Review: Lecture 4

- **Perceptron Learning Algorithm (PLA)**
  - Learning algorithm for linear threshold functions (LTF) (iterative)
  - Energy function: PLA implements a stochastic gradient algorithm
  - Novikoff’s theorem (algebraic proof bounding number of updates, depends on margins)
  - Version space (set of points where weights classify all training points correctly)
  - Solution is ill posed, formulate optimization problem based on margins
Optimum Margin Classifiers

Consider methods based on optimum margin classifiers or Support Vector Machines (SVM)

Here we consider a different algorithm that is well posed with a unique optimal solution. The solution is based on finding the largest minimal margin of all training points.
Optimal Marginal Classifiers

Given a set of points that are linearly separable:

which hyperplane should you choose to separate points?

Choose hyperplane that maximizes distance between two sets of points.
Finding Optimal Hyperplane

1) Draw convex hull around each set of points.
2) Find shortest line segment connecting two convex hulls.
3) Find midpoint of line segment.
4) Optimal hyperplane is perpendicular to segment at midpoint of line segment.
Alternative Characterization of Optimal Margin Classifiers

Maximizing margins equivalent to minimizing magnitude of weight vector.

\[ \mathbf{w}^T (\mathbf{u} - \mathbf{v}) = 2 \]
\[ \mathbf{w}^T (\mathbf{u} - \mathbf{v}) / \| \mathbf{w} \| = 2 / \| \mathbf{w} \| = 2m \]

\[ \mathbf{w} \mathbf{u} + b = 1 \]
\[ \mathbf{w} \mathbf{v} + b = -1 \]
Quadratic Programming Problem

Problem statement:
max \(w, b\) \(\min (||x-x(i)||) : x \in \mathbb{R}^n, w^T x + b = 0, 1 \leq i \leq l\)

which is equivalent to solving (QP) problem:
Min \(\frac{1}{2}||w||^2\)
subject to \(d_i (w^T x(i) + b) \geq 1, \forall i\)
Lagrange Multipliers

Constrained optimization can be dealt with by introducing Lagrange multipliers $\alpha(i) \geq 0$ and augmenting objective function

$$L(w,b,\alpha) = \frac{1}{2}||w||^2 - \sum \alpha(i) (d(i) (w^T x(i) + b) - 1)$$
Support Vectors

QP program is convex and note that at solution
\[ \frac{\partial L(w,b,\alpha)}{\partial w} = 0 \quad \text{and} \quad \frac{\partial L(w,b,\alpha)}{\partial b} = 0 \]
leading to
\[ \sum \alpha(i) d(i) = 0 \quad \text{and} \quad w = \sum \alpha(i) d(i) x(i) \]
Weight vector expressed in terms of subset of training vectors \( x_i \) and \( \alpha_i \) that are nonzero. These are support vectors. By KKT conditions we have that
\[ \alpha(i) (d(i) (w^T x(i) + b) - 1) = 0 \]
indicating that support vectors lie on margin.
Support Vectors

Points in red are support vectors.
Dual optimization representation

Solving the primal QP problem is equivalent to solving the dual QP problem. Primal variables $w$, $b$ are eliminated.

$$\max \ W(\alpha) = \sum \alpha(i) - \frac{1}{2} \sum \alpha(i) \alpha(j) \ d(i) \ d(j) (x(i)^T x(j))$$
subject to $\alpha(i) \geq 0$, and $\sum \alpha(i) \ d(i) = 0$.

The hyperplane decision function can be written as

$$f(x) = \text{sgn} \ (\sum \alpha(i) \ d(i) \ x^T x(i) + b)$$
Comments about SVM solution

- Solution can be found in primal or dual spaces. When we discuss kernels later there are advantages of finding solution in dual space.
- Threshold, $b$ is determined after weight $w$ is found.
- In version space, SVM solution is center of largest hypersphere that fits in version space.
- Solution can be modified to
  - Handle training data that is not linearly separable via slack variables
  - Implement nonlinear discriminant function via kernels
- Solving quadratic programming problems
Handling data that not linearly separable

- Data is often noisy or inconsistent resulting in training data being not linearly separable.
- PLA algorithm can be modified (pocket algorithm), however there are problems with algorithm termination.
- SVM can easily handle this case by adding slack variables to handle pattern recognition.
Training examples below margin

margins

Optimal hyperplane
Formulation with slack variables

Optimal margin classifier with slack variables and kernel functions described by Support Vector Machine (SVM).

\[
\min_{(w, \xi)} \frac{1}{2}||w||^2 + \gamma \sum \xi(i) \\
\text{subject to } \xi(i) \geq 0 \ \forall i, \ d(i) (w^T x(i) + b) \geq 1 - \xi(i), \ \forall i, \text{ and } \gamma > 0.
\]

In dual space

\[
\max W(\alpha) = \sum \alpha(i) - \frac{1}{2} \sum \alpha(i) \alpha(j) d(i) d(j) x(i)^T x(j) \\
\text{subject to } \gamma \geq \alpha(i) \geq 0, \text{ and } \sum \alpha(i) d(i) = 0.
\]

Weights can be found by \( w = \sum \alpha(i) d(i) x(i) \).
Solving QP Problem

- Quadratic programming problem with linear inequality constraints.
- Optimization problem involves searching space of feasible solutions (points where inequality constraints satisfied).
- Can solve problem in primal or dual space.
QP software for SVM

- Matlab (easy to use, choose primal or dual space, slow): quadprog() (need optimization toolbox)
  - Primal space (w,b, \( \xi^+, \xi^- \))
  - Dual space (\( \alpha \))
- Sequential Minimization Optimization (SMO) (specialized for solving SVM, fast): decomposition method, chunking method
- SVM light (fast): decomposition method
- LIBSVM software; National Taiwan Univ., Chih-Chung Chan, Chih-Jen Lin; C++, Java interfaces with Python, Matlab, Octave, R
Example

Drawn from Gaussian data $\text{cov}(X) = I$

20 + pts. Mean = (.5,.5)
20 - pts. Mean = -(.5,.5)
Example continued

Primal Space (matlab)
x = randn(40,2);
d = [ones(20,1); -ones(20,1)];
x = x + d * [.5 .5];
H = diag([0 1 1 zeros(1,80)]);
gamma = 1;
f = [zeros(43,1); gamma*ones(40,1)];
Aeq = [d x.*(d*[1 1]) -eye(40) eye(40)];
beq = ones(40,1);
A = zeros(1,83);
b = 0;
lb = [-inf*ones(3,1); zeros(80,1)];
ub = [inf*ones(83,1)];
[w,fval] = quadprog(gamma*H,f,A,b,Aeq,beq,lb,ub);
Example continued

Dual Space (matlab)

\[ \mathbf{x}_n = \mathbf{x} \ast (\mathbf{d} \ast [1 \ 1]) \]

\[ k = \mathbf{x}_n \ast \mathbf{x}_n' \]

\[ \gamma = 1 \]

\[ f = -\text{ones}(40,1) \]

\[ \mathbf{A}_\text{eq} = \mathbf{d}' \]

\[ \mathbf{b}_\text{eq} = 0 \]

\[ \mathbf{A} = \text{zeros}(1,40) \]

\[ \mathbf{b} = 0 \]

\[ \mathbf{lb} = \text{zeros}(40,1) \]

\[ \mathbf{ub} = [\gamma \ast \text{ones}(40,1)] \]

\[ [\alpha, \mathbf{fvala}] = \text{quadprog}(k, f, \mathbf{A}, \mathbf{b}, \mathbf{A}_\text{eq}, \mathbf{b}_\text{eq}, \mathbf{lb}, \mathbf{ub}) \]
Example continued

• \( w = (1.4245, 0.4390)^T \) \( b = 0.1347 \)
• \( w = \sum \alpha(i) d(i) x(i) \) (26 support vectors, 3 lie on margin hyperplane)
  - \( \alpha(i) = 0 \), \( x(i) \) above margin
  - \( 0 \leq \alpha(i) \leq \gamma \), \( x(i) \) lie on margin hyperplanes
  - \( \alpha(i) = \gamma \), \( x(i) \) lie below margin hyperplanes
• Hyperplane can be represented in
  - Primal space: \( w^T x + b = 0 \)
  - Dual space: \( \sum \alpha(i) d(i) x^T x(i) + b = 0 \)
• Regularization parameter \( \gamma \) controls balance between margin and errors.