Announcements:

- Project Milestones coming up
- No class Thurs, Nov 10
- HW2
  - Let’s figure it out...
- HW3 posted this week.
  - Let’s get state of the art on MNIST!
  - It’ll be collaborative
- Today:
  - Review: Kernels
  - SVMs
  - Generalization/review
What if the data is not linearly separable?

Use features of features of features of features....

$$\Phi(x) : \mathbb{R}^m \mapsto F$$

Feature space can get really large really quickly!
Common kernels

- Polynomials of degree exactly $d$
  $$K(u, v) = (u \cdot v)^d$$
- Polynomials of degree up to $d$
  $$K(u, v) = (u \cdot v + 1)^d$$
- Gaussian (squared exponential) kernel
  $$K(u, v) = \exp \left( -\frac{|u - v|^2}{2\sigma^2} \right)$$
- Sigmoid
  $$K(u, v) = \tanh(\eta u \cdot v + \nu)$$

Mercer’s Theorem

- When do we have a Kernel $K(x, x')$?
- Definition 1: when there exists an embedding
  $$K(x, x') = \phi(x) \cdot \phi(x')$$
- Mercer’s Theorem:
  - $K(x, x')$ is a valid kernel if and only if $K$ is a positive semi-definite.
  - PSD in the following sense:
    $$\forall x \in \mathcal{X}, \forall f \in \mathcal{F}, \quad \sum_{i,j} M_{ij} f_i(x) f_j(x') \geq 0$$
Linear classifiers – Which line is better?
Pick the one with the largest margin!

"confidence" = \( y^j (w \cdot x^j + w_0) \)

Maximize the margin

\[
\max_{\gamma, w, w_0} \gamma \quad \text{s.t.} \quad \|w\| = 1.
\]

\[
y^j (w \cdot x^j + w_0) \geq \gamma, \forall j \in \{1, \ldots, N\}
\]
But there are many planes...

Review: Normal to a plane

\[ \mathbf{x}^j = \bar{\mathbf{x}}^j + \alpha \frac{\mathbf{w}}{||\mathbf{w}||} \]
A Convention: Normalized margin – Canonical hyperplanes

\[ x^j = \bar{x}^j + \alpha \frac{w}{||w||} \]

Margin maximization using canonical hyperplanes

Unnormalized problem:

\[
\max_{\gamma, w, w_0} \gamma \quad y^j (w \cdot x^j + w_0) \geq \gamma, \forall j \in \{1, \ldots, N\} 
\]

Normalized Problem:

\[
\min_{w, w_0} ||w||_2^2 \\
y^j (w \cdot x^j + w_0) \geq 1, \forall j \in \{1, \ldots, N\}
\]
Support vector machines (SVMs)

\[ \min_{w, w_0} \|w\|^2 \quad \text{s.t.} \quad y_j (w \cdot x_j + w_0) \geq 1, \forall j \in \{1, \ldots, N\} \]

- Solve efficiently by many methods, e.g.,
  - quadratic programming (QP)
  - Well-studied solution algorithms
  - Stochastic gradient descent
- Hyperplane defined by support vectors

What if the data is not linearly separable?

Use features of features of features of features....
What if the data is still not linearly separable?

\[
\min_{w, w_0} \|w\|_2^2
\]
\[
y^j(w \cdot x^j + w_0) \geq 1, \quad \forall j
\]

- If data is not linearly separable, some points don’t satisfy margin constraint:
  - How bad is the violation?
- Tradeoff margin violation with \(\|w\|\):

SVMs for Non-Linearly Separable meet my friend the Perceptron…

- Perceptron was minimizing the hinge loss:
  \[
  \sum_{j=1}^{N} (-y^j (w \cdot x^j + w_0))_{+}
  \]
- SVMs minimizes the regularized hinge loss!!
  \[
  \|w\|_2^2 + C \sum_{j=1}^{N} (1 - y^j (w \cdot x^j + w_0))_{+}
  \]
Stochastic Gradient Descent for SVMs

- Perceptron minimization:
  \[ \sum_{j=1}^{N} (-y_j(w \cdot x_j + w_0))_+ \]
- SGD for Perceptron:
  \[ w(t+1) \leftarrow w(t) + \eta \left[ y^{(t)}(w^{(t)} \cdot x^{(t)}) \leq 0 \right] y^{(t)}x^{(t)} \]

- SVMs minimization:
  \[ \|w\|_2^2 + C \sum_{j=1}^{N} (1 - y_j(w \cdot x_j + w_0))_+ \]
- SGD for SVMs:

SVMs vs logistic regression

- We often want probabilities/confidences (logistic wins here)
- For classification loss, they are comparable
- Multiclass setting:
  - Softmax naturally generalizes logistic regression
  - SVMs have generalization win for logistic
- What about good old least squares?
Multiple Classes

- One can generalize the hinge loss
  - If no error (by some margin) \( \rightarrow \) no loss
  - If error, penalize what you said against the best

- SVMs vs logistic regression
  - We often want probabilities/confidences (logistic wins here)
  - For classification loss, they are

- Latent SVMs
  - When you have many classes it's difficult to do logistic regression

- 2) Kernels
  - Warp the feature space

SVMs can be reasonable with many classes (if partition difficult to compute for softmax)
What method should I use?

- Linear regression, logistic, SVMs?
- No regularization? Ridge? L1?

- I ran SGD without any regularization and it was ok?
Generalization

- You get N samples.
- You learn a classifier/regression $f^\wedge$.
- How close are you to optimal?
  $$L(f^\wedge) - L(f^*) < ???$$
- (We can look at the above in expectation or with ‘high’ probability).

Finite Case:

- You get N samples.
- You learn a classifier/regressor $f^\wedge$ among K classifiers:
  $$L(f^\wedge) - L(f^*) < \sqrt{\log(1/S)}$$
  with probability $\geq 1 - S$
Linear Regression

- N samples, d dimensions.
- L is the square loss.
- \( w^\) is the least squares estimate.

\[ L(w^\) - L(w*) < O(d/N) \]

- Need about N=O(d) samples

Sparse Linear Regression

- N samples, d dimensions, L is the square loss.
- \( f^\) is best fit line which only uses k features (computationally intractable)

\[ L(w^\) - L(w*) < k \log(d)/N \]

- true of Lasso under stronger assumptions: "incoherence"
- When do like sparse regression??
  - When we believe there are a few of GOOD features.
Learning a Halfspace

- You get $N$ samples, in $D$ dimensions.
- $L$ is the 0/1 loss.
- $f^*$ is the empirical risk minimizer (computationally infeasible to compute)

$$L(w^*) - L(w^*) < \sqrt{d \log(N)/N}$$

- Need $N=O(d)$ samples

What about Regularization?

- Let’s look at (dual) constrained problem
- Minimize:

$$\min L^*(w)$$

such that $||w|| < W_+$

- Where $L^*$ is our training error.
Optimization and Regularization?

- I did SGD without regularization and it was fine?
- “Early stopping” implicitly regularizes (in L2)

L2 Regularization

- Assume $\|w\|_2 < W_2$ $\|x\|_2 < R_2$
- L is some convex loss (logistic, hinge, square)
- $w^\wedge$ is the constrained minimizer (computationally tractable to compute)

$$L(w^\wedge) - L(w^*) < W_2 R_2 / \sqrt{N}$$

- DIMENSION FREE “margin” Bound!
L1 Regularization

- Assume $||w||_1 < W_1$ $||x||_\infty < R_\infty$
- L is some convex loss (logistic, hinge, square)
- $w^*$ is the constrained minimizer (computationally tractable to compute)

$$L(w^*)-L(w^*) < \frac{W_1 R_\infty \log(d)}{\sqrt{N}}$$

- Promotes sparsity, one can think of $W_1$ as the “sparsity level/k” (mild dimension dependence, $\log(d)$).