Foundations of Machine Learning

•

Online Perceptron and Linear SVM

Lecturer: Yann Chevaleyre
Scribe: Alexandre NGAU

Lecture n°7
09/11/2023

1 Linear Discrimination

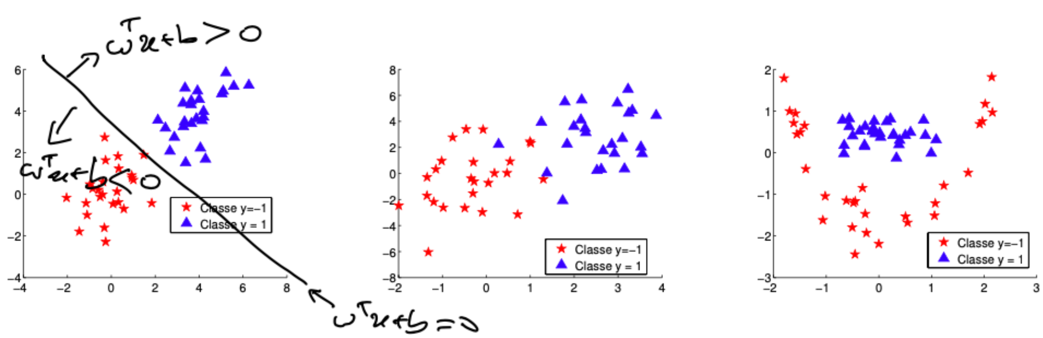
1.1 Formulation

Let $D = \{(x_i, y_i) \in X \times \{-1, 1\}\}_{i=1}^{n}$ be a set of labeled points. The goal is to build from D a function $f : X \to \{-1, 1\}$ or $f : X \to \mathbb{R}$ which predicts the class -1 or 1 of a point $x \in X$.

Definition 1. ***Scoring Function***
We assume the input space $X = R^d$.
*The **scoring function** : $f : R^d \to R$ such that if $f(x) < 0$, assign x to class -1, and if $f(x) > 0$, assign x to class 1.*
*The **linear decision function** : $f(x) = w > x + b$, where w is a d-dimensional weight vector and b is a scalar bias term.*

Definition 2. ***Linearly Separable Problem***
The points (x_i, y_i) are linearly separable if there exists a hyperplane that can correctly discriminate the entire dataset. Otherwise, we refer to them as linearly non-separable examples. In this lecture, we choose the one that maximizes the margin (see Figure 1).

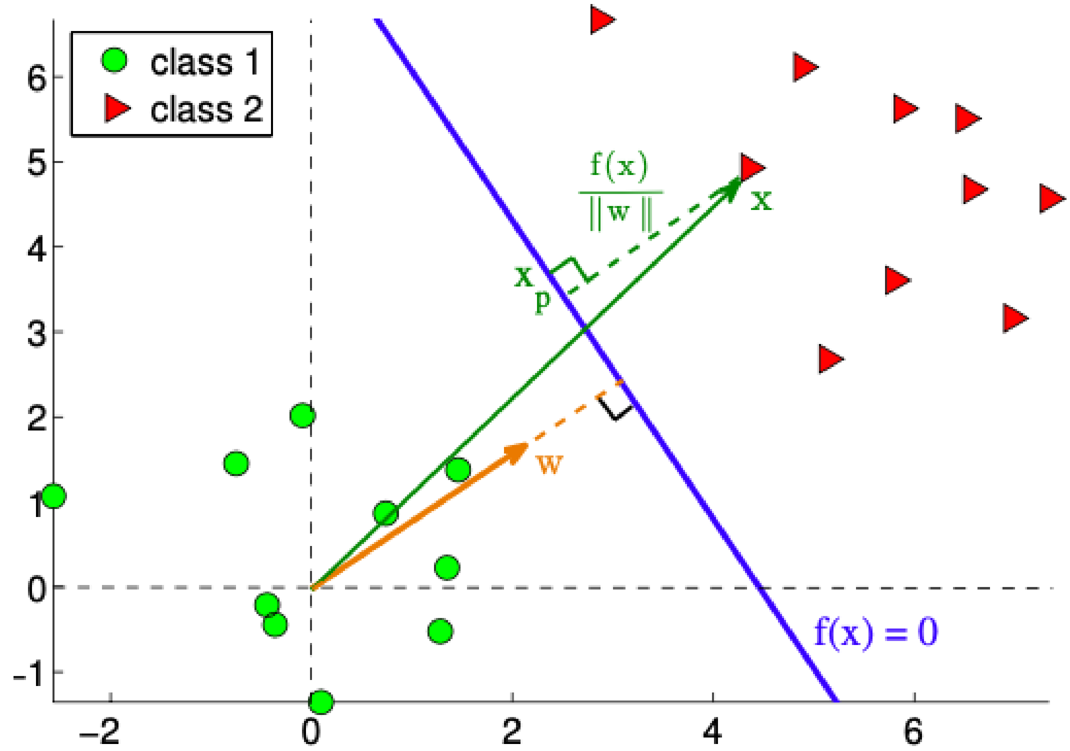
Figure 1: Separable and Non-separable Linear Problems



1.2 Linear Separator and Margin Maximization

Definition 3. ***Distance from a Point to the Decision Boundary***
Let $H(w, b) = \{z \in \mathbb{R}^d \mid f(z) = w^T z + b = 0\}$ be a hyperplane, and let $x \in \mathbb{R}^d$. The distance from the point x to the hyperplane H is $d(x, H) = |w^T x + b| = |f(x)|$ (see Figure 2).

Figure 2: Distance from a Point to the Decision Boundary



Proof.
Let $x = x_p + \frac{w}{\|w\|} \times d$ where $d = \frac{f(x)}{\|w\|}$.
Take the dot product of x with w :
$w^T w = w^T x_p + \frac{w^T w}{\|w\|} d$ where $\frac{w^T w}{\|w\|} d = \|w\| d$
So, $\|w\| d = w^T x - w^T x_p = (w^T x + b) - (w^T x_p + b) = w^T x + b$ because $w^T x_p + b = 0$
Finally, $d = \frac{w^T x + b}{\|w\|}$ □

Definition 4.
Canonical Hyperplane
A hyperplane is said to be canonical with respect to the data $\{x_1, \dots, x_N\}$ *if* $\min_{x_i} |w^T x_i + b| = 1$.

Margin
The geometrical margin is $M = \frac{2}{\|w\|}$

Optimal Canonical Hyperplane
The optimal canonical hyperplane maximizes the margin, and classes correctly each point i.e. $\forall i, y_i f(x_i) > 1$

1.3 Perceptron Algorithm

The following **Perceptron Algorithm** is for homogenous linear classifiers $f(x) = w^T x$ (with no bias b for the moment.

Algorithm 1: The Perceptron Algorithm (online setting)

Data:
$t \leftarrow 0$
$w_0 \leftarrow 0$
repeat
  Receive x_t;
  Predict $\hat{y}_t = \text{sign}(w_t^T x_t)$;
  Receive $y_t \in \{-1; 1\}$;
  if $y_t \neq \hat{y}_t$ **then**
    Update $w_{t+1} \leftarrow w_t + y_t w_t$
  else
    Update $w_{t+1} \leftarrow w_t$
until *convergence*;

Theorem 1. ***Block, Norikoff***
Assume : $\forall t, \|x_t\| < R, y_t \in \{-1; 1\}$.
Assume there exists a canonical hyperplane w^* classifying data perfectly, and passing through the origin with a half margin $\rho = \frac{1}{\|w^*\|}$.
Then, the number of mistakes of perceptron is at most $\frac{R^2}{\rho^2}$.

Proof.
Step 1
After an update (a prediction error), w_{t+1} is "more aligned to w^*".

$$
\begin{aligned}
< w_{t+1}, w^* > &= < w_t + y_t x_t, w^* > \\
&= < w_t, w^* > + y_t < x_t, w^* > \\
&\geq < w_t, w^* > + 1 \text{ because } y_t < x_t, w^* > \geq 1 \text{ (} w^* \text{ is canonical}
\end{aligned}
$$

Unrolling, we get $< w_t, w^* > \geq t$
Step 2
After an update (classification error) :

$$
\begin{aligned}
\|w_{t+1}\|^2 &= < w_t + y_t x_t, w_t + y_t x_t > \\
&= \|w_t\|^2 + 2y_t < w_t, x_t > + \|y_t x_t\|^2 \\
&\leq \|w_t\|^2 + R^2 \text{because } 2y_t < w_t, x_t > \leq 0 \\
&\implies \|w_t\|^2 \leq tR^2
\end{aligned}
$$

Step 3

$$
\begin{aligned}
t &\leq < w_t, w^* > \\
&\leq \|w_t\|\|w^*\|\text{Cauchy-Schwarz} \\
&\leq \sqrt{t}R\|w^*\| \\
&= \sqrt{t}\frac{R}{\rho} \\
&\implies sqrtt \leq \frac{R}{\rho} \\
t &\leq \frac{R^2}{\rho^2}
\end{aligned}
$$

□

1.4 Perceptron Algorithm as an SGD Online Learner

Let $s_t = w_t^T x$

<u>Perceptron Algorithm Update :</u>

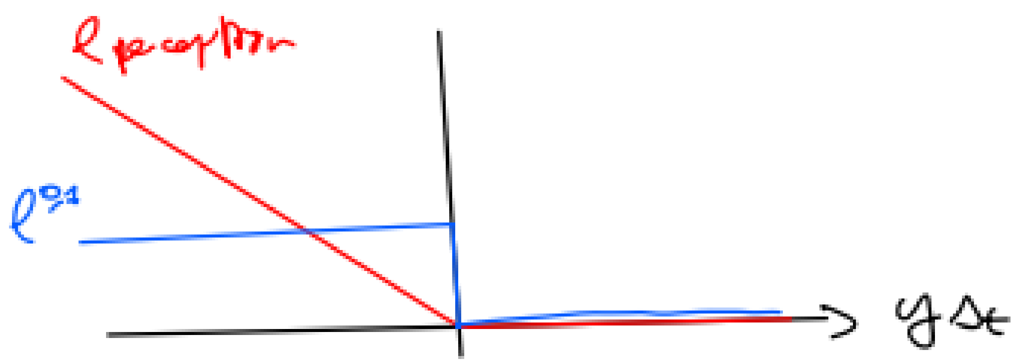
if $y_t s_t < 0$ **then**
 Update $w_{t+1} \leftarrow w_t + y_t w_t$
else
 Update $w_{t+1} \leftarrow w_t$

<u>SGD Update :</u>

$$w_{t+1} \leftarrow w_t - \alpha \nabla_w l^{perceptron}(s_t, y) \text{ with } l^{perceptron}(s_t, y) = \begin{cases} 0 & \text{if } y_t s_t \geq 0 \\ -y_t x_t & \text{otherwise} \end{cases} = max(0, -ys_t)$$

If $\alpha = 1$ then applying SGD here gives : $w_{t+1} \leftarrow w_t - \alpha \begin{cases} 0 & \text{if } y_t s_t \geq 0 \\ -y_t x_t & \text{otherwise} \end{cases}$

Figure 3: Graph of the Perceptron loss and the 0/1 loss



Definition 5.
VC Bound
Risk R on a class of functions H with a probability $1-\delta$ is :
$R(h) \leq R_{emp}(h) + C\sqrt{\frac{D(\log(2N/D)+1)+\log(4\delta)}{N}}$ *where D is the VC-dimension of H.*

VC-dimension of the Class of Linear Functions with Margin ρ
Let H be the class of functions $f(x) = w^T x + b$ with a margin ρ from the learning examples. Then $D \leq 1 + \min(d, \frac{R^2}{\rho^2})R$, where R is the radius of a ball containing the training data.

Definition 6. ***SVM and Formulation of the Maximisation Problem***
Let $D = \{(x_i, y_i) \in \mathbb{R}^d \times \{-1, 1\}\}_{i=1}^n$ be a set of linearly separable points.
The goal is to find a decision function $f(x) = w^T x + b$ that maximizes the margin and correctly discriminates the points in D i.e. $min \frac{1}{2}\|w\|^2$ subject to $y_i(w^T x_i + b) \geq 1$ for all $i = 1, \ldots, n$ (all points correctly classified).

2 Solving the SVM Problem

2.1 Primal Problem and Lagrangian

Definition 7. ***Primal Problem of SVM***
$min_{w\in\mathbb{R}^d, b\in\mathbb{R}} \frac{1}{2}\|w\|^2$ *subject to* $y_i(w \cdot x_i + b) \geq 1, \; \forall \; i = 1, \ldots, n$

We then introduce Lagrange multipliers $\alpha_i \geq 0$ associated with the n inequality constraints, i.e., n parameters α_i.

Finally, the Lagrangian of the problem is : $L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^n \alpha_i \left(y_i(w^T x_i + b) - 1\right)$.

2.2 SVM Dual Problem

Stationarity Condition

$$\frac{\partial L(w, b, \alpha)}{\partial b} = 0 \quad \text{and} \quad \frac{\partial L(w, b, \alpha)}{\partial w} = 0$$

So,

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad \text{and} \quad w = \sum_{i=1}^n \alpha_i y_i x_i$$

Definition 8. ***Dual Problem of SVM : quadratic programming problem***
By substituting these values into the Lagrangian, we obtain :

$$\begin{aligned} \max_{\{\alpha_i\}} \quad & \sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j \\ s.t. \quad & \alpha_i \geq 0, \quad \forall i = 1, \ldots, n \\ & \sum_{i=1}^n \alpha_i y_i = 0 \end{aligned}$$

With ***complementary slackness*** *:* $\alpha_i \left(y_i(w^T x_i + b) - 1\right) = 0$

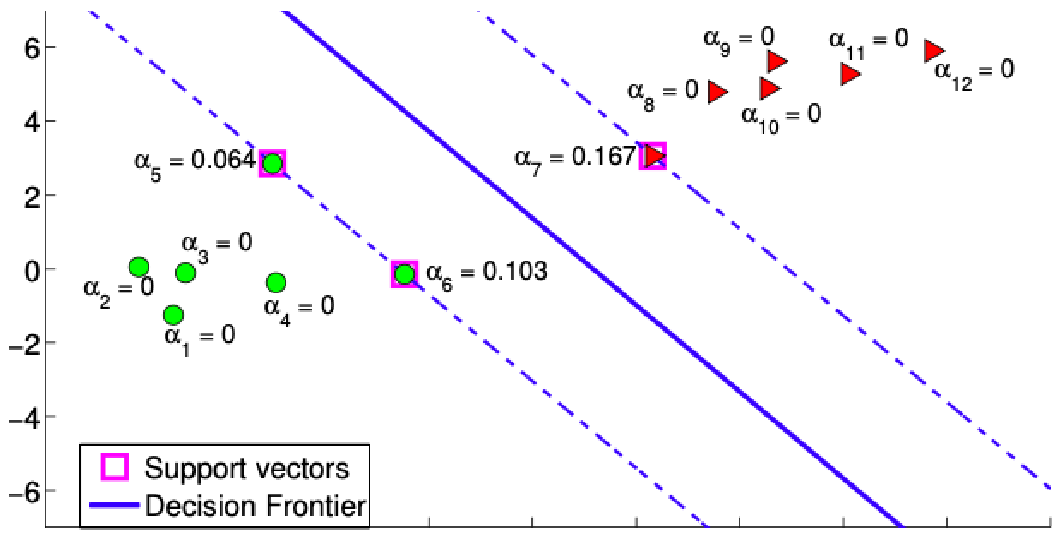
Problem Resolution

First, solve the dual to find the n parameters $\{\alpha_i\}_{i=1}^n$. Two types of parameters α_i are subsequently found :

- For a point x_j, if $y_j(w^T x_j + b) > 1$, then $\alpha_j = 0$.
- For a point x_i, if $y_i(w^T x_i + b) = 1$, then $\alpha_i \geq 0$.

The solution is then : $w = \sum_{i=1}^n \alpha_i y_i x_i$, where w is defined only for the points such that $y_i(w^T x_i + b) = 1$. These points are called **support vectors**.

Figure 4: SVM in the Linearly Separable Case



In practice (for the linearly separable case (see Figure 4)

- **Calculation of w**
  Use the data $D = \{(x_i, y_i)\}_{i=1}^n$ to solve the dual. We obtain the parameters $\{\alpha_i\}_{i=1}^n$. Therefore, deduce the solution $w = \sum_{i=1}^n \alpha_i y_i x_i$.
- **Calculation of b**
  The $\alpha_i > 0$ correspond to the support points that satisfy the relationship $y_i(w^T x_i + b) = 1$. Therefore, deduce the value of b.
- The **Score Function** is then $f(x) = w^T x + b = \sum_{i=1}^n \alpha_i y_i x_i^T x + b$.

3 SVM for Linearly Non-separable Problems

What happens if the data is not linearly separable?
Well, you will have to relax the constraints by allowing $y_i(w^T x_i + b) \geq 1 - \xi_i$, where $\xi_i \geq 0$ is the 'error' term, and include the sum of these 'errors' ($\sum_{i=1}^n \xi_i$) in the SVM problem.

Primal Problem

$$\begin{aligned}
\min_{w,b,\{\xi_i\}} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \\
\text{s.t.} \quad & y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \forall i = 1, \ldots, n \\
& \xi_i \geq 0, \quad \forall i = 1, \ldots, n
\end{aligned}$$

$C > 0$ is a regularization parameter (trade-off between error and margin), the value of which is to be determined by the use.

Dual Problem

The Lagrangian becomes $L(w, b, \xi, \alpha, \nu) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i(y_i(w^T x_i + b) - 1 + \xi_i) - \sum_{i=1}^{n}\nu_i\xi_i$.
The dual problem is then formulated as such :

$$\begin{aligned}
\max_{\{\alpha_i\}} \quad & \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j x_i^T x_j \\
\text{s.t.} \quad & 0 \geq \alpha_i \geq C, \quad \forall i = 1, \ldots, n \\
& \sum_{i=1}^{n}\alpha_i y_i = 0
\end{aligned}$$

Optimality Conditions of Stationarity

$$\frac{\partial L(w, b, \xi_i, \alpha)}{\partial b} = 0 \quad \text{and} \quad \frac{\partial L(w, b, \xi_i, \alpha)}{\partial w} = 0 \quad \text{and} \quad \frac{\partial L(w, b, \xi_i, \alpha)}{\partial \xi_k} = 0$$

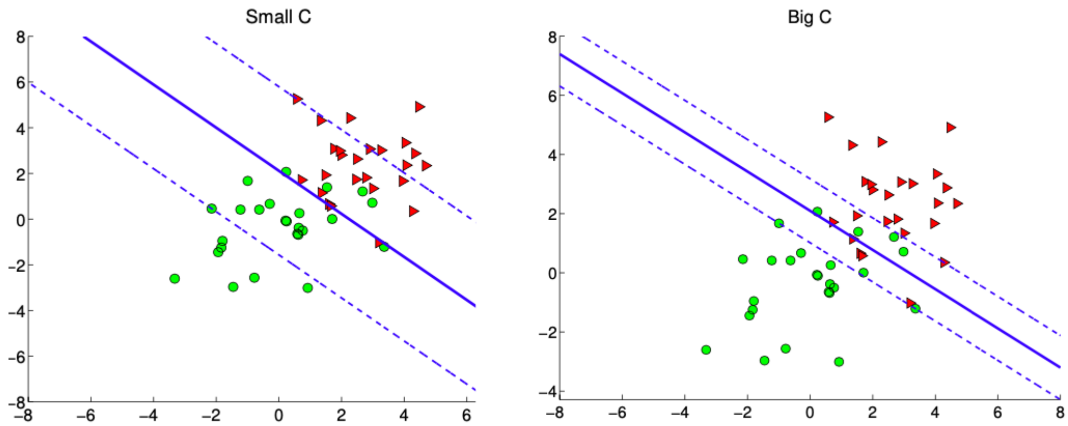
Which gives :

$$\sum_{i=1}^{n}\alpha_i y_i = 0 \qquad w = \sum_{i=1}^{n}\alpha_i y_i x_i \qquad C - \alpha_i - \nu_i = 0, \forall i = 1, \ldots, n$$

Theorem 2. ***Solution of a linear SVM: non-separable case***
Consider a non-separable linear SVM problem with the decision function $f(x) = w^T x + b$. The vector w is defined as $w = \sum_{i=1}^{n}\alpha_i y_i x_i$, where the coefficients α_i are solutions to the dual problem above.
What has changed? Nothing except the constraints on α_i which are now $0 \leq \alpha_i \leq C$.

In Figure 5, we resolve an SVM problem for $C = 0.01$ small, and $C = 1000$ large. The choice of C influences the solution : small C results in a large margin, while a large C results in a small margin.

Figure 5: SVM in the Linearly Separable Case



4 SVM in Practice

N.B.: some widespread SVM solvers that you can use are LibSVM[1] *and Scikit-Learn*[2]

In practice :

Input elements:

Labeled data: $\{(x_i, y_i) \in \mathbb{R}^d \times \{-1, 1\}\}_{i=1}^n\}$

Methodology:

1. Center the data: $\{x_i\}_{i=1}^n \leftarrow \{x_i - \bar{x}\}_{i=1}^n\}$
2. Set the parameter $C > 0$ for the SVM.
3. Use a solver to solve the dual problem and obtain $\alpha_i \neq 0$, the corresponding support points x_i, and the bias b.
4. Deduce the decision function: $f(x) = \sum_{i \in \text{SV}} \alpha_i y_i x_i^T x + b$.
5. Evaluate the generalization error of the obtained SVM (cross-validation, etc.).
6. Restart from step 2 if it is not satisfactory.

Model Selection : tuning of C:

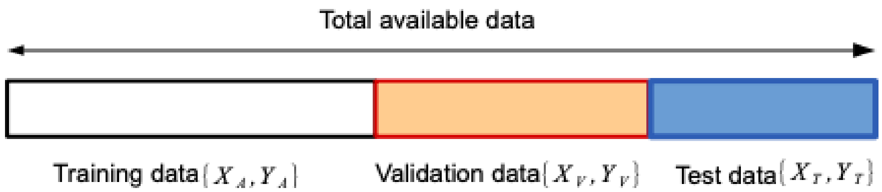
function $C \leftarrow \text{tuneC}(X, Y, \text{options})$

1. Split the data $(X_a, Y_a, X_v, Y_v) \leftarrow \text{SplitData}(X, Y, \text{options})$
2. For different values of C:
   - $(w, b) \leftarrow \text{TrainLinearSVM}(X_a, Y_a, C, \text{options})$

[1] `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`
[2] `http://scikit-learn.org/stable/modules/svm.html`

- error $\leftarrow$ EvaluateError(X_v, Y_v, w, b)

3. $C \leftarrow \text{argmin}_C \text{error}$



(a) Training set to calculate w and b

(b) Validation set to evaluate the classification error for different values of C

(c) Test set to evaluate the 'best model'

Figure 6: C Parameter Tuning Procedure

Relation between soft-SVM, Hinge loss, and Hinge loss perceptron :

Soft-SVM (SVM with slack variables(non-separable conditions)

$$\begin{cases} \min_{w,b,\{\xi_i\}} \quad \frac{1}{2}\|w\|^2 & + \quad C\sum_{i=1}^{n}\xi_i \\ y_i(<w, x_i> +b) & \geq \quad 1-\xi_i \\ \xi_i & \geq \quad 0 \end{cases}$$

The constraints on ξ_i are then : $\begin{cases} \xi_i & \geq \quad 0 \\ \xi_i & \geq 1 - y_i(<w, x_i> +b) = 1 - y_i s_i \text{ with } s_i =<w, x_i> +b' \end{cases}$
which is equivalent to : $\xi_i \geq max(0, 1 - y_i s_i)$.

Consider this optimization sub-problem :

$$\begin{cases} min & \sum_{i=1}^{n}\xi_i \\ \text{s.t.} & \xi \geq max(0, 1-y_i s_i) \end{cases} \xrightarrow{\text{solution}} \underbrace{\xi_i = \max(0, 1-y_i s_i)}_{\text{this is also the solution on } \xi_i \text{ to the original problem}}$$

The problem becomes : $\min_{w,b,\{\xi_i\}} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\max(0, 1 - y_i(<w, x_i> +b))$

$$\Leftrightarrow$$

$\min_{w,b,\{\xi_i\}} \frac{1}{2C}\|w\|^2 + \sum_{i=1}^{n} l^{hinge}(<w, x_i> +b, y_i)$ where $l^{hinge}(s_i, y_i) = \max(0, 1 - y_i s_i)$

SGD of soft-SVM on the objective function

$$\nabla_w\left(\frac{1}{2C}\|w\|^2 + \sum_{i=1}^{n} l^{hinge}(s_i, y_i)\right) = \frac{w}{C} + \sum_{i=1}^{n}\begin{cases} 0 & \text{if } y_i s_i > 1 \\ -y_i x_i & \text{otherwise} \end{cases}$$

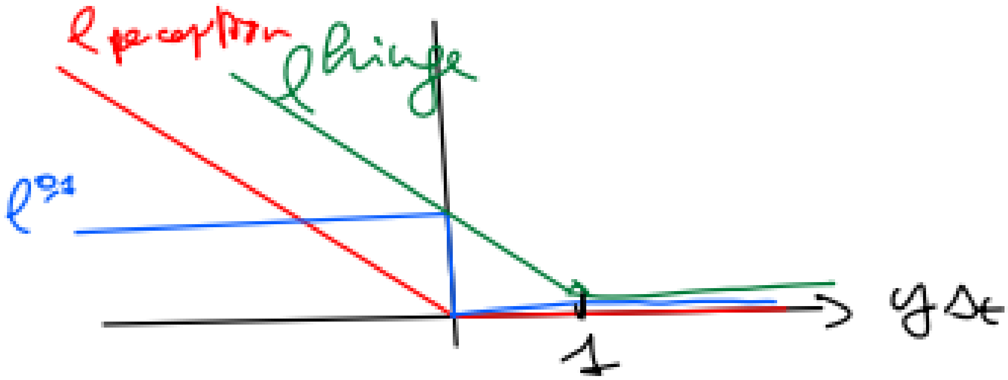
if $y_t < w_t, y_t >< 1$ **then**

  Update $w_{t+1} \leftarrow w_t + \alpha y_t x_t - \alpha \frac{w_t}{C}$

else

  Update $w_{t+1} \leftarrow w_t - \alpha \frac{w_t}{C}$

Figure 7: Graph of the Perceptron, the 0/1, and the Hinge loss



☞ As we can see in Figure 7, $l^{hinge}(\ldots) \geq l^{0,1}(\ldots)$.

5 Conclusions

In this lecture, we learned :

- To build an optimal hyperplane
- Maximizing the margin is the goal
- In-depth theoretical analysis reveals that maximizing the margin is equivalent to minimizing a bound on the generalization error
- The non-linear case, where a non-linear decision function is sought, can be handled through kernels
- Possible extension to the case of multiple classes
- Widely used classification algorithm in practice