Faculty Development Program

on

Machine Learning and Image Processing

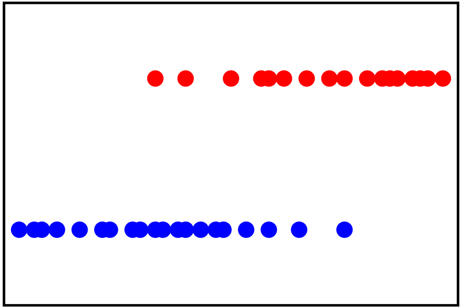
Support Vector Machine

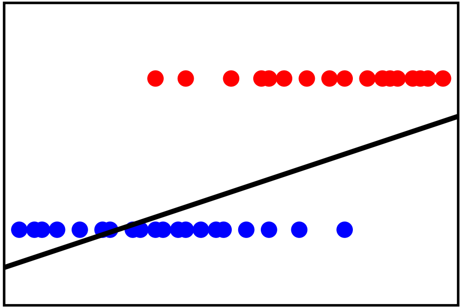
Logistic regression

- Responses may be qualitative (categorical)
  - Example: $\langle$Hours of study, pass/fail$\rangle$, $\langle$MRI scan, benign/malignant$\rangle$
  - Output should be 0 or 1
- Predicting qualitative response is known as classification
- Linear regression does not help

Issues with linear regression

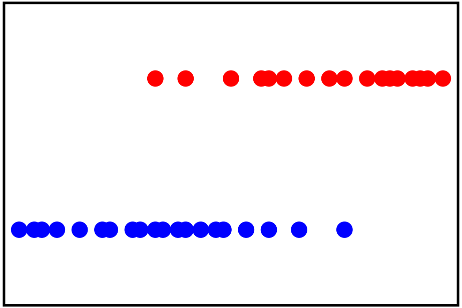
- **Linear regression can predict values as ∞ or $-\infty$**

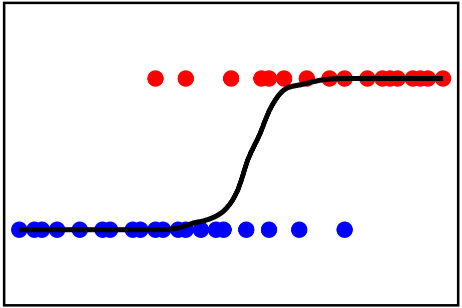




Logistic regression

- **Predicted value should be within** 0 **and** 1





Logistic model

- **Linear regression model to represent probability** $p(x) = w_0 + w_1 x$
- **To avoid problem, we use function** $p(x) = \dfrac{e^{w_0 + w_1 x}}{1 + e^{w_0 + w_1 x}}$
- **Quantity** $\frac{p(x)}{1-p(x)} = e^{w_0 + w_1 x}$ **is known as odds**
- **Taking** $\log$ **on both the sides, we get** $\log\left(\dfrac{p(x)}{1-p(x)}\right) = w_0 + w_1 x$
- **Coefficient can be determined using maximum likelihood**
  - $l(w_0, w_1) = \prod_{i:y_i=1} p(x_i) \prod_{j:y_j=0} p(x_j)$

Logistic model (contd.)

- **Similar to linear regression except the output is mapped between 0 and 1 ie.**

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \sigma(\boldsymbol{\theta}^T \boldsymbol{x})$$

where $\sigma(x) = \dfrac{1}{1 + \exp(-x)}$ **(Sigmoid function)**

Support Vector Machine

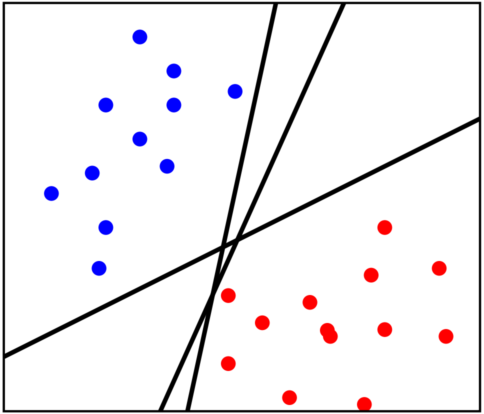
- An approach for classification
- Developed in 1990s
- Generalization of maximum margin classifier
  - Mostly limited to linear boundary
- Support vector classifier — broad range of classes
- SVM — Non-linear class boundary

Hyperplane

- **In n dimensional space a hyperplane is a flat affine subspace of dimension $n-1$**
- **Mathematically it is defined as**
  - **For 2 dimensions —** $w_0 + w_1x_1 + w_2x_2 = 0$
  - **For n dimensions —** $w_0 + w_1x_1 + \ldots + w_nx_n = 0$

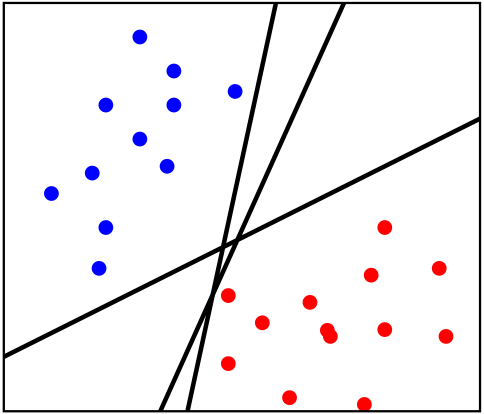
Classification using Hyperplane

- **Assume, *m* training observation in *n* dimensional space**



Classification using Hyperplane

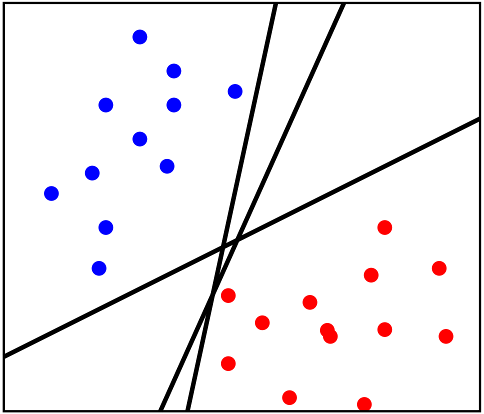
- **Assume, m training observation in n dimensional space**
- **Separating hyperplane has the property**
  - $w_0 + w_1 x_1 + \ldots + w_n x_n > 0$ **if** $y_i = 1$
  - $w_0 + w_1 x_1 + \ldots + w_n x_n < 0$ **if** $y_i = -1$



Classification using Hyperplane

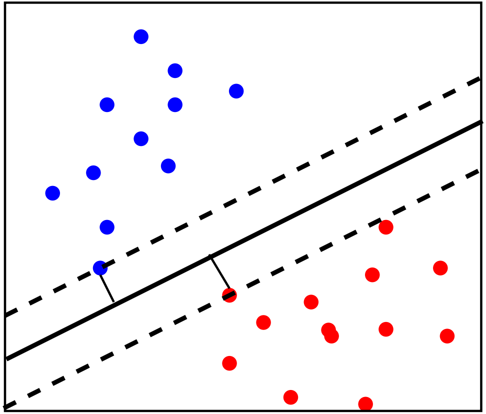
- Assume, m training observation in n dimensional space
- Separating hyperplane has the property
  - $w_0 + w_1x_1 + \ldots + w_nx_n > 0$ **if** $y_i = 1$
  - $w_0 + w_1x_1 + \ldots + w_nx_n < 0$ **if** $y_i = -1$
- Hence, $y_i(w_0 + w_1x_1 + \ldots + w_nx_n) > 0$
- Classification of test observation x^* is done based on the sign of

  $$f(\boldsymbol{x}^*) = w_0 + w_1x_1^* + \ldots + w_nx_n^*$$
- Magnitude of $f(\boldsymbol{x}^*)$
  - Far from 0 — Confident about prediction
  - Close to 0 — Less certain



Maximal margin classifier

- Also known as optimal separating hyper-plane
- Separating hyperplane farthest from training observation
  - Compute perpendicular distance from training point to the hyperplane
  - Smallest of these distances represents the margin
- Target is to find the hyperplane for which the margin is the largest



Construction of maximal margin classifier

- **Input — m points in n dimension space ie. $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$**
- **Input — labels $y_1, y_2, \ldots, y_m$ for each point $\boldsymbol{x}_i$ where $y_i \in \{-1, 1\}$**
- **Need to solve the following optimization problem**

$$\max_{w_0, w_1, \ldots, w_n, M} \quad M$$

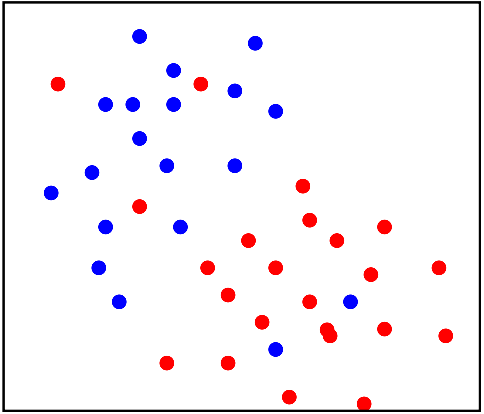
subject to

$$y_i(w_0 + w_1 x_{i1} + w_{i2} x_{i2} + \ldots + w_{in} x_{in}) \geq M \quad \forall i = 1, \ldots, m$$

$$\sum_{i=1}^{n} w_i^2 = 1$$

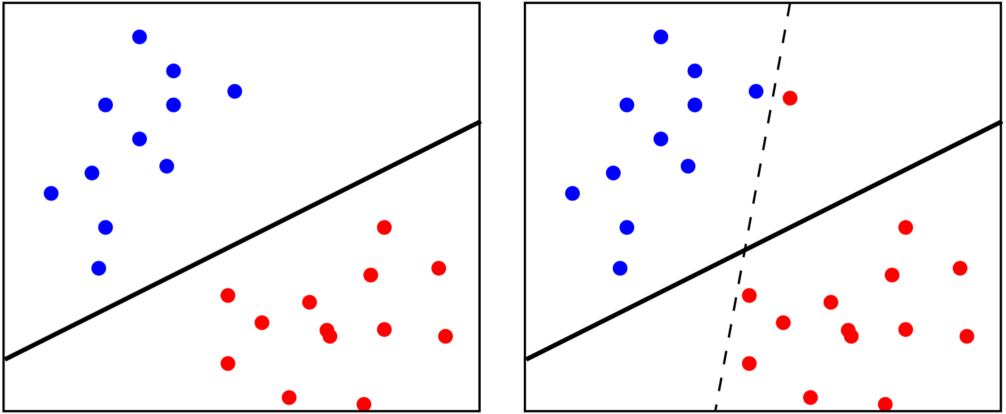
Issues

- **Maximal margin classifier fails to provide classification in case of overlap**



Issues

- **Single observation point can change the hyperplane drastically**

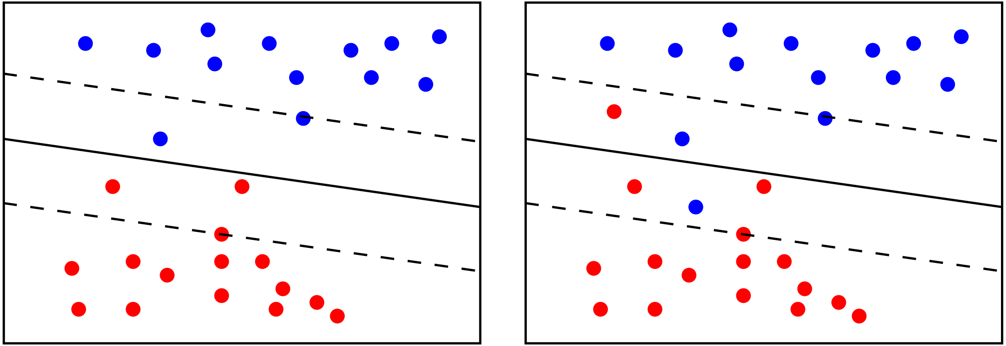


Support Vector Classifier

- Provides greater robustness to individual observations
- Better classification of most of the training observations
- Worthwhile to misclassify a few training observations
- Also known as soft margin classifier

Support Vector Classifier

- **Points can lie within the margin or wrong side of hyperplane**



Optimization with misclassification

- **Input —** $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$ **and** $y_1, y_2, \ldots, y_m$
- **Need to solve the following optimization problem**

$$\max_{w_0, w_1, \ldots, w_n, M} \quad M$$

subject to

$$y_i(w_0 + w_1 x_{i1} + \ldots + w_{in} x_{in}) \geq M(1 - \epsilon_i) \quad \forall i = 1, \ldots, m$$

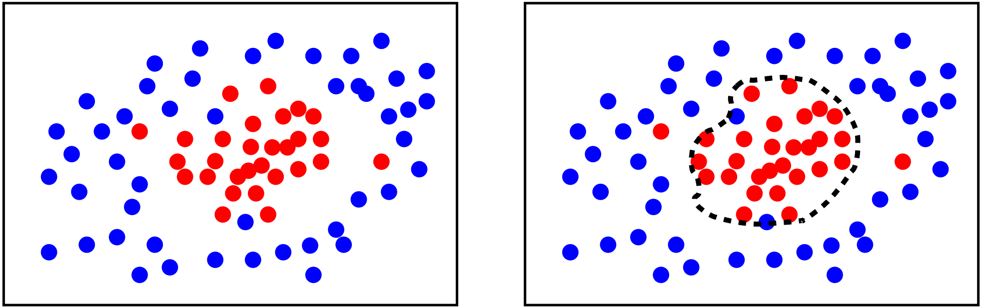
$$\sum_{i=1}^{n} w_i^2 = 1, \quad \sum_{i=1}^{m} \epsilon_i = C$$

- C **is non-negative tuning parameter,** ϵ_i **- slack variable**
- **Classification of test observation remains the same**

Observations

- $\epsilon_i = 0$ — ith observation is on the correct side of margin
- $\epsilon_i > 0$ — ith observation is on the wrong side of margin
- $\epsilon_i > 1$ — ith observation is on the wrong side of hyperplane
- C — budget for the amount that the margin can be violated by m observations
    - $C = 0$ — No violation, ie. maximal margin classifier
    - $C > 0$ — No more than C observation can be on the wrong side of hyperplane
    - C is small — Narrow margin, highly fit to data, low bias and high variance
    - C is large — Fitting data is less hard, more bias and may have less variance

Classification with non-linear boundaries



Classification with non-linear boundaries

- Performance of linear regression can suffer for non-linear data
- Feature space can be enlarged using function of predictors
  - For example, instead of fitting with $x_1, x_2, \ldots, x_n$ features we could use $x_1, x_1^2, x_2, x_2^2 \ldots, x_n, x_n^2$ as features
- Optimization problem becomes

$$\max_{w_0, w_{11}, w_{12} \ldots, w_{n1}, w_{n2}, \epsilon_i, M} \quad M$$

subject to

$$y_i \left(w_0 + \sum_{j=1}^{n} w_{j1} x_{ij} + \sum_{j=1}^{n} w_{j2} x_{ij}^2 \right) \geq M(1 - \epsilon_i) \quad \forall i = 1, \ldots, m$$

$$\sum_{i=1}^{n} \sum_{j=1}^{2} w_{ij}^2 = 1, \quad \sum_{i=1}^{m} \epsilon_i \leq C, \quad \epsilon_i \geq 0$$

Support Vector Machine

- **Extension of support vector classifier that results from enlarging feature space**
- **It involves inner product of the observations** $f(x) = w_0 + \sum_{i=1}^{m} \alpha_i \langle \mathbf{x}, \mathbf{x}_i \rangle$ **where** α_i **- one per training example**
  - **To estimate** α_i **and** w_0**, we need** $m(m-1)/2$ **inner products,** $\langle \mathbf{x}_i, \mathbf{x}_{i'} \rangle$
- **It turns out that** $\alpha_i \neq 0$ **for support vectors**

  $f(x) = w_0 + \sum_{i \in S} \alpha_i \langle \mathbf{x}, \mathbf{x}_i \rangle$ **where** S **- set of support vectors**

Support Vector Machine

- Inner product is replaced with kernel, K or $K(\boldsymbol{x}_i, \boldsymbol{x}_{i'})$
- Kernel quantifies similarity between observations $K(\boldsymbol{x}_i, \boldsymbol{x}_{i'}) = \sum_{j=1}^{n} x_{ij}x_{i'j}$
  - Above one is Linear kernel ie. Pearson correlation
- Polynomial kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_{i'}) = \left(1 + \sum_{j=1}^{n} x_{ij}x_{i'j}\right)^d$ where d is positive integer > 1
- Support vector classifier with non-linear kernel is known as support vector machine and the function will look

$$f(x) = w_0 + \sum_{i \in S} \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i)$$

- Radial kernel: $K(\boldsymbol{x}_i, \boldsymbol{x}_{i'}) = \exp\left(-\gamma \sum_{i=1}^{n} (x_{ij} - x_{i'j})^2\right)$ where $\gamma > 0$

Summary

- SVM is good for the data that are linearly separable
- Kernel trick helps to classify non-linear data
- It is one of the preferred choice for classification problem