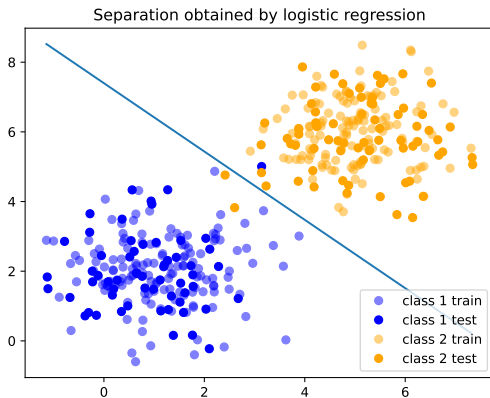


# Machine learning I, supervised learning: logistic regression



## 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.

## Using scikit

### Exercise 1 :

Use scikit to perform a logistic regression on the **digits** dataset.

template : `logistic_regression/logistic_regression_digits.py`

# Subsets

## Exercise 2: Combinatorial problem

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

## 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  $f = \text{sign} \circ g$  writes :

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

## Several solutions

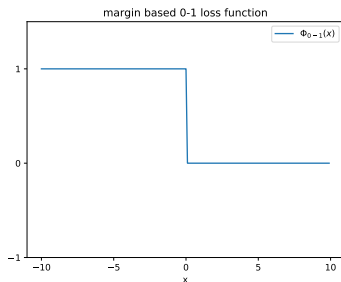
There might be many optimal maps  $g$ , i.e : such that  $\text{sign}(g(x)) = f^*(x)$  for all  $x$ .

(Here,  $f^*(x)$  is the Bayes predictor, which is the optimal predictor : it minimizes the generalization error (risque réel))



## Margin based 0-1 loss function $\Phi_{0-1}$

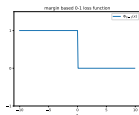
$$\begin{aligned} R(g) &= E \left[ 1_{\text{sign}(g(x)) \neq y} \right] \\ &= E \left[ 1_{yg(x) < 0} \right] \\ &= E \left[ \Phi_{0-1}(yg(x)) \right] \end{aligned} \tag{2}$$



## Empirical risk minimization

The corresponding empirical risk writes :

$$\frac{1}{n} \sum_{i=1}^n \Phi_{0-1}(y_i g(x_i)) \quad (3)$$



Issue with this objective function ?

- ▶ non-convex
- ▶ not continuous

## Convex surrogate

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

Natural question : but does minimizing the  $\Phi$ -risk lead to a good "0-1" loss prediction ? Answering this question requires an advanced study.

## Most common convex surrogates

### Définition

Logistic loss

$$\Phi(u) = \log(1 + e^{-u}) \quad (4)$$

With linear predictors ( $g(x_i) = \langle \theta, x_i \rangle$ ), this loss will lead to **logistic regression** (which is classification despite its name).

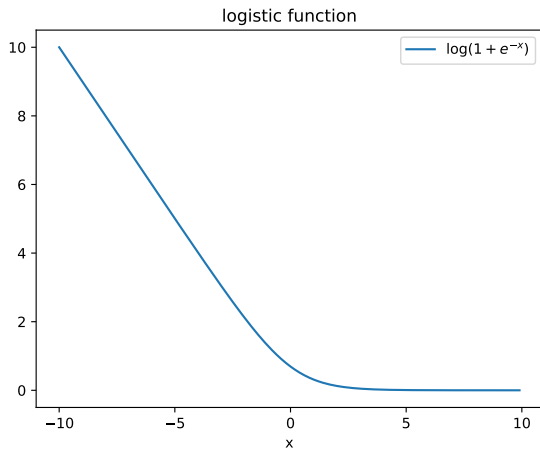
## Most common convex surrogates

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

$$l(\hat{y}, y) = y \log(1 + e^{-\hat{y}}) + (1 - y) \log(1 + e^{\hat{y}}) \quad (5)$$

(cross entropy loss)

# Logistic function



## Most common convex surrogates

### Définition

Hinge loss

$$\Phi(u) = \max(1 - u, 0) \quad (6)$$

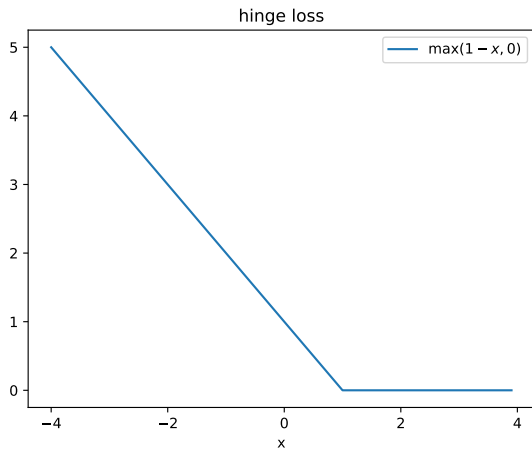
With linear predictors, this loss will lead to **Support vector machines**.

### Définition

Squared hinge loss

$$\Phi(u) = (\max(1 - u, 0))^2 \quad (7)$$

# Hinge loss

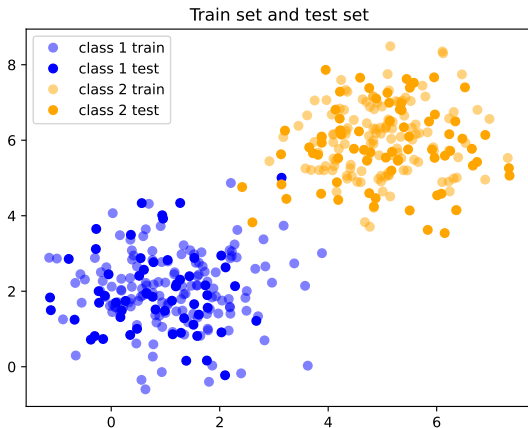




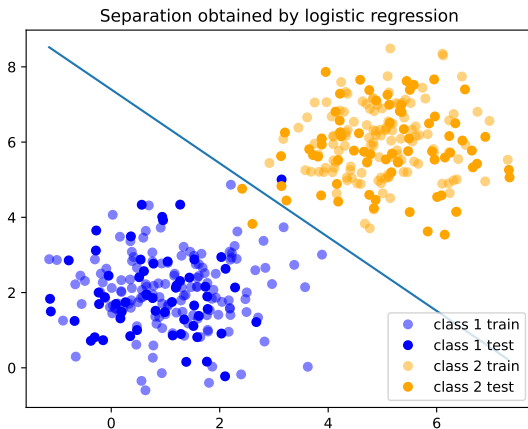
# Logistic regression

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

## Application on a toy dataset



## Application on a toy dataset



## Logistic regression

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

► prediction :  $\hat{y} = x^T \theta$

$$l(\hat{y}, y) = y \log(1 + e^{-\hat{y}}) + (1 - y) \log(1 + e^{\hat{y}}) \quad (8)$$

(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

We can show that the logistic loss is stricly convex in  $\theta$  :

$$\theta \mapsto y \log(1 + e^{-x^T \theta}) + (1 - y) \log(1 + e^{x^T \theta}) \quad (9)$$

This means that if we manage to find  $\theta$  that cancels the **gradient** of the empirical risk,  $\theta$  is a global minimizer.

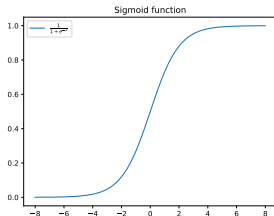
# Sigmoid

## Définition

Sigmoid function

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

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (10)$$



## 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)

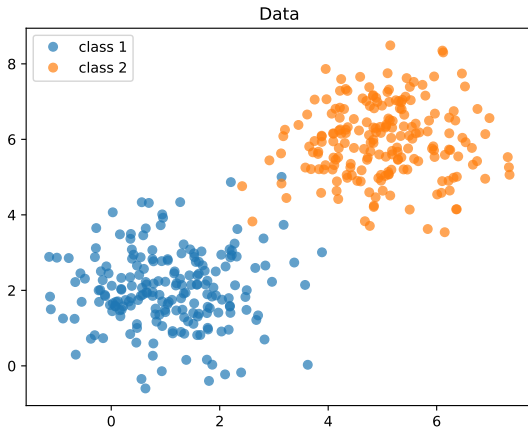


## Practical usage of logistic regression

In practice, it is common practice to :

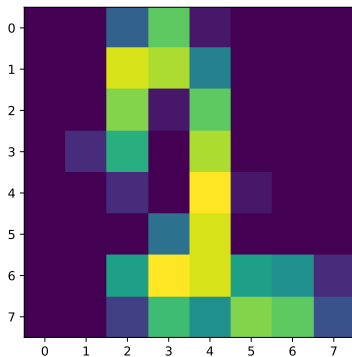
- ▶ regularize the logistic loss to avoid overfitting, for instance with a  $L2$  penalty (as in ridge regression)
- ▶ use feature maps and classify with  $\phi(x)$  instead of  $x$ .

Application to a simple 2D dataset.



Application to the digits dataset.

[https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load\\_digits.html](https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_digits.html)



# Data transformation (feature maps)

