# The Online Perceptron Algorithm And Linear Support Vector Machine

Scribe: Matteo Sammut

November 16, 2023

## 1 Linear Discrimination

### 1.1 Formulation

Let $D = \{(x_i, y_i) \in \mathcal{X} \times \{-1, 1\}\}_{i=1}^n$ be a set of labeled points. From $D$ we want to construct a function $f : \mathcal{X} \to \{-1, 1\}$ or $f : \mathcal{X} \to \mathbb{R}$ that predicts the class $-1$ or 1 of a point $x \in \mathcal{X}$.

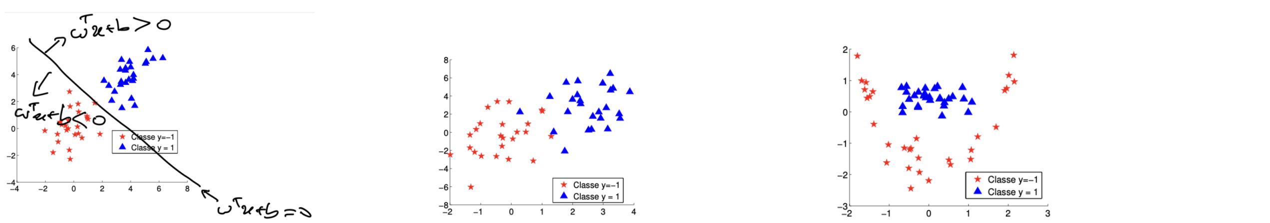
Let the input space be $\mathcal{X} = \mathbb{R}^d$. We can construct a scoring function: $f : \mathbb{R}^d \to \mathbb{R}$ such that:

$$f(x) = \begin{cases} f(x) < 0 & \text{assign } x \text{ to class } -1 \\ f(x) > 0 & \text{assign } x \text{ to class } 1 \end{cases}$$

A linear scoring function has the following expression: $f(x) = w^T x + b$, where $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$.

**Definition 1.1** *Linearly Separable Problem*
*The points $\{(x_i, y_i)\}$ are linearly separable if there exists a hyperplane that correctly discriminates the entire set of data. Otherwise, the points are non-linearly separable examples.*

Some examples are shown on the figure 1.




(a) Linearly separable

(b) Non-linearly separable

(c) Non-linearly separable

Figure 1: Linear separability examples

### 1.2 Linear Separator and Maximization of the Margin

**Proposition 1.0.1** *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 $d(x, H) = \frac{|w^T x + b|}{\|w\|} = \frac{|f(x)|}{\|w\|}$.*

**Proof 1.0.1** *Let's denote $x_p$ the projection of $x$ on the hyperplane, and suppose that $x - x_p$ is on the*

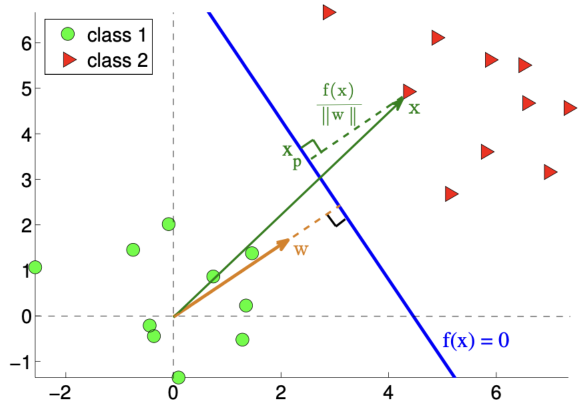



Figure 2: Distance from a Point to the Decision Boundary

*same direction than $w$. As we can see on the figure 2:*

$$
\begin{aligned}
x &= x_p + \frac{w}{\|w\|} \times d(x,H) \\
w^T x &= w^T x_p + w^T \frac{w}{\|w\|} \times d(x,H) \\
\|w\| \times d(x,H) &= w^T x - w^T x_p \\
\|w\| \times d(x,H) &= (w^T x + b) - (w^T x_p + b) \\
d(x,H) &= \frac{w^T x + b}{\|w\|}
\end{aligned}
$$

*If now we suppose that $x - x_p$ is on the opposite direction than $w$, we can conclude:*

$$d(x,H) = \frac{|w^T x + b|}{\|w\|}$$

**Definition 1.2** *Canonical Hyperplane*
*An hyperplane is said to be canonical with respect to the data $\{x_1, \ldots, x_N\}$ if $\min_i |w^T x_i + b| = 1$.*

**Definition 1.3** *Geometric Margin*
*The geometric margin is $M = \frac{2}{\|w\|}$*

**Definition 1.4** *Optimal Canonical Hyperplane*
*An optimal canonical hyperplane respects the following properties (cf figure 3:*

- *It maximizes the margin*
- *It correctly classifies each point: $\forall i, y_i f(x_i) \geq 1$*

## 1.3 Perceptron Algortihm

We first consider an homogeneous linear classifier: $f(x) = w^T x$. The perceptron algorithm can be written as follow:

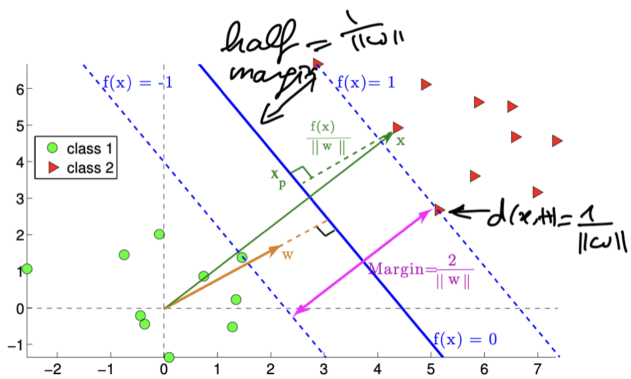



Figure 3: Example of an optimal canonical hyperplane

**Algorithm 1** Perceptron algorithm

$w_0 \leftarrow 0$
**for** $t = 1$ `to` $T$ **do**
    `receive` $x_t$
    `predict` $\hat{y_t} = sign(w_t^T x_t)$
    `receive` $y_t \in \{-1, 1\}$
    **if** $\hat{y_t} \neq y_t$ **then**
        $w_{t+1} \leftarrow w_t + y_t x_t$
    **else**
        $w_{t+1} \leftarrow w_t$
    **end if**
**end for**

**Theorem 1.1** *Block Norikoff*
*Assume* $\|x_t\| < R$ *for all* $t$ *and* $y_t \in \{-1, 1\}$.
*Assume there exists a canonical hyperplane* $w^\star$ *classyfing data perfectly, passing through the origin with half a margin* $\rho = \frac{1}{\|w^\star\|}$.
*Then, then number of mistakes of perceptron is at most of* $\frac{R^2}{\rho^2}$.

**Proof 1.1.1** *Step 1)*
*After an update (a prediction error),* $w_{t+1}$ *is more aligned' to* $w^\star$*:*

$$\langle w_{t+1}, w^\star \rangle = \langle w_t + y_t x_t, w^\star \rangle$$
$$\langle w_{t+1}, w^\star \rangle = \langle w_t, w^\star \rangle + y_t \langle x_t, w^\star \rangle$$

$w^\star$ *is a canonical hyperplane, then* $y_t \langle x_t, w^\star \rangle \geq 1$.

$$\langle w_{t+1}, w^\star \rangle \geq \langle w_t, w^\star \rangle + 1$$

*By unrolling we get:* $\langle w_t, w^\star \rangle \geq t_e$ *with* $t_e$ *the number of mistakes.*

*Step 2)*
*After an update (classification error) we have:*

$$\|w_{t+1}\|^2 = \langle w_t + y_t x_t, w_t + y_t x_t \rangle$$
$$\|w_{t+1}\|^2 = \|w_t\|^2 + 2y_t \langle w_t, y_t \rangle + \|y_t x_t\|^2$$

*The misclassification at this step leads to* $2y_t \langle w_t, y_t \rangle \leq 0$.*:*

$$\|w_{t+1}\|^2 \leq \|w_t\|^2 + R^2$$

*By unrolling we get:* $\|w_t\|^2 \leq t_e R^2$

*Step 3)*
*Using Cauchy-Scharwtz inequality:*

$$t_e \leq \langle w_t, w^\star \rangle \leq \|w_t\| \|w^\star\| \leq \sqrt{t_e} R \|w^\star\|$$
$$\implies \sqrt{t_e} \leq \frac{R}{\rho}$$
$$\implies t \leq \frac{R^2}{\rho^2}$$

### 1.3.1 Perceptron as a 'SGD' online learner

Perceptron algorithm can be rewrite as an SGD algorithm.
Let $S_t = w_t^T x_t$:

$$l(s_t, y_t) = \begin{cases} 0 & \text{if } y_t s_t \geq 0 \\ -y_t s_t & \text{otherwise} \end{cases}$$

Applying SGD algorithm here gives:

$$w_{t+1} \leftarrow w_t - \alpha \begin{cases} 0 & \text{if } y_t s_t \geq 0 \\ -y_t s_t & \text{otherwise} \end{cases}$$

Which is equivalent to the perceptron algorithm and $l^{perceptron}(s_t, y) = \max(0, 1 - y s_t)$. One can compare $l^{perceptron}$ and $l^{0,1}$ in Figure 4.

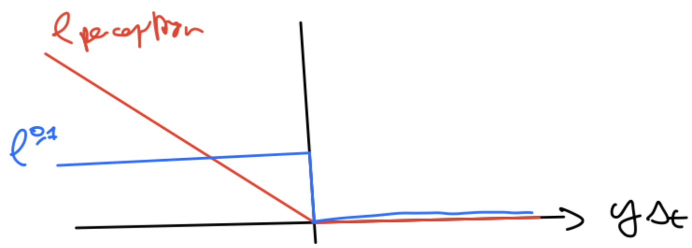



Figure 4: Comparison of $l^{perceptron}$ and $l^{0,1}$

### 1.3.2 Margin and Generalization Bound

Considering the VC generalization bound on a function class $\mathcal{H}$, with probability $1 - \delta$:

$$R(h) \leq R_{emp}(h) + C\sqrt{\frac{D(log(2N/D) + 1 + log(4\delta)}{N}}$$

where D is the VC dimension of $\mathcal{H}$.

If we consider $\mathcal{H}$ as the class of linear function $f(x) = w^T x + b$ with a margin $\rho$ to the training set, we can born the relative VC dimension as follow:

$$D \leq 1 + \min(d, \frac{R^2}{\rho^2})$$

where $R$ is the radius of a ball containing the training data.

The main idea here is that increasing the margin allows to reduce the VC dimension D. Hence, a large margin is a good way to prevent from overfitting.

# 2 Solving the SVM problem

## 2.1 Linearly separable problems

We first suppose in this section that the points $D = \{(x_i, y_i) \in \mathbb{R}^d \times \{-1, 1\}\}_{i=1}^n$ are linearly separable.

Our objective is to find a decision function $f(x) = w^T x + b$ that maximizes the margin and correctly discriminates the points in $D$.

The formulation of this problem is given as follow :

$$\begin{aligned} \min_{w \in \mathbb{R}, b \in \mathbb{R}} \quad & \frac{1}{2}\|w\|^2 \\ \text{s.t.} \quad & y_i(w^T x_i + b) \geq 1 \ \forall i = 1, ..., n \end{aligned}$$

The Lagrangian of this problem is given by:

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i(y_i(w^T x_i + b) - 1)$$

The stationary conditions gives us :

- $\frac{\partial L(w,b,\alpha)}{\partial b} = 0$ • $\frac{\partial L(w,b,\alpha)}{\partial w} = 0$

wich can be written as:

- $\sum_{i=1}^n \alpha_i y_i = 0$ • $w = \sum_{i=1}^n \alpha_i y_i x_i$

By substituting into the Lagrangian, the dual problem is written as:

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

The condition of complementary slackness is written as:

$$\alpha_i(y_i(w^T x_i + b) - 1) = 0$$

By solving the dual problem to find the $n$ parameters $\{\alpha_i\}$, two cases are obtained:

- 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 $\alpha_i \geq 0$

Hence, the solution $w = \sum_{i=1}^n \alpha_i y_i x_i$ is uniquely defined by points such as $y_i(w^T x_i + b) = 1$. This is what we called the **support vectors**. In other words, the hyperplane is entirely defined by a linear combination of support vectors (cf figure 5)

## 2.2 Non-linearly separable problems

Linearly separable problems is a too restrictive hypothesis. One way to consider non-linearly separable problems, is to allow misclassification.

- Relaxing $y_i(w^T x_i + b) = \geq 1$

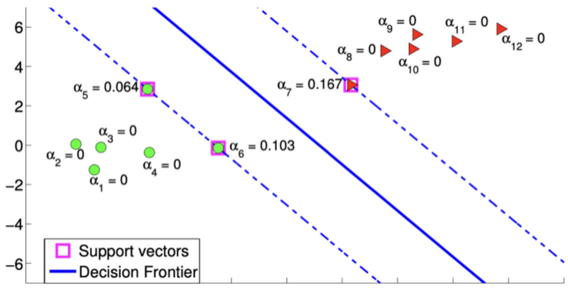



Figure 5: Example of the result of the SVM problem. Here the hyperplane is defined by this linear combination: $w = 0.167x_7 - 0.064x_5 - 0.103x_6$. Here, $x_5$, $x_6$ and $x_7$ are the support vectors.

- Accept $y_i(w^T x_i + b) \geq 1 - \epsilon_i$ with $\epsilon_i$ the error term.

- Include the sum of errors $\sum_{i=1}^{n} \epsilon_i$ in the SVM problem.

The non-linearly separable SVM problem can be formulize as follow :

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

where $C > 0$ is a regularisation parameter defined by the user.

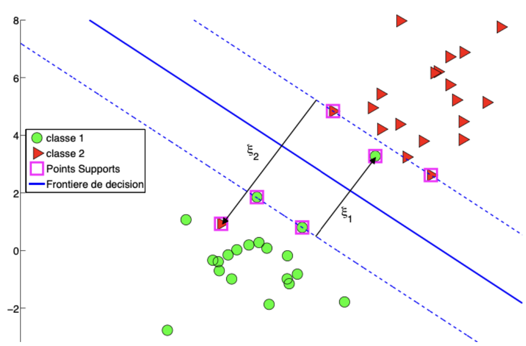



Figure 6: Example of a non-linearly separable SVM problem. The support vectors are indicates by the purple bounding boxes

We consider the Lagrangian:

$$L(w, b, \epsilon, \alpha, \nu) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \epsilon_i - \sum_{i=1}^{n} \alpha_i(y_i(w^T x_i + b) - 1 + \epsilon_i) - \sum_{i=1}^{n} \nu_i \epsilon_i$$

where $\alpha_i$, $\nu_i \geq 0 \ \forall i = 1, ..., n$ .

The stationary conditions gives us :

- $\frac{\partial L(w,b,\epsilon_i,\alpha)}{\partial b} = 0$
- $\frac{\partial L(w,b,\epsilon_i,\alpha)}{\partial w} = 0$
- $\frac{\partial L(w,b,\epsilon_i,\alpha)}{\partial \epsilon_k} = 0$

wich can be written as:

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

By substituting into the Lagrangian, the dual problem is written as:

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

**Theorem 2.1** *Solution of a linear SVM: no-separable case*
*Consider a linear non-separable SVM problem with a 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 $\alpha_i$ are the solutions of the dual problem above.*

Compared to the previous separable case, very few things have changed. The condition on $\alpha_i$ is now different since we have $0 \leq \alpha_i \leq C$. The influence of the $C$ parameter is shown on the figure 7.

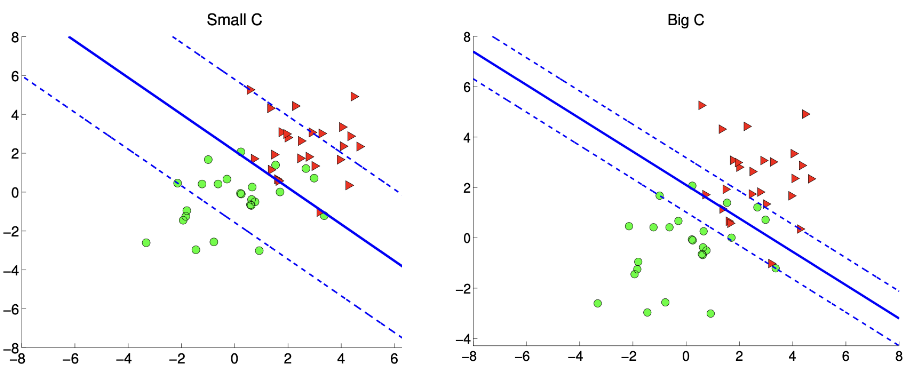



Figure 7: Example of the influence of $C$ parameter. If $C$ is small (left) then the margin is big and we accept a lot of errors. If $C$ is big (right) then the margin is small and we accept a small amount of errors.

In practice, given labelled data $\{(x_i, y_i) \in \mathbb{R}^d \times \{-1, 1\}\}_{i=1}^n$, the methodology is as follow :

1. Centered the data
2. Choose parameter $C > 0$ of SVM
3. Use a solver to solve the dual problem an obtain the $\alpha_i \neq 0$, corresponding support vectors $x_i$, and the bias $b$
4. Evaluate the generalization error of the obtained SVM model (cross validation...)
Restart the procedure from step 2 if needed.

# 3 Relation Between soft SVM, Hinge-loss and Hinge-loss Perceptron

The soft-SVM optimization problem is written as follow :

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

We writing the constraints on $\xi_i$ with $s_i = \langle w, x_i \rangle + b$ we obtain :

$$\xi_i \geq \max(0, 1 - y_i s_i)$$

The soft SVM problem become :

$$\min_{w \in \mathbb{R}, b \in \mathbb{R}, \{\epsilon_i\}} \quad \frac{1}{2}\|w\|^2 + C \sum_{i=1}^{n} \max(0, 1 - y_i(\langle w, x_i \rangle + b))$$

which is equivalent to:

$$\min_{w \in \mathbb{R}, b \in \mathbb{R}, \{\epsilon_i\}} \quad \frac{1}{2}\|w\|^2 + C \sum_{i=1}^{n} l^{\text{hindge}}(\langle w, x_i \rangle + b, y_i)$$

where $l^{\text{hindge}}(s_i, y_i) = \max(0, 1 - y_i s_i)$.

One can compare visually the difference between $l^{\text{hindge}}$, $l^{0,1}$ and $l^{\text{perceptron}}(s_t, y_t) = \max(0, -y_t s_t)$ in Figure 8. Notice that $l^{\text{hindge}} \geq l^{0,1}$.

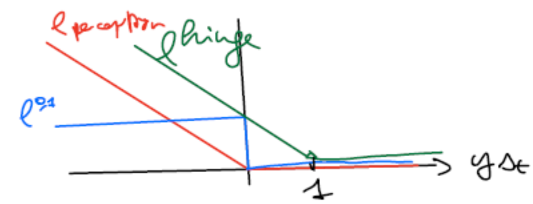



Figure 8: Losses comparison

# 4 Conclusion

- Construction of an optimal hyperplane for Margin Maximization
- A thorough theoretical analysis shows 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 addressed using kernels
- Generalization is possible to cases with multiple classes
- This is a classification algorithm widely used in practice...