks** : ϕ is **learned**.

We will have a dedicated course on both these methods.

Ridge regression

- Summary of OLS

- Ridge regression estimator

- Cross validation

- Numerical resolution of OLS and Ridge regression

Feature maps

Classification

- Problem statement

- Convexification of the risk and calibration

- Logistic regression

- Maximum likelihood

General classification problem

- ▶ $\mathcal{X} = \mathbb{R}^d$
- ▶ $\mathcal{Y} = \{-1, 1\}$ or $\mathcal{Y} = \{0, 1\}$.
- ▶ $l(y, z) = 1_{y \neq z}$ ("0-1" loss)
- ▶ $F = \mathcal{Y}^{\mathcal{X}}$

Problem

Optimizing on $F = \mathcal{Y}^{\mathcal{X}}$ is equivalent to optimizing in the set of subsets of \mathcal{X} .

We cannot differentiate on this hypothesis space and it is not clear how to regularize.

Subsets

Exercise 3 : Combinatorial problem

If we wanted to try all applications in $\mathcal{Y}^{\mathcal{X}}$, if $|\mathcal{X}| = n$, how many applications would there be ?

Bayes predictor

Proposition

Law of total expectation

$$E_{X,Y}[I(X, Y)] = E_X[E(I(X, Y)|X)] \quad (17)$$

Bayes predictor

Hence,

$$f^*(x) = \arg \min_{z \in \mathcal{Y}} E \left[I(y, z) | X = x \right] \quad (18)$$

Bayes predictor

Reminder : if we assume the knowledge of the joint distribution (X, Y) , the Bayes predictor can be explicitly computed.

$$\begin{aligned} f^*(x) &= \arg \min_{z \in \mathcal{Y}} E \left[I(y, z) | X = x \right] \\ &= \arg \min_{z \in \mathcal{Y}} P(Y \neq z | X = x) \\ &= \arg \min_{z \in \mathcal{Y}} 1 - P(Y = z | X = x) \\ &= \arg \max_{z \in \mathcal{Y}} P(Y = z | X = x) \end{aligned} \tag{19}$$

The optimal classifier selects the most probable output given $X = x$.

Bayes risk

We note $\eta(x) = P(Y = 1|X = x)$.

$$\begin{aligned} R^* &= E\left[l(Y, f^*(X))\right] \\ &= E\left[P(Y \neq f^*(X))\right] \\ &= E_X\left[E_Y(P(Y \neq f^*(X)|X = x))\right] \\ &= E_X\left[P(Y \neq f^*(x)|X = x)\right] \\ &= E_X\left[\min(\eta(x), 1 - \eta(x))\right] \end{aligned} \tag{20}$$

Exercise 4: Same random variable (X, Y) as in lecture 3, with $p = 1/3$, $q = 3/4$.

- ▶ $\mathcal{X} = \{0, 1\}$, $\mathcal{Y} = \{0, 1\}$.
- ▶ $X \sim B(\frac{1}{2})$,

$$Y = \begin{cases} B(1/3) & \text{if } X = 1 \\ B(3/4) & \text{if } X = 0 \end{cases}$$

With $B(p)$ a Bernoulli law with parameter p .

Compute the Bayes estimator and the bayes risk.

Bayes estimator

Bayes estimator

▶ $f^*(0) = 1$

▶ $f^*(1) = 0$

▶ $\eta(1) = \frac{1}{3}$

▶ $\eta(0) = \frac{3}{4}$

$$R^* = \frac{7}{24} \quad (21)$$

Real-valued function

Instead of an application in $\mathcal{Y}^{\mathcal{X}}$, we will learn $g : \mathcal{X} \rightarrow \mathbb{R}$ and define $f(x) = \text{sign}(g(x))$ with

$$\text{sign}(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases}$$

Risk

The risk (generalization error) of $f = \text{sign} \circ g$ is defined as

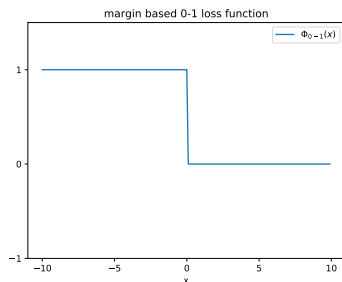
$$\begin{aligned} R(g) &= P(\text{sign}(g(x)) \neq y) \\ &= E \left[1_{\text{sign}(g(x)) \neq y} \right] \\ &= E \left[1_{yg(x) < 0} \right] \end{aligned} \tag{22}$$

Several solutions

There might be many optimal functions g , i.e : such that $\text{sign}(g(x)) = f^*(x)$.

Margin based 0-1 loss function Φ_{0-1}

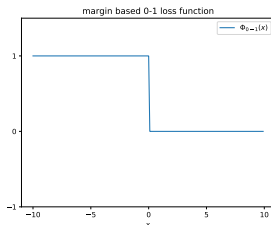
$$\begin{aligned} R(g) &= E \left[1_{\text{sign}(g(x)) \neq y} \right] \\ &= E \left[1_{yg(x) < 0} \right] \\ &= E \left[\Phi_{0-1}(yg(x)) \right] \end{aligned} \tag{23}$$



Empirical risk minimization

The corresponding empirical risk writes :

$$\frac{1}{n} \sum_{i=1}^n \Phi_{0-1}(y_i g(x_i)) \quad (24)$$

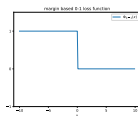


What is the issue with this objective function ?

Empirical risk minimization

The corresponding empirical risk writes :

$$\frac{1}{n} \sum_{i=1}^n \Phi_{0-1}(y_i g(x_i)) \quad (25)$$



What is the issue with this objective function ?

- ▶ non-convex
- ▶ not continuous

Convex surrogate

Key idea : replace Φ_{0-1} by another function Φ that is easier to optimize (convexity) but still represents the correctness of the classification.

Definition

The Φ -risk is defined as

$$R_{\Phi}(g) = E \left[\Phi(yg(x)) \right] \quad (26)$$

The empirical Φ -risk is defined as

$$R_{\Phi,n}(g) = \frac{1}{n} \sum_{i=1}^n \Phi(y_i g(x_i)) \quad (27)$$

Convex surrogate

Key idea : replace Φ_{0-1} by another function Φ that is easier to optimize (convexity) but still represents the correctness of the classification.

Definition

The Φ -risk is defined as

$$R_{\Phi}(g) = E \left[\Phi(yg(x)) \right] \quad (28)$$

The empirical Φ -risk is defined as

$$R_{\Phi,n}(g) = \frac{1}{n} \sum_{i=1}^n \Phi(y_i g(x_i)) \quad (29)$$

Key question : does minimizing the Φ -risk lead to a good "0-1" loss prediction ?

Most common convex surrogates

Definition

Logistic loss

$$\Phi(u) = \log(1 + e^{-u}) \quad (30)$$

With linear predictors, this loss will lead to **logistic regression** (which is classification despite its name).

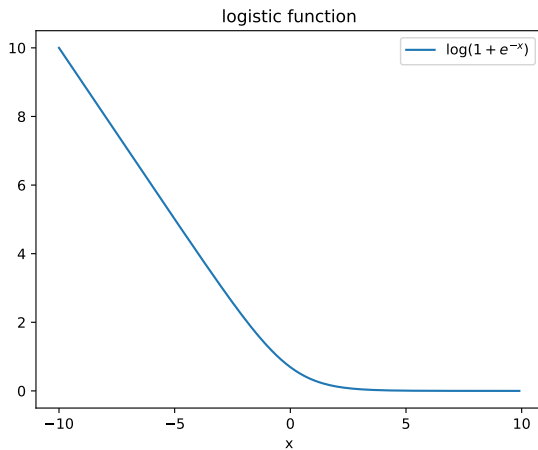
Most common convex surrogates

If $\mathcal{Y} = \{0, 1\}$, \hat{y} is the prediction and y is the correct label, then we sometimes write :

$$l(\hat{y}, y) = y \log(1 + e^{-\hat{y}}) + (1 - y) \log(1 + e^{\hat{y}}) \quad (31)$$

(cross entropy loss)

Logistic function



Most common convex surrogates

Definition

Hinge loss

$$\Phi(u) = \max(1 - u, 0) \quad (32)$$

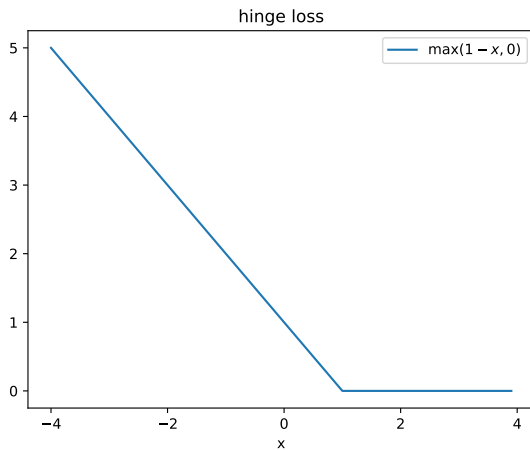
With linear predictors, this loss will lead to **Support vector machines**.

Definition

Squared hinge loss

$$\Phi(u) = (\max(1 - u, 0))^2 \quad (33)$$

Hinge loss



Φ -risk minimization

- ▶ We come back to the question : does minimizing the empirical Φ -risk lead to a good "0-1" loss prediction ?
- ▶ The Bayes predictor stays the same, but several Φ can be used. Hence, several minimizers can be obtained, since the minimizer or the Φ -risk depends on the choice of Φ .

Phi-risk minimization

- ▶ Testing error

$$R(g) = E\left[\Phi_{0-1}(yg(x))\right] \quad (34)$$

- ▶ Testing loss

$$R_{\Phi}(g) = E\left[\Phi(yg(x))\right] \quad (35)$$

Conditional Φ -risk

Definition

Conditional Φ -risk

$$E\left[\Phi(yg(x))|x\right] = \eta(x)\Phi(g(x)) + (1-\eta(x))\Phi(-g(x)) = C_{\eta(x)}(g(x)) \quad (36)$$

with

$$C_{\eta}(\alpha) = \eta\Phi(\alpha) + (1-\eta)\Phi(-\alpha) \quad (37)$$

Calibrated Φ

We say that Φ is *calibrated* if :

- ▶ $\eta > \frac{1}{2} \Leftrightarrow \arg \min_{\alpha \in \mathbb{R}} C_{\eta}(\alpha) \subset \mathbb{R}_{+}^{*}$
- ▶ $\eta < \frac{1}{2} \Leftrightarrow \arg \min_{\alpha \in \mathbb{R}} C_{\eta}(\alpha) \subset \mathbb{R}_{-}^{*}$

This means that the optimal $\forall x$, taken independently, the optimal $g(x)$ obtained by minimizing the conditional Φ -risk leads to the same prediction as the Bayes predictor.

Necessary and sufficient condition

Proposition

Let $\Phi : \mathbb{R} \rightarrow \mathbb{R}$ convex.

Φ is calibrated $\Leftrightarrow \Phi$ is differentiable at 0 and $\Phi'(0) < 0$.

Necessary and sufficient condition

Proposition

Let $\Phi : \mathbb{R} \rightarrow \mathbb{R}$ convex.

Φ is calibrated $\Leftrightarrow \Phi$ is differentiable at 0 and $\Phi'(0) < 0$.

The conditions are verified for the logistic loss and the hinge loss.

Calibration function

To know if minimizing $R_\Phi(g)$ leads to minimizing $R(g)$, it would be sufficient to have a monotonic function H (calibration function), such that

$$R(g) - R^* \leq H\left[R_\Phi(g) - R_\Phi^*\right] \quad (38)$$

Logistic regression

- ▶ $g(x) = \langle x, \theta \rangle = x^T \theta$.
- ▶ $f(x) = \text{sign}(\langle x^T \theta \rangle)$
- ▶ It can be seen as "linear regression applied to classification".

Logistic regression

In this section we use the setting $\mathcal{Y} = \{0, 1\}$.

► prediction : $\hat{y} = x^T \theta$

$$l(\hat{y}, y) = y \log(1 + e^{-\hat{y}}) + (1 - y) \log(1 + e^{\hat{y}}) \quad (39)$$

(cross entropy loss)

Logistic regression estimator

If l is the logistic loss, it is defined as

$$\hat{\theta}_{logit} = \arg \min_{\theta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^n l(x_i^T \theta, y_i)$$

Logistic regression

Exercise 5 : Convexity

Show that the logistic loss is stricly convex in θ :

$$\theta \mapsto y \log(1 + e^{-x^T \theta}) + (1 - y) \log(1 + e^{x^T \theta}) \quad (40)$$

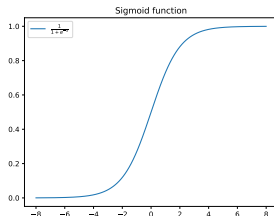
Sigmoid

Definition

Sigmoid function

$\sigma : \mathbb{R} \rightarrow \mathbb{R}$.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (41)$$



No closed-form solution

Since the loss is convex, to minimize it is sufficient to look for the cancellation of the gradient. However, the corresponding equation has no closed-form solution.

We thus need to use iterative algorithms (Gradient descent, Newton's method)

Practical usage of logistic regression

In practice, it is common practice to :

- ▶ regularize the logistic loss to avoid overfitting, for instance with a $L2$ penalty (as in ridge regression)
- ▶ use feature maps and classify with $\phi(x)$ instead of x .

Likelihood

Let $\mathcal{P} = \{p_\theta, \theta \in \Theta\}$ be a parametric model. Given $y \in \mathcal{Y}$, the **likelihood** of θ is defined as the function $\theta \mapsto p_\theta(y)$.

The likelihood $L(.|D_n)$ of a dataset $D_n = (y_1, \dots, y_n)$ is defined as

$$L(.|D_n) : \theta \mapsto \prod_{i=1}^n p_\theta(y_i)$$

Likelihood

Since the samples y_i are assumed to be independent, the likelihood corresponds to the probability of observing the dataset according to p_θ . We can define a loss function as the **negative log-likelihood**.

$$\Theta \times \mathcal{Y} \mapsto -\log(p_\theta(y))$$

Given this loss, the risk writes :

$$R(\theta) = E_Y[-\log(p_\theta(y))]$$

and the empirical risk (ER) :

$$R_n(\theta) = -\frac{1}{n} \sum_{i=1}^n \log(p_\theta(y_i))$$

Maximum likelihood

Finding the parameter with maximum likelihood means finding the parameter that minimizes $R_n(\theta)$.

$$R_n(\theta) = -\frac{1}{n} \sum_{i=1}^n \log(p_\theta(y_i))]$$

Link with logistic regression

Let us now consider the probabilistic model such that

$$p_{\theta}(1|x) = \sigma(\theta^T x)$$

which makes sense since $\sigma(\theta^T x) \in [0, 1]$, and can thus be interpreted as a probability.

Link with logistic regression

Let us now consider the probabilistic model such that

$$p_{\theta}(1|x) = \sigma(\theta^T x)$$

which makes sense since $\sigma(\theta^T x) \in [0, 1]$, and can thus be interpreted as a probability.

Equivalently, this model can be written (remember that $y = 0$ or $y = 1$)

$$p_{\theta}(y|x) = (\sigma(\theta^T x))^y (1 - \sigma(\theta^T x))^{1-y}$$

Link with logistic regression

Let us now consider the probabilistic model such that

$$p_{\theta}(1|x) = \sigma(\theta^T x)$$

We will show that the parameter θ with maximum likelihood is the logistic regression estimator θ_{logit} .

$$\begin{aligned} R_n(\theta) &= -\frac{1}{n} \sum_{i=1}^n \log(p_{\theta}(y_i|x_i)) \\ &= -\frac{1}{n} \sum_{i=1}^n \log\left((\sigma(\theta^T x_i))^{y_i} (1 - \sigma(\theta^T x_i))^{1-y_i}\right) \\ &= -\frac{1}{n} \sum_{i=1}^n y_i \log(\sigma(\theta^T x_i)) + (1 - y_i) \log(\sigma(-\theta^T x_i)) \\ &= \frac{1}{n} \sum_{i=1}^n y_i \log(1 + e^{\theta^T x_i}) + (1 - y_i) \log(1 - e^{\theta^T x_i}) \\ &= \frac{1}{n} \sum_{i=1}^n l(\theta^T x_i, y_i) \end{aligned}$$

Empirical risk minimization for the log-likelihood with this model and the logistic regression are the same.