ECE 5984: Introduction to Machine Learning

Topics:
- SVM
  - Lagrangian Duality
  - SVM dual & kernels

Readings: Barber 17.5

Dhruv Batra
Virginia Tech
HW1 Graded

- Mean 60/55 = 109%

![Histogram](image.png)

Max: 74; Min: 41; Avg: 60
HW1 Extra Credit Distribution
Administrativia

• HW2
  – Due: Friday 03/06, 03/15, 11:55pm
  – Implement linear regression, Naïve Bayes, Logistic Regression
  – Solutions available
  – Kaggle discussion:
Administrativia

• Mid-term
  – Solutions available

• Feedback on Midterm Exam?
  – Too hard? Too easy? Just right?
  – Too long? Too short?
Recap of Last Time
Generative vs. Discriminative

• Generative Approach (Naïve Bayes)
  – Estimate $p(x|y)$ and $p(y)$
  – Use Bayes Rule to predict $y$

• Discriminative Approach
  – Estimate $p(y|x)$ directly (Logistic Regression)
  – Learn “discriminant” function $f(x)$ (Support Vector Machine)
Linear classifiers – Which line is better?

\[ w \cdot x = \sum_j w^{(j)} x^{(j)} \]
Margin

\[ w \cdot x^+ + b = +1 \]
\[ w \cdot x^- + b = -1 \]
\[ w \cdot x = 0 \]

Margin

\[ x^+ = x^- + \lambda w \]
\[ w \cdot x^+ + b = 1 \]
\[ w \cdot (x^- + \lambda \frac{w}{||w||}) + b = 1 \]

\[ \lambda = \frac{2}{||w||} \]
\[ \gamma = \frac{1}{\sqrt{w \cdot w}} \]

(C) Dhruv Batra
Slide Credit: Carlos Guestrin
Support vector machines (SVMs)

\[ w \cdot x + b = +1 \]
\[ w \cdot x + b = -1 \]
\[ w \cdot x + b = 0 \]

\text{margin } 2\gamma

\begin{align*}
\text{minimize}_{w, b} & \quad w \cdot w \\
\text{subject to} & \quad (w \cdot x_j + b) y_j \geq 1, \quad \forall j
\end{align*}

- Solve efficiently by quadratic programming (QP)
  - Well-studied solution algorithms
- Hyperplane defined by support vectors
What if the data is not linearly separable?
What if the data is not linearly separable?

\[
\text{minimize}_{w,b} \quad w \cdot w \quad \left( w \cdot x_j + b \right) y_j \geq 1, \forall j
\]

- Minimize \( w \cdot w \) and number of training mistakes
  - 0/1 loss
  - Slack penalty \( C \)
  - Not QP anymore
  - Also doesn’t distinguish near misses and really bad mistakes

Slide Credit: Carlos Guestrin
Slack variables – Hinge loss

\[
\text{minimize}_{w, b} \quad w \cdot w + C \sum_j \xi_j \\
(w \cdot x_j + b) y_j \geq 1 - \xi_j, \quad \forall j \\
\xi_j \geq 0, \quad \forall j
\]

- If margin $\geq 1$, don’t care
- If margin $< 1$, pay linear penalty
Soft Margin SVM

• Effect of C
  – Matlab demo by Andrea Vedaldi
Side note: What’s the difference between SVMs and logistic regression?

**SVM:**

$$\text{minimize}_{\mathbf{w}, b} \quad \mathbf{w} \cdot \mathbf{w} + C \sum_j \xi_j$$

$$(\mathbf{w} \cdot \mathbf{x}_j + b) y_j \geq 1 - \xi_j, \quad \forall j$$

$$\xi_j \geq 0, \quad \forall j$$

**Logistic regression:**

$$P(Y = 1 \mid x, \mathbf{w}) = \frac{1}{1 + e^{-(\mathbf{w} \cdot \mathbf{x} + b)}}$$

Log loss:

$$-\ln P(Y = 1 \mid x, \mathbf{w}) = \ln \left(1 + e^{-(\mathbf{w} \cdot \mathbf{x} + b)}\right)$$

(C) Dhruv Batra

Slide Credit: Carlos Guestrin
Harder 1-dimensional dataset

That's wiped the smirk off SVM's face.

What can be done about this?
Harder 1-dimensional dataset

Remember how permitting non-linear basis functions made linear regression so much nicer?

Let’s permit them here too

\[ z_k = (x_k, x_k^2) \]
Harder 1-dimensional dataset

Remember how permitting non-linear basis functions made linear regression so much nicer?

Let’s permit them here too

\[ z_k = (x_k, x_k^2) \]
Does this always work?

• In a way, yes

**Lemma**

Let \((x_i)_{i=1,\ldots,n}\) with \(x_i \neq x_j\) for \(i \neq j\). Let \(\varphi : \mathbb{R}^k \rightarrow \mathbb{R}^m\) be a feature map. If the set \(\varphi(x_i)_{i=1,\ldots,n}\) is linearly independent, then the points \(\varphi(x_i)_{i=1,\ldots,n}\) are linearly separable.

**Lemma**

If we choose \(m > n\) large enough, we can always find a map \(\varphi\).
Caveat

Caveat: We can separate any set, not just one with “reasonable” $y_i$:

There is a fixed feature map $\varphi : \mathbb{R}^2 \to \mathbb{R}^{20001}$ such that – no matter how we label them – there is always a hyperplane classifier that has 0 training error.
Kernel Trick

• One of the most interesting and exciting advancement in the last 2 decades of machine learning
  – The “kernel trick”
  – High dimensional feature spaces at no extra cost!

• But first, a detour
  – Constrained optimization!
Constrained Optimization

• Lagrangian Multiplier Method

• \( \text{min } f(w) \)
  \[ \text{st } h(w) = 0 \]

• Define Lagrangian
Intuition

\[ f(x,y) \]
Intuition

\[ f(x,y) = d_1 \]

\[ g(x,y) = c \]

\[ f(x,y) = d_2 \]
Lagrangian Duality

• On paper