15-859(B) Machine Learning Theory

More on why large margins are good for learning. Kernels and general similarity functions. $L_1 - L_2$ connection.

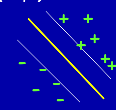
Avrim Blum

02/19/14

Margins

If data is separable by large margin γ, then that's a good thing. Need sample size only $\tilde{O}(1/\gamma^2)$ to learn to constant error rate.

$\boxed{|w \cdot x| \geq \gamma,\ \|w\| = 1, \|x\| = 1}$



Some ways to see it:

1. The perceptron algorithm does well: makes only $1/\gamma^2$ mistakes.
2. Margin bounds: whp all consistent large-margin separators have low true error.
3. Really-Simple-Learning + boosting…
4. Random projection… Today: 3 & 4.

A really simple learning algorithm

Suppose data is separable by margin γ. Here is another way to see why this is good for learning.

Consider the following simple algorithm…

1. Pick a random linear separator.
2. See if it is any good.
3. If it is a weak hypothesis (error rate $\leq \frac{1}{2} - \gamma/4$), plug into boosting. Else don't. Repeat.

Claim: if $\exists$ a large margin separator, then $\geq c\gamma$ chance that random separator is weak hyp.

Can pick random separators before seeing data, so can view as $MAJ_k(H)$ for $k = O(1/\gamma^2), |H| = O(k/\gamma)$

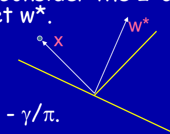
A really simple learning algorithm

Claim: if data has a separator of margin γ, there's a reasonable chance a random linear separator will have error $\leq \frac{1}{2} - \gamma/4$. [all hyperplanes through origin]

Proof: Consider random h s.t. $h \cdot w^* \geq 0$:

- Pick a (positive) example x. Consider the 2-d plane defined by x and target w*.
- $\Pr_h(h \cdot x \leq 0 \mid h \cdot w^* \geq 0)$ $\leq (\pi/2 - \gamma)/\pi = \frac{1}{2} - \gamma/\pi$.
- So, $E_h[err(h) \mid h \cdot w^* \geq 0] \leq \frac{1}{2} - \gamma/\pi$.
- Since err(h) is bounded between 0 and 1, there must be an $\Omega(\gamma)$ chance of success.

QED



Another way to see why large margin is good

Johnson-Lindenstrauss Lemma:

Given n points in R^n, if project randomly to R^k, for $k = O(\varepsilon^{-2} \log n)$, then whp all pairwise distances preserved up to $1 \pm \varepsilon$ (after scaling by $(n/k)^{1/2}$).

Cleanest proofs: IndykMotwani98, DasguptaGupta99

JL Lemma, cont

Given n points in R^n, if project randomly to R^k, for $k = O(\varepsilon^{-2} \log n)$, then whp all pairwise distances preserved up to $1 \pm \varepsilon$ (after scaling).
Cleanest proofs: IM98, DG99

Proof easiest for slightly different projection:

- Pick k vectors $u_1, \ldots, u_k$ iid from n-diml gaussian.
- Map $p \to (p \cdot u_1, \ldots, p \cdot u_k)$.
- What happens to $v_{ij} = p_i - p_j$?
  - Becomes $(v_{ij} \cdot u_1, \ldots, v_{ij} \cdot u_k)$
  - Each component is iid from 1-diml gaussian, scaled by $|v_{ij}|$.
  - For concentration on sum of squares, plug in version of Hoeffding for RVs that are squares of gaussians.
- So, whp all lengths apx preserved, and in fact not hard to see that whp all angles are apx preserved too.

Random projection and margins

Natural connection [ArriagaVempala99]:

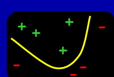
- Suppose we have a set S of points in R^n, separable by margin γ.
- JL lemma says if project to random k-dimensional space for $k=O(\gamma^{-2} \log |S|)$, whp still separable (by margin $\gamma/2$).
  - Think of projecting points and target vector w.
  - Angles between p_i and w change by at most $\pm\gamma/2$.
- Could have picked projection before sampling data.
- So, it's really just a k-dimensional problem after all. Do all your learning in this k-diml space.

So, random projections can help us think about why margins are good for learning. [note: this argument does NOT imply uniform convergence in original space]

OK, now to another way to view kernels…

Kernel function recap

- We have a lot of great algorithms for learning linear separators (perceptron, SVM, …). But, a lot of time, data is not linearly separable.
  - One option: use a more complicated algorithm.
  - Another option: use a kernel function!
- Many algorithms only interact with the data via dot-products.
  - So, let's just re-define dot-product.
  - E.g., $K(x,y) = (1 + x \cdot y)^d$.
    - $K(x,y) = \phi(x) \cdot \phi(y)$, where $\phi()$ is implicit mapping into an n^d-dimensional space.
  - Algorithm acts as if data is in "ϕ-space". Allows it to produce non-linear curve in original space.
  - Don't have to pay for high dimension if data is linearly separable there by a large margin.



Question: do we need the notion of an implicit space to understand what makes a kernel helpful for learning?

Can we develop a more intuitive theory?

- Match intuition that you are looking for a good measure of similarity for the problem at hand?
- Get the power of the standard theory with less of "something for nothing" feel to it?

And remove even need for existence of Φ?

Can we develop a more intuitive theory?

What would we intuitively want in a good measure of similarity for a given learning problem?

A reasonable idea:

- Say have a learning problem P (distribution D over examples labeled by unknown target f).
- Sim fn K:(,)→[-1,1] is good for P if:

  most x are on average more similar to random pts of their own label than to random pts of the other label, by some gap γ.

  E.g., most images of men are on average γ-more similar to random images of men than random images of women, and vice-versa.

  (Scaling so all values in [-1,1])

A reasonable idea:

- Say have a learning problem P (distribution D over examples labeled by unknown target f).
- Sim fn $K:(x,y)\to[-1,1]$ is (ε,γ)-good for P if at least a $1-\varepsilon$ fraction of examples x satisfy:

$$E_{y\sim D}[K(x,y)|\ell(y)=\ell(x)] \geq E_{y\sim D}[K(x,y)|\ell(y)\neq\ell(x)]+\gamma$$

E.g., most images of men are on average γ-more similar to random images of men than random images of women, and vice-versa.

(Scaling so all values in [-1,1])

A reasonable idea:

- Say have a learning problem P (distribution D over examples labeled by unknown target f).
- Sim fn $K:(x,y)\to[-1,1]$ is (ε,γ)-good for P if at least a $1-\varepsilon$ fraction of examples x satisfy:

$$E_{y\sim D}[K(x,y)|\ell(y)=\ell(x)] \geq E_{y\sim D}[K(x,y)|\ell(y)\neq\ell(x)]+\gamma$$

How can we use it?

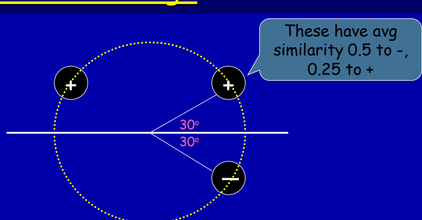
Just do "average nearest-nbr"

At least a $1-\varepsilon$ fraction of x satisfy:

$$E_{y\sim D}[K(x,y)|\ell(y)=\ell(x)] \geq E_{y\sim D}[K(x,y)|\ell(y)\neq\ell(x)]+\gamma$$

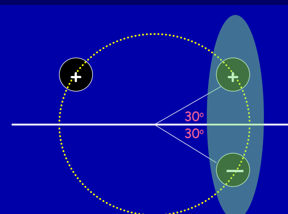
- Draw S^+ of $O((1/\gamma^2)\ln 1/\delta^2)$ positive examples.
- Draw S^- of $O((1/\gamma^2)\ln 1/\delta^2)$ negative examples
- Classify x based on which gives better score.
  - Hoeffding: for any given "good x", prob of error over draw of S^+,S^- at most δ^2.
  - So, at most δ chance our draw is bad on more than δ fraction of "good x".
- With prob $\geq 1-\delta$, error rate $\leq \varepsilon + \delta$.

But not broad enough



- K(x,y)=x·y has good separator but doesn't satisfy defn. (half of positives are more similar to negs that to typical pos)

But not broad enough



- Idea: would work if we didn't pick y's from top-left.
- Broaden to say: OK if ∃ large region R s.t. most x are on average more similar to y∈R of same label than to y∈R of other label. (even if don't know R in advance)

Broader defn…

- Ask that exists a set R of "reasonable" y (allow probabilistic) s.t. almost all x satisfy

$E_y[K(x,y)|\ell(x)=\ell(y), y \in R] \geq E_y[K(x,y)|\ell(x)\neq\ell(y), y \in R]+\gamma$

- Formally, say K is $(\varepsilon',\gamma,\tau)$-good if $E_x[\gamma\text{-hinge loss}(x)] \leq \varepsilon'$, and $\Pr(R_+), \Pr(R_-) \geq \tau$.
- **Thm 1:** this is a legitimate way to think about good kernels:
  - If kernel has margin γ in implicit space, then for any τ is (τ,γ^2,τ)-good in this sense.

Broader defn…

- Ask that exists a set R of "reasonable" y (allow probabilistic) s.t. almost all x satisfy

$E_y[K(x,y)|\ell(x)=\ell(y), y \in R] \geq E_y[K(x,y)|\ell(x)\neq\ell(y), y \in R]+\gamma$

- Formally, say K is $(\varepsilon',\gamma,\tau)$-good if $E_x[\gamma\text{-hinge loss}(x)] \leq \varepsilon'$, and $\Pr(R_+), \Pr(R_-) \geq \tau$.
- **Thm 2:** even if not a legal kernel, this is nonetheless sufficient for learning.
  - If K is $(\varepsilon',\gamma,\tau)$-good, $\varepsilon' \ll \varepsilon$, can learn to error ε with $O\left(\frac{1}{\epsilon\gamma^2}\log\frac{1}{\epsilon\gamma\tau}\right)$ labeled examples.

[and $\tilde{O}(1/(\gamma^2\tau))$ unlabeled examples]

How to use such a sim fn?

- Assume $\exists$ R s.t. $\Pr_y[R_+,R_-] \geq \tau$ and almost all x satisfy $E_y[K(x,y)|\ell(x)=\ell(y), y \in R] \geq E_y[K(x,y)|\ell(x)\neq\ell(y), y \in R]+\gamma$
  - Draw $S = \{y_1,\ldots,y_n\}$, $n \approx 1/(\gamma^2\tau)$. could be unlabeled
  - View as "landmarks", use to map new data: $F(x) = [K(x,y_1), \ldots, K(x,y_n)]$.
  - Whp, exists separator of good L_1 margin in this space: $w=[0,0,1/n_+,1/n_+,0,0,0,-1/n_-,0]$

    $(n_+ = \# y_i \in R_+, n_- = \# y \in R_-)$
  - So, take new set of examples, project to this space, and run good L_1 alg (Winnow).

Other notes

- So, large margin in implicit space $\Rightarrow$ satisfy this defn (with potentially quadratic penalty in margin).
- Can apply to similarity functions that are not legal kernels. E.g.,
  - K(x,y)=1 if x,y within distance d, else 0.
  - $K(s_1, s_2)$ = output of arbitrary dynamic-programming alg applied to s_1, s_2, scaled to [-1,1].
  - Nice work on using this in the context of edit-distance similarity fns for string data [Bellet-Sebban-Habrard 11]
- This def is really an L_1 style margin, so has nice properties:
  - E.g., given k similarity fns with hope that some convex combination is good: only log(k) blowup in sample size.