



# TutorialSii de5

## STATS5099: Data Mining

Support vector machines

Xiaochen Yang  
xiaochen.yang@glasgow.ac.uk

### This week's content

Three types of support vector machines (SVMs)

- (1) Hard-margin (linear) SVM
- (2) Soft-margin (linear) SVM
- (3) Nonlinear/kernel SVM

### Overview of SVM

The goal of SVM is to find a hyperplane that separates the two classes with the largest margin.

### Overview of SVM

The goal of SVM is to find a hyperplane that separates the two classes with the largest margin.

- Hyperplane: in a  $p$ -dimensional space, a hyperplane is defined by
 
$$\{x \in \mathbb{R}^p : w^T x + b = 0\},$$
 where  $w$  is a set of  $p$  coefficients and  $b$  is a scalar.
  - 2D – a line; 3D – a plane
  - SVM classification rule:
 
$$\hat{g}_{new} = \text{sign}(w^T x_{new} + \hat{b})$$

### Margin

### Margin

### Margin

Scalar projection

$$2\delta = (x_1 - x_2)^T \frac{w}{\|w\|}$$

$$= \frac{1}{\|w\|} w^T (x_1 - x_2)$$

$w^T x + b = 0$  def of hyperplane

### Margin

### Margin

### Margin

margin:  $\delta = \frac{1}{\|w\|}$  length of parameter

$x_1, x_2$ : support vectors

on the line support the margin

### Margin

An alternative view: How many units should we walk to go from  $x_0$  to  $x_1$ ?

### Optimisation problem

goal:  $\max_{b,w} \delta$

subject to  $w^T x + b \geq 1$  for  $g_i = 1$  (blue class)

$w^T x + b \leq -1$  for  $g_i = -1$  (green class)

### Optimisation problem

Given  $\delta = \frac{1}{\|w\|}$

$\min_{b,w} \frac{1}{2} \|w\|^2$  (hard margin)

subject to  $g_i(w^T x + b) \geq 1$

### Soft-margin SVM

soft-margin

margin  $\delta = 1$

### Soft-margin SVM

Introduce slack variables  $\xi_i$

$\min_{b,w} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i$

subject to  $g_i(w^T x + b) \geq 1 - \xi_i$

$\xi_i \geq 0$

$C$ : balance variable

$\xi_i$ : less violating

$C \downarrow$ : more violating

$\xi_i$ : large for some  $x_i$

### Nonlinear SVM

(kernel) project into higher

### Nonlinear SVM

enlarge the feature space and find a hyperplane in the new space

translate to nonlinear decision boundary in the original space

### Nonlinear SVM: kernel trick

Primal problem:

$$\min_{b,w} \frac{1}{2} \|w\|^2$$

subject to  $g_i(w^T x + b) \geq 1$

Dual problem:

$$\max_{\alpha} \left( \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j g_i g_j \phi(x_i)^T \phi(x_j) \right)$$

subject to  $\alpha_i \geq 0, \sum_{i=1}^n \alpha_i g_i = 0$

Prediction:

$$\hat{g}(x_{new}) = \text{sign} \left( \sum_{i=1}^n \hat{\alpha}_i g_i \phi(x_{new})^T \phi(x_i) + \hat{b} \right)$$

### Nonlinear SVM: kernel trick

Dual problem:

$$\max_{\alpha} \left( \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j g_i g_j \phi(x_i)^T \phi(x_j) \right)$$

subject to  $\alpha_i \geq 0, \sum_{i=1}^n \alpha_i g_i = 0$

Prediction:

$$\hat{g}(x_{new}) = \text{sign} \left( \sum_{i=1}^n \hat{\alpha}_i g_i \phi(x_{new})^T \phi(x_i) + \hat{b} \right)$$

### Nonlinear SVM: kernel trick

Kernel: a function  $k: X \times X \rightarrow \mathbb{R}$  and can be expressed as

$$k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

- linear kernel:  $k(x_i, x_j) = x_i^T x_j$  ( $\phi$  is identity function)
- polynomial kernel of degree  $d$ :  $k(x_i, x_j) = (1 + x_i^T x_j)^d$
- radial basis function (RBF) kernel / Gaussian kernel:  $k(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$
- sigmoid kernel:  $k(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$

What will be the effect of changing  $d$  and  $\gamma$  in polynomial and RBF kernel to SVM?

$d \uparrow$ , high dim. hyper, better classification, overfit

$\gamma \uparrow$ , smaller kernel, only consider around the test point (local behavior)

### Limitations, practical considerations and R

- very sensitive to hyperparameters (cost parameter  $C$ , kernel parameters)
- not scale-invariant if multiply  $x_i$  by constant,  $k(x_i, x_j)$  also changes, need to standardised data
- no uncertainty measure
- explicitly designed for binary classification

svm(Y~, data, type="C-classification", kernel="c(\*linear", "polynomial", "radial basis", "sigmoid"), degree, gamma, coef0, cost, ...)

tune.svm() #same syntax as svm()

deg=2,3,4,5

gamma=0.1, 0.5, 1, 2, 5, 10

gamma=0.1, 0.5, 1, 2, 5, 10

gamma=0.1, 0.5, 1, 2, 5, 10

### Summary

- (1) hard-margin (linear) SVM
  - hyperplane
  - margin, maximal margin
  - support vectors
- (2) soft-margin (linear) SVM
  - slack variables
  - understand the effect of  $C$
- (3) nonlinear/kernel SVM
  - kernel, kernelisation
  - understand the effect of changing parameters of kernel functions

This is a hard topic in this course. Feel free to ask questions.