Goals:
1. Develop geometric intuition for support vector machines, soft margins, kernel trick
2. Develop algorithm to learn decision trees and build ensemble learner

Statistical inference

\[ p(y | x; \theta) \]
\[ \text{Generative model builds } p(x, y; \theta). \]
(Lecture 4, 8, 11, 12, 13)

Discriminative model does not.
(today)

Support Vector Machine (SVM)

Given a point \( x \in \mathbb{R}^n \), output class \( y \in \{-1, +1\} \)

Strategy: pick hyperplane which separates points in training data \((x_1, y_1), \ldots, (x_n, y_n)\)

\[ \begin{align*}
  &w \cdot x + b = 1 \\
  &w \cdot x + b = 0 \\
  &w \cdot x + b = -1
\end{align*} \]

How to pick hyperplane? Maximize the margin (distance to closest point)

\[
\min_{w, b} \frac{1}{2} \|w\|^2 \\
\text{ s.t. } y_i (w \cdot x + b) > 1 \quad \forall i
\]

This is a quadratic program, but hard to solve.
Work on the dual problem:

\[
\max_{\alpha} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j [x_i \cdot x_j]
\]

s.t. \( \alpha_i \geq 0 \) \( \forall i \)

\[\sum_i \alpha_i y_i = 0 \]

From the opt. solution, recover opt. solution to primal problem:

\[w^* = \sum_i \alpha_i^* y_i x_i\]

Many \( \alpha_i = 0 \) in optimal solution. If \( \alpha_i > 0 \), \( x_i \) is a support vector (on the margin). Recover opt. \( b^* \) from any SV:

\[y_m (w \cdot x_m + b^*) = 1\]

**How to classify a new point?**

\[y' = \text{sign} (w^* \cdot x' + b^*)\]

**Generalization bound:** How well can classifier do on new data?

One bound: leave-one-out error.

**Theorem:** \( P(\text{L.O.O.E.}) < \frac{\# \text{SVs}}{n+1} \)

**Extensions:** ubiquitous in practice.

1. **Soft margin:** allow misclassifications in training data to improve generalization.

\[\min \frac{1}{2} ||w||^2 + C \sum \xi_i\]

\( \xi_i \) slack variables allow error

Tunable parameter (set by cross-validation)
Kernel trick: map data to a higher dimensional space where it can be separated by a hyperplane.

Problem: doing the transform takes additional time, space
Observation: we only ever need the dot product
Kernel function: \( K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \) in some space

Gaussian radial basis function kernel

\[
K(x_i, x_j) = \exp \left(-\gamma \|x_i - x_j\|^2\right)
\]

Claim: This is a dot product in an infinite-dimensional space!

Proof sketch: Manipulation of Taylor series for \( \exp(x) \).

There are conditions to rigorously prove \( K(\cdot, \cdot) \) is an inner product (Mercer's conditions), but hard to prove.

Decision Trees

ex.) Do 2 proteins interact?

Expression \( r^2 > 0.9 \)?

\[
\begin{array}{c}
\text{No} \\
\text{Yes}
\end{array}
\]

Shared localization?

\[
\begin{array}{c}
\text{No} \\
\text{Yes}
\end{array}
\]

Shared function?

\[
\begin{array}{c}
\text{No} \\
\text{Yes}
\end{array}
\]

\[
\text{No}
\]
Advantages: white box, interpretable model. Can recover which features are most important.

How to build a decision tree?

Given examples \((x_1, y_1), \ldots, (x_n, y_n)\):

- If should split sample:
  - Pick feature set that optimally splits
  - Recurse

How to pick feature set? Minimize impurity of partitions

Two common measures:

1. Entropy: \(\sum_k p(y = k) \log p(y = k)\)
2. Gini impurity: \(1 - \sum_k p(y = k)^2\)

How to terminate recursion?

1. All samples have the same label
2. Not enough gain from splitting

How to classify a new point?

- Traverse tree according to splits. At leaf, output majority label.

Problem: splits can be arbitrarily complicated

Solution: random projection: pick from random, fixed size subset at every level.

Problem: single tree may not capture all features

Solution: build many trees, take majority vote (Random Forest)

Problem: not enough examples to train many trees

Solution: sample examples with replacement (Bootstrap aggregating, bagging)