SVM: Kernels

Anca Ralescu
Machine Learning and Computational Intelligence Laboratory
Department of Computer Science
University of Cincinnati
Cincinnati, Ohio 45221, USA
ancaralescu@gmail.com

October 27, 2015

Recall that the decision rule for a linearly separable training set is

$$D_{\mathbf{w},b}(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$$

By mapping into a new feature space: $\mathbf{x} \mapsto \phi(\mathbf{x})$ we obtain

$$D_{\mathbf{w},b}(\mathbf{x}) = \mathbf{w} \cdot \phi(\mathbf{x}) + b$$

In the dual form

$$\sum_{i=1}^{n} \alpha_i y_i \mathbf{x_i} \cdot \mathbf{x} + b \text{ becomes } \sum_{i=1}^{n} \alpha_i y_i \phi(\mathbf{x_i}) \cdot \phi(\mathbf{x}) + b$$

or using $K(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x}) \cdot \phi(\mathbf{z})$ we obtain

$$\sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x_i}, \mathbf{x}) + b$$

How do we find/construct kernels?

- The dot product is a particular case of kernel: ϕ is the identity map;
- $K(\mathbf{x}, \mathbf{z}) = (x \cdot \mathbf{z} + c)^d$ Let us look at some particular cases of c and **especially** d **as it is** d **which determines the dimension of the new feature space**. I take $c = 1$. Assume the dimension of the original feature space is 2, that is $\mathbf{x} = (x_1, x_2)$
  - $d = 2$. Then

$$\begin{aligned} K(\mathbf{x}, \mathbf{z}) &= (\mathbf{x} \cdot \mathbf{z} + 1)^2 = \\ & (\mathbf{x} \cdot \mathbf{z})^2 + 2(\mathbf{x} \cdot \mathbf{z}) + 1 = \\ & ((x_1, x_2) \cdot (z_1, z_2))^2 + 2((x_1, x_2) \cdot (z_1, z_2)) + 1 = \\ & [x_1 z_1 + x_2 z_2]^2 + 2[x_1 z_1 + x_2 z_2] + 1 \\ & (x_1 z_1)^2 + 2x_1 z_1 x_2 z_2 + (x_2 z_2)^2 + 2(x_1 z_1) + 2(x_2 z_2) + 1 = \\ & (x_1)^2 (z_1)^2 + (\sqrt{2} x_1 x_2)(\sqrt{2} z_1 z_2) + (x_2)^2 (z_2)^2 + (\sqrt{2} x_1)(\sqrt{2} z_1) + (\sqrt{2} x_2)(\sqrt{2} z_2) + 1 \end{aligned}$$

  which is $\mathbf{X} \cdot \mathbf{Z}$ where

$$\mathbf{x} = (x_1, x_2) \mapsto \mathbf{X} = (x_1^2, \sqrt{2} x_1 x_2, x_2^2, \sqrt{2} x_1, \sqrt{2} x_2, 1)$$

  That is, ϕ maps $\mathbf{x} \in \Re^2$ into $\Re^6$.

The examples above show that a polynomial of the dot product is a kernel. Immediately it follows that a polynomial of a kernel is a kernel. Why? Because it will be a polynomial of the dot product! Let $p_k(u) = a_k u^k + a_{k-1}u^{k-1} + \cdots + a_0$ denote a polynomial of degree k. Then

- If K_1, and K_2 are each polynomials of the dot product then $K = K1 * K2$ is also a kernel for any operator $*$ such that $K_1 * K_2$ is a polynomial of the dot product!
- the composition $p_k \circ p_m$ is a polynomial of degree km: $(u^i)^j = u^{ij}$. Thus if K is a kernel, then $p_k(K)$ is a kernel for any $k >= 1$.

How can we construct other kernels from the model suggested above?

Theorem 1 *let $K_i, i = 1, 2$ be kernels over the same feature space $A \in \Re^n$, $a > 0, f : A \longrightarrow \Re$ and $\phi : X \longrightarrow \Re^m$ (usually $m >> n$), with kernel K_3. Then the following are also kernels:*

1. $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z}) + K_2(\mathbf{x}, \mathbf{z})$
2. $K(\mathbf{x}, \mathbf{z}) = aK_1(\mathbf{x}, \mathbf{z})$
3. $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z})K_2(\mathbf{x}, \mathbf{z})$
4. $K(\mathbf{x}, \mathbf{z}) = f(\mathbf{x})f(\mathbf{z})$
5. $K(\mathbf{x}, \mathbf{z}) = K_3(\phi(\mathbf{x}), \phi(\mathbf{z}))$

Proof

The proof is quite easy: (1) and (2) follow from the argument about polynomials above. For (3) - (5) use the particular case $n = 2$ and work out the formulae.

An immediate consequence of this theorem is the following

Corollary 1 *If $K_1(\mathbf{x}, \mathbf{z})$ is a kernel, p a polynomial with positive coefficients, then the following are also kernels:*

1. $K(\mathbf{x}, \mathbf{z}) = e^{K_1(\mathbf{x}, \mathbf{z})}$
2. $K(\mathbf{x}, \mathbf{z}) = e^{-\frac{\|\mathbf{x}-\mathbf{z}\|}{\sigma^2}}$

Proof

Part (1): use the fact that the exponential is a limit of polynomials with positive coefficients ($e^x = \sum_{n\geq 0} \frac{x^n}{n!}$)

Part (2): use $\|\mathbf{x} - \mathbf{z}\|^2 = \|\mathbf{x}\|^2 + \|\mathbf{z}\|^2 - 2(\mathbf{x} \cdot \mathbf{z})$. Then

$$e^{-\frac{\|\mathbf{x}-\mathbf{z}\|}{\sigma^2}} = \underbrace{e^{-\frac{\|\mathbf{x}\|}{\sigma^2}}}_{\text{real-valued function of } \mathbf{x}} \underbrace{e^{-\frac{\|\mathbf{z}\|}{\sigma^2}}}_{\text{real-valued function of } \mathbf{z}} \underbrace{e^{2\frac{\mathbf{x}\cdot\mathbf{z}}{\sigma^2}}}_{\text{part (1) of the corollary}}$$

1 Working in the feature space

An interesting point is that we can calculate distances in the (new) feature space directly.

We use $\phi(\mathbf{x})$ to represent the image of $\mathbf{x}$, where the mapping ϕ is not known.

$$\phi(\mathbf{x}) = (\phi_1(\mathbf{x}), \ldots, \phi_i(\mathbf{x}), \ldots)$$

$$\phi(\mathbf{X}) = \{\phi(\mathbf{x}) \mid \mathbf{x} \in X\}$$

Let P be a linear combination of points in $\phi(\mathbf{X})$, that is

$$P = \sum_{i=1}^{n} p_i \phi(x_i)$$

Then we can represent P as

$$P = (p_1\phi(\mathbf{x_1}), \ldots, p_n\phi(\mathbf{x_n}))$$

Let Q be another such point: linear combination of points in $\phi(\mathbf{X})$. That is

$$Q = (q_1\phi(\mathbf{z_1}), \ldots, q_n\phi(\mathbf{z_k}))$$

Let $F = co(\phi(\mathbf{X}))$ the space of linear combinations of points in $\phi(\mathbf{X})$.
The dot product in F, denoted by $\cdot_F$ is then

$$P \cdot_F Q = \sum_{i=1}^{n}\sum_{j=1}^{k} p_i q_j \phi(\mathbf{x_i}) \cdot \phi(\mathbf{z_j}) = \sum_{i=1}^{n}\sum_{j=1}^{k} p_i q_j K(\mathbf{x_i}, \mathbf{z_j}) \tag{1}$$

Then

$$\begin{aligned} \underbrace{\|P - Q\|^2}_{\text{in } F} &= (P-Q) \cdot_F (P-Q) \\ &= \textstyle\sum_{i=1}^{n}\sum_{j=1}^{k} p_i p_j K(\mathbf{x_i}, \mathbf{x_j}) - 2\sum_{i=1}^{n}\sum_{j=1}^{k} p_i q_j K(\mathbf{x_i}, \mathbf{z_j}) + \sum_{i=1}^{n}\sum_{j=1}^{k} q_i q_j K(\mathbf{z_i}, \mathbf{z_j}) \end{aligned}$$

What can be computed in the feature space

Let us see what we can compute in the feature space directly from kernels, that is without making use of the actual mapping $\mathbf{Q} = (\phi_1, \ldots, \phi_m)$, where m is the dimension of the original space.

Norm of linear combinations of points in the feature space

It follows from (1) that

$$\|\mathbf{P}\|_F = \mathbf{P} \cdot_\mathbf{F} \mathbf{P} =< \sum_{i=1}^{n} p_i\phi(\mathbf{x_i}), \sum_{i=j}^{n} p_j\phi(\mathbf{x_j}) >= \sum_{i,j=1}^{n} p_i p_j \mathbf{K}(\phi(\mathbf{x_i}), \phi(\mathbf{x_j}))$$

Distances between feature vectors

We start with $\mathbf{x}, \mathbf{z}$, and let $\phi(\mathbf{x}), \phi(\mathbf{z})$ denote their image in the feature space. Then

$$\begin{aligned} dist(\phi(\mathbf{x}), \phi(\mathbf{z})) &= \|\phi(\mathbf{x}) - \phi(\mathbf{z})\| \\ &= < \phi(\mathbf{x}) - \phi(\mathbf{z}), \phi(\mathbf{x}) - \phi(\mathbf{z}) > \\ &= < \phi(\mathbf{x}), \phi(\mathbf{x}) > -2 < \phi(\mathbf{x}), \phi(\mathbf{z}) > + < \phi(\mathbf{z}), \phi(\mathbf{z}) > \\ &= \mathbf{K}(\mathbf{x}, \mathbf{x}) - 2\mathbf{K}(\mathbf{x}, \mathbf{z}) + \mathbf{K}(\mathbf{z}, \mathbf{z}) \end{aligned}$$

Use these to calculate the norm of the center of mass (average) in the feature space

Recall that in the 1-dimensional case, given a sample of data $a_1, \ldots, a_n$, the sample mean (average), $\overline{a}$, satisfies the following

$$\overline{a} = argmin_X \sum_{i=1}^{n} [a_i - X]^2$$

and

$$\overline{a} = \frac{1}{n}\sum_{i=1}^{n} a_i$$

Let us now see what can we say/do about the mean of points in the feature space. Let $\phi(\mathbf{x_1}), \ldots, \phi(\mathbf{x_n})$ and the equation

$$g(\mathbf{\Phi}) = \sum_{i=1}^{n} \|\phi(\mathbf{x_i}) - \mathbf{\Phi}\|^2,$$

for some $\mathbf{\Phi}$ in the feature space. We want to find $\mathbf{\Phi}$ which minimizes g. Rewrite g as

$$g(\mathbf{\Phi}) = \sum_{i=1}^{n} \{\mathbf{K}(\mathbf{x_i}, \mathbf{x_i}) - 2 < \mathbf{\Phi}(\mathbf{x_i}), \mathbf{\Phi} > + < \mathbf{\Phi}, \mathbf{\Phi} >\} \tag{2}$$

Assume that $\mathbf{\Phi} = (\mathbf{\Phi}_1, \ldots, \mathbf{\Phi}_h)$ where h denotes the dimension ($h >> m$) of the feature space.

Then (14) becomes

$$g(\mathbf{\Phi}) = \sum_{i=1}^{n} \left\{ \mathbf{K}(\mathbf{x_i}, \mathbf{x_i}) - 2 \sum_{l=1}^{h} \mathbf{\Phi}(\mathbf{x_i})_l \mathbf{\Phi}_l + \sum_{l=1}^{h} \mathbf{\Phi}_l^2 \right\} \tag{3}$$

Take the partial derivatives with respect to $\mathbf{\Phi}_j$, set equal to zero and solve:

$$\frac{\delta g(\mathbf{\Phi})}{\delta \mathbf{\Phi}_j} = -2 \sum_{i=1}^{n} \phi(\mathbf{x_i})_j + 2\mathbf{\Phi}_j = 0$$

Therefore, $\mathbf{\Phi}_j = \frac{1}{n} \sum_{i=1}^{n} \phi(\mathbf{x_i})_j$. Let $\overline{\mathbf{\Phi}} = (\mathbf{\Phi}_1, \ldots, \mathbf{\Phi}_h)$. Then $\overline{\mathbf{\Phi}} = \frac{1}{n} \sum_{i=1}^{n} \phi(\mathbf{x_i})$. $\overline{\mathbf{\Phi}}$ is the point where g attains its minimum. Why? Note that $\mathbf{\Phi}$ is NOT necessarily the image through ϕ of a point in the original feature space. Why? Suppose it always is such an image. Then it follows that ϕ is linear which usually is not the case.

Exercise

Let $K(\mathbf{x}, \mathbf{z}) = e^{-\frac{\|\mathbf{x}-\mathbf{z}\|}{\sigma^2}}$ be the Gaussian kernel and let $K_1(\mathbf{x}, \mathbf{z})$ be any kernel on the feature space $X \times X$ for some input space X. How can one compute a Gaussian kernel of the features defined implicitly by K_1 and therefore use this as a kernel on $X \times X$?

Centering in the feature space

Centering of data is the procedure according to which the data is mapped into a new set whose mean/center of mass is 0. The usual way to accomplish is is by subtracting the mean of the data before centering.

In other words, $\{x_1, \ldots, x_n\}$ with mean $\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ is mapped into $\{x'_1, \ldots, x'_n\}$, where $x'_i = x_i - \overline{x}$.

Let us see what does it mean to center the data in the feature space. We create the data $\phi'(\mathbf{x}) = \phi(\mathbf{x}) - \overline{\mathbf{\Phi}} = \frac{1}{n} \sum_{1}^{n} \phi(\mathbf{x_i}$.

The kernel, $\mathbf{K}'$ in the transformed space is then

$$\begin{aligned} \mathbf{K}'(\mathbf{x}, \mathbf{z}) &= \langle \phi'(\mathbf{x}), \phi'(\mathbf{z}) \rangle = \left\langle \phi(\mathbf{x}) - \tfrac{1}{n} \textstyle\sum_{1}^{n} \phi(\mathbf{x_i}), \phi(\mathbf{z}) - \tfrac{1}{n} \textstyle\sum_{1}^{n} \phi(\mathbf{x_i}) \right\rangle \\ &= \cdots = \\ &= \mathbf{K}(\mathbf{x}, \mathbf{z}) - \tfrac{1}{n} \textstyle\sum_{i=1}^{n} \mathbf{K}(\mathbf{x}, \mathbf{x_i}) - \tfrac{1}{n} \textstyle\sum_{i=1}^{n} \mathbf{K}(\mathbf{z}, \mathbf{x_i}) + \tfrac{1}{n^2} \textstyle\sum_{i,j=1}^{n} \mathbf{K}(\mathbf{x_i}, \mathbf{x_j}) \end{aligned} \tag{4}$$

So, what does (4) say? It tells us how to calculate the kernel in the feature space when the data is centered in the feature space.

The smallest hypersphere containing a set of points

We have a set of points $\mathbf{S} = \{x_1, \ldots, x_n\}$, and a kernel K corresponding to some mapping $\phi : \mathbf{S} \subseteq \mathcal{X} \longrightarrow F$:

$$K(\mathbf{x}, \mathbf{y}) =< \phi(\mathbf{x}), \phi(\mathbf{y}) >$$

We want to find the smallest hypersphere containing $\mathbf{S}$, that is its center and its radius.

Based on the above, we have

- $\|\phi(\mathbf{x_i}) - \mathbf{c}\|$ is the distance from the image of a data point $\mathbf{x_i}$ to a point $\mathbf{c}$.
- The largest distance is

$$\max_{i=1,\ldots,n} \|\phi(\mathbf{x_i}) - \mathbf{c}\| \tag{5}$$

- We want to find $\mathbf{c}^*$ that minimizes (5), that is

$$\mathbf{c}^* = \text{argmin}_{\mathbf{c}} \max_{i=1,\ldots,n} \|\phi(\mathbf{x_i}) - \mathbf{c}\| \tag{6}$$

Put another way, if we denote by

$$r(\mathbf{c}) = \max_{i=1,\ldots,n} \|\phi(\mathbf{x_i}) - \mathbf{c}\|$$

we want to find the minimum of $r(\mathbf{c})$ and we denote by $\mathbf{c}^*$ the point where this minimum is attained.

We can rewrite this as an optimization problem

$$\begin{array}{ll} min_{\mathbf{c},r} & r^2 \\ \text{subject to} & \|\phi(\mathbf{x_i}) - \mathbf{c}\|^2 \leq r^2, i = 1, \ldots, n, \leftarrow \text{this states that all the points are within the sphere} \end{array} \tag{7}$$

The constraint

$$\|\phi(\mathbf{x_i}) - \mathbf{c}\|^2 \leq r^2$$

can be further rewritten as

$$h(\mathbf{c}, r) = \|\phi(\mathbf{x_i}) - \mathbf{c}\|^2 - r^2 \leq 0,\ i = 1, \ldots, n$$

Introduce $\alpha_i \geq 0$ for each of these constraints and form the Lagrangian:

$$L(\mathbf{c}, r) = r^2 + \sum_{i=1}^{n} \alpha_i \left[\|\phi(\mathbf{x_i}) - \mathbf{c}\|^2 - r^2\right] \tag{8}$$

Take the derivatives

$$\begin{aligned} \tfrac{\delta L(\mathbf{c},r)}{\delta \mathbf{c}} &= 2\textstyle\sum_{i=1}^{n} \alpha_i \left(\phi(\mathbf{x_i}) - \mathbf{c}\right) = 0, \text{and} \\ \tfrac{\delta L(\mathbf{c},r)}{\delta r} &= 2r\left(1 - \textstyle\sum_{i=1}^{n} \alpha_i\right) = 0 \end{aligned}$$

from which we obtain

$$\sum_{i=1}^{n} \alpha_i = 1 \text{ and, as a consequence, } \mathbf{c} = \sum_{i=1}^{n} \alpha_i \phi(\mathbf{x_i}) \tag{9}$$

Summarized, the two equations of (9) state that **the center of the smallest sphere is the convex combination/convex hull of the training points.**

Now to compute the actual coefficients α_i we plug the relations (9) into the Lagrangian to obtain:

$$\begin{aligned}
L(\alpha) &= L(\alpha_1,\ldots,\alpha_n)\\
&= r^2 + \textstyle\sum_{i=1}^n \alpha_i \left[\|\phi(\mathbf{x_i}) - \mathbf{c}\|^2 - r^2\right]\\
&= r^2 + \textstyle\sum_{i=1}^n \alpha_i < \phi(\mathbf{x_i}) - \mathbf{c}, \phi(\mathbf{x_i}) - \mathbf{c} > -r^2 \underbrace{\sum_{i=1}^n \alpha_i}_{=1}\\
&= r^2 + \textstyle\sum_{i=1}^n \alpha_i < \phi(\mathbf{x_i}) - \mathbf{c}, \phi(\mathbf{x_i}) - \mathbf{c} > -r^2\\
&= \textstyle\sum_{i=1}^n \alpha_i < \phi(\mathbf{x_i}) - \mathbf{c}, \phi(\mathbf{x_i}) - \mathbf{c} >\\
&= \textstyle\sum_{i=1}^n \alpha_i < \phi(\mathbf{x_i}) - \sum_{j=1}^n \alpha_j\phi(\mathbf{x_j}), \phi(\mathbf{x_i}) - \sum_{k=1}^n \alpha_k\phi(\mathbf{x_k}) > \qquad (10)\\
&= \cdots\\
&= \textstyle\sum_{i=1}^n \alpha_i \Big(K(\mathbf{x_i},\mathbf{x_i}) + \sum_{k,j=1}^n \alpha_j\alpha_k K(\mathbf{x_k},\mathbf{x_j}) - 2\sum_{j=1}^n \alpha_j K(\mathbf{x_i},\mathbf{x_j})\Big)\\
&= \textstyle\sum_{i=1}^n \alpha_i K(\mathbf{x_i},\mathbf{x_i}) + \underbrace{\sum_{i=1}^n \alpha_i}_{=1} \sum_{k,j=1}^n \alpha_j\alpha_k K(\mathbf{x_k},\mathbf{x_j}) - 2\sum_{i,j=1}^n \alpha_i\alpha_j K(\mathbf{x_i},\mathbf{x_j})\\
&= \textstyle\sum_{i=1}^n \alpha_i K(\mathbf{x_i},\mathbf{x_i}) - \sum_{i,j=1}^n \alpha_i\alpha_j K(\mathbf{x_i},\mathbf{x_j})
\end{aligned}$$

An interesting result is that the corresponding KK conditions must be obeyed by the solution to this problem.

That is, the optimal solution must satisfy

$$\alpha_i \left[\|\phi(\mathbf{x_i}) - \mathbf{c}\|^2 - r^2\right] = 0, \text{ for } i = 1,\ldots,n \ \textbf{(KKT)}$$

Because of $\sum_{i=1}^n \alpha_i = 1$ it follows that the α's we are interested must be $\neq 0$, that is, they correspond to training points for which

$$\|\phi(\mathbf{x_i}) - \mathbf{c}\|^2 - r^2 = 0$$

that means these $\mathbf{x_i}$ are on the surface of the sphere. We will call these, once again, **support vectors**.

Thus we have the following algorithm for finding the smallest (hyper)sphere enclosing a (training) set of points.

Input: training set $S = \{\mathbf{x_1},\ldots,\mathbf{x_n}\}$

1. Find α^* the solution of the following optimization problem:

$$\text{Maximize } L(\alpha) = \textstyle\sum_{i=1}^n \alpha_i K(\mathbf{x_i},\mathbf{x_i}) - \sum_{i,j=1}^n \alpha_i\alpha_j K(\mathbf{x_i},\mathbf{x_j})$$
$$\text{subject to } \textstyle\sum_{i=1}^n \alpha_i \text{ and } \alpha_i \geq 0,\ i = 1,\ldots,n$$

2. Set $r^* = \sqrt{L(\alpha^*)}$

3. Set $D = \sum_{i,j=1}^n \alpha_i^*\alpha_j^* K(\mathbf{x_i},\mathbf{x_j}) - r{*}^2$

4. $\mathbf{c}^* = \sum_{i=1}^n \alpha_i^*\phi(\mathbf{x_i})$

5. The decision rule is $f(\mathbf{x}) = \mathcal{H}\left[K(\mathbf{x},\mathbf{x}) - 2\sum_{i=1}^n \alpha_i^* K(\mathbf{x_i},\mathbf{x}) + D\right]$, where $\mathcal{H}(x) = 1$ if $x \geq 0$ and $= 0$ otherwise.

Output: $\mathbf{c}^*$ and f.

2 The Sequential Minimum Optimization (SMO) Algorithm

Let us take a break (sort of) from theory only and try to look at implementation issues.

As you can see in the algorithms for solving the SVM problem or the smallest (hyper)sphere problem we need to solve the associated optimization problem.

Recall that for the type of optimization problems we have, the KKT conditions are *necessary and sufficient*.

But the complexity of solving them in exponential in n, the size of the training set!

This means that using them is a real option only for a small training se.

However, this observation is at the same time very useful!

Why? Because it leads us to the idea that if we could select an appropriate small subset of the training set we could solve the problem.

QUESTION: How small a subset and what do we mean by "appropriate"?

ANSWER:

- The training subset can be very small: 2 elements
- By appropriate we mean that *the subset should be near the actual decision rule, that there should be at least one point from each class*. See figure 1 below.

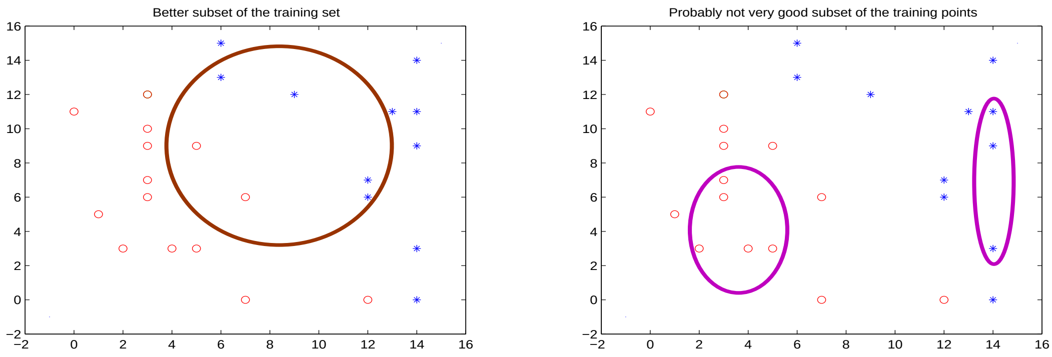

Figure 1: Examples of training subsets

Analytitcal Solution for Two Points

The dual optimization problem is

$$\begin{array}{ll} \text{maximize} & W(\alpha) = \sum \alpha_i - \frac{1}{2}\sum_{i,j=1}^{l} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \\ \text{subject to} & \sum \alpha_i y_i = 0, \\ & 0 \leq \alpha_i \leq C, i = 1, \ldots, l \end{array} \tag{11}$$

where $C = \infty$ for the hard margin SVM.

At each step SMO chooses two elements α_i, α_j to jointly optimize, finds the optimal values for them given that *all others are fixed* and updates the vector α.

So, we have two issues:

1. Select the two elements: *heuristic*
2. optimize: *analytic*

WLOG assume the chosen multipliers are α_1 and α_2. Assume that we have l data points. Thus, $\alpha = (\alpha_1, \ldots, \alpha_l)$.

Recall the $\sum_{i=1}^{l} \alpha_i y_i = 0$. Thus, keeping α_3, α_l fixed, we have that

$$\alpha_1 y_1 + \alpha_2 y_2 = -\sum_{i=3}^{l} \alpha_i y_i = ct. = \alpha_1^{old} y_1 + \alpha_2^{old} y_2$$

Thus, in the two dimensional space (α_1, α_2), these two multipliers lie on a line.

Further, assume that

$$0 \leq \alpha_1, \alpha_2 \leq C \tag{12}$$

(where C is a constant that bounds the values that the multipliers take.

Now the SVM optimization problem is a one dimensional problem which can be solved analytically as follows: In one step, we have

$$\alpha_1^{new} y_1 + \alpha_2^{new} y_2 = \alpha_1^{old} y_1 + \alpha_2^{old} y_2$$

from which, taking into account (12) we obtain the following

$$U \leq \alpha_2^{new} \leq V \tag{13}$$

where

$$U = \begin{cases} \max(0, \alpha_2^{old} - \alpha_1^{old}) & \text{if} \quad y_1 \neq y_2 \\ \max(0, \alpha_1^{old} + \alpha_2^{old} - C) & \text{if} \quad y_1 = y_2 \end{cases}$$

$$V = \begin{cases} \min(C, C - \alpha_1^{old} + \alpha_2^{old}) & \text{if} \quad y_1 \neq y_2 \\ \min(C, \alpha_1^{old} + \alpha_2^{old}) & \text{if} \quad y_1 = y_2 \end{cases}$$

Indeed, consider the following two cases:

Case 1: $y_1 = y_2$. From (12 it follows that

$$\alpha_2^{new} = \alpha_1^{old} + \alpha_2^{old} - \alpha_1^{new}$$

Since we assume $\alpha_1^{new} \geq C$, it follows

$$\alpha_2^{new} \geq \alpha_1^{old} + \alpha_2^{old} - C$$

But α_2^{new} must also be ≥ 0. Hence

$$\alpha_2^{new} \geq \max(0, \alpha_1^{old} + \alpha_2^{old} - C)$$

On the other hand, $\alpha_2^{new} \leq \alpha_1^{old} + \alpha_2^{old}$. But it must also be $\leq C$. Thus,

$$\alpha_2^{new} \leq \min(C\alpha_1^{old} + \alpha_2^{old})$$

Case 2: $y_1 \neq y_2$. From (12) it follows

$$\alpha_2^{new} = -\alpha_1^{old} + \alpha_2^{old} + \alpha_1^{new}$$

Thus,

$$\alpha_2^{new} \leq C - \alpha_1^{old} + \alpha_2^{old}$$

But α_2^{new} must also be $\leq C$. Thus,

$$\alpha_2^{new} \leq \min(C, C - \alpha_1^{old} + \alpha_2^{old})$$

Similarly,

$$\alpha_2^{new} \geq -\alpha_1^{old} + \alpha_2^{old}$$

But it must also be ≥ 0. Hence

$$\alpha_2^{new} \geq \max(0, \alpha_2^{old} - \alpha_1^{old})$$

Let

$$E_j = \sum_{i=1}^{l} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b - y_j,\ j = 1, 2$$

That is, E_1 is the difference between the output of the classifier and the target output value y_1 for the training data $\mathbf{x}_1$, and similarly for E_2.

Remark 1 *The magnitude of E_j may be large even if the point is correctly classified (e.g., $Classifier\ output = 5$, $y_1 = 1$, then $E_1 = 4$)*

Remark 2 *The second derivative of the objective function along the diagonal is $-\mathcal{K}$ where*

$$\mathcal{K} = K(\mathbf{x}_1, \mathbf{x}_1) + K(\mathbf{x}_2, \mathbf{x}_2) - 2K(\mathbf{x}_1, \mathbf{x}_2)$$

Theorem 2 *The maximum of the objective funtion W when only α_1 and α_2 are allowed to change, is achieved by the following procedure:*

Step 1. *Compute $\alpha_2^{new,unclipped} = \alpha_2^{old} + \frac{y_2(E_1 - E_2)}{\mathcal{K}}$.*

Step 2. *Clip $\alpha_2^{new,unclipped}$ to enforce the constraint $U \leq \alpha_2^{new} \leq V$:*

$$\alpha_2^{new} = \begin{cases} V & \text{if } \alpha_2^{new,unclipped} > V \\ \alpha_2^{new,unclipped} & \text{if } U \leq \alpha_2^{new,unclipped} \leq V \\ U & \text{if } \alpha_2^{new,unclipped} < U \end{cases}$$

Step 3. *Obtain the value of α_1^{new} from α_2^{new}:*

$$\alpha_1^{new} = \alpha_1^{old} + y_1 y_2 (\alpha_2^{old} - \alpha_2^{new})$$

Proof:

Let

$$v_i = \sum_{j=3}^{l} y_j \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) = \sum_{j=1}^{l} y_j \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) - \sum_{j=1}^{2} y_j \alpha_j K(\mathbf{x}_i, \mathbf{x}_j)\ i = 1, 2$$

Then we can rewrite the objective function as a function of α_1 and α_2 only as follows:

$$\begin{aligned} W(\alpha_1, \alpha_2) &= \alpha_1 + \alpha_2 - \tfrac{1}{2}K_{11}\alpha_1^2 - \tfrac{1}{2}K_{22}\alpha_2^2 \\ &\quad - y_1 y_2 K_{12}\alpha_1\alpha_2 - y_1\alpha_1 v_1 - y_2\alpha_2 v_2 + CONSTANT \end{aligned}$$

where $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$ for $i, j = 1, 2$; The term CONSTANT contains only $\alpha_i, i = 3, \ldots, l$.

Furthermore, from the condition $\sum_{i=1}^{l} y_i\alpha_i = 0$ we can write

$$y_1\alpha_1 + y_2\alpha_2 = const.$$

Multiplying both sides by y_1, and taking into account that $y_i^2 = 1$, we obtain

$$y_1^2\alpha_1 + y_1y_2\alpha_2 = \alpha_1 + s\alpha_2 = const. = \alpha_1^{old} + s\alpha_2^{old} = \gamma$$

where $s = y_1y_2$.
That is, we obtain the constraint

$$\alpha_1 + s\alpha_2 = \gamma \tag{14}$$

We solve (14) for α_1 and plug its value in $W(\alpha_1, \alpha_2)$ to obtain the new objective functiion

$$\begin{aligned} W(\alpha_2) &= \gamma - s\alpha_2 + \alpha_2 - \tfrac{1}{2}K_{11}(\gamma - s\alpha_2)^2 - \tfrac{1}{2}K_{22}\alpha_2^2 \\ &\quad -sK_{12}\gamma - s\alpha_2\alpha_2 - y_1(\gamma - s\alpha_2)v_1 - y_2\alpha_2v_2 + CONSTANT \end{aligned}$$

We take the derivative of W with resp[ect to α_2 and set it equal to 0:

$$\begin{aligned} W'(\alpha_2) &= 1 - s + sK_{11}(\gamma - s\alpha_2) - K_{22}\alpha_2 \\ &\quad +K_{12}\alpha_2 - sK_{12}(gamma - s\alpha_2) + y_2v_1 - y_2v_2 \\ &= 0 \end{aligned}$$

Solving for α_2 we obtain:

$$\begin{aligned} \alpha_2^{new,unclipped}(K_{11} + K_{22} - 2K_{12}) &= 1 - s + \gamma s(K_{11} - K_{12}) + y_2(v_1 - v2) \\ &= y_2\left[y_2 - y_1 + \gamma y_1(K_{11} - K_{12}) + v_1 - v_2\right] \end{aligned}$$

Using $\mathcal{K}$ we obtain

$$\alpha_2^{new,unclipped}\mathcal{K}y_2 = y_2\alpha_2\mathcal{K} + E_1 - E_2$$

and thus

$$\alpha_2^{new,unclipped} = \alpha_2^{old} + \frac{y_2(E_1 - E_2)}{\mathcal{K}}$$

Clip according to U and V if necessary (when $C < \infty$).