Machine Learning 10-701

Tom M. Mitchell
Machine Learning Department
Carnegie Mellon University

April 12, 2011

Today:

- Support Vector Machines
- Margin-based learning

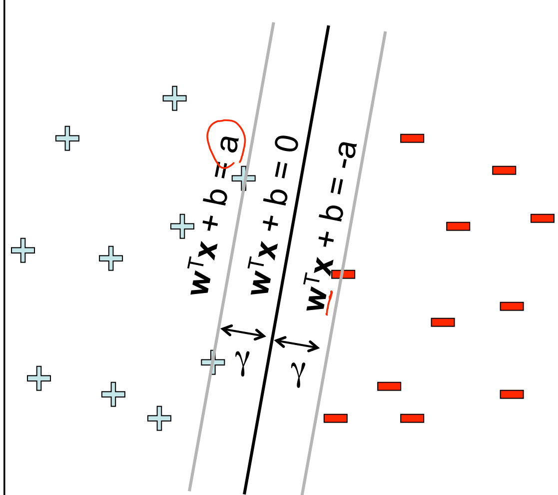
Readings:

Required:
SVMs: Bishop Ch. 7, through 7.1.2

Optional:
Remainder of Bishop Ch. 7

Thanks to Aarti Singh for several slides

SVM: Maximize the margin

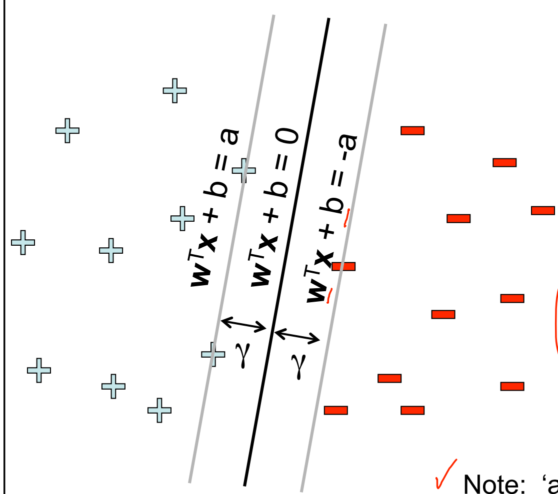


Margin = Distance of closest examples from the decision line/ hyperplane

margin = γ = a/‖w‖

Maximizing the margin

$Y \in \{-1, 1\}$



Margin = Distance of closest examples from the decision line/ hyperplane

margin = $\gamma = a/\|w\|$

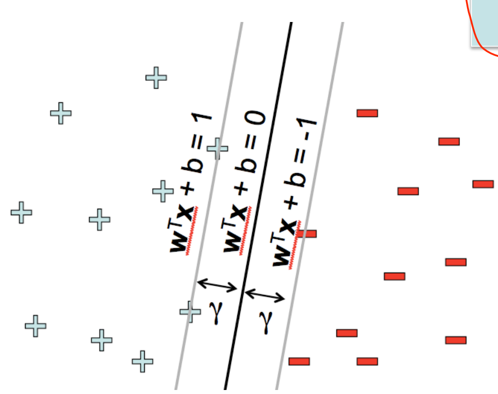
$$\max_{\mathbf{w},b} \ \gamma = a/\|\mathbf{w}\|$$

$a=1$

$$\text{s.t. } (\mathbf{w}^T\mathbf{x}_j + b)\, y_j \geq a \quad \forall j$$

Note: 'a' is arbitrary (can normalize equations by a)

Support Vector Machine (primal form)



$$\max_{\mathbf{w},b} \ \gamma = 1/\|\mathbf{w}\|$$

$$\text{s.t. } (\mathbf{w}^T\mathbf{x}_j + b)\, y_j \geq 1 \quad \forall j$$

Primal form:

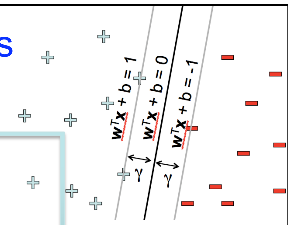
$$\min_{\mathbf{w},b} \ \mathbf{w}^T\mathbf{w}$$

$$\text{s.t. } (\mathbf{w}^T\mathbf{x}_j + b)\, y_j \geq 1 \quad \forall j$$

Solve efficiently by quadratic programming (QP)

- Well-studied solution algorithms

We can solve either primal or dual forms



Primal form: solve for $\mathbf{w}$, b

$$\min_{\mathbf{w},b} \quad \mathbf{w}^T\mathbf{w}$$
$$\text{s.t.} \quad y_l(\mathbf{w}^T\mathbf{x}_l + b) \geq 1 \quad \forall l \in \text{ training examples}$$

Classification test for new $\mathbf{x}$: $\mathbf{w}^T\mathbf{x} + b > 0$

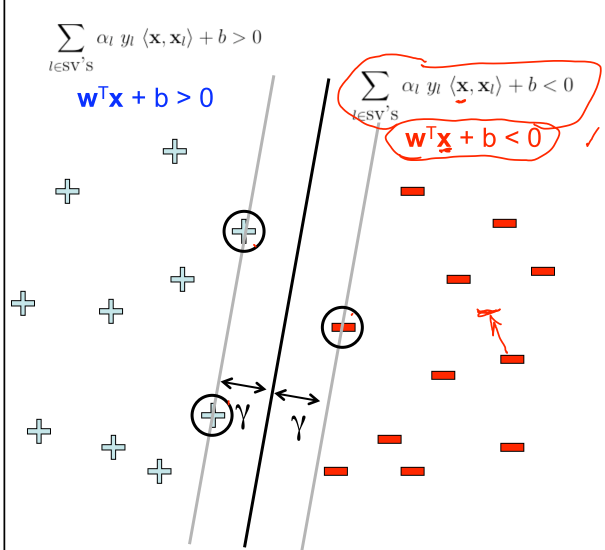
Dual form: solve for $\alpha_1 \ldots \alpha_M$

$$\max_{\alpha_1 \ldots \alpha_M} \quad \sum_{l=1}^{M} \alpha_l - \frac{1}{2}\sum_{j=1}^{M}\sum_{k=1}^{M} \alpha_j \alpha_k y_j y_k \langle \mathbf{x_j}, \mathbf{x_k} \rangle$$
$$\text{s.t.} \quad \alpha_l \geq 0 \quad \forall l \in \text{ training examples}$$
$$\sum_{l=1}^{M} \alpha_l y_l = 0$$

Classification test for new $\mathbf{x}$: $\sum_{l \in \text{sv's}} \alpha_l y_l \langle \mathbf{x}, \mathbf{x}_l \rangle + b > 0$

both are QP problems with a single local optimum!

Support Vectors



Linear hyperplane defined by "support vectors"

Moving other points a little doesn't effect the decision boundary

only need to store the support vectors to predict labels of new points

How many support vectors in linearly separable case, given d dimensions?

≤ d+1

Kernel SVM

And because the dual form depends only on inner products, we can apply the kernel trick to work in a (virtual) projected space $\Phi : X \rightarrow F$

Primal form: solve for $\mathbf{w}, b$ in the projected higher dim. space

$$\min_{\mathbf{w},b} \quad \mathbf{w}^T\mathbf{w}$$

$$\text{s.t.} \quad y_l(\mathbf{w}^T\Phi(\mathbf{x}_l) + b) \geq 1 \quad \forall l \in \text{ training examples}$$

Classification test for new $\mathbf{x}$ $\quad \mathbf{w}^T\Phi(\mathbf{x}) + b > 0$

Dual form: solve for $\alpha_1 ... \alpha_M$ in the original low dim. space

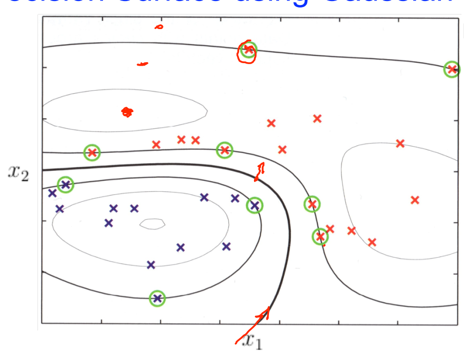
$$\max_{\alpha_1...\alpha_M} \quad \sum_{l=1}^M \alpha_l - \frac{1}{2}\sum_{j=1}^M \sum_{k=1}^M \alpha_j \alpha_k y_j y_k \kappa(\mathbf{x_j}, \mathbf{x_k})$$

$$\text{s.t.} \quad \alpha_l \geq 0 \quad \forall l \in \text{ training examples}$$

$$\sum_{l=1}^M \alpha_l y_l = 0$$

Classification test for new $\mathbf{x}$: $\sum_{l \in \text{sv's}} \alpha_l \; y_l \kappa(\mathbf{x}, \mathbf{x}_l) + b > 0$

SVM Decision Surface using Gaussian Kernel



[from Bishop, figure 7.2]

$$\hat{f}(\mathbf{x}) = \mathbf{w}^T\Phi(\mathbf{x}) + b$$

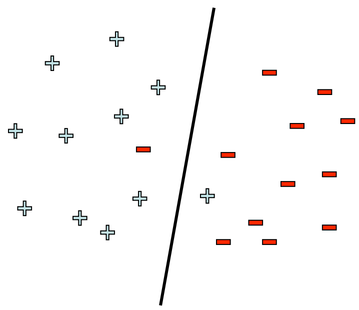
Circled points are the ***support vectors***: training examples with non-zero α_l

Points plotted in <u>original</u> 2-D space.

Contour lines show constant $\hat{f}(\mathbf{x})$

$$\hat{f}(\mathbf{x}) = b + \sum_{l=1}^M \alpha_l \; y_l \; \kappa(\mathbf{x}, \mathbf{x}_l) = b + \sum_{l=1}^M \alpha_l \; y_l \exp(-\|\mathbf{x} - \mathbf{x}_l\|^2 / 2\sigma^2)$$

What if data is not linearly separable?



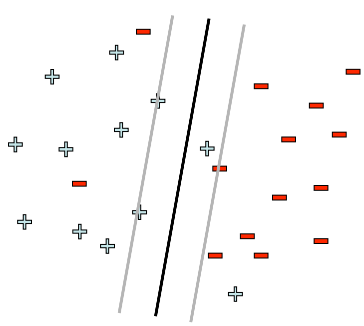
Use features of features of features of features….

$x_1^2, x_2^2, x_1x_2, ..., \exp(x_1)$

But run risk of overfitting!

What if data is still not linearly separable?

Allow "error" in classification



$$\min_{\mathbf{w},b} \ \mathbf{w}^T\mathbf{w} + C \ \#\text{mistakes}$$
$$\text{s.t. } (\mathbf{w}^T\mathbf{x}_j + b)\, y_j \geq 1 \quad \forall j$$

Maximize margin and minimize # mistakes on training data

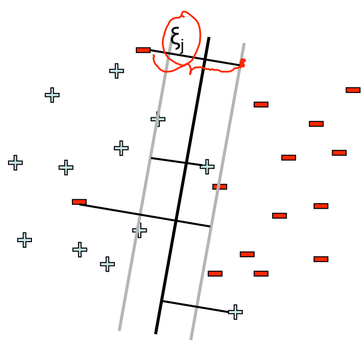
C - tradeoff parameter

Not QP ☹

0/1 loss (doesn't distinguish between near miss and bad mistake)

Support Vector Machine with soft margins

Allow "error" in classification



Soft margin approach

$$\min_{\mathbf{w},b} \; \mathbf{w}^T\mathbf{w} + C \sum_j \xi_j$$

$$\text{s.t. } (\mathbf{w}^T\mathbf{x}_j + b)\, y_j \geq 1 - \xi_j \quad \forall j$$

$$\xi_j \geq 0 \quad \forall j$$

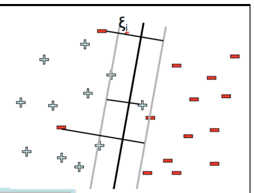
ξ_j - "slack" variables = (>1 if x_j misclassifed)

pay linear penalty if mistake

C - tradeoff parameter (chosen by cross-validation)

Still QP ☺

Primal and Dual Forms for Soft Margin SVM



Primal form: solve for $\mathbf{w}, b$ in the projected higher dim. space

$$\min_{\mathbf{w},b} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{l=1}^{M}\xi_l$$

$$\text{s.t.} \quad y_l(\mathbf{w}^T\Phi(\mathbf{x}_l) + b) \geq 1 - \xi_l \quad \forall l \in \text{ training examples}$$

$$\xi_l \geq 0 \quad \forall l \in \text{ training examples}$$

Dual form: solve for $\alpha_1 \ldots \alpha_M$ in the original low dim. space

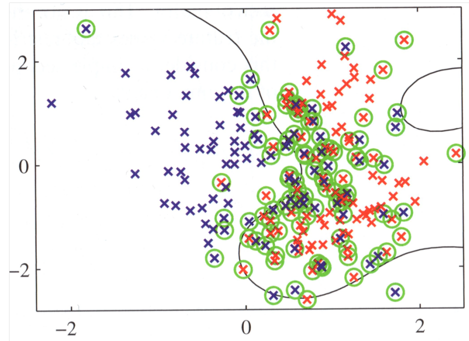
$$\max_{\alpha_1 \ldots \alpha_M} \quad \sum_{l=1}^{M}\alpha_l - \frac{1}{2}\sum_{j=1}^{M}\sum_{k=1}^{M}\alpha_j\alpha_k y_j y_k \; \kappa(\mathbf{x_j}, \mathbf{x_k})$$

$$\text{s.t.} \quad 0 \leq \alpha_l \leq C \quad \forall l \in \text{ training examples}$$

$$\sum_{l=1}^{M}\alpha_l y_l = 0$$

both are QP problems with a single local optimum ☺

SVM Soft Margin Decision Surface using Gaussian Kernel



[from Bishop, figure 7.4]

$$\hat{f}(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x}) + b$$

Circled points are the ***support vectors***: training examples with non-zero α_l

Points plotted in original 2-D space.

Contour lines show constant $\hat{f}(\mathbf{x})$

$$\hat{f}(\mathbf{x}) = b + \sum_{l=1}^{M} \alpha_l \; y_l \; \kappa(\mathbf{x}, \mathbf{x}_l) = b + \sum_{l=1}^{M} \alpha_l \; y_l \exp(-\|\mathbf{x} - \mathbf{x}_l\|^2 / 2\sigma^2)$$

SVM Summary

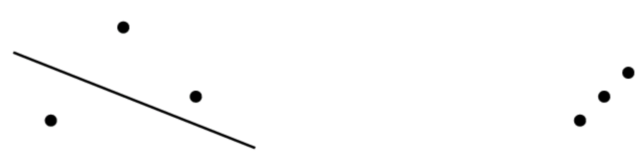
- Objective: maximize margin between decision surface and data
- Primal and dual formulations
  - dual represents classifier decision in terms of *support vectors*
- Kernel SVM's
  - learn linear decision surface in high dimension space, working in original low dimension space
- Handling noisy data: soft margin "slack variables"
  - again primal and dual forms
- SVM algorithm: Quadratic Program optimization
  - single global minimum

SVM: PAC Results?

VC dimension: examples

What is VC dimension of

- H_2 = { (($w_0 + w_1x_1 + w_2x_2$)>0 → y=1) }
  - $VC(H_2)=3$
- For H_n = linear separating hyperplanes in n dimensions, $VC(H_n)=n+1$

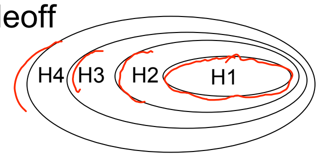


$$m \geq \frac{1}{\epsilon}(4\log_2(2/\delta) + 8VC(H)\log_2(13/\epsilon))$$

Structural Risk Minimization [Vapnik]

Which hypothesis space should we choose?

- Bias / variance tradeoff



SRM: choose H to minimize bound on expected true error!

$$error_{true}(h) < error_{train}(h) + \sqrt{\frac{VC(H)(\ln \frac{2m}{VC(H)} + 1) + \ln \frac{4}{\delta}}{m}}$$

* unfortunately a somewhat loose bound...

Margin-based PAC Results

[Shawe-Taylor, Langford, McCallester]

Consider a fixed distribution D on pairs $\langle x, y\rangle$ with $x \in R^d$ satisfying $||x|| = 1$ and $y \in \{-1, 1\}$. We are interested in finding a weight vector w with $||w|| = 1$ such that the sign of $w \cdot x$ predicts y. For $\gamma > 0$ the error rate of w on distribution D relative to safety margin γ, denoted $\ell_\gamma(w, D)$ is defined as follows.

$$\ell_\gamma(w, D) \equiv \mathrm{P}_{\langle x, y\rangle \sim D}\left[(w \cdot x)y \leq \gamma\right]$$

Let S be a sample of m pairs drawn IID from the distribution D. The sample S can be viewed as an empirical distribution on pairs. We are interested in bounding $\ell_0(w, D)$ in terms of $\ell_\gamma(w, S)$ and the margin γ. Bartlett and Shawe-Taylor use fat shattering arguments [2] to show that with probability at least $1 - \delta$ over the choice of the sample S we have the following simultaneously for all weight vectors w with $||w|| = 1$ and margins $\gamma > 0$.

$$\ell_0(w, D) \leq \ell_\gamma(w, S) + 27.18\sqrt{\frac{\log^2 m + 84}{m\gamma^2}} + O\left(\sqrt{\frac{\ln \frac{1}{\delta}}{m}}\right) \quad (1)$$

recall:

$$error_{true}(h) < error_{train}(h) + \sqrt{\frac{VC(H)(\ln \frac{2m}{VC(H)} + 1) + \ln \frac{4}{\delta}}{m}}$$

Maximizing Margin as an Objective Function

- We've talked about many learning algorithms, with different objective functions

- 0-1 loss
- sum sq error
- maximum log data likelihood
- MAP
- maximum margin

How are these all related?

Slack variables – Hinge loss

Complexity penalization

$$\xi_j = \text{loss}(f(x_j), y_j)$$

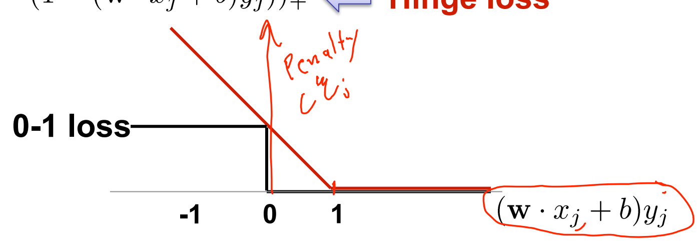
$$f(x_j) = \text{sgn}(\mathbf{w} \cdot \mathrm{x_j} + \mathrm{b})$$

$$\min_{\mathbf{w},b} \ \mathbf{w}^T\mathbf{w} + C \sum_j \xi_j$$

s.t. $(\mathbf{w}^T\mathbf{x}_j + b)\, y_j \geq 1 - \xi_j \quad \forall j$

$\xi_j \geq 0 \quad \forall j$

$$\xi_j = (1 - (\mathbf{w} \cdot x_j + b) y_j))_+$$ **Hinge loss**



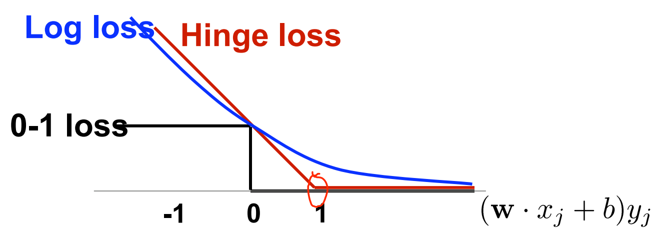
SVM vs. Logistic Regression

SVM : **Hinge loss**

$$\text{loss}(f(x_j), y_j) = (1 - (\mathbf{w} \cdot x_j + b)y_j))_+$$

Logistic Regression : **Log loss** (-ve log conditional likelihood)

$$\text{loss}(f(x_j), y_j) = -\log P(y_j \mid x_j, \mathbf{w}, b) = \log(1 + e^{-(\mathbf{w} \cdot x_j + b)y_j})$$



What you need to know

Primal and Dual optimization problems

Kernel functions

Support Vector Machines

- Maximizing margin
- Derivation of SVM formulation
- Slack variables and hinge loss
- Relationship between SVMs and logistic regression
  - 0/1 loss
  - Hinge loss
  - Log loss