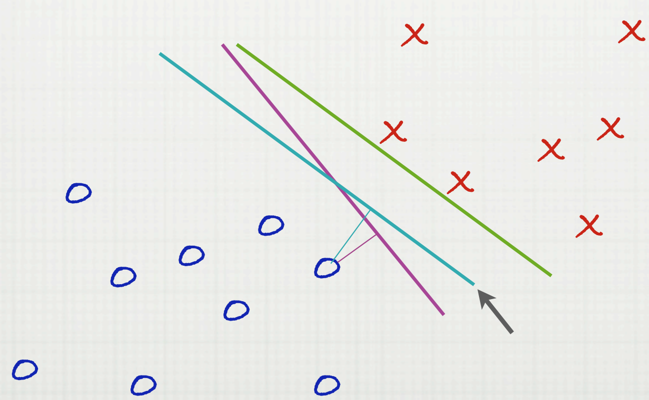
Support Vector Machines

SUE ANN HONG

10/18/2007

THE MOST FAMOUS SLIDE *EVER*



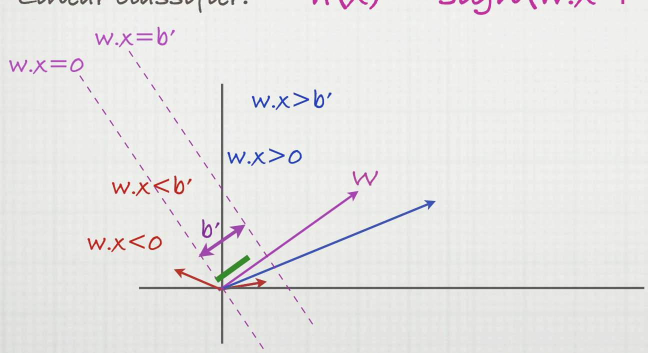
PRIMAL: THE "INTUITIVE" VERSION

$$\min \|w\|^2 + C \sum \xi$$
$$\text{s.t. } (w.x + b)y \geq 1 - \xi$$
$$\xi \geq 0$$

PRIMAL: THE "INTUITIVE" VERSION

$$\min \|w\|^2 + C \sum \xi$$
$$\text{s.t. } (w.x + b)y \geq 1 - \xi$$
$$\xi \geq 0$$

- Linear classifier: $h(x) = \text{sign}(w.x + b)$



- $|(w.x+b)|$ big if far away from the boundary "confidence"

PRIMAL: THE "INTUITIVE" VERSION

$$\min \|w\|^2 + C \sum \xi$$
$$\text{s.t. } (w.x + b)y \geq 1 - \xi$$
$$\xi \geq 0$$

1. regularizer
2. ~ 1/margin

"slack variable"

- $(w.x+b)y > 0$ iff $w.x+b$ and y same sign
- so want $(w.x+b)y$ to be as large as possible
- could set ξ's to anything big... no constraint?!
- need ξ's to be small... but set $\xi_i < 0$ for a confident point, ξ_j can be big for some other point
- $\xi \geq 0$ means no love shared

PRIMAL: THE "INTUITIVE" VERSION

$$\min \|w\|^2 + C \sum \xi$$
$$\text{s.t. } (w.x + b)y \geq 1 - \xi$$
$$\xi \geq 0$$

- How do we find w?
  - Quadratic programming
- how do we find C?
  - Cross-validation!
  - HW3! :) get your libsvm today!

DUAL: THE "SUPPORT VECTOR" VERSION

$$\max \sum \alpha - 1/2 \sum \alpha_i \alpha_j y_i y_j x_i x_j$$
$$\text{s.t. } \sum \alpha_i y_i = 0$$
$$C \geq \alpha_i \geq 0$$

- Where did this come from?
  - Remember Lagrange Multipliers
    - Let us "incorporate" constraints into objective
    - Then solve the problem in the "dual" space of lagrange multipliers

PRIMAL FEAR

$$\min ||w||^2 + C \sum \xi$$
$$\text{s.t. } (w.x + b)y \geq 1-\xi$$
$$\xi \geq 0$$

$$\max \sum \alpha - 1/2 \sum \alpha_i \alpha_j y_i y_j x_i x_j$$
$$\text{s.t. } \sum \alpha_i y_i = 0$$
$$C \geq \alpha_i \geq 0$$

- Number of parameters?
  - large # features?
  - large # examples?
    - for large # features, DUAL preferred
    - many α_i can go to zero!

DUAL: THE "SUPPORT VECTOR" VERSION

$$\max \sum \alpha - 1/2 \sum \alpha_i \alpha_j y_i y_j x_i x_j$$
$$\text{s.t. } \sum \alpha_i y_i = 0$$
$$C \geq \alpha_i \geq 0$$

- How do we find α?
  - Quadratic programming
- How do we find C?
  - Cross-validation!
- Wait... how do we predict y for a new point x??

  $y = \text{sign}(w.x+b)$
- How do we find w?

  $w = \Sigma_i \alpha_i y_i x_i$
- b? "intersection" (algebra I),

  $y = \text{sign}(\Sigma_i \alpha_i y_i x_i x_j + b)$

"SUPPORT VECTOR"S?

$$\max \sum \alpha - 1/2 \sum \alpha_i \alpha_j y_i y_j x_i x_j$$
$$\text{s.t. } \sum \alpha_i y_i = 0$$
$$C \geq \alpha_i \geq 0$$

$$\max \sum \alpha - \alpha_1 \alpha_2 (-1)(0+2)$$
$$- 1/2\, \alpha_1^2 (1)(0+1)$$
$$- 1/2\, \alpha_2^2 (1)(4+4)$$

$$\max \alpha_1 + \alpha_2 + 2\alpha_1\alpha_2 - \alpha_1^2/2 - 4\alpha_2^2$$
$$\text{s.t. } \alpha_1 - \alpha_2 = 0$$
$$C \geq \alpha_i \geq 0$$

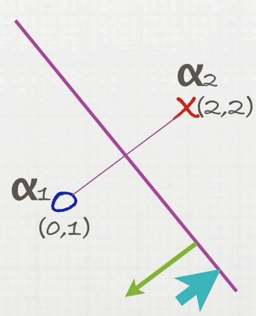
b

$y = w.x + b$

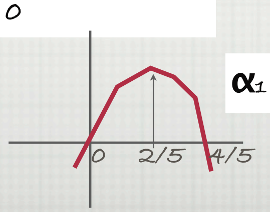
$b = y - w.x$

x1: $b = 1 - .4\,[-2\ -1][0\ 1]$

$= 1 + .4 = 1.4$



$$\alpha_1 = \alpha_2 = \alpha$$
$$\max 2\alpha - 5/2\alpha^2$$
$$\max 5/2\alpha(4/5 - \alpha)$$



$$\alpha_1 = \alpha_2 = 2/5$$

$$w = \Sigma_i \alpha_i y_i x_i$$
$$w = .4([0\ 1] - [2\ 2])$$
$$= .4[-2\ -1]$$

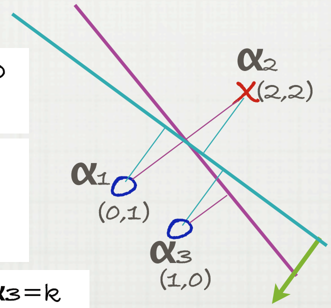
"SUPPORT VECTOR"S?

$\max \sum\alpha - 1/2 \sum\alpha_i\alpha_j y_i y_j x_i x_j$
s.t. $\sum\alpha_i y_i = 0$
$C \geq \alpha_i \geq 0$

b | $b = y - w.x\ldots$

$\max \sum\alpha - \alpha_1\alpha_2(-2) - \alpha_2\alpha_3(-2) - \alpha_3\alpha_1 0$
$- 1/2\ \alpha_1^2(1) - 1/2\ \alpha_2^2(8) - 1/2\ \alpha_3^2(1)$

$\max \alpha_1 + \alpha_2 + \alpha_3 + 2\alpha_1\alpha_2 + 2\alpha_2\alpha_3$
$- \alpha_1^2/2 - 4\alpha_2^2 - \alpha_3^2/2$
s.t. $\alpha_1 - \alpha_2 + \alpha_3 = 0$



$\alpha_2 = \alpha_1 + \alpha_3$ | let $\alpha_2 = k$ | $\alpha_1 + \alpha_3 = k$

$\max (\alpha_1 + \alpha_3) + \alpha_2 + 2\alpha_2(\alpha_1 + \alpha_3) - \alpha_1^2/2 - 4\alpha_2^2 - \alpha_3^2/2$
$\max 2k + 4k^2 - \alpha_1^2/2 - 4k^2 - \alpha_3^2/2$
$\max (\alpha_1^2 - \alpha_3^2)/2$
$\max \alpha_1^2 - (k - \alpha_1)^2$

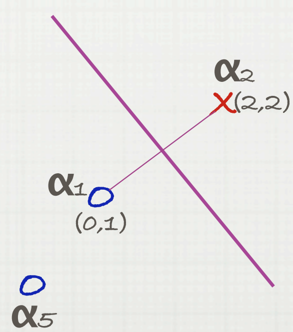
$\alpha_1 = k/2$ | $\alpha_3 = k/2$

$w = \Sigma_i\ \alpha_i y_i x_i$
$w = k\ (.5[0\ 1] - [2\ 2] + .5[1\ 0])$
$= k[-1.5\ -1.5]$

"SUPPORT VECTOR"S?

$\max \sum\alpha - 1/2 \sum\alpha_i\alpha_j y_i y_j x_i x_j$
s.t. $\sum\alpha_i y_i = 0$
$C \geq \alpha_i \geq 0$

What is α_5?
Try this at home

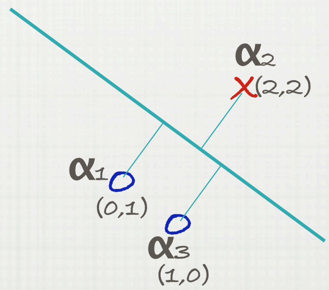


“SUPPORT VECTOR”’S?

$$\max \sum \alpha - 1/2 \sum \alpha_i \alpha_j y_i y_j x_i x_j$$

$$\text{s.t. } \sum \alpha_i y_i = 0$$

$$C \geq \alpha_i \geq 0$$

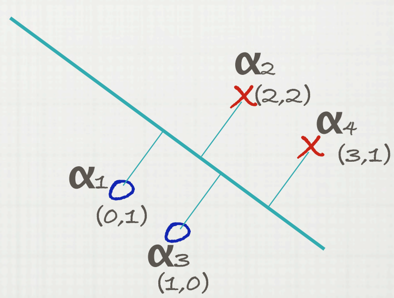


“SUPPORT VECTOR”’S?

$$\max \sum \alpha - 1/2 \sum \alpha_i \alpha_j y_i y_j x_i x_j$$

$$\text{s.t. } \sum \alpha_i y_i = 0$$

$$C \geq \alpha_i \geq 0$$



Which ones are support vectors?

Why?

intuition: how many points “define” a line in 2D?

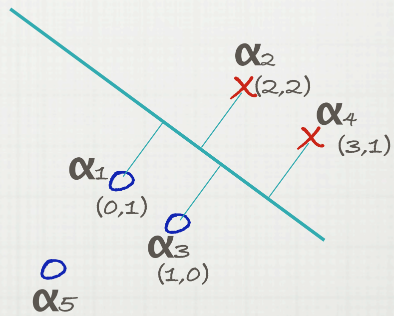
HW3

"SUPPORT VECTOR"S?

$$\max \sum \alpha - 1/2 \sum \alpha_i \alpha_j y_i y_j x_i x_j$$

$$\text{s.t. } \sum \alpha_i y_i = 0$$

$$c \geq \alpha_i \geq 0$$



HINGE LOSS YOUR LOSS

$$\min ||w||^2 + c \sum \xi$$

$$\text{s.t. } (w.x + b)y \geq 1 - \xi$$

$$\xi \geq 0$$

"Loss part": $c \sum \xi$ $||w||^2$ ~regularization

1. $\xi \geq 0$ only if $(w.x + b)y < 1$
2. we want: $\xi \geq 1-(w.x + b)y$ & minimize ξ

   $\Rightarrow \xi = 1-(w.x + b)y$

$\Rightarrow$ loss $= c(1-(w.x + b)y)$ only if $(w.x + b)y < 1$

HINGE LOSS YOUR LOSS

1-Z

- hinge loss $L = 1-(w.x + b)y$ only if $(w.x + b)y < 1$

C=1

HINGE LOSS YOUR LOSS

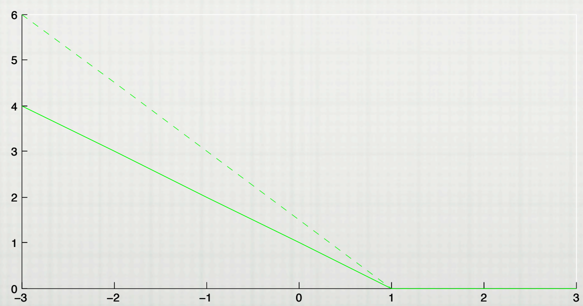
- hinge loss $L = 1-(w.x + b)y$ only if $(w.x + b)y < 1$

LOSS FOR POSITIVE CLASS

C=1

HINGE LOSS YOUR LOSS

- hinge loss $L = 1-(w.x + b)y$ only if $(w.x + b)y < 1$

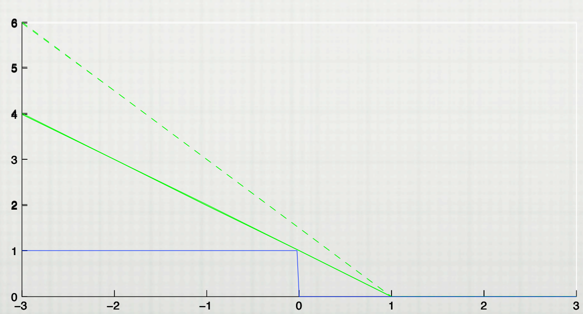


C=1

C=1.5

HINGE LOSS YOUR LOSS

- hinge loss $L = 1-(w.x + b)y$ only if $(w.x + b)y < 1$
- 0/1 loss $L = 1$ if $(w.x+b)y < 0$, 0 otherwise

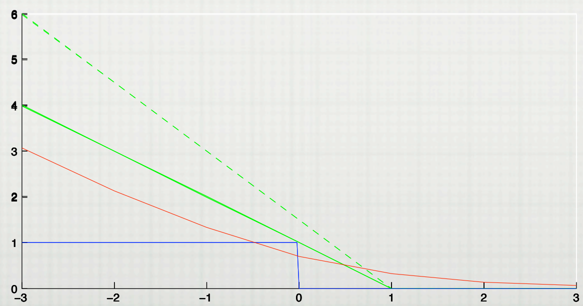


C=1

C=1.5

HINGE LOSS YOUR LOSS

- hinge loss $L = 1-(w.x + b)y$ only if $(w.x + b)y < 1$
- 0/1 loss $L = 1$ if $(w.x+b)y < 0$, 0 otherwise
- logistic loss $L = ||w||^2 + \sum \ln P(Y=1|x,w) = \ldots \ln(1+e\hat{}(-w.x+b))$



C=1

C=1.5

SVM

- Decision boundary: plain SVM is quite simple
- Why is the dual form useful? interesting?
- Support vectors are neat! (computationally, kernel trick, ...)
- SVM, LR, Boosting, ... all a family (diff loss)
- Which letters didn't we see? c b ξ α γ μ x y z w t f ?

LAST REMARKS

- I <3 Burges' tutorial -- READ IT!!!
  - first part (VC-dim, etc) might not make sense until learning theory lectures, but charge on
- <u>MIDTERM REVIEW ON TUESDAY</u>
  - 5-6:30, tentatively, somewhere