SVM

- Said to start in 1979 with Vladimir Vapnik’s paper
- Major developments throughout 1990’s
- Elegant theory
  - Has good generalization properties
  - Have been applied to diverse problems very successfully in the last 10-15 years
- One of the most important developments in pattern recognition in the last 10 years

Linear Discriminant Functions

- A discriminant function is linear if it can be written as
  \[ g(x) = w^T x + w_0 \]
- \( g(x) > 0 \Rightarrow x \in \text{class } 1 \)
- \( g(x) < 0 \Rightarrow x \in \text{class } 2 \)

- which separating hyperplane should we choose?

Linear Discriminant Functions

- Training data is just a subset of all possible data
- Suppose hyperplane is close to sample \( x_i \)
- If we see new sample close to sample \( i \), it is likely to be on the wrong side of the hyperplane

- Poor generalization (performance on unseen data)
**Linear Discriminant Functions**
- Hyperplane as far as possible from any sample
- New samples close to the old samples will be classified correctly
- Good generalization

**SVM: Linearly Separable Case**
- SVM: maximize the margin
  - Margin is twice the absolute value of distance $b$ of the closest example to the separating hyperplane
  - Better generalization (performance on test data)
    - In practice
    - In theory

**SVM**
- Idea: maximize distance to the closest example
- For the optimal hyperplane
  - Distance to the closest negative example = distance to the closest positive example
- Support vectors are the samples closest to the separating hyperplane
  - They are the most difficult patterns to classify
  - Optimal hyperplane is completely defined by support vectors
    - Of course, we do not know which samples are support vectors without finding the optimal hyperplane
**SVM: Formula for the Margin**

- \( g(x) = w^T x + w_0 \)
- absolute distance between \( x \) and the boundary \( g(x) = 0 \)
  \[
  \frac{w^T x + w_0}{||w||}
  \]
- distance is unchanged for hyperplane
  \[g_1(x) = \alpha g(x)\]
  \[
  \frac{\alpha w^T x + \alpha w_0}{||w||} = \frac{w^T x + w_0}{||w||}
  \]
- Let \( x_i \) be an example closest to the boundary. Set
  \[w^T x_i + w_0 = 1\]
- Now the largest margin hyperplane is unique

\[
\text{Let} \quad x^{(1)}_i = \text{example closest to the boundary. Set} \quad w^T x^{(1)}_i + w_0 = 1
\]

\[
\text{Now the largest margin hyperplane is unique}
\]

**SVM: Optimal Hyperplane**

- Maximize margin \( m = \frac{2}{||w||} \)
- subject to constraints
  \[
  \begin{align*}
  w^T x_i + w_0 & \geq 1 & \text{if} \ x_i \text{is positive example} \\
  w^T x_i + w_0 & \leq -1 & \text{if} \ x_i \text{is negative example}
  \end{align*}
  \]
- Let \( z_i = 1 \) if \( x_i \) is positive example
  \( z_i = -1 \) if \( x_i \) is negative example
- Can convert our problem to
  \[
  \text{minimize} \quad J(w) = \frac{1}{2}||w||^2
  \]
  \[
  \text{constrained to} \quad z_i (w^T x_i + w_0) \geq 1 \quad \forall i
  \]
- \( J(w) \) is a quadratic function, thus there is a single global minimum

**SVM: Formula for the Margin**

- For uniqueness, set \( |w^T x_i + w_0| = 1 \) for any example \( x_i \) closest to the boundary
- now distance from closest sample \( x_i \) to \( g(x) = 0 \) is
  \[
  \frac{|w^T x_i + w_0|}{||w||} = \frac{1}{||w||}
  \]
- Thus the margin is
  \[
  m = \frac{2}{||w||}
  \]

**SVM: Optimal Hyperplane**

- Use Kuhn-Tucker theorem to convert our problem to:
  \[
  \text{maximize} \quad L_0(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j z_i z_j x_i^T x_j
  \]
  \[
  \text{constrained to} \quad \alpha_i \geq 0 \quad \forall i \quad \text{and} \sum_{i=1}^{n} \alpha_i z_i = 0
  \]
- \( \alpha = \{\alpha_1, \ldots, \alpha_n\} \) are new variables, one for each sample
- Can rewrite \( L_0(\alpha) \) using \( n \) by \( n \) matrix \( H \):
  \[
  L_0(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j H_{ij}
  \]
  \[
  \text{where the value in the} \ i\text{th row and} \ j\text{th column of} \ H \text{is} \quad H_{ij} = z_i z_j x_i^T x_j
  \]
**SVM: Optimal Hyperplane**

- Use Kuhn-Tucker theorem to convert our problem to:
  
  \[
  \text{maximize} \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \mathbf{x}_i^T \mathbf{x}_j - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \mathbf{x}_i^T \mathbf{x}_j \\
  \text{subject to} \quad \alpha_i \geq 0 \quad \forall i \text{ and } \sum_{i=1}^{n} \alpha_i \mathbf{x}_i = 0
  \]

- \(\alpha = (\alpha_1, \ldots, \alpha_n)\) are new variables, one for each sample
- \(L_D(\alpha)\) can be optimized by quadratic programming
- \(L_D(\alpha)\) formulated in terms of \(\alpha\)
  - It depends on \(\mathbf{w}\) and \(\mathbf{w}_0\) indirectly

**SVM: Optimal Hyperplane**

- \(L_D(\alpha)\) depends on the number of samples, not on dimension of samples
- Samples appear only through the dot products \(\mathbf{x}_i^T \mathbf{x}_j\)
- This will become important when looking for a **nonlinear** discriminant function, as we will see soon
- Code available on the web to optimize

**SVM: Non Separable Case**

- Data is most likely to be not linearly separable, but linear classifier may still be appropriate

- Can apply SVM in non linearly separable case
  - Data should be “almost” linearly separable for good performance
SVM: Non Separable Case

- Use non-negative slack variables $\xi_1, \ldots, \xi_n$ (one for each sample)
- Change constraints from $z_i(w^Tx_i + w_0) \geq 1$ to $z_i(w^Tx_i + w_0) \geq 1 - \xi_i$ for all $i$
- $\xi_i$ is a measure of deviation from the ideal for sample $i$
  - $\xi_i > 1$ sample $i$ is on the wrong side of the separating hyperplane
  - $0 < \xi_i < 1$ sample $i$ is on the right side of separating hyperplane but within the region of maximum margin

Unfortunately this minimization problem is NP-hard due to discontinuity of functions $I(\xi_i)$

$$J(w, \xi_1, \ldots, \xi_n) = \frac{1}{2} ||w||^2 + \beta \sum_{i=1}^{n}(\xi_i > 0)$$

- where $I(\xi_i > 0) = \begin{cases} 1 & \text{if } \xi_i > 0 \\ 0 & \text{if } \xi_i \leq 0 \end{cases}$
- constrained to $z_i(w^Tx_i + w_0) \geq 1 - \xi_i$, and $\xi_i \geq 0$ for all $i$

Would like to minimize

$$J(w, \xi_1, \ldots, \xi_n) = \frac{1}{2} ||w||^2 + \beta \sum_{i=1}^{n}(\xi_i > 0)$$

- where $I(\xi_i > 0) = \begin{cases} 1 & \text{if } \xi_i > 0 \\ 0 & \text{if } \xi_i \leq 0 \end{cases}$
- constrained to $z_i(w^Tx_i + w_0) \geq 1 - \xi_i$, and $\xi_i \geq 0$ for all $i$
**SVM: Non Separable Case**

- Instead we minimize
  \[ J(w; \zeta) = \frac{1}{2} \| w \|^2 + \sum_i \xi_i \]  
  \( \xi_i \geq 0 \) (a measure of misclassified examples)

  - constrained to \[ \xi_i (w^t x_i + w_0) \geq 1 - \eta_i \quad \forall i \]
  \[ \xi_i \geq 0 \]

- Can use Kuhn-Tucker theorem to convert to

\[
\begin{align*}
\text{maximize } & \quad L(a) = \sum_i a_i - \frac{1}{2} \sum_{i,j} a_i a_j z_i z_j x_i^t x_j \\
\text{subject to } & \quad 0 \leq a_i \leq \beta_i \quad \forall i \\
& \quad \sum_i a_i z_i = 0
\end{align*}
\]

- find \( w \) using \( w = \sum_i a_i z_i x_i \)
- solve for \( w_0 \) using any \( 0 < a_i < \beta_i \) and \( a_i [w^t x_i + w_0] - 1 = 0 \)

---

**Non Linear Mapping**

- To solve a non-linear classification problem with a linear classifier
  1. Project data \( x \) to high dimension using function \( \phi(x) \)
  2. Find a linear discriminant function for transformed data \( \phi(x) \)
  3. Final non-linear discriminant function is \( g(x) = w^t \phi(x) + w_0 \)

\[
\phi(x) = (x, x^2)
\]

- In 2D, discriminant function is linear

\[
g(x) = [w_1, w_2]^t x^{(n)} + w_0
\]

- In 1D, discriminant function is not linear

\[
g(x) = w_1 x + w_2 x^2 + w_3
\]

---

**Non Linear Mapping**

- Cover's theorem:
  - “pattern-classification problem cast in a high dimensional space non-linearly is more likely to be linearly separable than in a low-dimensional space”

- One dimensional space, not linearly separable

<table>
<thead>
<tr>
<th>-3</th>
<th>-2</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>5</th>
</tr>
</thead>
</table>

- Lift to two dimensional space with \( \phi(x) = (x, x^2) \)
Non Linear SVM

- Can use any linear classifier after lifting data into a higher dimensional space. However we will have to deal with the "curse of dimensionality"
  1. poor generalization to test data
  2. computationally expensive
- SVM avoids the "curse of dimensionality" problems by
  1. enforcing largest margin permits good generalization
  2. computation in the higher dimensional case is performed only implicitly through the use of kernel functions

Non Linear SVM: Kernels

- Recall SVM optimization
  \[ \text{maximize } L_0(a) = \sum \alpha_i - \frac{1}{2} \sum \alpha_i \alpha_j K(x_i, x_j) \]
  \[ \text{subject to } \sum \alpha_i y_i = 0 \]

- Note this optimization depends on samples \( x_i \) only through the dot product \( x_i \cdot x_j \)
- If we lift \( x_i \) to high dimension using \( \phi(x) \), need to compute high dimensional product \( \phi(x_i) \cdot \phi(x_j) \)
  \[ \text{maximize } L_0(a) = \sum \alpha_i - \frac{1}{2} \sum \alpha_i \alpha_j \phi(x_i) \cdot \phi(x_j) \]

- Idea: find kernel function \( K(x_i, x_j) \) s.t.
  \[ K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \]

Non Linear SVM: Kernels

- Suppose we have 2 features and \( K(x,y) = (x^T y)^2 \)
- Which mapping \( \phi(x) \) does it correspond to?
  \[ K(x,y) = (x^T y)^2 = \left( \begin{array}{c} x^T \\ y^T \end{array} \right)^2 = \left( \begin{array}{c} x^T y \\ y^T \end{array} \right)^2 = \left( \begin{array}{c} x^T y \end{array} \right)^2 \]
  \[ = (x^T y)^2 + 2(x^T y)(x^T y) + (x^T y)^2 \]
  \[ = \left( \begin{array}{c} x^T \\ y^T \end{array} \right)^2 \cdot \sqrt{2} x^T y \cdot x^T y \cdot (y^T)^2 \]

- Thus
  \[ \phi(x) = \left( \begin{array}{c} x^T \\ \sqrt{2} x^T y \\ (y^T)^2 \end{array} \right) \]
Non Linear SVM: Kernels

- How to choose kernel function $K(x_i, x_j)$?
  - $K(x_i, x_j)$ should correspond to product $\varphi(x_i)^T \varphi(x_j)$ in a higher dimensional space
  - Mercer’s condition tells us which kernel function can be expressed as dot product of two vectors
  - Kernel’s not satisfying Mercer’s condition can be sometimes used, but no geometrical interpretation
- Some common choices (satisfying Mercer’s condition):
  - Polynomial kernel $K(x_i, x_j) = (x_i^T x_j + 1)^p$
  - Gaussian radial Basis kernel (data is lifted in infinite dimension)
    $$K(x_i, x_j) = \exp \left(-\frac{1}{2\sigma^2} \|x_i - x_j\|^2\right)$$

Non Linear SVM

- Will not use notation $a = [w_0 \ w]$, we’ll use old notation $w$ and seek hyperplane through the origin $w \varphi(x) = 0$
- If the first component of $\varphi(x)$ is not 1, the above is equivalent to saying that the hyperplane has to go through the origin in high dimension
  - removes only one degree of freedom
  - But we have introduced many new degrees when we lifted the data in high dimension

Non Linear SVM

- search for separating hyperplane in high dimension $w \varphi(x) + w_0 = 0$
- Choose $\varphi(x)$ so that the first ("0"th) dimension is the augmented dimension with feature value fixed to 1 $\varphi(x) = [1 \ x^{(1)} \ x^{(2)} \ x^{(1)}x^{(2)}]$
- Threshold parameter $w_0$ gets folded into the weight vector $w$
  $$\begin{bmatrix} w_0 \\ w \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} = 0$$

Non Linear SVM Recepie

- Start with data $x_1, \ldots, x_n$ which lives in feature space of dimension $d$
- Choose kernel $K(x_i, x_j)$ or function $\varphi(x_j)$ which takes sample $x_j$ to a higher dimensional space
- Find the largest margin linear discriminant function in the higher dimensional space by using quadratic programming package to solve:
  $$\text{maximize} \quad L(a) = \sum_i a_i - \frac{1}{2} \sum_i \sum_j a_i a_j K(x_i, x_j)$$
  $$\text{constrained to} \quad 0 \leq a_i \leq \beta \quad \forall i \quad \text{and} \quad \sum_i a_i = 0$$
Non Linear SVM Recipe

- Weight vector \( w \) in the high dimensional space:
  \[
  w = \sum_{i \in S} \alpha_i \varphi(x_i)
  \]
  where \( S \) is the set of support vectors \( S = \{x_i | \alpha_i > 0\} \)
- Linear discriminant function of largest margin in the high dimensional space:
  \[
  g(x) = w^T \varphi(x) = \left( \sum_{i \in S} \alpha_i \varphi(x_i) \right)^T \varphi(x)
  \]
- Non linear discriminant function in the original space:
  \[
  g(x) = \left( \sum_{i \in S} \alpha_i \varphi(x_i) \right)^T \varphi(x) = \sum_{i \in S} \alpha_i \varphi(x_i) \varphi(x) = \sum_{i \in S} \alpha_i \varphi(x_i) K(x,x)
  \]
- decide class 1 if \( g(x) > 0 \), otherwise decide class 2

Non Linear SVM

- Nonlinear discriminant function
  \[
  g(x) = \sum_{i \in S} \alpha_i z_i K(x_i,x)
  \]
- \( g(x) \) is the sum of the weights of the support vectors from \( x \) to \( x_i \) that are most important training samples, i.e. support vectors.
- \( K(x_i,x) = \exp\left(-\frac{1}{2d^2} \| x_i - x \| \right) \)

SVM Example: XOR Problem

- Class 1: \( x_1 = [1,-1], x_2 = [-1,1] \)
- Class 2: \( x_3 = [1,1], x_4 = [-1,-1] \)
- Use polynomial kernel of degree 2:
  \[
  K(x_i,x_j) = (x_i^T x_j + 1)^2
  \]
- This kernel corresponds to mapping \( \varphi(x) = [x^{(0)}, \sqrt{2} x^{(0)} x^{(0)}, (x^{(0)})^T (x^{(0)})] \)

- Need to maximize
  \[
  L_0(\alpha) = \sum_{i \in S} \alpha_i - \frac{1}{2} \sum_{i \neq j} \alpha_i \alpha_j z_i z_j (x_i^T x_j + 1)
  \]
  constrained to \( 0 \leq \alpha_i \) \( \forall i \) and \( \alpha_i + \alpha_j - \alpha_i - \alpha_j = 0 \)

Solution to the above is \( \alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25 \)
- satisfies the constraints \( \forall i, 0 \leq \alpha_i \) and \( \alpha_i + \alpha_j - \alpha_i - \alpha_j = 0 \)
- all samples are support vectors
**SVM Example: XOR Problem**

\[ \varphi(x) = [\sqrt{2}x^{(1)}, \sqrt{2}x^{(2)}, \sqrt{2}x^{(3)}x^{(4)} (x^{(3)})^2 (x^{(4)})^2] \]

- Weight vector \( w \) is:
  \[ w = \sum_{i} \alpha_i \varphi(x_i) = 0.25(\varphi(x_1) + \varphi(x_2) + \varphi(x_3) - \varphi(x_4)) \]
  \[ = \begin{bmatrix} 0 & 0 & -2 & 0 & 0 \end{bmatrix} \]

- Thus the nonlinear discriminant function is:
  \[ g(x) = w \varphi(x) = \sum_{i} w_i \varphi_i(x) = -2\varphi(x_3)x^{(3)} - 2x^{(4)}x^{(4)} \]

---

**Degree 3 Polynomial Kernel**

- In linearly separable case (on the left), decision boundary is roughly linear, indicating that dimensionality is controlled
- Nonseparable case (on the right) is handled by a polynomial of degree 3

---

**SVM Example: XOR Problem**

**SVM Summary**

- **Advantages:**
  - Based on nice theory
  - Excellent generalization properties
  - Objective function has no local minima
  - Can be used to find non-linear discriminant functions
  - Complexity of the classifier is characterized by the number of support vectors rather than the dimensionality of the transformed space

- **Disadvantages:**
  - Tends to be slower than other methods
  - Quadratic programming is computationally expensive
  - Not clear how to choose the Kernel