Lecture Notes on

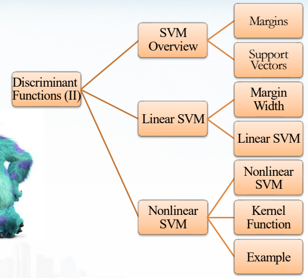
Pattern Recognition

Session 07(B): Discriminant Functions (II)

Gang Li
School of Information Technology
Deakin University, VIC 3125, Australia

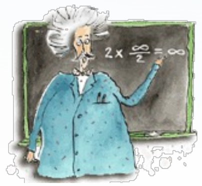
1

Road map



2

SVM Overview



- Best Linear Classifiers
- Margins
- Support Vectors
- VC Dimension

3

Learning through empirical risk minimization

- Estimate *g(x)* from a finite set of observations by minimizing some kind of error function, for example, the ***empirical risk (sample risk)***:

$$R_{emp}(w, w_0) = \frac{1}{n}\sum_{k=1}^{n}[z_k - g(x_k, w, w_0)]^2$$

class labels: $z_k = \begin{cases} +1 \; if \; \mathbf{x}_k \in \omega_1 \\ -1 \; if \; \mathbf{x}_k \in \omega_2 \end{cases}$

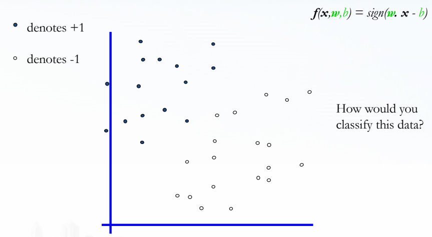
4

Learning through empirical risk minimization

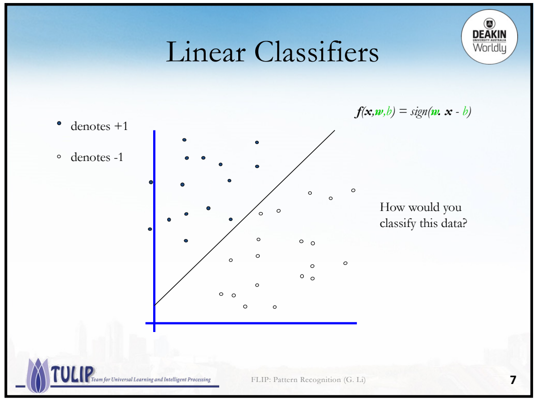
- Conventional ***empirical risk*** minimization over the training data **does not** imply good generalization to novel test data.
  - There could be a number of different functions which all approximate the training data set well.
  - Difficult to determine a function which **best** captures the true underlying structure of the data distribution
    - i.e., has good generalization capabilities

5

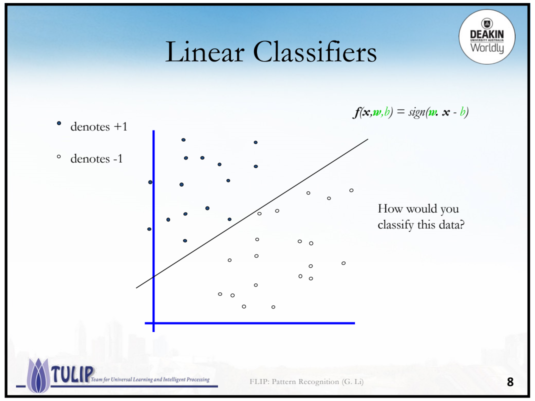
Linear Classifiers



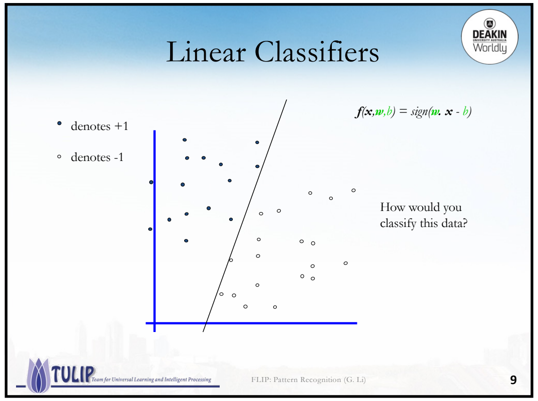
6


Linear Classifiers
f(x,w,b) = sign(w. x - b)
denotes +1
denotes -1
How would you classify this data?
FLIP: Pattern Recognition (G. Li)
7

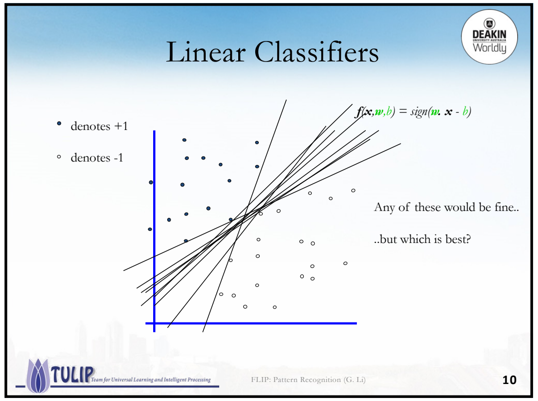
7


Linear Classifiers
f(x,w,b) = sign(w. x - b)
denotes +1
denotes -1
How would you classify this data?
FLIP: Pattern Recognition (G. Li)
8

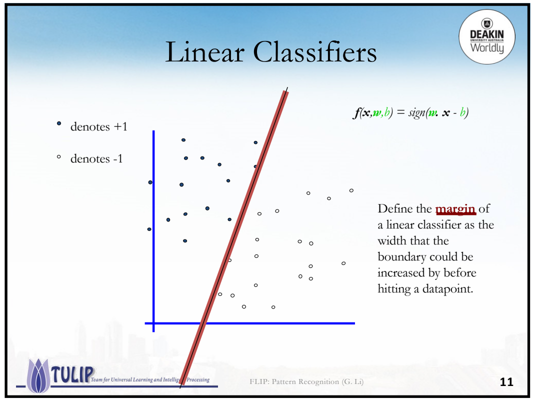
8


Linear Classifiers
f(x,w,b) = sign(w. x - b)
denotes +1
denotes -1
How would you classify this data?
FLIP: Pattern Recognition (G. Li)
9

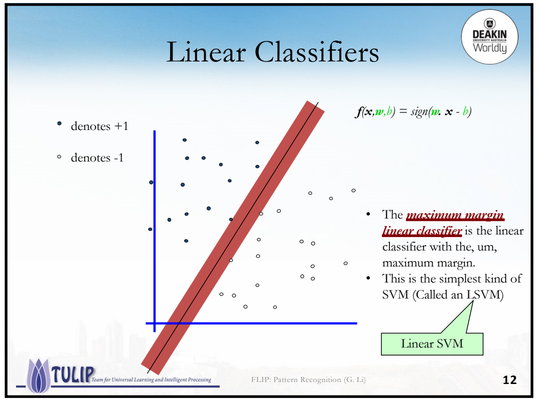
9


Linear Classifiers
f(x,w,b) = sign(w. x - b)
denotes +1
denotes -1
Any of these would be fine..
..but which is best?
FLIP: Pattern Recognition (G. Li)
10

10


Linear Classifiers
f(x,w,b) = sign(w. x - b)
denotes +1
denotes -1
Define the margin of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.
FLIP: Pattern Recognition (G. Li)
11

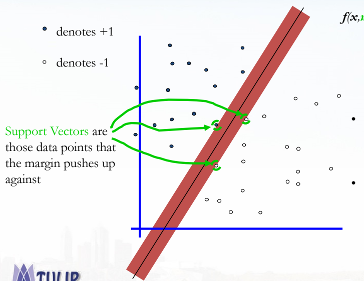
11


Linear Classifiers
f(x,w,b) = sign(w. x - b)
denotes +1
denotes -1
The maximum margin linear classifier is the linear classifier with the, um, maximum margin.
This is the simplest kind of SVM (Called an LSVM)
Linear SVM
FLIP: Pattern Recognition (G. Li)
12

12

Linear Classifiers

$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$



- The ***maximum margin linear classifier*** is the linear classifier with the, um, maximum margin.
- This is the simplest kind of SVM (Called an LSVM)

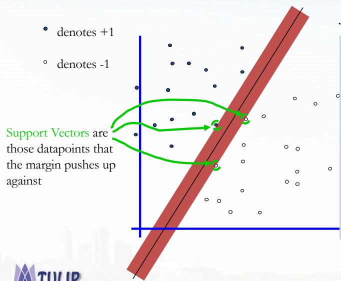
Linear SVM

FLIP: Pattern Recognition (G. Li) 13

13

Linear Classifiers

$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$



1. Intuitively this feels safest.
2. If we have made a small error in the location of the boundary (it's been jolted in its perpendicular direction) this gives us least chance of causing a misclassification.
3. LOOCV is easy since the model is immune to removal of any non-support-vector data points.
4. There's some theory (using VC dimension) that is related to (but not the same as) the proposition that this is a good thing.
5. Empirically it works very very well.

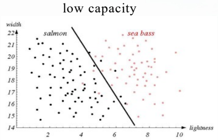
FLIP: Pattern Recognition (G. Li) 14

14

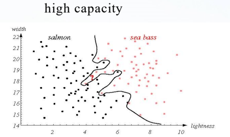
Statistical Learning: Capacity and VC dimension

- To guarantee good generalization performance, the capacity of the learned functions must be controlled.
- Functions with high capacity are more complicated (i.e., have many degrees of freedom).

low capacity



high capacity



FLIP: Pattern Recognition (G. Li) 15

15

Statistical Learning: Capacity and VC dimension

- In statistical learning, the ***Vapnik-Chervonenkis (VC) dimension*** is one of the most popular measures of **capacity**.
  - The **VC dimension** can predict a probabilistic upper bound on the test error (generalization error) of a classification model.

FLIP: Pattern Recognition (G. Li) 16

16

Statistical Learning: Capacity and VC dimension

- A function that
  1. minimizes the empirical risk and
  2. has low VC dimension

will generalize well regardless of the dimensionality of the input space with probability (1-δ):

$$err_{true} \leq err_{train} + \sqrt{\frac{VC(\log(2n/VC)+1)-\log(\delta/4)}{n}}$$

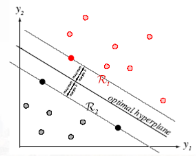
n: training set size

FLIP: Pattern Recognition (G. Li) 17

17

VC dimension and margin of separation

- Vapnik has shown that **maximizing** the margin of separation between classes is equivalent to **minimizing** the VC dimension.
  - The optimal hyper-plane is the one giving the **largest margin** of separation between the classes.

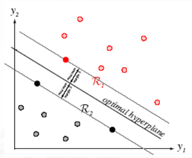


FLIP: Pattern Recognition (G. Li) 18

18

Margin of separation and support vectors

- The margin (i.e., empty area around the decision boundary) is defined by the distance to the nearest training patterns which we refer to as ***support vectors.***
  - Intuitively speaking, these are the most difficult patterns to classify.

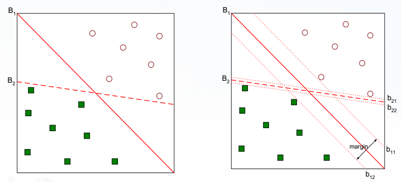


19

Margin of separation and support vectors

different solutions

corresponding margins



20

SVM Overview

- SVMs perform ***structural risk minimization*** to achieve good generalization performance.
- The optimization criterion is the **margin** of separation between classes.
- Training is equivalent to solving a **quadratic programming** problem with **linear constraints**.
- Primarily **two-class** classifiers but can be extended to **multiple** classes.

21

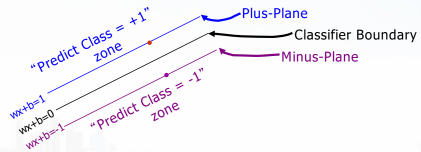
Linear SVM

- Margin Width
- Linear SVM

22

Specifying a line and margin

- How do we represent this mathematically?
  - …in *m* input dimensions?

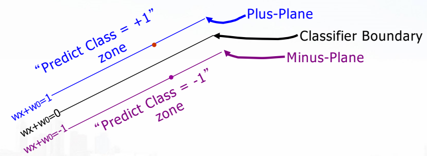


23

Specifying a line and margin

- Plus-plane = $\{ \mathbf{x} : \mathbf{w} . \mathbf{x} + w_0 = +1 \}$
- Minus-plane = $\{ \mathbf{x} : \mathbf{w} . \mathbf{x} + w_0 = -1 \}$

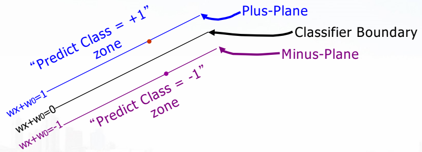
Classify as.. +1 if $\mathbf{w} . \mathbf{x} + w_0 >= 1$
-1 if $\mathbf{w} . \mathbf{x} + w_0 <= -1$
☹ if $-1 < \mathbf{w} . \mathbf{x} + w_0 < 1$



24

Specifying a line and margin

- Plus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = +1 \,\}$
- Minus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = -1 \,\}$
- Claim: The vector w is perpendicular to the Plus Plane. Why?



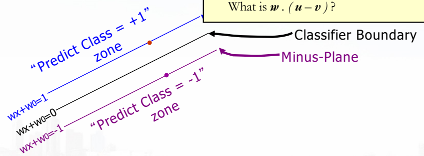
25

Specifying a line and margin

- Plus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = +1 \,\}$
- Minus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = -1 \,\}$
- Claim: the vector w is perpendicular to the Plus Plane. Why?

And so of course the vector **w** is also perpendicular to the Minus Plane

Let **u** and **v** be two vectors on the Plus Plane. What is $\boldsymbol{w} \,.\, (\boldsymbol{u} - \boldsymbol{v})$?

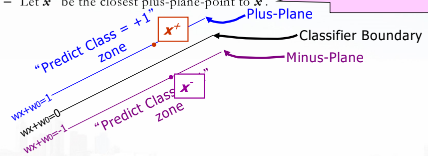


26

Specifying a line and margin

- Plus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = +1 \,\}$
- Minus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = -1 \,\}$
- The vector **w** is perpendicular to the Plus Plane
  - Let $\boldsymbol{x}^-$ be any point on the minus plane
  - Let $\boldsymbol{x}^+$ be the closest plus-plane-point to $\boldsymbol{x}^-$.

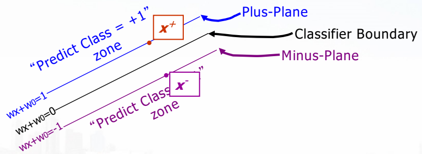
Any location in R^m: not necessarily a data point



27

Specifying a line and margin

- Plus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = +1 \,\}$
- Minus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = -1 \,\}$
- The vector **w** is perpendicular to the Plus Plane
- Claim: $\boldsymbol{x}^+ = \boldsymbol{x}^- + \lambda\, \boldsymbol{w}$ for some value of λ. Why?

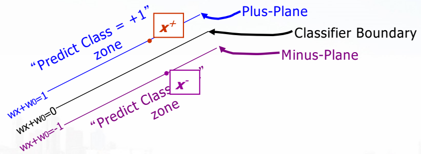


28

Specifying a line and margin

- Plus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = +1 \,\}$
- Minus-plane = $\{\, \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + \boldsymbol{w_0} = -1 \,\}$
- The vector **w** is perpendicular to the Plus Plane
- Claim: $\boldsymbol{x}^+ = \boldsymbol{x}^- + \lambda\, \boldsymbol{w}$ for some value of λ. Why?

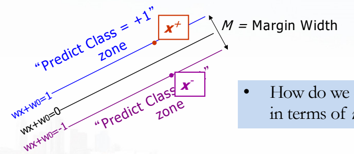
- The line from $\boldsymbol{x}^-$ to $\boldsymbol{x}^+$ is perpendicular to the planes.
- So to get from $\boldsymbol{x}^-$ to $\boldsymbol{x}^+$ travel some distance in direction $\boldsymbol{w}$.



29

Specifying a line and margin

- What we know:
  - $\boldsymbol{w} \,.\, \boldsymbol{x}^+ + \boldsymbol{w0} = +1$
  - $\boldsymbol{w} \,.\, \boldsymbol{x}^- + \boldsymbol{w0} = -1$
  - $\boldsymbol{x}^+ = \boldsymbol{x}^- + \lambda\, \boldsymbol{w}$
  - $|\, \boldsymbol{x}^+ - \boldsymbol{x}^- \,| = M$



- How do we compute M in terms of $\boldsymbol{w}$ and b?

30

Specifying a line and margin

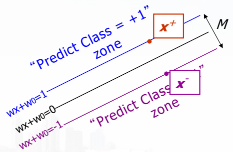
- What we know:
  - $\boldsymbol{w} \cdot \boldsymbol{x}^+ + w_0 = +1$
  - $\boldsymbol{w} \cdot \boldsymbol{x}^- + w_0 = -1$
  - $\boldsymbol{x}^+ = \boldsymbol{x}^- + \lambda \boldsymbol{w}$
  - $|\boldsymbol{x}^+ - \boldsymbol{x}^-| = M$

$\boldsymbol{w} \cdot (\boldsymbol{x}^- + \lambda \boldsymbol{w}) + w_0 = 1$

$\Rightarrow \boldsymbol{w} \cdot \boldsymbol{x}^- + w_0 + \lambda \boldsymbol{w} \cdot \boldsymbol{w} = 1$

$\Rightarrow -1 + \lambda \boldsymbol{w} \cdot \boldsymbol{w} = 1$

$\Rightarrow \lambda = \dfrac{2}{\mathbf{w}.\mathbf{w}}$



- How do we compute M in terms of $\boldsymbol{w}$ and b?

31

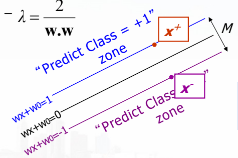
Specifying a line and margin

- What we know:
  - $\boldsymbol{w} \cdot \boldsymbol{x}^+ + w_0 = +1$
  - $\boldsymbol{w} \cdot \boldsymbol{x}^- + w_0 = -1$
  - $\boldsymbol{x}^+ = \boldsymbol{x}^- + \lambda \boldsymbol{w}$
  - $|\boldsymbol{x}^+ - \boldsymbol{x}^-| = M$
  - $\lambda = \dfrac{2}{\mathbf{w}.\mathbf{w}}$

$M = |\boldsymbol{x}^+ - \boldsymbol{x}^-| = |\lambda \boldsymbol{w}|$

$= \lambda |\mathbf{w}| = \lambda \sqrt{\mathbf{w}.\mathbf{w}}$

$= \dfrac{2\sqrt{\mathbf{w}.\mathbf{w}}}{\mathbf{w}.\mathbf{w}} = \dfrac{2}{\sqrt{\mathbf{w}.\mathbf{w}}}$

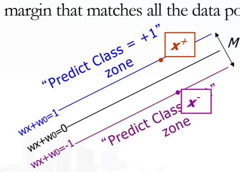


- How do we compute M in terms of $\boldsymbol{w}$ and b?

32

Specifying a line and margin

- Given a guess of $\boldsymbol{w}$ and w_0 we can
  - Compute whether all data points in the correct half-planes
  - Compute the width of the margin
- So now we need to search the space of $\mathbf{w}$'s to find the widest margin that matches all the data points. ***How?***



33

Linear SVM: separable case

- Linear discriminant

$$g(\mathbf{x}) = \mathbf{w}^t \mathbf{x} + w_0$$

Decide ω_1 if $g(\mathbf{x}) > 0$ and ω_2 if $g(\mathbf{x}) < 0$

- Class labels

$$z_k = \begin{cases} +1 \ \text{if} \ \mathbf{x}_k \in \omega_1 \\ -1 \ \text{if} \ \mathbf{x}_k \in \omega_2 \end{cases}$$

- Normalized version

$$z_k g(\mathbf{x}_k) > 0 \quad or \quad z_k(\mathbf{w}^t \mathbf{x}_k + w_0) > 0, \quad for \ k = 1,2,\ldots,n$$

34

Linear SVM: separable case

- The distance of a point $\mathbf{x}_k$ from the separating hyper-plane should satisfy the ***constraint***:

$$\frac{z_k g(\mathbf{x}_k)}{\|w\|} \geq b, \quad b > 0$$

- To ensure uniqueness, impose:

$$b\|w\| = 1$$

- The above ***constraint*** becomes:

$$z_k g(\mathbf{x}_k) \geq 1 \ or \ z_k(\mathbf{w}^t \mathbf{x}_k + w_0) > 1 \quad \text{where } b = \frac{1}{\|w\|}$$

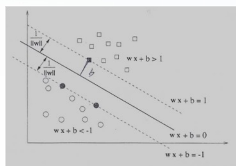
35

Linear SVM: separable case

- Optimization:
  - Maximize the margin $\dfrac{2}{\sqrt{\mathbf{w}.\mathbf{w}}}$
  - With constraints

Problem 1: Minimize $\frac{1}{2}\|w\|^2$

subject to $z_k(w^t x_k + w_0) \geq 1, \quad k = 1, 2, \ldots, n$



- Quadratic programming problem !

36

Linear SVM: separable case

- Use ***Langrange*** optimization, we seek to minimize:

$$L(\mathbf{w}, w_0, \lambda) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{k=1}^{n} \lambda_k [z_k(\mathbf{w}^t \mathbf{x}_k + w_0) - 1], \quad \lambda_k \geq 0$$

- Easier to solve the "dual" problem
  - ***Kuhn-Tucker construction***:

Problem 2: Maximize $\sum_{k=1}^{n} \lambda_k - \frac{1}{2} \sum_{k,j}^{n} \lambda_k \lambda_j z_k z_j x_j^t x_k$

subject to $\sum_{k=1}^{n} z_k \lambda_k = 0, \quad \lambda_k \geq 0, \ k = 1, 2, \ldots, n$

Linear SVM: separable case

- The solution is given by:

$$w = \sum_{k=1}^{n} z_k \lambda_k \mathbf{x}_k, \qquad w_0 = z_k - w^t \mathbf{x}_k$$

$$g(\mathbf{x}) = \mathbf{w}^t \mathbf{x} + w_0$$

$$g(\mathbf{x}) = \sum_{k=1}^{n} z_k \lambda_k (\mathbf{x}_k^t \mathbf{x}) + w_0 = \sum_{k=1}^{n} z_k \lambda_k (\mathbf{x} . \mathbf{x}_k) + w_0$$

- dot product

- It can be shown that if $\mathbf{x}_k$ is **not** a support vector, then the corresponding λ_k=0.
  - Only support vectors contribute to the solution!

Linear SVM: non-separable case

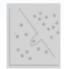
- Allow miss-classifications (i.e., soft margin classifier) by introducing positive ***error (slack)*** variables ψ_k :

$$z_k(w^t \mathbf{x}_k + w_0) \geq 1 - \psi_k, \quad \psi_k \geq 0, \ k = 1, 2, \ldots, n$$

Problem 3: Minimize $\frac{1}{2}\|w\|^2 + c \sum_{k=1}^{n} \psi_k$

subject to $z_k(w^t x_k + w_0) \geq 1 - \psi_k, \quad k = 1, 2, \ldots, n$

- The result is a hyperplane that minimizes the sum of errors ψ_k while maximizing the margin for the correctly classified data.



  - constant c controls the trade-off between the margin and misclassification errors.
  - Aims to prevent outliers from affecting the optimal hyperplane.

Linear SVM: non-separable case

- Easier to solve the "dual" problem
  - ***Kuhn-Tucker construction***

Problem 4: Maximize $\sum_{k=1}^{n} \lambda_k - \frac{1}{2} \sum_{k,j}^{n} \lambda_k \lambda_j z_k z_j x_j^t x_k$

subject to $\sum_{k=1}^{n} z_k \lambda_k = 0$ and $0 \leq \lambda_k \leq c, \ k = 1, 2, \ldots, n$

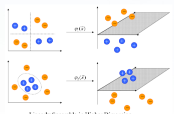
where the use of error variables ψ_k constraint the range of the Lagrange coefficients from 0 to c.

Nonlinear SVM

- Nonlinear SVM
- Kernel Functions
- Example

Nonlinear SVM

- Extending these concepts to the non-linear case involves mapping the data to a high-dimensional space *h*:



Linearly Separable in Higher Dimension

$$\mathbf{x}_k \rightarrow \mathbf{\Phi}(\mathbf{x}_k) = \begin{bmatrix} \varphi_1(\mathbf{x}_k) \\ \varphi_2(\mathbf{x}_k) \\ \ldots \\ \varphi_h(\mathbf{x}_k) \end{bmatrix}$$

- Mapping the data to a sufficiently high dimensional space is likely to cast the data ***linearly separable*** in that space.

Nonlinear SVM

- linear SVM: $g(\mathbf{x}) = \sum_{k=1}^{n} z_k \lambda_k (\mathbf{x} . \mathbf{x}_k) + w_0$

$$\Downarrow$$

- non-linear SVM: $g(\mathbf{x}) = \sum_{k=1}^{n} z_k \lambda_k (\Phi(\mathbf{x}) . \Phi(\mathbf{x}_k)) + w_0$

- Decide ω_1 if $g(\mathbf{x}) > 0$ and ω_2 if $g(\mathbf{x}) < 0$

43

Nonlinear SVM

- The ***disadvantage*** of this approach is that the mapping might be very computationally intensive to compute!

$$\mathbf{x}_k \to \Phi(\mathbf{x}_k)$$

- non-linear SVM: $g(\mathbf{x}) = \sum_{k=1}^{n} z_k \lambda_k (\Phi(\mathbf{x}) . \Phi(\mathbf{x}_k)) + w_0$

44

Nonlinear SVM

- **The kernel trick**
  - Compute dot products using a kernel function

$$K(\mathbf{x}, \mathbf{x}_k) = \Phi(\mathbf{x}) . \Phi(\mathbf{x}_k)$$

$$g(\mathbf{x}) = \sum_{k=1}^{n} z_k \lambda_k (\Phi(\mathbf{x}) . \Phi(\mathbf{x}_k)) + w_0$$

$$\Downarrow$$

$$g(\mathbf{x}) = \sum_{k=1}^{n} z_k \lambda_k K(\mathbf{x}, \mathbf{x}_k) + w_0$$

45

Nonlinear SVM

- **The kernel trick**
  - Compute dot products using a kernel function
  - Kernel functions which can be expressed as a dot product in some space satisfy the ***Mercer's condition***
    - The Mercer's condition does not tell us how to construct $\Phi()$ or even what the high dimensional space is.
    - see Burges' paper
- Advantages of kernel trick
  - no need to know $\Phi()$
  - computations remain feasible even if the feature space has high dimensionality.

46

Polynomial Kernel

- $K(x,y)=(x . y)^{d}$

* It can be shown for the case of polynomial kernels that the data is mapped to a space of dimension $h = \binom{p+d-1}{d}$ where p is the original dimensionality.

* Suppose p=256 and $d = 4$, then h=183,181,376 !!

* A dot product in the high dimensional space would require $O(h)$ computations while the kernel requires only $O(p)$ computations.

47

Choice of Φ is not unique

Example: consider $x \in R^2$, $\Phi(x) = \begin{pmatrix} x_1^2 \\ \sqrt{2} x_1 x_2 \\ x_2^2 \end{pmatrix} \in R^3$, and $K(x, y) = (x . y)^2$

$$(x . y)^2 = (x_1 y_1 + x_2 y_2)^2$$

$$\Phi(x) . \Phi(y) = x_1^2 y_1^2 + 2 x_1 y_1 x_2 y_2 + x_2^2 y_2^2 = (x_1 y_1 + x_2 y_2)^2$$

- Note that neither the mapping $\Phi()$ nor the high dimensional space are unique.

$$\Phi(x) = \frac{1}{\sqrt{2}} \begin{pmatrix} (x_1^2 - x_2^2) \\ 2 x_1 x_2 \\ (x_1^2 + x_2^2) \end{pmatrix} \in R^3 \quad \text{or} \quad \Phi(x) = \begin{pmatrix} x_1^2 \\ x_1 x_2 \\ x_1 x_2 \\ x_2^2 \end{pmatrix} \in R^4$$

48

Kernel functions

- By using different kernel functions, SVM implement a variety of learning machines, some of which coincide with classical architectures

$$polynomial: \; K(x, x_k) = (x.\,x_k)^d$$

$$sigmoidal: \; K(x, x_k) = tanh(v_k(x.\,x_k) + c_k)$$

(corresponds to a two-layer sigmoidal neural network)

$$Gaussian: \; K(x, x_k) = exp\left(\frac{-\|x - x_k\|^2}{2\sigma_k^2}\right)$$

(corresponds to a radial basis function (RBF) neural network)

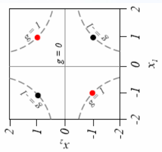
49

Example

- Consider the XOR problem which is non-linear separable

(1,1) and (-1, -1) belong to ω_1

(1,-1) and (-1, 1) belong to ω_2



50

Example

- Consider the following mapping (among others)

$$y = \Phi(x) = \begin{pmatrix} x_1^2 \\ \sqrt{2}x_1 \\ \sqrt{2}x_1x_2 \\ \sqrt{2}x_2 \\ x_2^2 \\ 1 \end{pmatrix}$$

h=6

51

Example

- The above transformation maps x_k to a 6-dimensional space

$$y_1 = \Phi(x_1) = \begin{pmatrix} 1 \\ \sqrt{2} \\ \sqrt{2} \\ \sqrt{2} \\ 1 \\ 1 \end{pmatrix} \quad y_3 = \Phi(x_3) = \begin{pmatrix} 1 \\ -\sqrt{2} \\ \sqrt{2} \\ -\sqrt{2} \\ 1 \\ 1 \end{pmatrix}$$

$$y_2 = \Phi(x_2) = \begin{pmatrix} 1 \\ \sqrt{2} \\ -\sqrt{2} \\ -\sqrt{2} \\ 1 \\ 1 \end{pmatrix} \quad y_4 = \Phi(x_4) = \begin{pmatrix} 1 \\ -\sqrt{2} \\ -\sqrt{2} \\ \sqrt{2} \\ 1 \\ 1 \end{pmatrix}$$

52

Example

- We seek to maximize

$$\sum_{k=1}^{4} \lambda_k - \frac{1}{2}\sum_{k,j}^{4} \lambda_k \lambda_j z_k z_j \Phi(x_j^t)\Phi(x_k)$$

$$\text{subject to } \sum_{k=1}^{4} z_k \lambda_k = 0, \;\; \lambda_k \geq 0, \; k = 1, 2, \ldots, 4$$

- The solution turns out to be:

$$\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = \frac{1}{8}$$

- Since all $\lambda_k \neq 0$, all x_k are support vectors !

53

Example

- We now compute w

$$w = \sum_{k=1}^{4} z_k \lambda_k \Phi(x_k) = \frac{1}{8}\begin{pmatrix} 1 \\ \sqrt{2} \\ \sqrt{2} \\ \sqrt{2} \\ 1 \\ 1 \end{pmatrix} - \frac{1}{8}\begin{pmatrix} 1 \\ \sqrt{2} \\ -\sqrt{2} \\ -\sqrt{2} \\ 1 \\ 1 \end{pmatrix} + \frac{1}{8}\begin{pmatrix} 1 \\ -\sqrt{2} \\ \sqrt{2} \\ -\sqrt{2} \\ 1 \\ 1 \end{pmatrix} - \frac{1}{8}\begin{pmatrix} 1 \\ -\sqrt{2} \\ -\sqrt{2} \\ \sqrt{2} \\ 1 \\ 1 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} 0 \\ 0 \\ \sqrt{2} \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

- The solution for w_0 can be determined using any support vector, e.g., x_1:

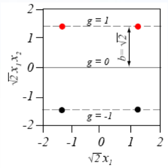
$w^t\Phi(x_1) + w_0 = z_1$ or w0=0

54

Example

- The margin b is computed as follows

$$w = \frac{1}{2}\begin{pmatrix}0\\0\\\sqrt{2}\\0\\0\\0\end{pmatrix} \qquad b = \frac{1}{\|w\|} = \sqrt{2}$$



55

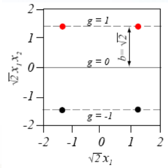
Example

- The discriminant function is as follows

$$g(x) = w^t\Phi(x) + w_0 = x_1x_2$$

where we decide ω_1 if $g(x) > 0$ and ω_2 if $g(x) < 0$

$$w = \frac{1}{2}\begin{pmatrix}0\\0\\\sqrt{2}\\0\\0\\0\end{pmatrix} \qquad \Phi(x) = \begin{pmatrix}x_1^2\\\sqrt{2}x_1\\\sqrt{2}x_1x_2\\\sqrt{2}x_2\\x_2^2\\1\end{pmatrix}$$



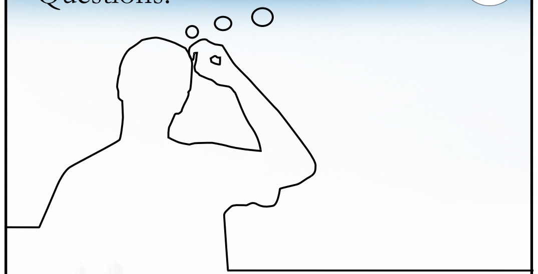
56

Summary of SVM

- SVM is based on exact optimization, not on approximate methods
  - i.e., global optimization method, ***no local optima***
- Avoid ***overfitting*** in high dimensional spaces and ***generalize*** well using a small training set.
- Performance depends on the choice of the ***kernel*** and its parameters.
- Its ***complexity*** depends on the number of support vectors, **not** on the dimensionality of the transformed space.

57

Questions?



59