1 SVM (Support Vector Machines):

Constructing optimally seperating hyperplane between two classes for seperation. We review the mathematical basis of SVM and how to construct it. We will also discuss the application of SVM in the context of classification. We will also discuss the application of SVM in the context of regression and generalizing to the nonseperable case where the classes may not be seperable by a linear boundary.

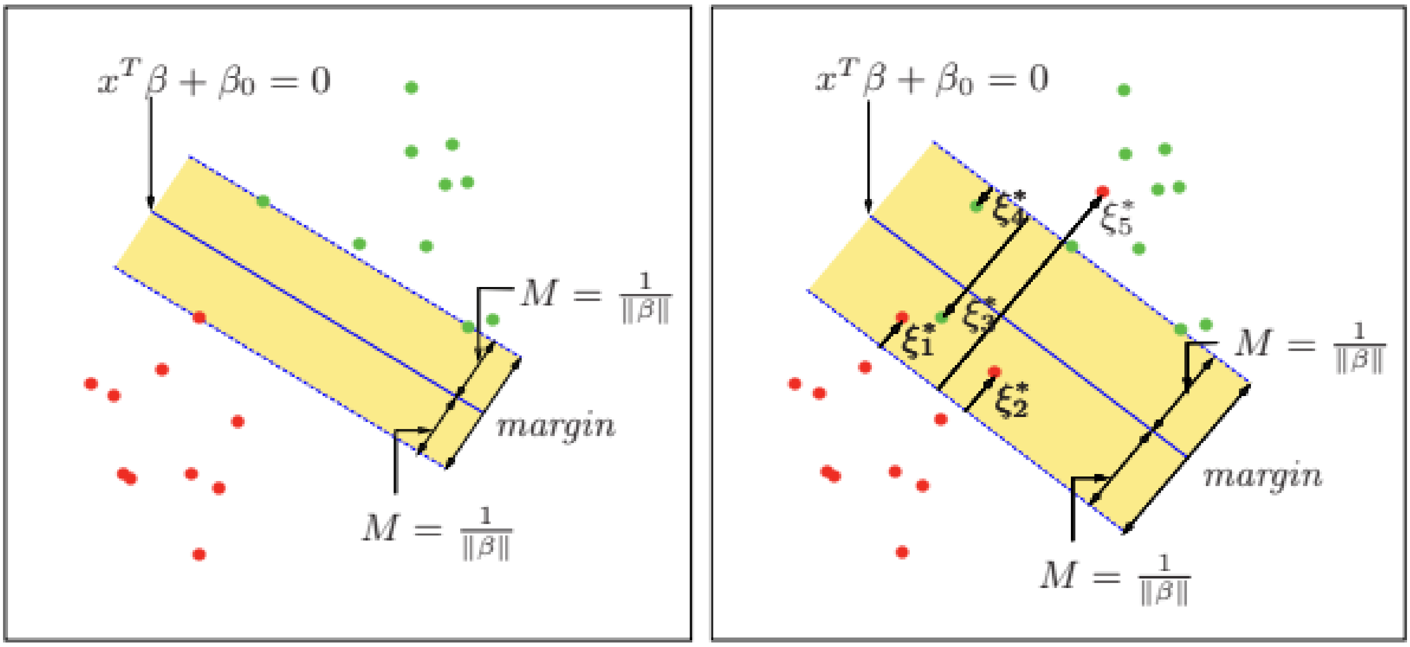

Figure 1: FIGURE SVM

Our training data consists of two classes of points of N pairs of $(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)$ with $x_i \in \mathbb{R}^P$ and $y_i \in \{-1, 1\}$. Define a hyperplane by

$$\{x : f(x) = x^T\beta + \beta_0 = 0\},$$

where β is an unit vector: $||\beta||$ = 1 and β_0 is the intercept. A classification rule induced by $f(x)$ is

$$G(x) = sign[x^T\beta + \beta_0]$$

Now we can find $x^T\beta + \beta_0 = 0$ and $y_i f(x_i) > 0$ for all i. The optimization problem

$$max_{\beta,\beta_0,\|\beta\|=1} M$$

Subject to $y_i f(x_i) \geq M, \forall i \in \{1, 2, ..., N\}$

more conveniently written as

$$min_{\beta,\beta_0} \|\beta\|$$

Subject to $y_i f(x_i) \geq 1, \forall i \in \{1, 2, ..., N\}$

This convex optimization problem is basically quadratic programming with the constraint that the hyperplane is a separating hyperplane.

2 Linear SVM:

From the figure above we get three hyperplanes as:

$$x : f(x) = x^T\beta + \beta_0 = -1$$

$$x : f(x) = x^T\beta + \beta_0 = 0$$

$$x : f(x) = x^T\beta + \beta_0 = 1$$

The distance between the hyperplanes is (under consideration of the of $\|\beta\| \neq 1$): Assuming $\beta = w$ and $\beta_0 = b$, we get the disance as

$$D = \frac{1-b}{\|w\|} - \frac{-1-b}{\|w\|}$$

$$D = \frac{2}{\|w\|}$$

where D is nothing but twice of the margin. So margin can be written as $\frac{1}{\|w\|}$. Then the SVM objective is boiled down to the fact of minimizing the term $\frac{1}{\|w\|}$.

$$max\frac{1}{\|w\|}$$

or $min\|w\|$

As, l2 optimization ar often more stable than l1 optimization

$$min\frac{\|w\|^2}{2}$$

such that $y_i(w.x_i + b) \geq 0 \forall i \in \{1, 2, ..., N\}$

3 Nonlinear SVM:

3.1 Critical points:

Example:

Find critical points of $f(x, y) = x^2 + y^2$

Sol: $\frac{\partial f}{\partial x} = 2x = 0, \frac{\partial f}{\partial y} = 2y = 0$

so, minimum of $f(x, y)$ is at $(0, 0)$

4 Constrained Optimization:

Similar example:

Find critical points of $f(x, y) = x^2 + y^2$ such that $x + y = 3:$

Sol:$l = x^2 + y^2 - \lambda(x + y - 3)\frac{\partial l}{\partial x} = 2x - \lambda = 0, \frac{\partial l}{\partial y} = 2y - \lambda = 0, \frac{\partial l}{\partial \lambda} = x + y - 3 = 0$

so, minimum of $f(x, y)$ is at $(\frac{3}{2}, \frac{3}{2}) and \lambda = 3$

Suppose the classes is now overlapping in feature space.The way to deal with this is to maximize M, but allow some points to be on the wrong side of the margin. Define the slack variables as $\zeta = \{\zeta_1, \zeta_2, ..., \zeta_N\}$. The two natural ways to modify the constraint are

$$y_i(x_i^T\beta + \beta_0) \geq M - \zeta_i, \forall i \in \{1, 2, ..., N\},$$

$$or$$

$$y_i(x_i^T\beta + \beta_0) \geq M(1 - \zeta_i)$$

4.1 Computating the Support Vector Classifier:

The problem is quadratic with linear inequality constraints. We can solve this problem using the quadratic programming algorithm. It is conventionally re-express in the form

$$min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2 + C\sum_{i=1}^{N}\zeta_i$$

$$\text{subject to } \zeta_i \geq 0, y_i(x_i^T\beta + \beta_0) \geq 1 - \zeta_i, \forall i \in \{1, 2, ..., N\}$$

Now the Langrangian (primal) function is

$$L_p = \frac{1}{2}\|\beta\|^2 + C\sum_{i=1}^{N}\zeta_i - \sum_{i=1}^{N}\alpha_i[y_i(x_i^T\beta + \beta_0) - (1 - \zeta_i)] - \sum_{i=1}^{N}\mu_i\zeta_i$$

which we minimize w.r.t ζ_I,β and β_0.we get,

$$\beta = \sum_{i=1}^{N}\alpha_i y_i x_i,$$

$$0 = \sum_{i=1}^{N}\alpha_i y_i,$$

$$\alpha_i = C - \mu_i$$

After substituting it into the Lagrangian (Dual), we get

$$L_D = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j x_i x_j$$

we see that the solution $f(x)$ is given by

$$f(x) = h(x)^T\beta + \beta_0 = \sum_{i=1}^{N}\alpha_i y_i\langle h(x)h(x_i)\rangle + \beta_0$$

So, both the requires the inner product of the kernel function with the data points.In fact we need not specify the transoformation $h(x)$ explicitly, but requires only the knowledge of the kernel function.

$$K\langle x, x_i\rangle = \langle h(x), h(x')\rangle$$

There are popular choices of kernel functions. These are:

- $K(x, x') = (1 + \langle x, x'\rangle)^d$,
- $K(x, x') = e^{-\gamma\|x - x\|^2}$,

- $K(x, x') = tanh(\kappa_1 \langle x, x' \rangle + \kappa_2)$

Considering an example,

$$K(x, x') = (1 + \langle x, x' \rangle)^2$$

$$(1 + x_1 x'_1 + x_2 x'_2)^2$$

$$1 + 2x_1 x'_1 + 2x_2 x'_2 + (x_1 x'_1)^2 + (x_2 x'_2)^2 + 2x_1 x_2 x'_1 x'_2$$

Then M = 6 and if we choose $h_1(x) = 1$,$h_2(x) = \sqrt{2}x_1$,$h_3(x) = \sqrt{2}x_2$,$h4(x) = x_1$,$h_5(x) = x_1^2$,$h_6(x) = x_2^2$ then the kernel function is $\langle h(x), h(x') \rangle$

$$\hat{f}(x) = \sum_{i=1}^{N} \hat{\alpha}_i y_i \langle h(x), h(x_i) \rangle + \hat{\beta}_0$$

5 The SVM as penalization Method:

with $f(x) = h(x)^T$ β+β_0,consider the following example:

$$min_{\beta,\beta_0} \sum_{i=1}^{N} [1 - y_i f(x_i)] + \frac{\lambda}{2} \|\beta\|^2$$

5.1 Table for all the methods:

Loss Function	$L[y, f(x))]$	Minimizing Function
Binomial Deviance	$\log[1 + e^{-yf(x))}$	$f(x) = \log \frac{Pr(Y=+1\|x)}{pr(Y=-1\|x)}$
SVM hinge loss	$[1 - f(x)]_+$	$f(x) = sign[Pr(Y = +1\|x) - \frac{1}{2}]$
Squared Error loss	$[y - f(x)]^2 = [1 - yf(x)]^2$	$f(x) = 2Pr(Y = +1\|x) - 1$