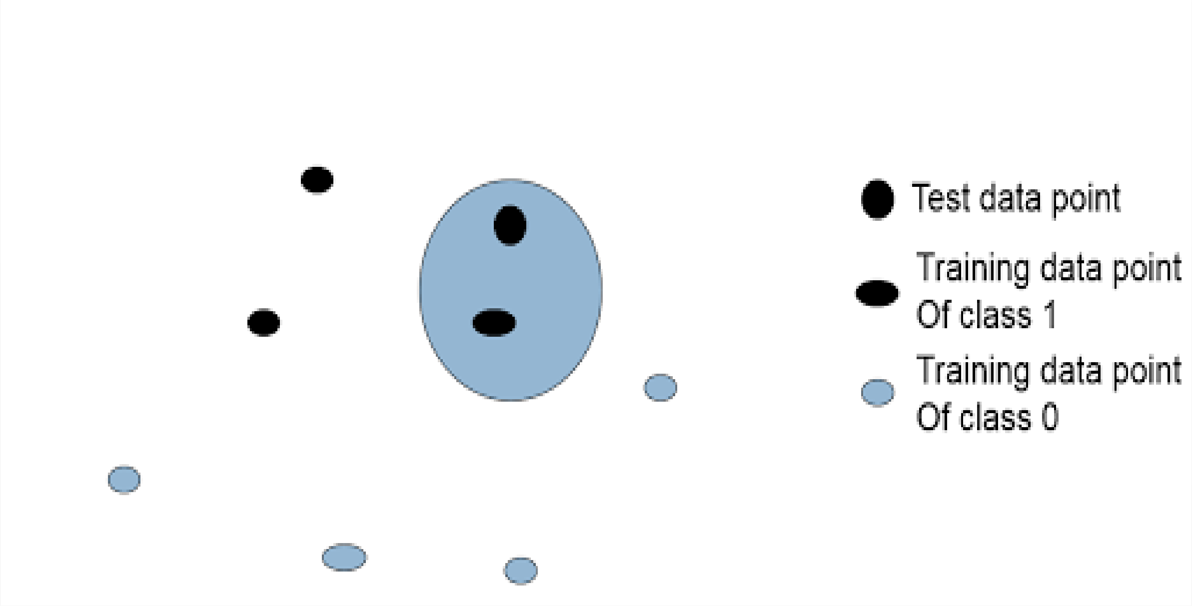
Supervised Learning

Week 8
KNN
SVM

KNN

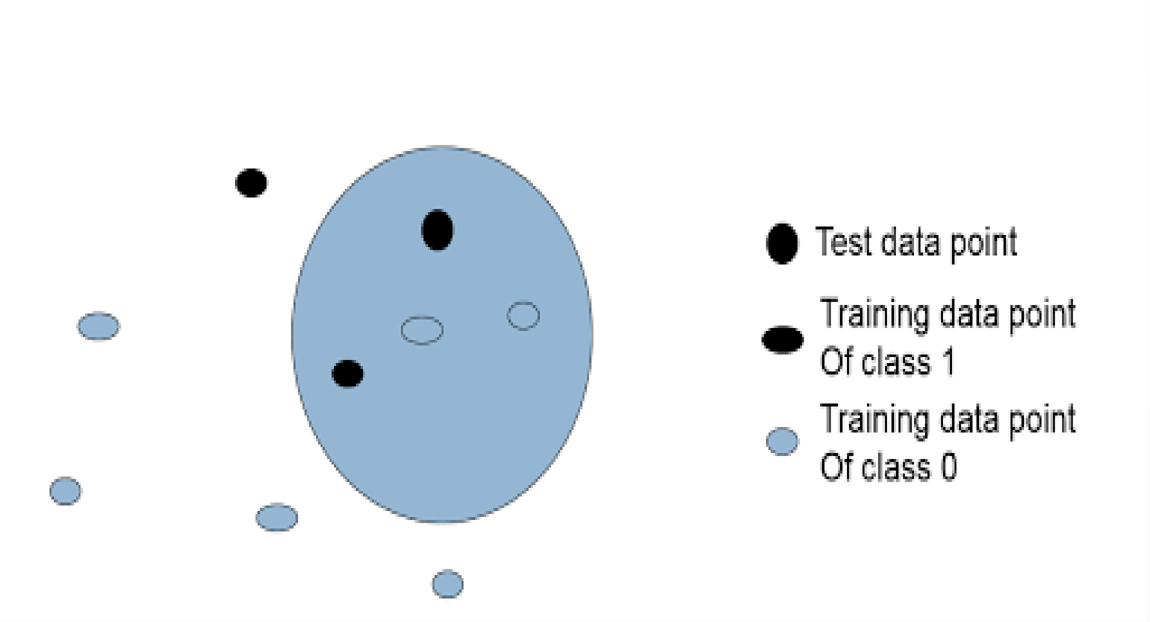
Algorithm & Variants
Best K

KNN
Algorithm & Variants

- For both classification and regression
  - A useful technique is to assign weights based on the neighbours.
  - The nearer neighbors contribute more to the average than the more distant ones.
- The basic idea is to label the test data point is the same as the nearest neighbor.
- 

KNN
Algorithm & Variants…

- How many neighbors?
  - K=?



- Lets say someone would like to check K nearest neighbours of the test point in order to make the decision.
  - Label a test instance same as the majority label of the K-nearest neighbours.
- The figure is an example a of 3−NN classification.

KNN
Algorithm & Variants…

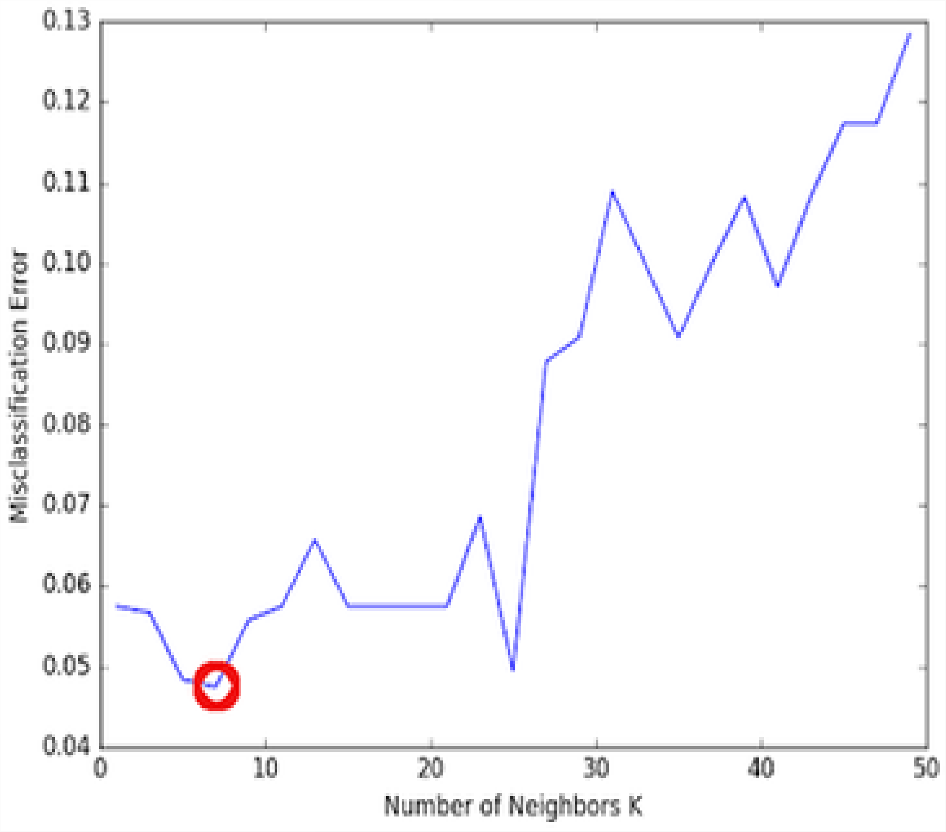
- How to make the majority of decisions?
  - Mode of the class labels
    - Discrete cases
  - Average or mean distances
    - Continuous cases
  - Distance-weighted nearest neighbor algorithm (Shepard's method)
    - Assign weights to the neighbours based on their distance from the test point.
    - Weight may be inverse square of the distances (*$1/D^2$*)
      - Higher the distance of the neighbour, lower its weight.

KNN
Best number of neighbors (K)

- What is the importance of the variable K?
  - K controls the shape of the decision boundary
- Small values of K
  - Restrains the region of a given prediction
  - Forces classifier to be more focused on the close regions and neighbours
    - This will result in a low bias and high variance
- Higher values of K
  - Asking for more information from distant training points
  - Smoother decision boundaries
    - Lower variance but increases bias

KNN
Best number of neighbors (K)...

- Finding the best K
  - There is no rule of thumb in selecting K_{max} since it depends on your desired rate of exploration for K
  - A simple and handy method
    - Cross-validation to partition your data into test and training samples
    - Evaluate model with different ranges of K values $K = 1, .., K_{max}$
    - The misclassification error can be used as a measurement of performance



Remarks

- Learning is very simple (actually, no learning involved).
- Classification is very time consuming because we need to find distance with all the training instances.

Support Vector Machine (SVM)

Support Vector Machine (SVM)

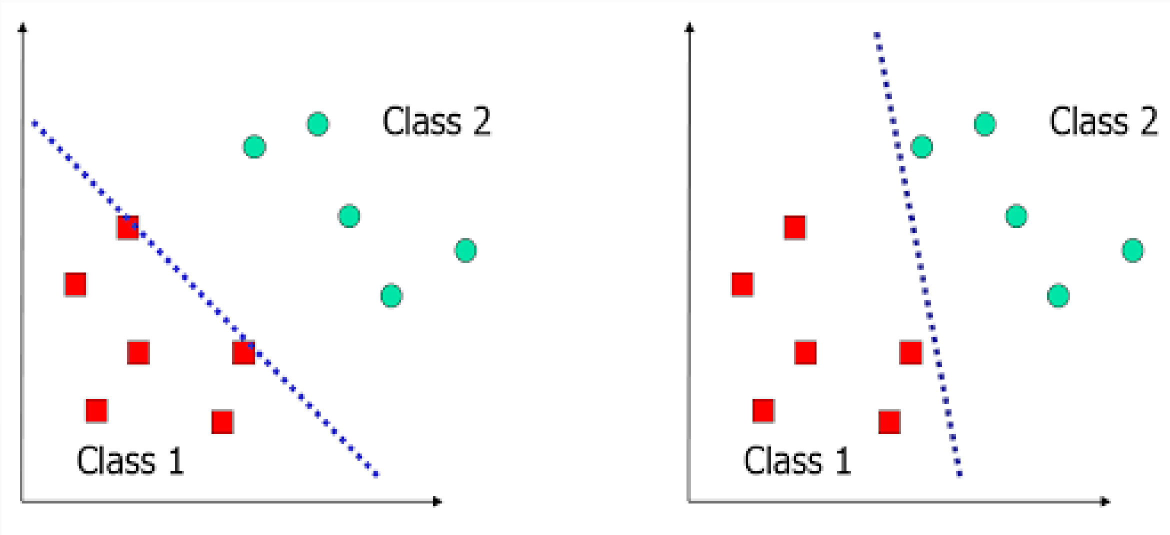
- SVMs can represent non-linear functions and they have an efficient training algorithm.
- Derived from statistical learning theory by Vapnik and Chervonenkis (COLT-92).
- With sufficiently large training data, SVMs can achieve high classification accuracy.
  - For example, ≈99% accuracy for handwritten digits recognition.

Support Vector Machine (SVM)

- Linear SVMs
  - Perfectly separable data points.
  - Almost separable data points.
- Nonlinear SVMs.

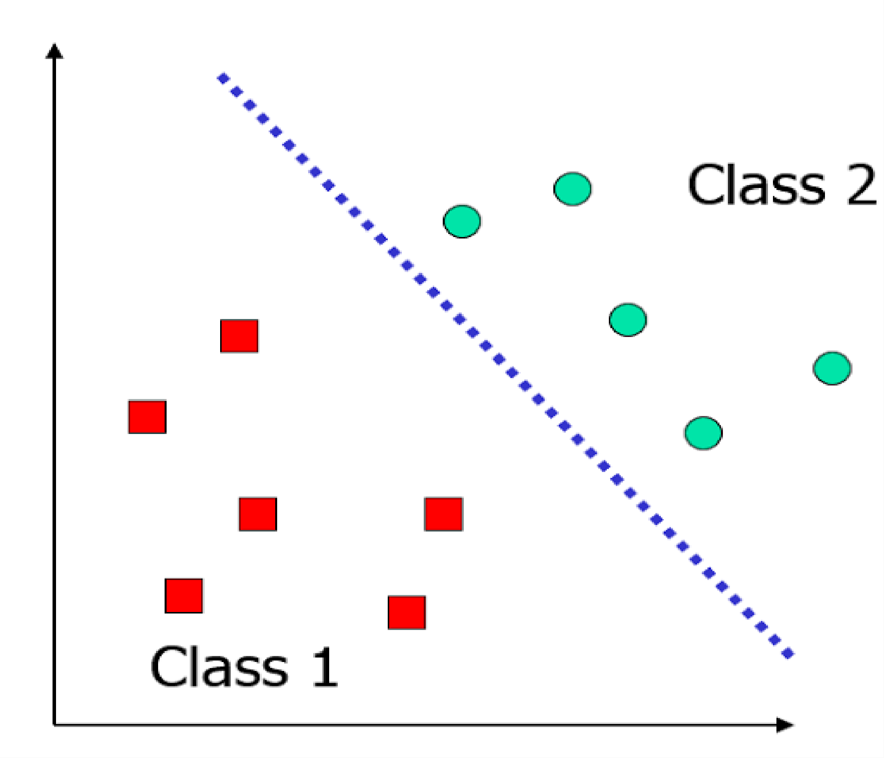
Support Vector Machine (SVM)
Revisiting linear binary classification...

- Consider the following figure as an illustration of boundary in a two class binary classification problem.
- Many decision boundaries can separate these two classes as you can see in the figure.
- Which one should we choose?



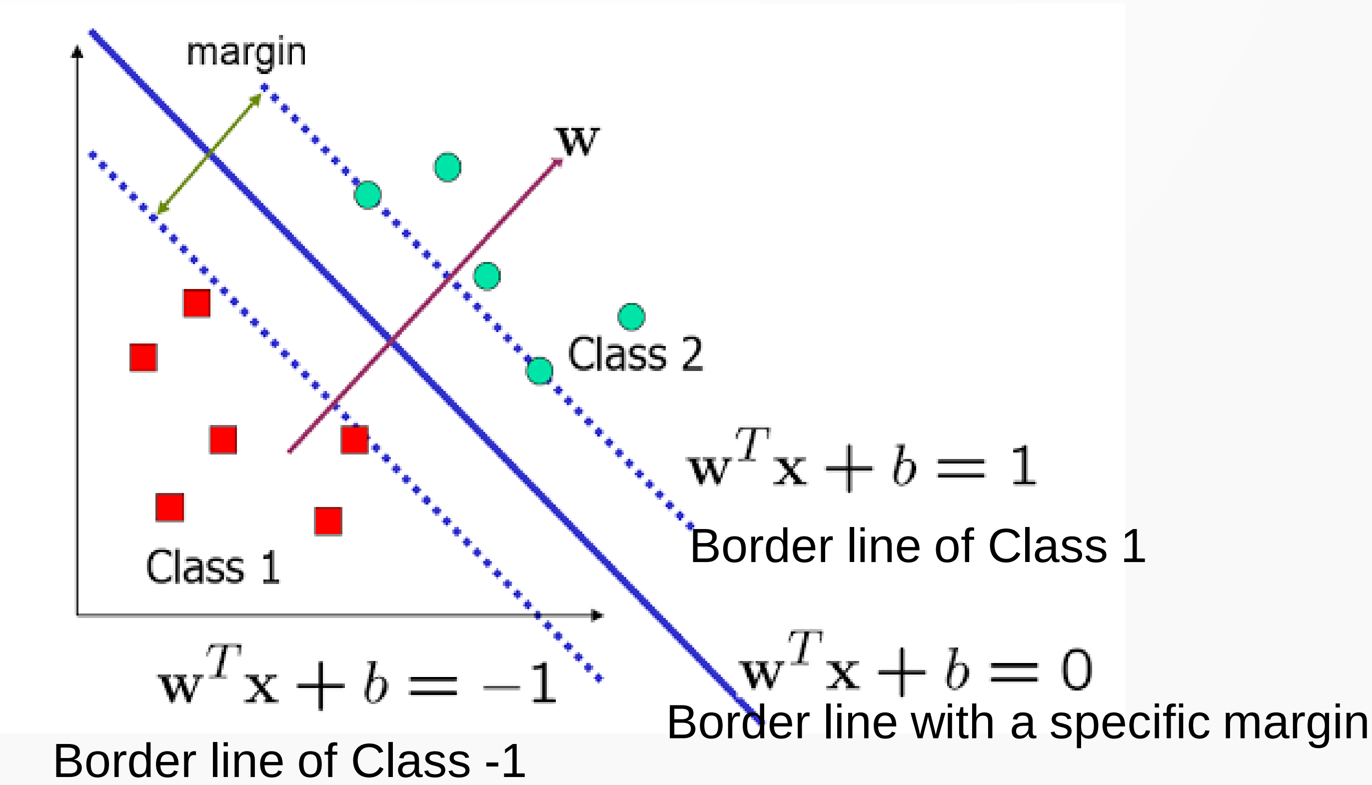
Support Vector Machine (SVM)
Revisiting linear binary classification...

- In SVM, given the labelled data, we do not choose this line randomly.
- We aim for maximizing the margin for this boundary line.
- Consider the figure as another example:
  - Can you see the difference now?
- The proposed line looks to be in the middle of the data points with different classes.



Support Vector Machine (SVM)
Concept of margin

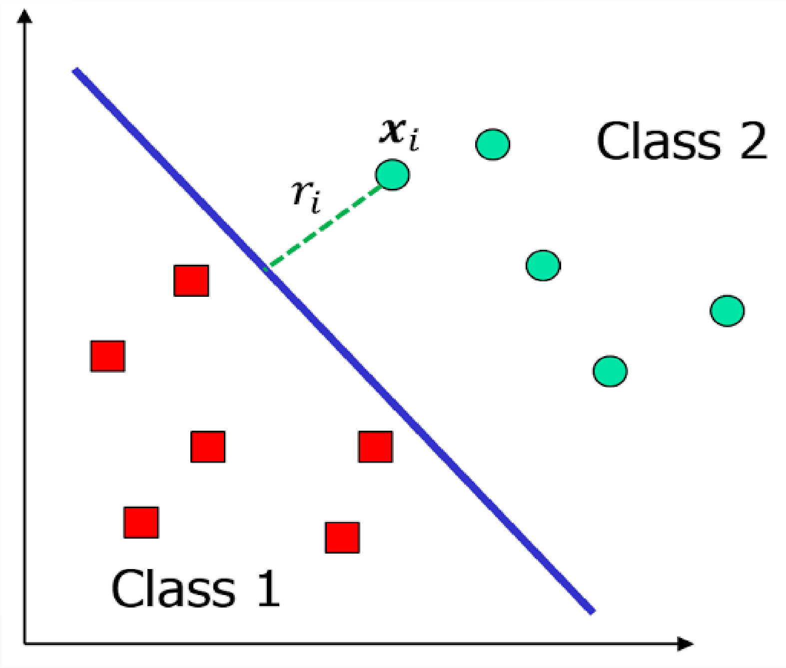
- The basic idea is that the decision boundary should be as far away from the data of both classes as possible.
- The aim is to minimise possible conflicts which may happen during classification.
  - Maximise the margin



Support Vector Machine (SVM)
Concept of margin...

- What is the Euclidean distance from a point x to the decision boundary?
  - Let us denote our training instances as $\{\mathbf{x}_i, y_i\}, \quad i = 1, ..., n$ where $y_i \in \{-1,1\}$.
- The shortest distance between a point and a hyperplane is perpendicular to the plane
  - The distance of x_i to the separating hyperplane is :

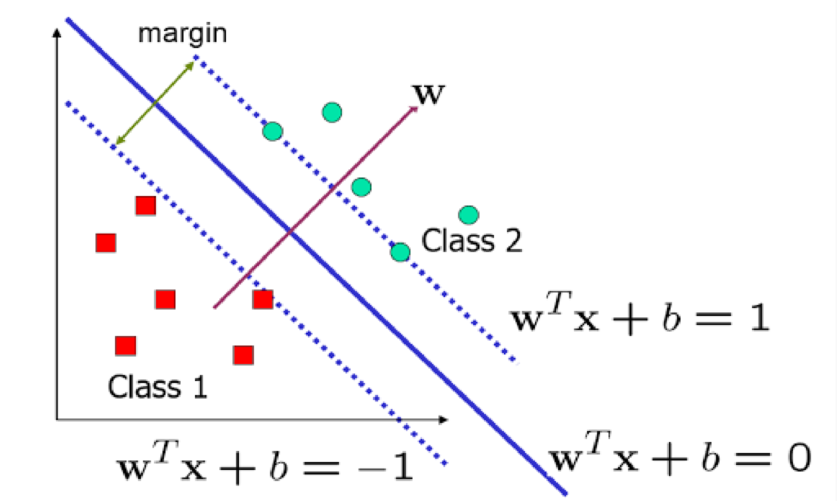
$$r_i = y_i \frac{\mathbf{w}^T \mathbf{x}_i + b}{\|\mathbf{w}\|}$$



Support Vector Machine (SVM)
Concept of margin…

- Let us assume that the minimum distance of the hyperplane from any instance is r.

$$r_i = y_i \frac{\mathbf{w}^T \mathbf{x}_i + b}{\|\mathbf{w}\|} \geq r$$



- Instances that are closest to the hyperplane are:
  - Support vectors - at r distance from the hyperplane.
- The margin is defined as the distance between the support vectors and is given as:

$$m = 2r = \frac{2}{\|\mathbf{w}\|}$$

- From those two equations, we can derive: $y_i \frac{\mathbf{w}^T \mathbf{x}_i + b}{\|\mathbf{w}\|} \geq \frac{1}{\|\mathbf{w}\|} = y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1$

Linear SVM
SVM formulation (linearly separable data)

- SVM aims to find a hyperplane (w, b)
  - The margin $\frac{2}{||\mathbf{w}||}$ is maximised
  - Satisfying the constraint $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1$
- SVM formulation therefore solves the following optimisation problem:

$$Minimize\ \frac{1}{2}||\mathbf{w}||^2$$

$$\text{subject to}\quad \mathrm{y}_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \quad \forall i$$

- We need to optimise a quadratic function in *w* subject to linear constraints.

Linear SVM
SVM formulation (linearly separable data)

- This problem is well known in optimisation community and is called quadratic programming
  - Do not get confused with the word programming.
  - It just means optimisation.
- The optimisation problem is often solved by constructing an equivalent problem called dual problem.
  - In quadratic programming, the original optimisation problem is called the primal problem.
  - The solution to the dual problem provides a lower bound to the solution of the primal (minimisation) problem.

Linear SVM
Dual optimisation problem

- Primal problem in SVM which is maximising the margin (or minimising 1/margin):

$$Minimize \ \frac{1}{2}\|\mathbf{w}\|^2$$
$$subject\ to\ \ y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \quad \forall i$$

- Using Lagrange multipliers
  - Convert a constrained optimisation into an unconstrained optimisation problem.
- The Lagrange function to minimise is:

$$\mathrm{L}(\mathbf{w}, \mathrm{b}, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{n} \alpha_i(1 - y_i(\mathbf{w}^T\mathbf{x}_i + b))$$

where $\alpha = [\alpha_1, \dots, \alpha_n]$ are Lagrange multipliers.

- The Lagrange function to minimise is:

Dual problem

$$\min_\alpha \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{i=1}^{n} \alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j$$
$$\text{subject to: } \sum_{i=1}^{n} \alpha_i y_i = 0 \ \ and \ \ \alpha_i \geq 0 \ \ for\ all\ i$$

Linear SVM
Dual optimisation problem...

- If you carefully look at the derived problem:
  - x_i and x_j as our data points are multiplied with dot product
    - Represents the similarity of these two vectors
- With the help of the Lagrange multipliers
  - A dual problem for minimising $\frac{1}{2}||\mathrm{w}||^2$
    - Maximising the margin in SVM

Linear SVM
Dual optimisation problem…

- Now, let's say we have found the α values, so given a solution to $\alpha = [\alpha_1, \ldots, \alpha_n]$ the hyperplane w is given as:

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

$$b = y_k - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i^T \mathbf{x}_k$$

using any k such that αk>0

- There is one α_i corresponding to each x_i.
  - The x_i corresponding to each non-zero α_i is a support vector.

Linear SVM
Dual optimisation problem...

- Given w and b, we can write classification function as:

$$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b$$

- Solving α and b
  - Used training data only in the form of dot products $\mathbf{x}_i^T \mathbf{x}_j$
- The classification function
  - Uses a dot product between x and support vectors x_i.

Linear SVM
Dual optimisation problem...

- We are not covering the details of the optimisation here because it requires deeper understanding of quadratic programming.
- However, you should know that the primal problem has computational requirements of the order $O(d^3)$, whereas the dual problem requires an order $O(n^3)$.

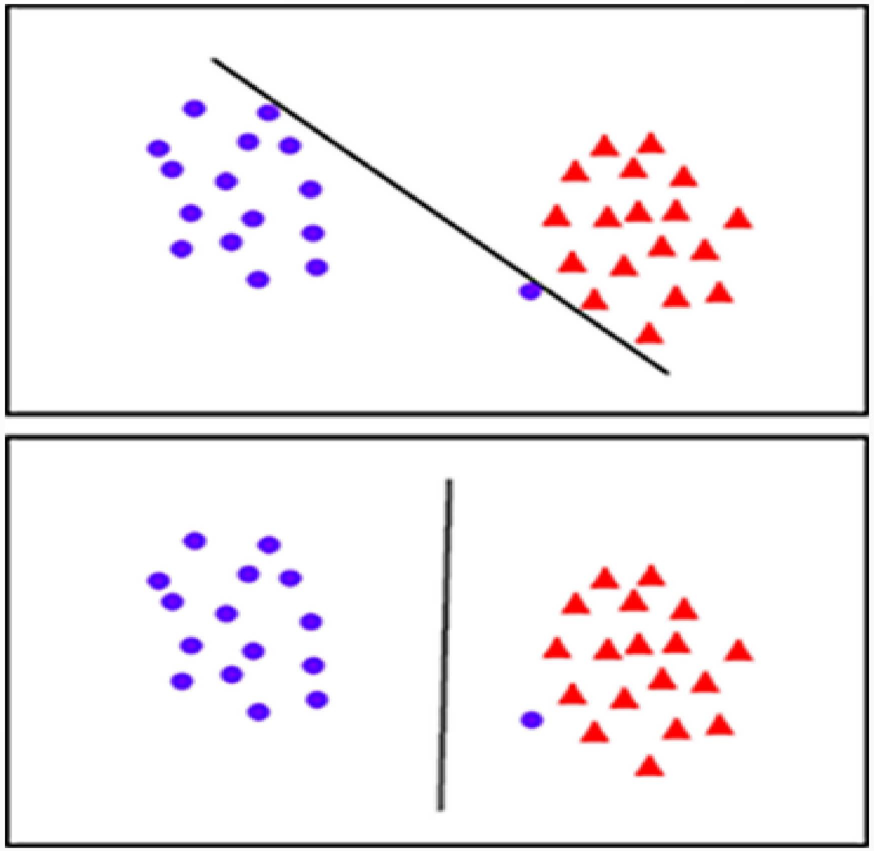
  Where d is the dimension of feature space and n is the number of instances in the training set.
- Dual problem is popular
  - It allows the use of arbitrary Kernels (How?)
    - SVM boundaries can be significantly nonlinear

Support Vector Machine (SVM)

- Linear SVMs
  - Perfectly separable data points.
  - Almost separable data points.
- Nonlinear SVMs.

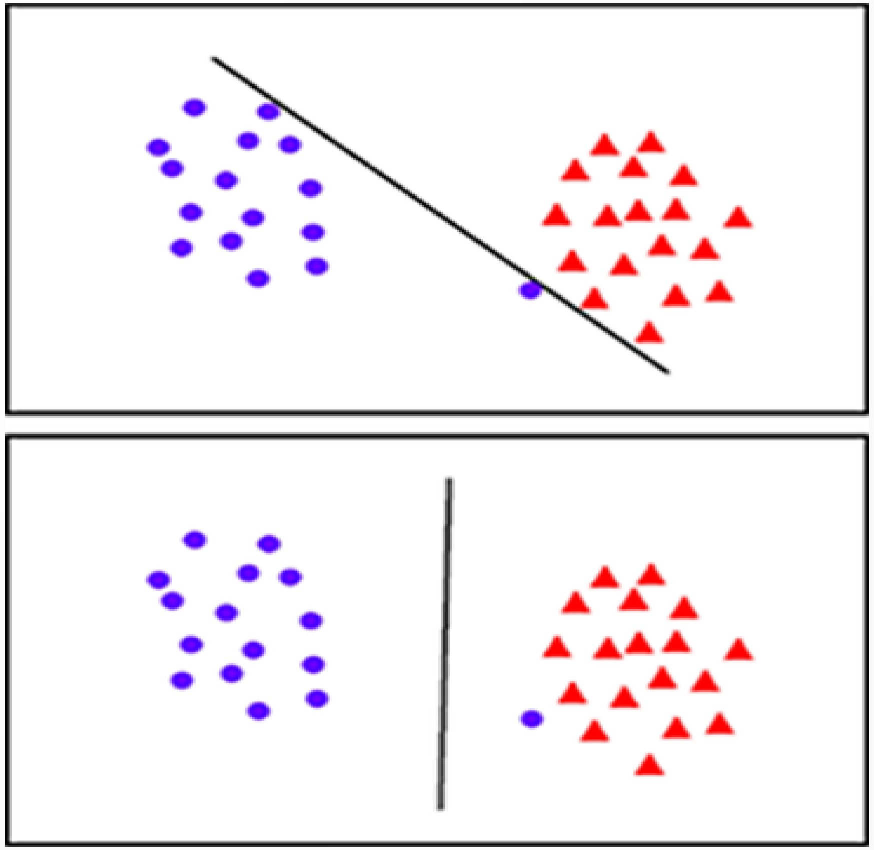
Linear SVM
SVM formulation (linearly non-separable data)

- In SVM, we have so far assumed that data is linearly separable.
- What approach should we take when data can be linearly separable but with a narrow margin?



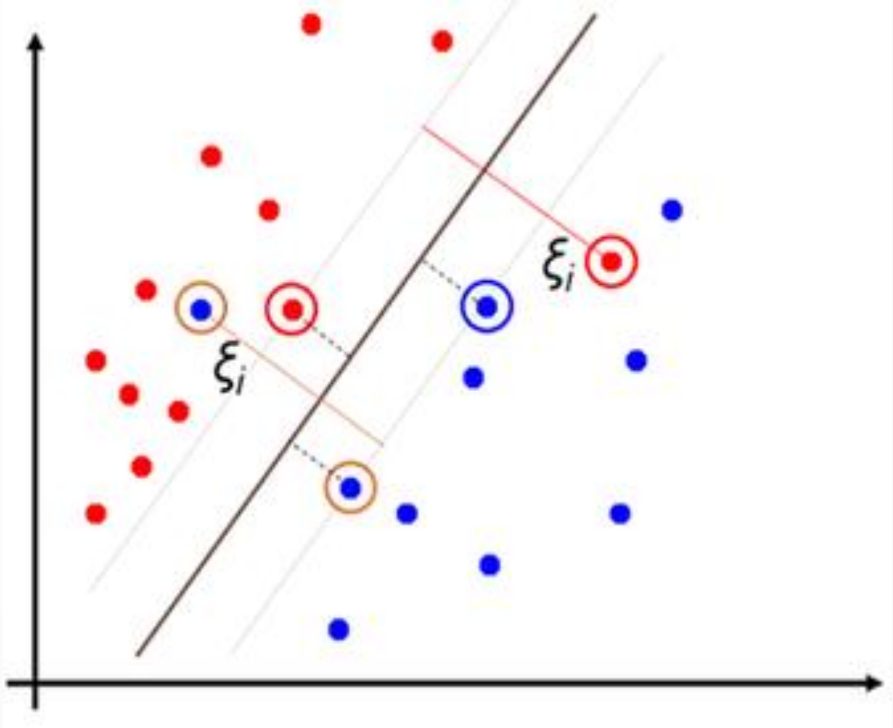
Linear SVM
SVM formulation (linearly non-separable data)

- No interference with the boundary even with small noisy data points or outliers.
- Acceptable to have large margins even though some of the constraints are violated.
- In practice, we need a trade-off between the margin and the number of errors in classifying the training instances.



Linear SVM
SVM formulation (linearly non-separable data)

- Margin – classification error trade-off
  - Soft margin concept
    - allowing some of the data points to cross the borders
      and to be in the wrong side of the boundary or to be mis-classified
    - Slack variables ζ_i are added to allow mis-classification of outliers, noisy or difficult to classify instances.



Linear SVM
SVM formulation (linearly non-separable data)

- Allow some of the training instances to be mis-classified
  - Minimize the sum of slack variables
  - Data points for which $\zeta_i \neq 0$
    - mis-classified
- SVM with soft margin uses the following formulation:

Find $\mathbf{w}$ and b such that

$\Phi(\mathbf{w}) = \mathbf{w}^T\mathbf{w} + C\Sigma\xi_i$ is minimized

and for all $(\mathbf{x}_i, y_i)$, $i=1..n$: $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i$, $\xi_i \geq 0$

Linear SVM
SVM formulation (linearly non-separable data)

- The parameter C can be used as a way to achieve the trade-off between large margin and fitting training data.
  - High values of C
    - Highly penalize the mis-classification.
  - Small values of C
    - Allows more mis-classifications.

Linear SVM
Soft margin dual problem

- The dual problem is given as:

$$\min_\alpha \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{i=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

$$Subject\ to: \sum_{i=1}^n \alpha_i y_i = 0 \quad and \quad 0 \leq \alpha_i \leq C \quad \forall i$$

- Given a solution to $\alpha = [\alpha_1, \dots, \alpha_n]$ the hyperplane w is given as:

$$\mathbf{w} = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i$$

$$b = y_k(1 - \zeta_k) - \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i^T \mathbf{x}_k \ using\ any\ k\ such\ that\ \alpha_k > 0$$

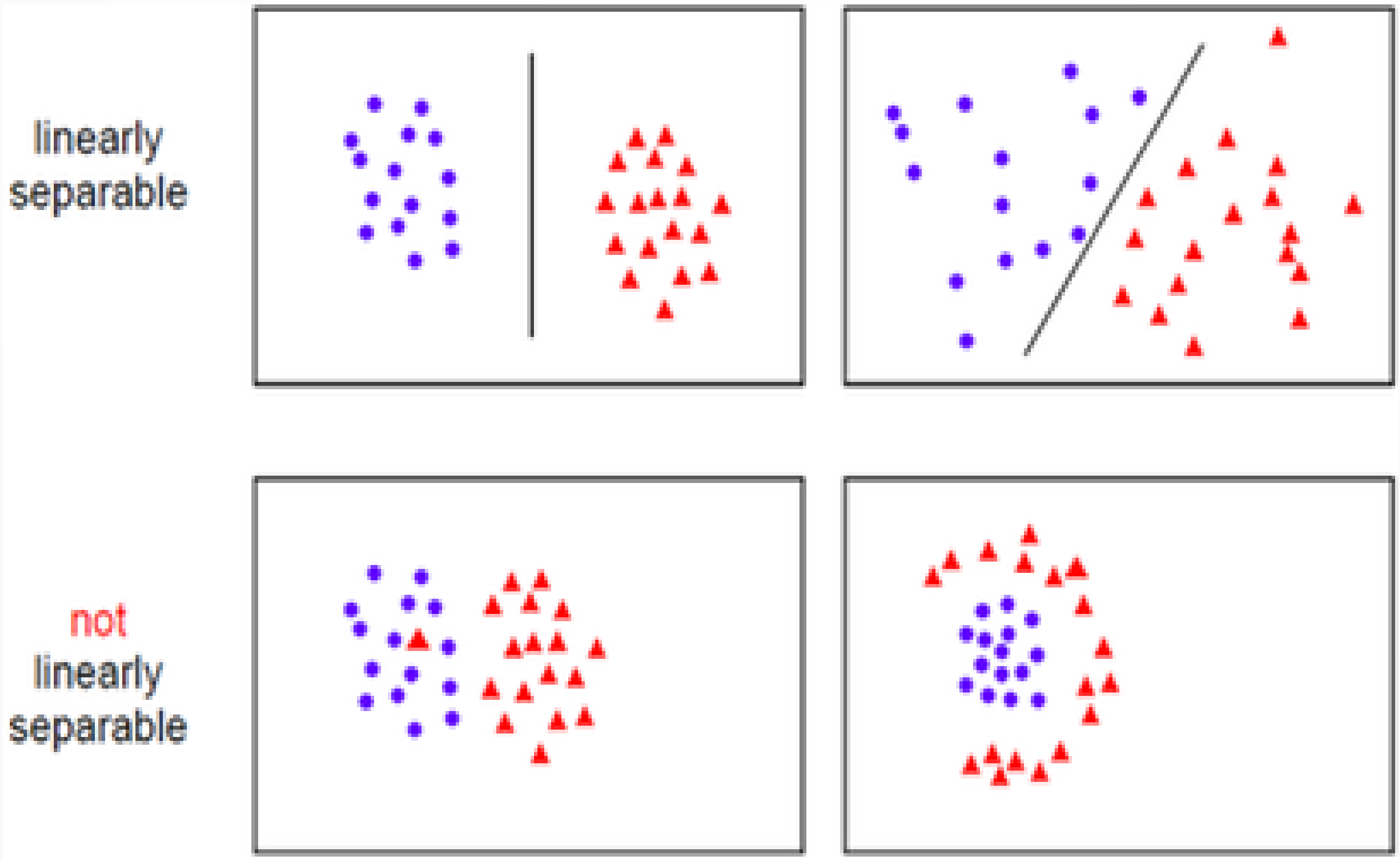
- Given w and b, we can write the classification function as:

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b$$

Support Vector Machine (SVM)

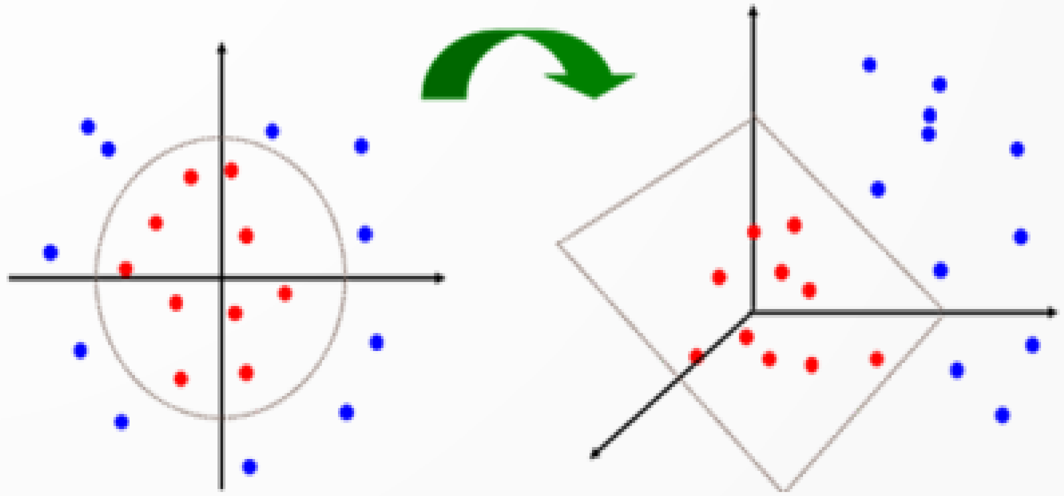
- Linear SVMs
  - Perfectly separable data points.
  - Almost separable data points.
- Nonlinear SVMs.

Nonlinear SVM
Kernel trick and non-linear SVM...



Nonlinear SVM
Kernel trick and non-linear SVM…

- The key idea:
  - To handle non-linearity
    - transform the features to a higher dimensional space where data is linearly separable (see figure below).
  - This figure illustrates a 2D space in which the data points can only be separated through a nonlinear curve.
  - However by transforming these data points to a 3D space, it looks that data points are now linearly separable!



Nonlinear SVM
Kernel trick and non-linear SVM...

- A kernel function is a function that is used to compute dot products in a high dimensional feature space.
- SVM performs the required transformation <u>implicitly</u> by using <u>kernels</u> $\Phi : \mathbf{x} \to \phi(\mathbf{x})$
  - Since data participates in computations only in the form of dot products
    - all the computations are performed via kernels.
- Dual problem of Nonlinear SVM
  - Find $\alpha_1, \alpha_2, ..., \alpha_n$ such that

$$\mathbf{Q}(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

$$\min_\alpha \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{i=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$
$$\text{subject to: } \sum_{i=1}^n \alpha_i y_i = 0 \quad and \quad \alpha_i \geq 0 \quad for\ all\ i$$

Nonlinear SVM
Kernel function…

- What functions are kernel functions?
- Mercer's theorem says that:
  - "every positive, semi-definite and symmetric function is a kernel function"
- Kernel functions when evaluated on each pair of data instances give rise to a matrix called a Gram matrix.
- This matrix (denoted as K) is a positive semi-definite and symmetric matrix.

$$K = \begin{array}{|c|c|c|c|c|}\hline k(\mathbf{x}_1,\mathbf{x}_1) & k(\mathbf{x}_1,\mathbf{x}_2) & k(\mathbf{x}_1,\mathbf{x}_3) & \cdots & k(\mathbf{x}_1,\mathbf{x}_n) \\ \hline k(\mathbf{x}_2,\mathbf{x}_1) & k(\mathbf{x}_2,\mathbf{x}_2) & k(\mathbf{x}_2,\mathbf{x}_3) & & k(\mathbf{x}_2,\mathbf{x}_n) \\ \hline & & & & \\ \hline \cdots & \cdots & \cdots & \cdots & \cdots \\ \hline k(\mathbf{x}_n,\mathbf{x}_1) & k(\mathbf{x}_n,\mathbf{x}_2) & k(\mathbf{x}_n,\mathbf{x}_3) & \cdots & k(\mathbf{x}_n,\mathbf{x}_n) \\ \hline \end{array}$$

Nonlinear SVM
Kernel function...

- Some popular kernel functions where $k(\mathbf{x}_i, \mathbf{x}_j) = \phi^T(\mathbf{x}_i)\phi(\mathbf{x}_j)$
  - Linear Kernel:

    $$k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j, Mapping \qquad \phi(\mathbf{x}) = x$$

  - Polynomial Kernel with degree p:

    $$k(\mathbf{x}_i, \mathbf{x}_j) = (1 + \mathbf{x}_i^T \mathbf{x}_j)^p$$

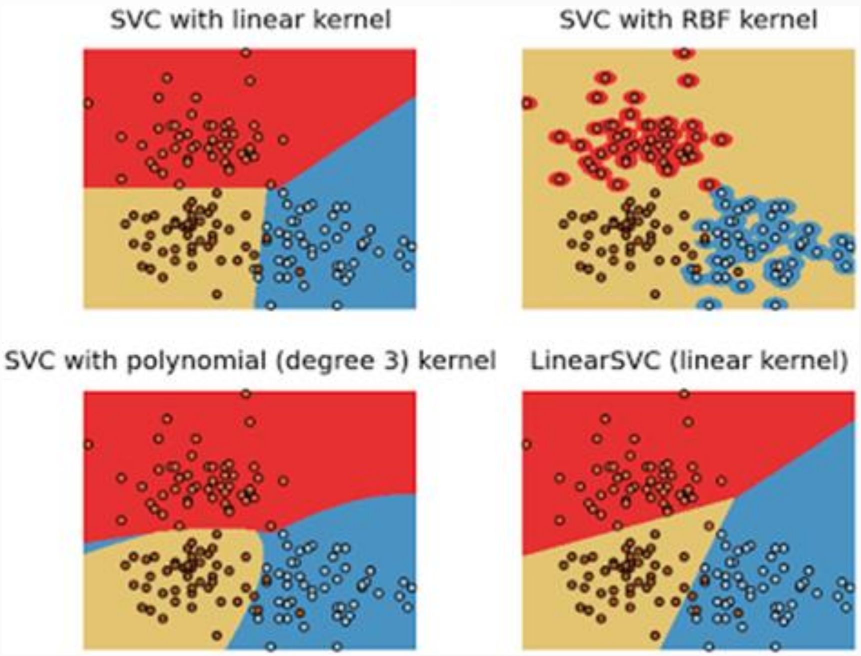
  - Radial basis function (RBF) kernel:

    $$k(\mathbf{x}_i, \mathbf{x}_j) = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}}$$

    Mapping φ(x) is infinite dimensional.

Nonlinear SVM
Summary

- SVM fits a linear hyperplane in high dimensional space
  - In original space the boundaries are nonlinear
- Use of the linear kernel
  - generates linear boundaries
- Polynomial kernel with degree 3
  - the SVM came up with curve boundaries.

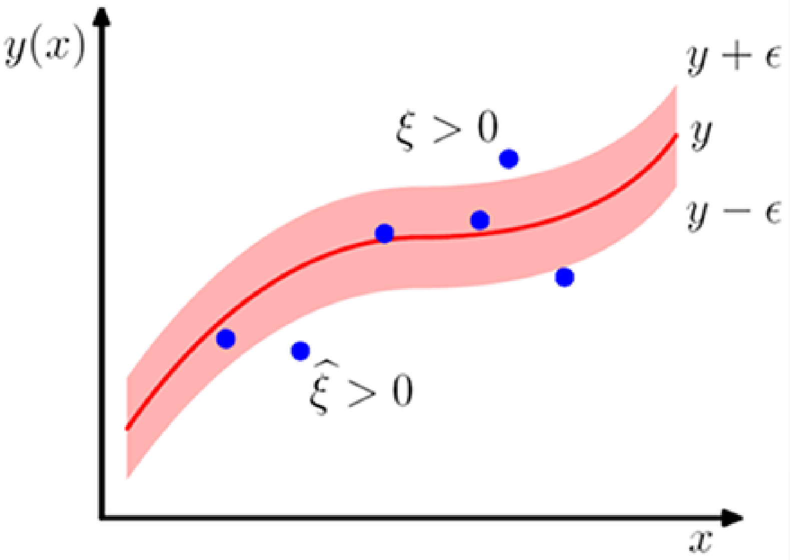


All these outputs are on the Iris data set which is a multivariate data set introduced by the British statistician and biologist Ronald Fisher.

Support vector regression

Nonlinear SVM
Support vector regression…

- Support Vector Regression (SVR)
  - The classifier still defines a margin (looks like a tube in the image)
  - If the data points are in the tube
    - they are clear and fine,
  - If a data point goes outside of the tube or margin
    - the point is a deviation.
- This is similar to the concept in almost separable data points in previous lesson.



Nonlinear SVM
Support vector regression…

- Uses a ζ and try to look for a right complexity and also accurate results.
- Enforcing a very small numbers of deviations or mis-classified points
    - Will create a margin or a tube
        - Contains most of the points
        - Increases model complexity and the risk of overfitting.

Nonlinear SVM
Support vector regression…

- The SVR formulation uses the following linear model:

$$f(x) =< \mathbf{w}, \mathbf{x}w, x > +b \quad with\ w \in X, b \in \mathbb{R}$$

- It assumes that small errors ϵ are acceptable and minimises the flatness of the function through:

$$Minimize \quad \frac{1}{2}||\mathbf{w}||^2$$

$$subject\ to \quad \begin{cases} y_i - < \mathbf{w}, \mathbf{x}w, x_i > -b, & \leq \epsilon \\ < \mathbf{w}, \mathbf{x}w, x_i > +b - y_i & \leq \epsilon \end{cases}$$

-

Statistical Learning Theory of SVM

Nonlinear SVM
Statistical learning theory of SVM

- Theoretically, does maximum margin make sense ?
- Structural risk minimization
  - seeks to prevent over-fitting by incorporating a penalty on the model complexity.
- Also the general idea is to minimize the structural risk as:

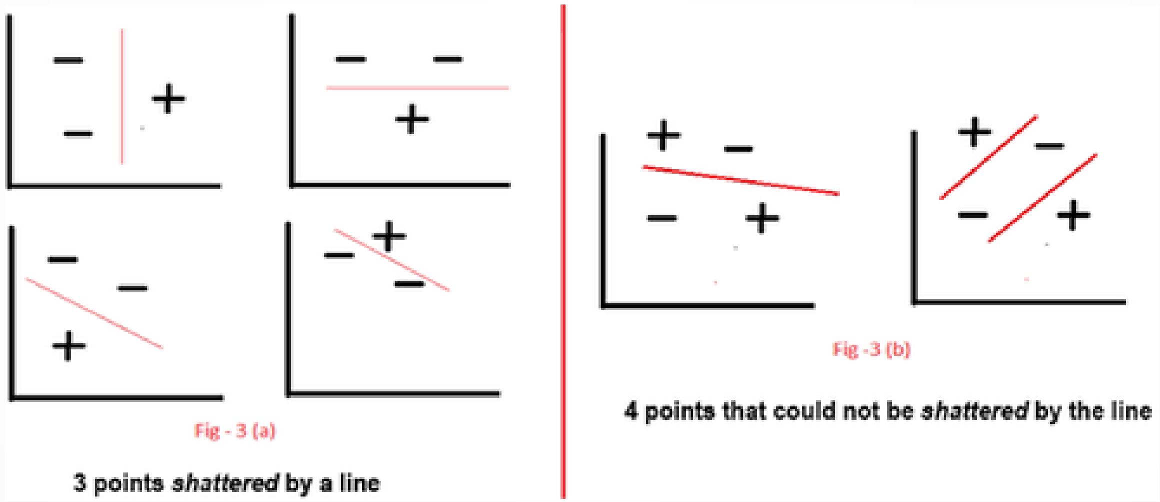
$$R_{str}(f) = R_{emp}(f) + \lambda h(f)$$

  where $h(f)$ is the complexity of hypothesis function f and λ is a penalty parameter.
- So we would like choose a model with small error and less complex.

Nonlinear SVM
Vapnik-Chervonenkis (VC) Dimension

- Suppose we pick n instances and assign labels of + and – to them randomly.
- If our hypothesis class is rich enough to learn any association of labels to the data, it is sufficiently complex.
- How about we characterize the complexity of the hypothesis class by looking at how many instances it can shatter(i.e. can fit perfectly for all possible label assignments).
- The number of instances a hypothesis class can shatter is called its Vapnik-Chervonenkis (VC) Dimension.

Nonlinear SVM
An Illustration of VC Dimension



- In the left figure:
  - 3 points with any combination of labels can be separated by a line.
    - No matter even you change the labels of the data points.
- But in the right figure
  - a single line cannot separate these data points.
- Therefore, VC dimension of a line in 2−dimension is 3
  - In d−dimension: d+1

Nonlinear SVM
Summary of VC Dimension

- The theoretical justification for maximum margin has shown by vapnik in the following results.
- The class of optimal linear separators have VC dimension h, bounded as:

$$\mathrm{h} \leq min\{d, \left\lceil \frac{D^2}{\rho^2} \right\rceil\} + 1$$

Where ρ is the margin, D is the diameter of the smallest sphere that can enclose all of the training examples, and d is the dimensionality.

- Intuitively, this implies that regardless of dimensionality d, we can minimize the model complexity (VC dimension) by maximizing the margin ρ.
- If ρ is maximized or in other words,
  - a classifier with high margins
    - a smaller value for the complexity of model h

Thank You.