Support Vector Machines

Prof. Ziping Zhao

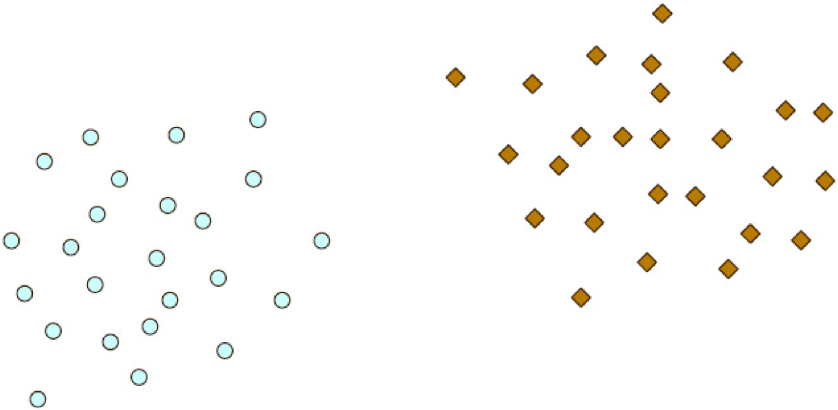
School of Information Science and Technology
ShanghaiTech University, Shanghai, China

CS182: Introduction to Machine Learning (Fall 2021)
http://cs182.sist.shanghaitech.edu.cn

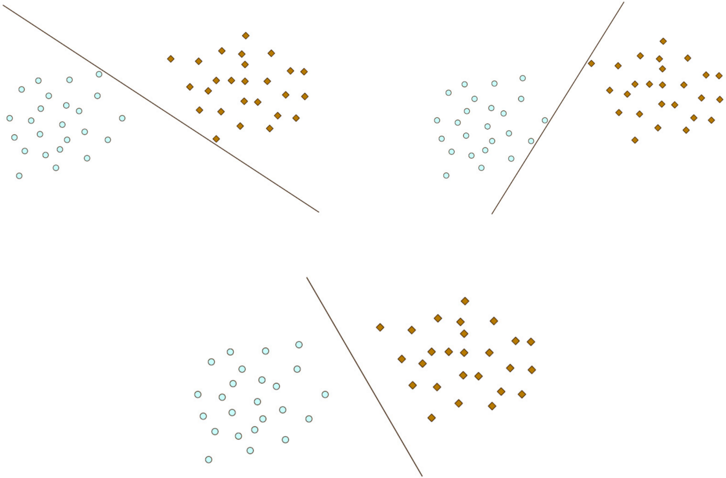
Outline

Outline

Given a Data Set ...

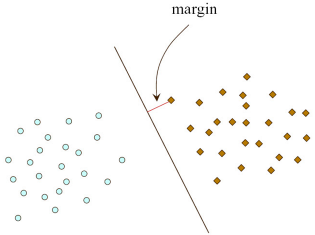


… Which Separating Hyperplane is the Best?



Optimal Separating Hyperplane

- A data point is represented as a vector in the space.
- We are using the hypothesis class of lines denoted as separating hyperplanes.
- Margin of a separating hyperplane: distance to the separating hyperplane from the data point closest to it on either side.

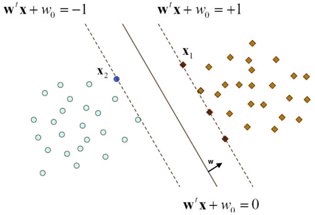


- Relationship between margin and generalization:
  There exist theoretical results from statistical learning theory showing that the separating hyperplane with the largest margin generalizes best (i.e., has smallest generalization error).

Outline

Canonical Optimal Separating Hyperplane

- Hard-margin case: data points from the two classes are assumed to be linearly separable.
- Note that $(c\mathbf{w})^T\mathbf{x} + cw_0 = 0$ with $c \neq 0$ defines the same hyperplane as $\mathbf{w}^T\mathbf{x} + w_0 = 0$.
- With proper scaling of $\mathbf{w}$ and w_0, the points closest to the hyperplane satisfy $|\mathbf{w}^T\mathbf{x} + w_0| = 1$. Such a hyperplane is called a canonical separating hyperplane.
- The one that maximizes the margin is called the canonical optimal separating hyperplane.



Canonical Optimal Separating Hyperplane (2)

- Let $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ be two closest points, one on each side of the hyperplane.
- Note that

$$\mathbf{w}^T\mathbf{x}^{(1)} + w_0 = +1$$
$$\mathbf{w}^T\mathbf{x}^{(2)} + w_0 = -1$$

  Hence the margin can be given by

$$\gamma = \frac{|\mathbf{w}^T\mathbf{x}^{(1)} + w_0|}{\|\mathbf{w}\|} = \frac{|\mathbf{w}^T\mathbf{x}^{(2)} + w_0|}{\|\mathbf{w}\|} = \frac{1}{\|\mathbf{w}\|}$$

- Maximizing the margin is equivalent to minimizing $\|\mathbf{w}\|$.

Inequality Constraints

- Let us start again with two classes and use labels $-1/+1$ for the two classes.
- The sample is $\mathcal{X} = \{(\mathbf{x}^{(\ell)}, r^{(\ell)})\}$ where $r^{(\ell)} = +1$ if $\mathbf{x}^{(\ell)} \in C_1$ and $r^{(\ell)} = -1$ if $\mathbf{x}^{(\ell)} \in C_2$.
- For all data points in the sample $\mathcal{X}$, we want $\mathbf{w}$ and w_0 to satisfy

$$\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0 \begin{cases} \geq +1 & \text{if } r^{(\ell)} = +1 \\ \leq -1 & \text{if } r^{(\ell)} = -1 \end{cases}$$

  which are equivalent to the following inequality constraints:

$$r^{(\ell)}(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) \geq +1 \tag{1}$$

- Instead of simply using inequality constraints

$$r^{(\ell)}(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) \geq 0$$

  which only require the data points to lie on the right side of the hyperplane, the constraints in (1) also want them some distance away for better generalization.

Optimization Problem

- Optimization problem (the primal problem):

$$\begin{aligned} \underset{\mathbf{w}\in\mathbf{R}^d,\, w_0}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{subject to} \quad & r^{(\ell)}(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) \geq 1, \quad \forall \ell \end{aligned}$$

- This is a quadratic programming (QP) problem (or a quadratic program), a type of convex optimization problem, the complexity of which depends on d.
- This QP can be solved directly via QP numerical solving methods to find $\mathbf{w}$ and w_0, i.e., the canonical optimal separating hyperplane.
- On both sides of the hyperplane, there will be instances that are $\frac{1}{\|\mathbf{w}\|}$ away from the hyperplane and the total margin will be $\frac{2}{\|\mathbf{w}\|}$.

Optimization Problem (2)

- As discussed in previous lectures, if the classification problem is not linearly separable, instead of fitting a nonlinear function, one trick is to map the problem to a new space by using nonlinear basis functions.
- It is generally the case that this new space has more dimensions than the original space (i.e., a larger d), and, in such a case, we are interested in a method whose complexity does not depend on the input dimensionality.
- In optimization theory, it is very common and sometimes advantageous to turn a primal problem into a dual problem and then solve the latter instead.
- In our case, it also turns out to be more convenient to solve the dual problem (whose complexity depends on the sample size N) rather than the primal problem directly (whose complexity depends on the dimensionality d). The dual problem also makes it easy for a nonlinear extension using kernel functions.

Lagrangian

- Lagrangian:

$$\begin{aligned}\mathcal{L}(\mathbf{w}, w_0, \{\alpha_\ell\}) =& \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{\ell=1}^{N} \alpha_\ell \left[r^{(\ell)}(\mathbf{w}^T \mathbf{x}^{(\ell)} + w_0) - 1 \right] \\ =& \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{\ell=1}^{N} \alpha_\ell r^{(\ell)}(\mathbf{w}^T \mathbf{x}^{(\ell)} + w_0) + \sum_{\ell=1}^{N} \alpha_\ell \\ =& \frac{1}{2}\mathbf{w}^T \mathbf{w} - \mathbf{w}^T \sum_\ell \alpha_\ell r^{(\ell)} \mathbf{x}^{(\ell)} - w_0 \sum_\ell \alpha_\ell r^{(\ell)} + \sum_\ell \alpha_\ell\end{aligned}$$

  with Lagrange multipliers $\alpha_\ell \geq 0$.
- The inequality constraints of the primal problem are incorporated into the second term of the Lagrangian. So it is no longer necessary to enforce them explicitly.
- The optimal solution is a saddle point which minimizes L_p w.r.t. the primal variables $\mathbf{w}, w_0$ and maximizes L_p w.r.t. the dual variables α_ℓ.

Eliminating Primal Variables

- Setting the gradients of $\mathcal{L}$ w.r.t. $\mathbf{w}$ and w_0 to 0:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \mathbf{0} \quad \Rightarrow \quad \mathbf{w} = \sum_\ell \alpha_\ell r^{(\ell)} \mathbf{x}^{(\ell)} \tag{2}$$

$$\frac{\partial \mathcal{L}}{\partial w_0} = 0 \quad \Rightarrow \quad \sum_\ell \alpha_\ell r^{(\ell)} = 0 \tag{3}$$

- Plugging (2) and (3) into $\mathcal{L}$ gives the objective function G for the dual problem:

$$\begin{aligned} G(\{\alpha_\ell\}) &= -\frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_\ell \alpha_\ell \\ &= -\frac{1}{2}\sum_\ell \sum_{\ell'} \alpha_\ell \alpha_{\ell'} r^{(\ell)} r^{(\ell')} (\mathbf{x}^{(\ell)})^T \mathbf{x}^{(\ell')} + \sum_\ell \alpha_\ell \end{aligned}$$

Dual Optimization Problem

- Dual optimization problem:

$$\begin{aligned} \underset{\{\alpha_\ell\}}{\text{maximize}} \quad & \sum_\ell \alpha_\ell - \frac{1}{2} \sum_\ell \sum_{\ell'} \alpha_\ell \alpha_{\ell'} r^{(\ell)} r^{(\ell')} (\mathbf{x}^{(\ell)})^T \mathbf{x}^{(\ell')} \\ \text{subject to} \quad & \sum_\ell \alpha_\ell r^{(\ell)} = 0 \\ & \alpha_\ell \geq 0, \quad \forall \ell \end{aligned}$$

- This is also a QP problem, and its complexity depends on the sample size N (rather than the input dimensionality d):
  - Time complexity: $O(N^3)$ (for generic QP solvers)
  - Space complexity: $O(N^2)$
- Define $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_N]^T$, $\mathbf{r} = [r^{(1)}, \ldots, r^{(N)}]^T$, and the symmetric matrix $\mathbf{H} \in \mathbf{R}^{N \times N}$ with $h_{ij} = r^{(i)} r^{(j)} (\mathbf{x}^{(i)})^T \mathbf{x}^{(j)}$, we get the equivalent reformulation

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{maximize}} \quad & \boldsymbol{\alpha}^T \mathbf{1} - \frac{1}{2} \boldsymbol{\alpha}^T \mathbf{H} \boldsymbol{\alpha} \\ \text{subject to} \quad & \boldsymbol{\alpha}^T \mathbf{r} = 0, \ \ \boldsymbol{\alpha} \geq \mathbf{0} \end{aligned}$$

Support Vectors

- Most of the dual variables vanish with $\alpha_\ell = 0$. They are points lying beyond the margin (sufficiently away from the hyperplane), i.e., $r^{(\ell)}(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) > 1$, with no effect on the hyperplane. (use the KKT complementarity slackness condition)
  - Even if any subset of them are removed or moved around, we would still get the same solution.
  - It is possible to use a simpler classifier to filter out a large portion of such instances, i.e., decreasing N, thereby decreasing the complexity of the optimization.
- Support vectors (SVs): $\mathbf{x}^{(\ell)}$ with $\alpha_\ell > 0$, i.e., $r^{(\ell)}(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) = 1$, hence the name support vector machine (SVM).
  - Solution is determined by the data on the margin.

Support Vectors (2)

- Computation of primal variables, i.e., $\mathbf{w}$ and w_0:
  - From (2) we get

$$\mathbf{w} = \sum_{\ell=1}^{N} \alpha_\ell r^{(\ell)} \mathbf{x}^{(\ell)} = \sum_{\mathbf{x}^{(\ell)} \in \mathcal{SV}} \alpha_\ell r^{(\ell)} \mathbf{x}^{(\ell)}$$

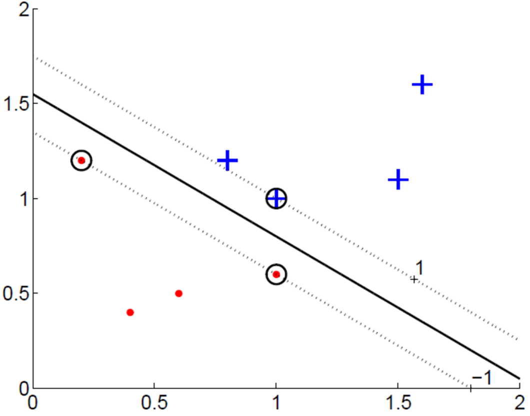
    where $\mathcal{SV}$ denotes the set of support vectors.
  - The support vectors must lie on the margin, so they should satisfy

$$r^{(\ell)}(\mathbf{w}^T \mathbf{x}^{(\ell)} + w_0) = 1 \qquad \text{or} \qquad w_0 = r^{(\ell)} - \mathbf{w}^T \mathbf{x}^{(\ell)}$$

    For numerical stability, in practice all support vectors are used to compute w_0:

$$w_0 = \frac{1}{|\mathcal{SV}|} \sum_{\mathbf{x}^{(\ell)} \in \mathcal{SV}} \left(r^{(\ell)} - \mathbf{w}^T \mathbf{x}^{(\ell)} \right)$$

Hard-Margin Support Vector Machine



Discriminant function

- Discriminant function:

$$
\begin{aligned}
g(\mathbf{x}) &= \mathbf{w}^T \mathbf{x} + w_0 \\
&= \left[\sum_{\mathbf{x}^{(\ell)} \in \mathcal{SV}} \alpha_\ell r^{(\ell)} \mathbf{x}^{(\ell)} \right]^T \mathbf{x} + \frac{1}{|\mathcal{SV}|} \sum_{\mathbf{x}^{(\ell)} \in \mathcal{SV}} \left(r^{(\ell)} - \mathbf{w}^T \mathbf{x}^{(\ell)} \right)
\end{aligned}
$$

- During testing, we do not enforce a margin and obtain the classification rule :

$$
\text{Choose } \begin{cases} C_1 & \text{if } g(\mathbf{x}) > 0 \\ C_2 & \text{otherwise} \end{cases}
$$

Generalization to $K > 2$ Classes

- One way to handle multiple classes is to define K two-class problems, each separating one class from all other classes combined, i.e., the one-vs.-all approach.
- An SVM $g_i(\mathbf{x})$ is learned for each two-class problem.
- Classification rule during testing:

$$\text{Choose } C_j \text{ if } j = \arg\max_k g_k(\mathbf{x})$$

- We can also define pairwise separation of classes by training $K(K-1)/2$ SVMs, i.e., the one-vs.-one approach.

Outline

Relaxing the Constraints

- In practice, a separating hyperplane may not exist, possibly due to the fact that the data is not linearly separable or a high noise level which causes a large overlap of the classes.
- Even if a separating hyperplane exists, it is not always the best solution to the classification problem when there exist outliers in the data.
  - A mislabeled example can become an outlier which affects the location of the separating hyperplane.

Slack Variables

- A soft-margin SVM allows for the possibility of violating the inequality constraints

$$r^{(\ell)}(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) \geq 1$$

  by introducing slack variables

$$\xi_\ell \geq 0, \quad \ell = 1, \ldots, N$$

  which store the deviation from the margin.
- Relaxed separation constraints:

$$r^{(\ell)}(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) \geq 1 - \xi_\ell$$

Penalty

- By making ξ_ℓ large enough, the constraint on $(\mathbf{x}^{(\ell)}, r^{(\ell)})$ can always be met.
- In order not to obtain the trivial solution where all ξ_ℓ take on large values, we should penalize them in the objective function.
- Three cases for ξ_ℓ:
  - $\xi_\ell = 0$: no problem with $\mathbf{x}^{(\ell)}$ (no penalty)
  - $0 < \xi_\ell < 1$: $\mathbf{x}^{(\ell)}$ lies on the right side of the hyperplane but in the margin (small penalty)
  - $\xi_\ell > 1$: $\mathbf{x}^{(\ell)}$ lies on the wrong side of the hyperplane (large penalty)
- Number of misclassifications: $\#\{\xi_\ell > 1\}$
- Number of nonseparable instances: $\#\{\xi_\ell > 0\}$
- Soft error as additional penalty term:

$$\sum_{\ell=1}^{N} \xi_\ell$$

Primal Optimization Problem

- Primal optimization problem:

$$\begin{aligned} \underset{\mathbf{w},\, w_0,\, \{\xi_\ell\}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{\ell=1}^{N}\xi_\ell \\ \text{subject to} \quad & r^{(\ell)}(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) \geq 1 - \xi_\ell, \quad \forall\ell \\ & \xi_\ell \geq 0, \quad \forall\ell \end{aligned}$$

  where $C \geq 0$ is a regularization parameter (which trades off model complexity in terms of the number of support vectors and data misfit in terms of the number of nonseparable points).
- Both the misclassified instances and the ones in the margin are penalized for better generalization, though the latter ones would be correctly classified during testing.
- For the same reason as before, we will resort to the dual problem.

Lagrangian

- Lagrangian:

$$
\begin{aligned}
&\mathcal{L}(\mathbf{w}, w_0, \{\xi_\ell\}, \{\alpha_\ell\}, \{\mu_\ell\}) \\
&\qquad = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{\ell=1}^{N}\xi_\ell - \sum_{\ell=1}^{N}\alpha_\ell\Big[r^{(\ell)}(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) - 1 + \xi_\ell\Big] - \sum_{\ell=1}^{N}\mu_\ell\xi_\ell
\end{aligned}
$$

where the new Lagrange multipliers $\mu_\ell \geq 0$.

Eliminating Primal Variables

- Setting the gradients of $\mathcal{L}$ w.r.t. $\mathbf{w}$, w_0, and $\{\xi_\ell\}$ to 0:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \mathbf{0} \quad \Rightarrow \quad \mathbf{w} = \sum_\ell \alpha_\ell r^{(\ell)} \mathbf{x}^{(\ell)} \tag{4}$$

$$\frac{\partial \mathcal{L}}{\partial w_0} = 0 \quad \Rightarrow \quad \sum_\ell \alpha_\ell r^{(\ell)} = 0 \tag{5}$$

$$\frac{\partial \mathcal{L}}{\partial \xi_\ell} = 0 \quad \Rightarrow \quad \mu_\ell = C - \alpha_\ell, \quad \forall \ell \tag{6}$$

- Plugging (4), (5), and (6) into $\mathcal{L}$ gives the objective function G to maximize for the dual problem:

$$G(\{\alpha_\ell\}) = -\frac{1}{2} \sum_\ell \sum_{\ell'} \alpha_\ell \alpha_{\ell'} r^{(\ell)} r^{(\ell')} (\mathbf{x}^{(\ell)})^T \mathbf{x}^{(\ell')} + \sum_\ell \alpha_\ell$$

- Since $\mu_\ell \geq 0$, $\forall \ell$, (6) implies that $0 \leq \alpha_\ell \leq C$, $\forall \ell$.

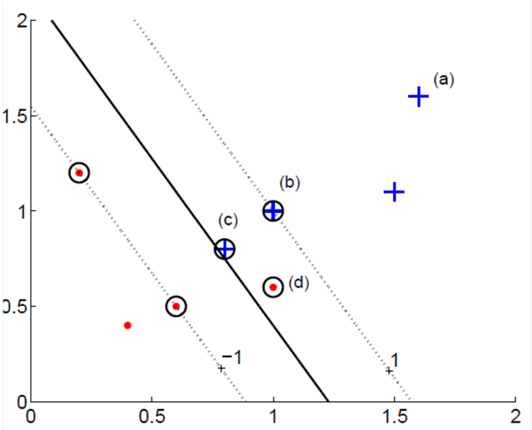
Dual Optimization Problem

- Dual optimization problem:

$$\begin{aligned} \underset{\{\alpha_\ell\}}{\text{maximize}} \quad & \sum_\ell \alpha_\ell - \frac{1}{2} \sum_\ell \sum_{\ell'} \alpha_\ell \alpha_{\ell'} r^{(\ell)} r^{(\ell')} (\mathbf{x}^{(\ell)})^T \mathbf{x}^{(\ell')} \\ \text{subject to} \quad & \sum_\ell \alpha_\ell r^{(\ell)} = 0 \\ & 0 \leq \alpha_\ell \leq C, \quad \forall \ell \end{aligned}$$

- Similar to the hard-margin case (i.e., the separable case), instances that are not support vectors (lie on the correct side of the boundary with sufficient margin) vanish with $\alpha_\ell = 0$.
- The primal variables $\mathbf{w}$ and w_0 can be computed similarly based on the SVs.
  - The SVs have their $\alpha_\ell > 0$ and they define $\mathbf{w}$.
  - Of SVs, those whose $\alpha_\ell < C$ are the ones that are on the margin which can be used to calculate w_0 (they have $\xi_\ell = 0$ and satisfy $r^{(\ell)}(\mathbf{w}^T \mathbf{x}^{(\ell)} + w_0) = 1$).
  - Those instances that are in the margin or misclassified have their $\alpha_\ell = C$.

Soft-Margin Support Vector Machine



Support Vectors

- The nonseparable instances that we store as support vectors are the instances that we would have trouble correctly classifying if they were not in the training set; they would either be misclassified or classified correctly but not with enough confidence.
- An important result from Vapnik's statistical learning theory is that the expected test error rate has an upper bound which depends on the number of support vectors:

$$E_N[P(\text{error})] \leq \frac{E_N[\#\text{ of SVs}]}{N}$$

  where $E_N[\cdot]$ denotes the expectation over training sets of size N.
- It shows that the error rate depends on the number of support vectors and not on the input dimensionality.

Hinge Loss

- In the soft-margin SVM, we define an error ξ_ℓ if the instance $(\mathbf{x}^{(\ell)}, r^{(\ell)})$ is nonseparable, which can be described as a hinge loss as

$$L_{\text{hinge}}(y^{(\ell)}, r^{(\ell)}) = (1 - r^{(\ell)} y^{(\ell)})_+ = \begin{cases} 0 & \text{if } r^{(\ell)} y^{(\ell)} \geq 1 \\ 1 - y^{(\ell)} r^{(\ell)} & \text{otherwise} \end{cases}$$

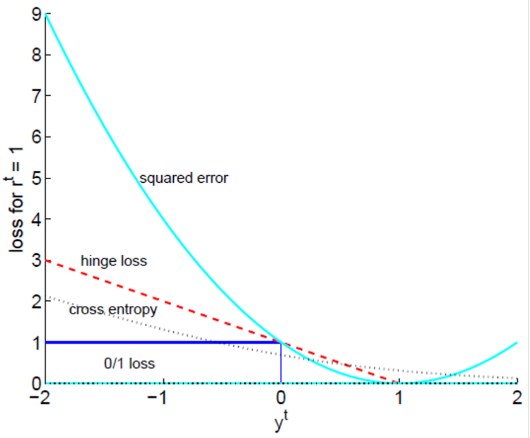
where $y^{(\ell)} = \mathbf{w}^T \mathbf{x}^{(\ell)} + w_0$.

- The soft-margin SVM problem can be equivalently formulated as

$$\begin{aligned} \underset{\mathbf{w},\, w_0}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{\ell=1}^{N} (1 - r^{(\ell)} y^{(\ell)})_+ \\ \text{subject to} \quad & y^{(\ell)} = \mathbf{w}^T \mathbf{x}^{(\ell)} + w_0, \quad \forall \ell \end{aligned}$$

- The hinge loss, again, reveals the nature of solution sparsity in SVM, i.e., predictions only depend on a subset of the training data.

More Loss Functions



Remark on SVMs

- The SVM problem can be case as convex programming problem (every local solution to a convex programming problem is a globally optimal solution), which is contrast to neural networks, where many local minima usually exist.
- In both training and testing, training data only appear in the form of dot products between vectors, which will become important later on.

Outline

Key Ideas of Kernel Methods

- Instead of defining a nonlinear model in the original (input) space, the problem is mapped to a new (feature) space by performing a nonlinear transformation using suitably chosen basis functions.
- A linear model is then applied in the new space.
- This approach can be used in both classification and regression problems.
- In the particular case of support vector machines, it leads to certain simplifications, where the basis functions are often defined implicitly via defining kernel functions directly.

Basis Functions

- Basis Functions:

$$\mathbf{z} = \phi(\mathbf{x}) \quad \text{where } z_j = \phi_j(\mathbf{x}),\ j = 1, \ldots, k$$

  mapping from the d-dimensional $\mathbf{x}$-space to the k-dimensional $\mathbf{z}$-space.

- Discriminant function:

$$\begin{aligned} g(\mathbf{z}) =& \mathbf{w}^T \mathbf{z} + w_0 \\ g(\mathbf{x}) =& \mathbf{w}^T \phi(\mathbf{x}) + w_0 = \sum_{j=1}^{k} w_j \phi_j(\mathbf{x}) + w_0 \end{aligned}$$

- Usually, $k \gg d$, N (in fact k can even be infinite). The dual form is preferred because its complexity depends on N but that of the primal form depends on k.

Primal Optimization Problem

- We use the general case of soft-margin nonlinear SVM because we have no guarantee that the problem is linearly separable in this new space.
- Primal optimization problem:

$$
\begin{aligned}
\underset{\mathbf{w},\, w_0,\, \{\xi_\ell\}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{\ell=1}^{N}\xi_\ell \\
\text{subject to} \quad & r^{(\ell)}(\mathbf{w}^T\phi(\mathbf{x}^{(\ell)}) + w_0) \geq 1 - \xi_\ell, \quad \forall \ell \\
& \xi_\ell \geq 0, \quad \forall \ell
\end{aligned}
$$

  where $C \geq 0$.
- We will resort to the dual problem.

Lagrangian

- Lagrangian:

$$\mathcal{L}(\mathbf{w}, \{\xi_\ell\}, \{\alpha_\ell\}, \{\mu_\ell\}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{\ell=1}^{N}\xi_\ell - \sum_{\ell=1}^{N}\alpha_\ell\left[r^{(\ell)}(\mathbf{w}^T\phi(\mathbf{x}^{(\ell)}) + w_0) - 1 + \xi_\ell\right] - \sum_{\ell=1}^{N}\mu_\ell\xi_\ell$$

where the Lagrange multipliers $\alpha_\ell,\ \mu_\ell \geq 0$.

Dual Optimization Problem

- Setting the gradients of $\mathcal{L}$ w.r.t. $\mathbf{w}$, w_0, and $\{\xi_\ell\}$ to 0:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \mathbf{0} \quad \Rightarrow \quad \mathbf{w} = \sum_\ell \alpha_\ell r^{(\ell)} \phi(\mathbf{x}^{(\ell)}) \tag{7}$$

$$\frac{\partial \mathcal{L}}{\partial w_0} = 0 \quad \Rightarrow \quad \sum_\ell \alpha_\ell r^{(\ell)} = 0 \tag{8}$$

$$\frac{\partial \mathcal{L}}{\partial \xi_\ell} = 0 \quad \Rightarrow \quad \mu_\ell = C - \alpha_\ell, \quad \forall \ell \tag{9}$$

- Plugging (7) and (8) into $\mathcal{L}$ gives the objective function G for the dual problem:

$$G(\{\alpha_\ell\}) = -\frac{1}{2} \sum_\ell \sum_{\ell'} \alpha_\ell \alpha_{\ell'} r^{(\ell)} r^{(\ell')} \phi(\mathbf{x}^{(\ell)})^T \phi(\mathbf{x}^{(\ell')}) + \sum_\ell \alpha_\ell$$

Dual Optimization Problem (2)

- Dual optimization problem:

$$\begin{aligned}
\underset{\{\alpha_\ell\}}{\text{maximize}} \quad & \sum_\ell \alpha_\ell - \frac{1}{2}\sum_\ell \sum_{\ell'} \alpha_\ell \alpha_{\ell'} r^{(\ell)} r^{(\ell')} \phi(\mathbf{x}^{(\ell)})^T \phi(\mathbf{x}^{(\ell')}) \\
\text{subject to} \quad & \sum_\ell \alpha_\ell r^{(\ell)} = 0 \\
& 0 \leq \alpha_\ell \leq C, \ \forall \ell
\end{aligned}$$

Kernel Functions

- In kernel SVM, we have $K(\mathbf{x}^{(\ell)}, \mathbf{x}^{(\ell')}) \equiv \phi(\mathbf{x}^{(\ell)})^T \phi(\mathbf{x}^{(\ell')})$ which is a kernel function (a.k.a. positive definite kernel, Mercer kernel, or reproducing kernel).

$$
\begin{aligned}
\underset{\{\alpha_\ell\}}{\text{maximize}} \quad & \sum_\ell \alpha_\ell - \frac{1}{2} \sum_\ell \sum_{\ell'} \alpha_\ell \alpha_{\ell'} r^{(\ell)} r^{(\ell')} K(\mathbf{x}^{(\ell)}, \mathbf{x}^{(\ell')}) \\
\text{subject to} \quad & \sum_\ell \alpha_\ell r^{(\ell)} = 0 \\
& 0 \leq \alpha_\ell \leq C, \ \forall \ell
\end{aligned}
$$

- Instead of mapping two instances $\mathbf{x}^{(\ell)}$ and $\mathbf{x}^{(\ell')}$ to the $\mathbf{z}$-space and doing a dot product there, we directly apply the kernel function in the original $\mathbf{x}$-space.
- Kernel matrix (a.k.a. Gram matrix):

$$
\mathbf{K} = \left[K(\mathbf{x}^{(\ell)}, \mathbf{x}^{(\ell')}) \right]_{\ell, \ell' = 1}^{N}
$$

which, like a covariance matrix, is symmetric and positive semidefinite.

Kernel Functions (2)

- Solution:

$$\mathbf{w} = \sum_{\ell} \alpha_\ell r^{(\ell)} \mathbf{z}^{(\ell)} = \sum_{\mathbf{x}^{(\ell)} \in \mathcal{SV}} \alpha_\ell r^{(\ell)} \phi(\mathbf{x}^{(\ell)})$$

- Discriminant function:

$$g(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + w_0 = \sum_{\mathbf{x}^{(\ell)} \in \mathcal{SV}} \alpha_\ell r^{(\ell)} \phi(\mathbf{x}^{(\ell)})^T \phi(\mathbf{x}) + w_0 = \sum_{\mathbf{x}^{(\ell)} \in \mathcal{SV}} \alpha_\ell r^{(\ell)} K(\mathbf{x}^{(\ell)}, \mathbf{x}) + w_0$$

where the kernel function also shows up in the discriminant.

Some Common Kernel Functions

- Polynomial kernel:

$$K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T \mathbf{x}' + 1)^q$$

where q is the degree.
E.g., when $q = 2$ and $d = 2$,

$$\begin{aligned} K(\mathbf{x}, \mathbf{x}') =& (\mathbf{x}^T \mathbf{x}' + 1)^2 \\ =& (x_1 x_1' + x_2 x_2' + 1)^2 \\ =& 1 + 2x_1 x_1' + 2x_2 x_2' + 2x_1 x_1' x_2 x_2' + (x_1)^2 (x_1')^2 + (x_2)^2 (x_2')^2 \end{aligned}$$

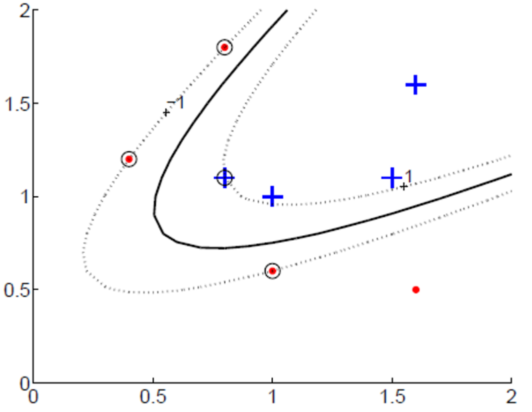
which corresponds to the inner product of the basis function

$$\phi(\mathbf{x}) = \left(1, \sqrt{2}x_1, \sqrt{2}x_2, \sqrt{2}x_1 x_2, (x_1)^2, (x_2)^2\right)^T$$

When $q = 1$, we have the linear kernel corresponding to the original formulation.

Some Common Kernel Functions (2)

- Polynomial kernel of degree 2:



Some Common Kernel Functions (3)

- Radial basis function (RBF) kernel (or Gaussian radial kernel):

$$K(\mathbf{x}, \mathbf{x}') = \exp\left[-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2s^2}\right]$$

  which is a spherical kernel where $\mathbf{x}'$ is the center and s, supplied by the user, defines the radius.
- The feature space of the RBF kernel has an infinite number of dimensions.
- It can be generalized to

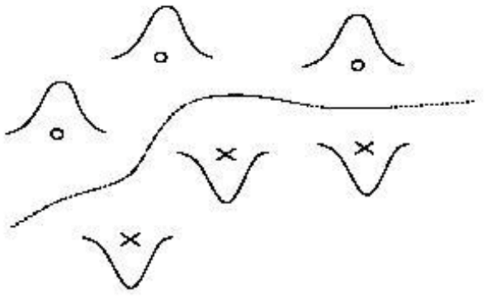
$$K(\mathbf{x}, \mathbf{x}') = \exp\left[-\frac{\mathcal{D}(\mathbf{x}, \mathbf{x}')}{2s^2}\right]$$

  where $\mathcal{D}(\cdot, \cdot)$ is some distance function.
- When taking the Mahalanobis distance, we have the Mahalanobis kernel

$$K(\mathbf{x}, \mathbf{x}') = \exp\left[-\frac{(\mathbf{x} - \mathbf{x}')^T \mathbf{S}^{-1} (\mathbf{x} - \mathbf{x}')}{2s^2}\right]$$

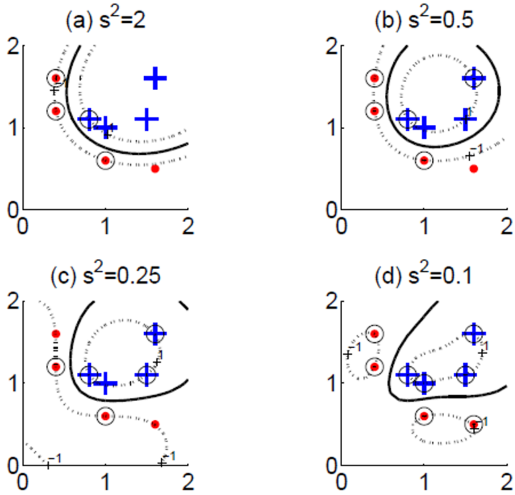
Some Common Kernel Functions (4)

- Discriminant function with RBF kernel: amounts to putting bumps of various sizes on the training set



Some Common Kernel Functions (5)

- Gaussian kernel with different spread values, s^2:



Some Common Kernel Functions (5)

- Sigmoidal kernel (or hyperbolic tangent kernel):

$$K(\mathbf{x}, \mathbf{x}') = \tanh(\kappa \mathbf{x}^T \mathbf{x}' + \theta)$$

  which, strictly speaking, is not positive semidefinite for certain parameter values κ and θ.
- This is similar to multilayer perceptrons that we discussed in last lecture.

Outline

ℓ_2 Loss Function

- We start with a linear model for regression as

$$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$$

and we have used the squared loss in ordinary linear regression

$$E_2^{(\ell)}(r^{(\ell)}, f(\mathbf{x}^{(\ell)})) = |r^{(\ell)} - f(\mathbf{x}^{(\ell)})|^2$$

- Total loss:

$$E_2 = \sum_{\ell} E_2^{(\ell)}(r^{(\ell)}, f(\mathbf{x}^{(\ell)})) = \sum_{\ell} |r^{(\ell)} - f(\mathbf{x}^{(\ell)})|^2$$

- Squared regression (or least squares regression):

$$\underset{\mathbf{w},\, w_0}{\text{minimize}} \quad \frac{1}{N}\sum_{\ell=1}^{N} |r^{(\ell)} - f(\mathbf{x}^{(\ell)})|^2$$

ϵ-Insensitive Loss Function

- In order for the sparseness property of support vectors in SVM for classification to carry over to regression, we do not use the squared loss but the ϵ-insensitive loss function:

$$E_\epsilon^{(\ell)}(r^{(\ell)}, f(\mathbf{x}^{(\ell)})) = (|r^{(\ell)} - f(\mathbf{x}^{(\ell)})| - \epsilon)_+ = \begin{cases} 0 & \text{if } |r^{(\ell)} - f(\mathbf{x}^{(\ell)})| \leq \epsilon \\ |r^{(\ell)} - f(\mathbf{x}^{(\ell)})| - \epsilon & \text{otherwise} \end{cases}$$

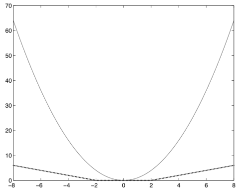
- Two characteristics:
  - Errors are tolerated up to a threshold of ϵ, i.e., no loss for point lying inside an ϵ-tube around the prediction.
  - Errors beyond ϵ have a linear (rather than quadratic) effect so that the model is more more tolerant to noise and robust against noise.
- Total loss:

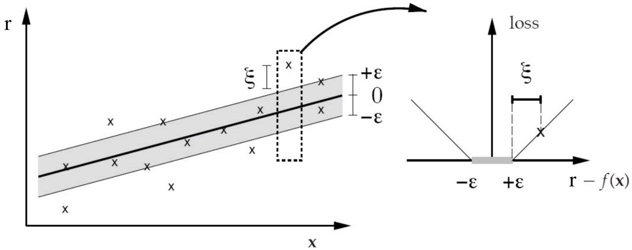
$$E_\epsilon = \sum_\ell E_\epsilon^{(\ell)}(r^{(\ell)}, f(\mathbf{x}^{(\ell)})) = \sum_\ell (|r^{(\ell)} - f(\mathbf{x}^{(\ell)})| - \epsilon)_+$$

- Tube regression:

$$\underset{\mathbf{w},\, w_0}{\text{minimize}} \quad \frac{1}{N} \sum_{\ell=1}^{N} (|r^{(\ell)} - f(\mathbf{x}^{(\ell)})| - \epsilon)_+$$

ϵ-Insensitive Loss Function (2)





Support Vector Regression

- Support vector (machine) regression (SVR) is given as

$$\underset{\mathbf{w},\, w_0}{\text{minimize}} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{\ell}(|r^{(\ell)} - f(\mathbf{x}^{(\ell)})| - \epsilon)_+$$

  where C trades off the model complexity (i.e., the flatness of the model) and data misfit.
- The value of ϵ determines the width of the tube (a smaller value indicates a lower tolerance for error) and also affects the number of support vectors and, consequently, the solution sparsity.
  - If ϵ is decreased, the boundary of the tube is shifted inward. Therefore, more datapoints are around the boundary indicating more support vectors.
  - Similarly, increasing ϵ will result in fewer points around the boundary.
- A convex problem, but not a standard QP.
- We will rewrite it to a form similar to SVM which can be QP-solvable.

Primal Optimization Problem

- We introduce slack variables ξ_ℓ^+ and ξ_ℓ^- to account for deviations out of the ϵ-zone.
- Primal optimization problem:

$$
\begin{aligned}
\underset{\mathbf{w},\, w_0,\, \{\xi_\ell^+\},\, \{\xi_\ell^-\}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_\ell (\xi_\ell^+ + \xi_\ell^-) \\
\text{subject to} \quad & r^{(\ell)} - (\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) \le \epsilon + \xi_\ell^+, \quad \forall \ell \\
& (\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) - r^{(\ell)} \le \epsilon + \xi_\ell^-, \quad \forall \ell \\
& \xi_\ell^+, \xi_\ell^- \ge 0, \quad \forall \ell
\end{aligned}
$$

  which is a standard QP.
- Two types of slack variables:
  - ξ_ℓ^+: for positive deviation such that $r^{(\ell)} - (\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) > \epsilon$.
  - ξ_ℓ^-: for negative deviation such that $(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) - r^{(\ell)} > \epsilon$.
- If $r^{(\ell)} - (\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) \le \epsilon$ and $(\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0) - r^{(\ell)} \le \epsilon$, then $\xi_\ell^+ = \xi_\ell^- = 0$, contributing no cost to the objective function.

Lagrangian

- Similar to SVM for classification, the optimization problem for SVR can also be rewritten in the dual form.
- Lagrangian:

$$
\begin{aligned}
&\mathcal{L}(\mathbf{w}, w_0, \{\xi_\ell^+\}, \{\xi_\ell^-\}, \{\alpha_\ell^+\}, \{\alpha_\ell^-\}, \{\mu_\ell^+\}, \{\mu_\ell^-\}) \\
=&\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_\ell(\xi_\ell^+ + \xi_\ell^-) \\
&- \sum_\ell \alpha_\ell^+ \left[\epsilon + \xi_\ell^+ - r^{(\ell)} + (\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0)\right] - \sum_\ell \alpha_\ell^- \left[\epsilon + \xi_\ell^- + r^{(\ell)} - (\mathbf{w}^T\mathbf{x}^{(\ell)} + w_0)\right] \\
&- \sum_\ell (\mu_\ell^+\xi_\ell^+ + \mu_\ell^-\xi_\ell^-)
\end{aligned}
$$

where α_ℓ^+, α_ℓ^-, μ_ℓ^+, $\mu_\ell^- > 0$.

Eliminating Primal Variables

- Setting the gradients of $\mathcal{L}$ w.r.t. $\mathbf{w}$, w_0, $\{\xi_\ell^+\}$, and $\{\xi_\ell^-\}$ to 0:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \mathbf{0} \quad \Rightarrow \quad \mathbf{w} = \sum_\ell (\alpha_\ell^+ - \alpha_\ell^-)\mathbf{x}^{(\ell)} \tag{10}$$

$$\frac{\partial \mathcal{L}}{\partial w_0} = 0 \quad \Rightarrow \quad \sum_\ell (\alpha_\ell^+ - \alpha_\ell^-) = 0 \tag{11}$$

$$\frac{\partial \mathcal{L}}{\partial \xi_\ell^+} = 0 \quad \Rightarrow \quad \mu_\ell^+ = C - \alpha_\ell^+, \quad \forall \ell \tag{12}$$

$$\frac{\partial \mathcal{L}}{\partial \xi_\ell^-} = 0 \quad \Rightarrow \quad \mu_\ell^- = C - \alpha_\ell^-, \quad \forall \ell \tag{13}$$

- Plugging (9), (10), (11), and (12) into $\mathcal{L}$ gives the objective function G for the dual problem:

$$G(\{\alpha_\ell^+\}, \{\alpha_\ell^-\}) = -\frac{1}{2}\sum_\ell \sum_{\ell'} (\alpha_\ell^+ - \alpha_\ell^-)(\alpha_{\ell'}^+ - \alpha_{\ell'}^-)(\mathbf{x}^{(\ell)})^T \mathbf{x}^{(\ell')} - \epsilon \sum_\ell (\alpha_\ell^+ + \alpha_\ell^-) + \sum_\ell r^{(\ell)} (\alpha_\ell^+ - \alpha_\ell^-)$$

Dual Optimization Problem

- Dual optimization problem:

$$\begin{aligned}
\underset{\{\alpha_\ell^+\},\{\alpha_\ell^-\}}{\text{maximize}} \quad & -\frac{1}{2}\sum_\ell \sum_{\ell'} (\alpha_\ell^+ - \alpha_\ell^-)(\alpha_{\ell'}^+ - \alpha_{\ell'}^-)(\mathbf{x}^{(\ell)})^T \mathbf{x}^{(\ell')} \\
& - \epsilon \sum_\ell (\alpha_\ell^+ + \alpha_\ell^-) + \sum_\ell r^{(\ell)} (\alpha_\ell^+ - \alpha_\ell^-) \\
\text{subject to} \quad & \sum_\ell (\alpha_\ell^+ - \alpha_\ell^-) = 0 \\
& 0 \le \alpha_\ell^+ \le C, \ \forall \ell \\
& 0 \le \alpha_\ell^- \le C, \ \forall \ell
\end{aligned}$$

- Instances in the ϵ-tube ($\alpha_\ell^+ = \alpha_\ell^- = 0$) are instances fitted with enough precision.
- The support vectors satisfy either $\alpha_\ell^+ > 0$ or $\alpha_\ell^- > 0$ and are of two types.
  - instances on the boundary of the ϵ-tube (either $0 < \alpha_\ell^+ < C$ or $0 < \alpha_\ell^- < C$), and we use these to calculate w_0
  - instances outside the ϵ-tube are instances for which we do not have a good fit (either $\alpha_\ell^+ = C$ or $\alpha_\ell^- = C$)

Dual Optimization Problem (2)

- We have the fitted line as a weighted sum of the support vectors:

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0 = \sum_{\mathbf{x}^{(\ell)} \in \mathcal{SV}} (\alpha_\ell^+ - \alpha_\ell^-)(\mathbf{x}^{(\ell)})^T \mathbf{x} + w_0$$

- Due to the sparseness property of the ϵ-insensitive loss function, only a small fraction of the training instances are support vectors which are used in defining the regression function (like the discriminant function for classification).
- Nonlinear (kernel) extension is possible by introducing appropriate kernel functions.

SVR