ECE 5984: Introduction to Machine Learning

Topics:
  – SVM
    – soft & hard margin
    – comparison to Logistic Regression

Readings: Barber 17.5

Dhruv Batra
Virginia Tech
New Topic
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?

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

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

Margin

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

\[ \lambda = \frac{2}{||w||} \]
\[ \gamma = \frac{1}{\sqrt{w \cdot w}} \]
Support vector machines (SVMs)

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

- Margin \( 2\gamma \)
- Solve efficiently by quadratic programming (QP)
  - Well-studied solution algorithms
- Hyperplane defined by support vectors

\[
\begin{align*}
    \text{minimize}_{w,b} & \quad w \cdot w \\
    \left( w \cdot x_j + b \right) y_j & \geq 1, \quad \forall j
\end{align*}
\]
What if the data is not linearly separable?
What if the data is not linearly separable?

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

- 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
Slack variables – Hinge loss

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

- 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:**

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

**Logistic regression:**

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

Log loss:

\[-\ln P(Y = 1 \mid x, w) = \ln \left(1 + e^{-(w \cdot x + b)}\right)\]
Harder 1-dimensional dataset

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

What can be done about this?

\[ x = 0 \]
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 \((x_i)_{i=1,...,n}\) 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 \(\varphi(x_i)_{i=1,...,n}\) is linearly independent, then the points \(\varphi(x_i)_{i=1,...,n}\) 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 $\varphi : \mathbb{R}^2 \rightarrow \mathbb{R}^{20001}$ 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!