Recap of linear SVM algorithm

Given training set

$S = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_m, y_m) \mid (\mathbf{x}_i, y_i) \in \Re^n \times \{+1, -1\}$

Apply a quadratic programming procedure to find a hyperplane ($\mathbf{w}$, w_0), such that ($\mathbf{w}$,w_0) is the hyperplane maximizing the margin of *S*.

We can write

$$\mathbf{w} = \sum_i y_i \alpha_i \ \mathbf{x}_i$$

where the $\mathbf{x}_i$s are support vectors, y_i is the true class of $\mathbf{x}_i$, and α_i's are coefficients such that

$$\sum_i y_i \alpha_i \ = 0$$

- Now, given a new instance, $\mathbf{x}$, find the classification of $\mathbf{x}$ by computing

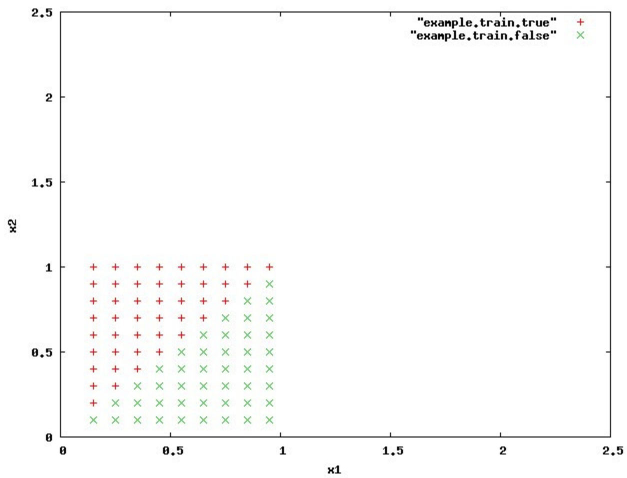
$$\text{class}(\boldsymbol{x}) = \text{sgn}\left(\mathbf{w} \cdot \mathbf{x} + w_0\right)$$
$$= \text{sgn}\left(\sum_i y_i \alpha_i \left(\boldsymbol{x} \cdot \boldsymbol{x}_i\right) + w_0\right)$$

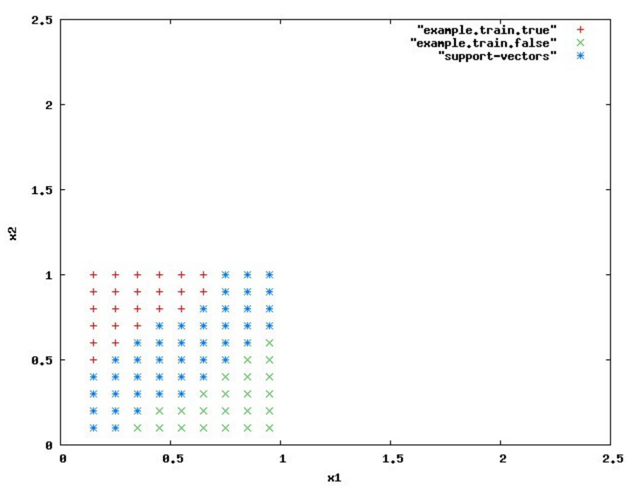
Simple Example
(from textbook's practice problems)

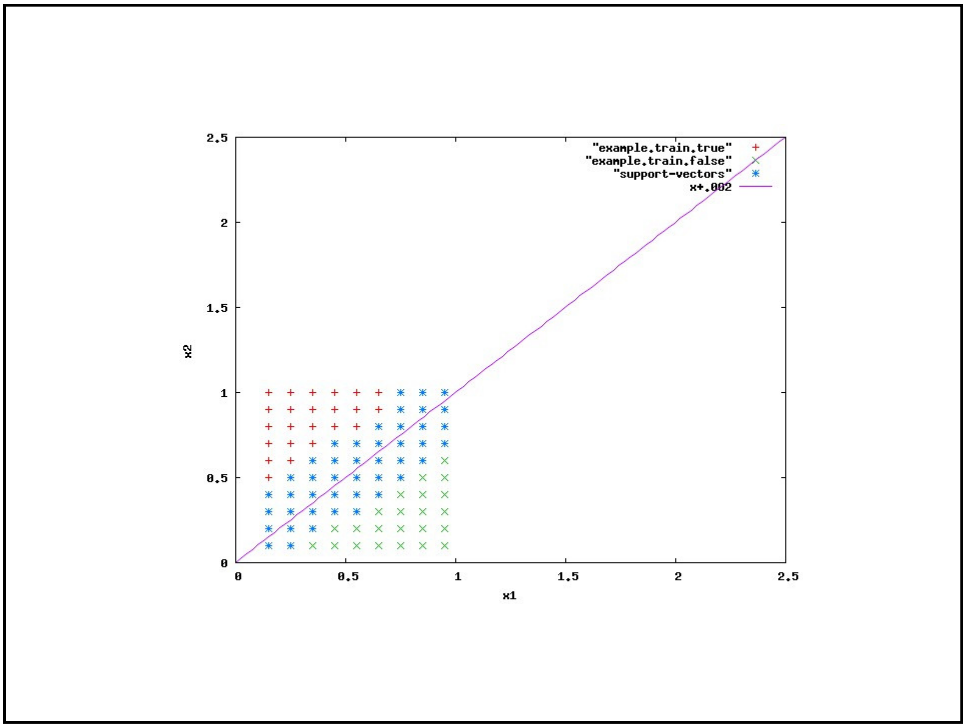
Class 1: (1, 1), (1,2) (2,1)
Class 2: (0, 0), (1,0), (0, 1)

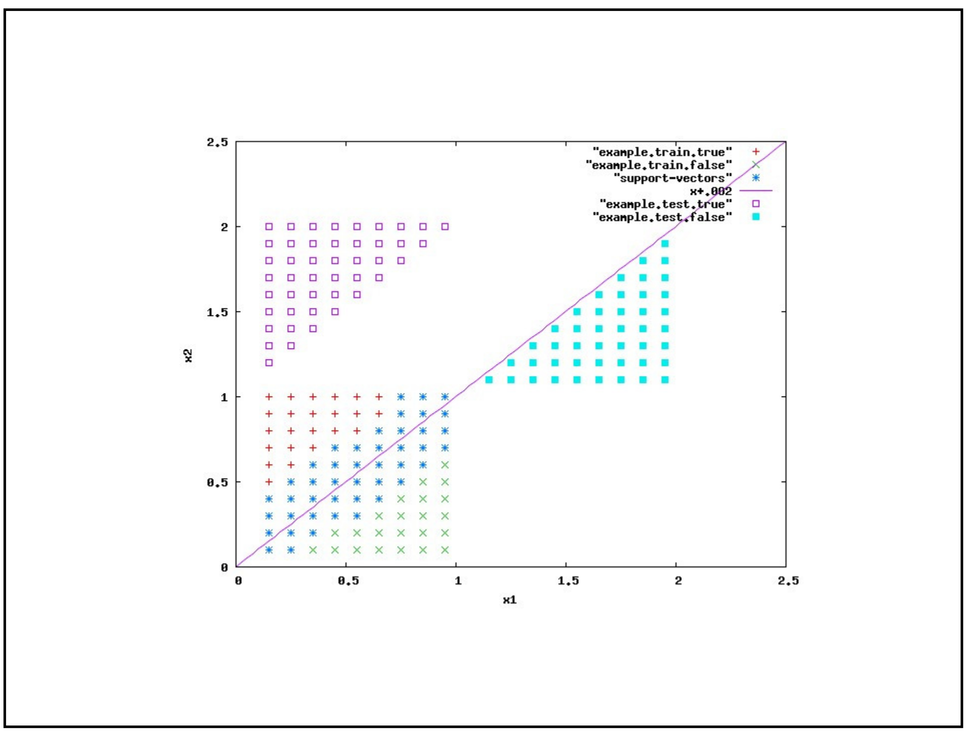
Plot, find optimal seperating line, margin, support vectors, alphas
Classify new point: (3, 1)

Simple example using SVM_light


"example.train.true"
"example.train.false"
x1
x2
0
0.5
1
1.5
2
2.5


"example.train.true"
"example.train.false"
"support-vectors"
x1
x2
0
0.5
1
1.5
2
2.5


"example.train.true"
"example.train.false"
"support-vectors"
x+.002
x1
x2


"example.train.true"
"example.train.false"
"support-vectors"
x+.002
"example.test.true"
"example.test.false"
x1
x2

Recap of **nonlinear** SVM algorithm

Given training set

$S = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_m, y_m) \mid (\mathbf{x}_i, y_i) \in \Re^n \times \{+1, -1\}$

1. Choose a map $\Phi: \Re^n \rightarrow F$, which maps $\mathbf{x}_i$ to a higher dimensional feature space. (Solves problem that X might not be linearly separable in original space.)

2. Choose a cheap-to-compute kernel function

$$k(\mathbf{x},\mathbf{z}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{z})$$

(Solves problem that in high dimensional spaces, dot products are very expensive to compute.)

3. Map all the $\mathbf{x}_i$'s to feature space F by computing $\Phi(\mathbf{x}_i)$.

4. Apply quadratic programming procedure (using the kernel function k) to find a hyperplane ($\mathbf{w}$, w_0), such that

$$\mathbf{w} = \sum_i y_i \alpha_i \, \Phi(\boldsymbol{x}_i)$$

where the $\Phi(\mathbf{x}_i)$'s are support vectors, the α_i's are coefficients, and w_0 is a threshold, such that ($\mathbf{w}$,w_0) is the hyperplane maximizing the margin of S in F.

- Now, given a new instance, **x**, find the classification of **x** by computing

$$\text{class}(\boldsymbol{x}) = \text{sgn}\left(\sum_i y_i \alpha_i (\Phi(\boldsymbol{x}) \cdot \Phi(\boldsymbol{x}_i)) + w_0) \right)$$

$$= \text{sgn}\left(\sum_i y_i \alpha_i \, k(\boldsymbol{x}, \boldsymbol{x}_i) + w_0) \right)$$

Most commonly used kernels

- Linear

$$K(\mathbf{x}, \mathbf{x}_i) = \mathbf{x} \cdot \mathbf{x}_i$$

- Polynomial: parameter d

$$K(\mathbf{x}, \mathbf{x}_i) = [(\mathbf{x} \cdot \mathbf{x}_i) + 1]^d$$

- Gaussian (or "radial basis function"): parameter γ

$$K(\mathbf{x}, \mathbf{x}_i) = e^{-\gamma |\mathbf{x} - \mathbf{x}_i|^2}$$

- Sigmoid: parameters a, b

$$K(\mathbf{x}, \mathbf{x}_i) = \tanh(a\mathbf{x} \cdot \mathbf{x}_i + b)$$

Example: Polynomial Kernel

- Let $\mathbf{x} = (x_1, x_2)$, $\mathbf{y} = (y_1, y_2)$
- Let $k(\mathbf{x},\mathbf{y}) = (1 + \mathbf{x} \cdot \mathbf{y})^2$
- Let $\varphi(\mathbf{x}) = (1,\ x_1^{\ 2}, \sqrt{2}x_1x_2, x_2^{\ 2}, \sqrt{2}x_1, \sqrt{2}x_2)$
- Show $k(\mathbf{x},\mathbf{y}) = \varphi(\mathbf{x}) \cdot \varphi(\mathbf{y})$

$$(1 + x_1 y_1 + x_2 y_2)^2$$

SVM Homework

Study Sheet for Quiz Wed. (Jan 26)

1. You should know what the sigmoid output function is and how to take its derivative.

2. You should know the definition of "learning rate" and "momentum", and what the purpose of each is.

3. You should be able to briefly describe the k-fold cross validation algorithm.

4. You should know what an "auto-associative" network is.

5. You should understand the neural network vision system as far as it was described in class.

6. You should know the definition of "margin" in the context of SVMs.

7. You should be able to show that if

$$y = \mathbf{x}_i \cdot \mathbf{w} + b \geq +1 \text{ for all positive instances } (y_i = +1)$$
$$y = \mathbf{x}_i \cdot \mathbf{w} + b \leq -1 \text{ for all negative instances } (y_i = -1)$$

then

$$M = \frac{1}{\|w\|}$$

8. You should know what is the input to and output of a linear support vector machine, and what formula it uses to classify a new instance.

9. You should know the role of a "kernel" function in SVMs, and be able to give an example of one.