

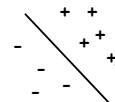
15-859(B) Machine Learning Theory

Lecture 13: Margins, kernels, and similarity functions

Avrim Blum
02/27/07

Basic Supervised learning setting

- w Examples are points x in instance space, like \mathbb{R}^n .
- w Labeled + or -.
- w Assume drawn from some probability distribution:
 - n Distribution D over x , labeled by target function c .
 - n Or distribution P over (x, l)
 - n Will call P (or (c, D)) our "learning problem".
- w Given labeled training data, want algorithm to do well on new data.



Margins

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

$$|w \cdot x| / |x| \geq \gamma, \quad |w| = 1$$



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...

A really simple learning algorithm

Suppose our problem has the property that whp a sufficiently large sample S would be separable by margin γ . Here is another way to see why this is good for learning. [Nina mentioned this last time].

Consider the following simple algorithm...

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

Claim: if data has a large margin separator, there's a reasonable chance a random hyperplane will be a weak-learner.

A really simple learning algorithm

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

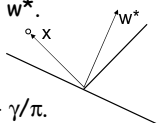
Proof:

w Pick a (positive) example x . Consider the 2-d plane defined by x and target w^* .

$$\Pr_h(h \cdot x \leq 0 \mid h \cdot w^* \geq 0) \leq (\pi/2 - \gamma)/\pi = \frac{1}{2} - \gamma/\pi.$$

$$\text{So, } E_h[\text{err}(h) \mid h \cdot w^* \geq 0] \leq \frac{1}{2} - \gamma/\pi.$$

w Since $\text{err}(h)$ is bounded between 0 and 1, there must be a reasonable chance of success.



Application to Semi-Supervised learning

- w As Nina mentioned, in Co-Training, under assumption of independence given the label, can boost weak h from unlabeled data.
 - n Given (x_1, x_2) , compute $h(x_1)$ and use as label for x_2 .
 - n As long as problem is learnable from data with noisy labels, then this will do it.
- w Simple alg shows: if target is large-margin separator, can randomly choose initial hyps, use unlabeled data to bootstrap, and then use labeled data to pick.
 - o Only requires $O(1)$ labeled examples (in fact, just 1).

Another way to see why large margin is good

Johnson-Lindenstrauss Lemma:

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

Cleanest proofs: IM98, DG99

JL Lemma

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

Cleanest proofs: IM98, DG99

Proof intuition:

- w Consider a random unit-length vector $(x_1, x_2, \dots, x_n) \in \mathbb{R}^n$. What does x_i coordinate look like?
- w $E[x_i^2] = 1/n$. Usually $\leq c/n$.
- w If indep, $\Pr[|(x_1^2 + \dots + x_k^2) - k/n| \geq \epsilon k/n] \leq e^{-O(\epsilon^2 k)}$.
- w So, at $k = O(\epsilon^{-2} \log n)$, with prob $1 - 1/\text{poly}(n)$, projection to 1st k coordinates has length $(k/n)^{1/2} (1 \pm \epsilon)$.
- w Now, apply this to vector $v_{ij} = p_i - p_j$, projecting onto random k -diml space.
Whp all v_{ij} project to length $(k/n)^{1/2}(1 \pm \epsilon)|v_{ij}|$

JL Lemma, cont

Proof easiest for slightly different projection:

- w Pick k vectors u_1, \dots, u_k iid from n -diml gaussian.
- w Map $p \rightarrow (p \cdot u_1, \dots, p \cdot u_k)$.
- w What happens to $v_{ij} = p_i - p_j$?
 - Becomes $(v_{ij} \cdot u_1, \dots, v_{ij} \cdot u_k)$
 - 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.
- w 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 [AV99]:

- w Suppose we have a set S of points in \mathbb{R}^n , separable by margin γ .
- w 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 $\pm \gamma/2$.
- w Could have picked projection before sampling data.
- w 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 doesn't imply uniform convergence in original space]

OK, now on to kernels...

Generic problem

- w Given a set of images: , want to learn a linear separator to distinguish men from women.
- w Problem: pixel representation no good.

Old style advice:

- w Pick a better set of features!
- w But seems ad-hoc.

New style advice:

- w Use a Kernel! $K(\text{img}_1, \text{img}_2) = \Phi(\text{img}_1) \cdot \Phi(\text{img}_2)$. Φ is implicit, high-dimensional mapping.
- w Many algorithms only interact with data through dot-products, so can be "kernelized". If data is separable in Φ -space by large margin, don't have to pay for dim.

Generic problem

New style advice:

- w Use a Kernel! $K(\begin{matrix} \text{img} \\ \text{img} \end{matrix}) = \Phi(\begin{matrix} \text{img} \\ \text{img} \end{matrix}) \cdot \Phi(\begin{matrix} \text{img} \\ \text{img} \end{matrix})$. Φ is implicit, high-dimensional mapping.
- w Many algorithms only interact with data through dot-products, so can be "kernelized". If data is separable in Φ -space by large margin, don't have to pay for dim.
- w E.g., $K(x,y) = (1+x_1y_1)(1+x_2y_2)\dots(1+x_ny_n)$.
 - $\Phi: (n\text{-diml space}) \rightarrow (2^n\text{-diml space})$.
- w Conceptual warning: You're not really "getting all the power of the high dimensional space without paying for it". As we saw from JL lemma, assumption of large margin means it's really an $\tilde{O}(1/\gamma^2)$ -dimensional problem after all.

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

Kernel fns have become very popular

...but there's something a little funny:

- w On the one hand, operationally a kernel is just a similarity function: $K(x,y) \in [-1,1]$, with some extra requirements. [here I'm scaling to $|\Phi(x)| = 1$]



- w And in practice, people think of a good kernel as a good measure of similarity between data points.
- w But Theory talks about margins in implicit high-dimensional Φ -space. $K(x,y) = \Phi(x) \cdot \Phi(y)$.

I want to use ML to classify protein structures and I'm trying to decide on a similarity fn to use. Any help?

It should be pos. semidefinite, and should result in your data having a large margin separator in implicit high-diml space you probably can't even calculate.

Umm... thanks, I guess.

It should be pos. semidefinite, and should result in your data having a large margin separator in implicit high-diml space you probably can't even calculate.

Kernel fns have become very popular

- w Theory talks about margins in implicit high-dimensional Φ -space. $K(x,y) = \Phi(x) \cdot \Phi(y)$.
 - Not great for intuition (do I expect this kernel or that one to work better for my kind of data)
 - Has a something-for-nothing feel to it. "All the power of the high-dim'l implicit space without having to pay for it". More prosaic explanation? (We saw evidence in JL lemma).

Goal: notion of "good similarity function" that...

1. Talks in terms of more intuitive properties (no implicit high-diml spaces, no requirement of positive-semidefiniteness, etc)
2. If K satisfies these properties for our given problem, then has implications to learning
3. Is broad: includes usual notion of "good kernel" (one that induces a large margin separator in Φ -space).

[Recent work with Nina, with extensions by Nati Srebro]

Defn satisfying (1) and (2):

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

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

- w Note: you can have this property without being a legal kernel.
- w Q: how could you use this to learn?

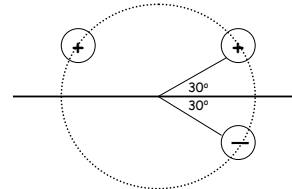
How to use it

At least a $1-\epsilon$ prob mass of x satisfy:

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

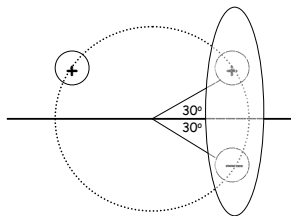
- w Draw S^+ of $O((1/\gamma^2) \ln 1/\delta^2)$ positive examples.
- w Draw S^- of $O((1/\gamma^2) \ln 1/\delta^2)$ negative examples.
- w Classify x based on which gives better score.
 - Hoeffding: for any given "good x", prob of error over draw of S^+, S^- at most δ^2 .
 - So, at most δ chance our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

But not broad enough



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

But not broad enough



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

Broader defn...

- w Say $K:(x,y) \rightarrow [-1,1]$ is an (ϵ,γ) -good similarity function for P if exists a weighting function $w(y) \in [0,1]$ s.t. at least $1-\epsilon$ frac. of x satisfy:

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

- w Can still use for learning:
 - Draw $S^+ = \{y_1, \dots, y_n\}$, $S^- = \{z_1, \dots, z_n\}$. $n = \tilde{O}(1/\gamma^2)$
 - Use to "triangulate" data: $F(x) = [K(x,y_1), \dots, K(x,y_n), K(x,z_1), \dots, K(x,z_n)]$.
 - Whp, exists good separator in this space: $w = [w(y_1), \dots, w(y_n), -w(z_1), \dots, -w(z_n)]$

Broader defn...

w Say $K:(x,y) \rightarrow [-1,1]$ is an (ϵ, γ) -good similarity function for P if exists a weighting function $w(y) \in [0,1]$ s.t. at least $1-\epsilon$ frac. of x satisfy:

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

- n So, take new set of examples, project to this space, and run your favorite linear separator learning algorithm.*
- n Whp, exists good separator in this space: w
- *Takes $\frac{1}{\epsilon} \ln \frac{1}{\gamma}$ samples and $\frac{1}{\epsilon} \ln \frac{1}{\gamma}$ iterations. First definition to penalize examples more that fail the inequality badly...

Main Definition, Implications

Algorithm

w Draw $S=\{y_1, \dots, y_d\}$, $S'=\{z_1, \dots, z_d\}$, $d=O((1/\gamma^2) \ln(1/\delta^2))$.

w Use to "triangulate" data: $F(x) = [K(x,y_1), \dots, K(x,y_d), K(x,z_1), \dots, K(x,z_d)]$.

Guarantee: with prob. $\geq 1-\delta$, exists linear separator of error $\leq \epsilon + \delta$ at margin $\gamma/4$.

Implications

K arbitrary sim. function

$$\bar{K}(x, x') = \sum_{i=1}^d K(x, y_i)K(x', y_i) + \sum_{i=1}^d K(x, z_i)K(x', z_i)$$

legal kernel

(ϵ, γ) -good sim. function

$(\epsilon + \delta, \gamma/4)$ -good kernel function



Interesting property of definition

- n An (ϵ, γ) -good kernel [at least $1-\epsilon$ fraction of x have margin $\geq \gamma$] is an (ϵ', γ') -good sim fn under this definition.
- n But our current proofs suffer a penalty: $\epsilon' = \epsilon + \epsilon_{\text{extra}}$, $\gamma' = \gamma^3 \epsilon_{\text{extra}}$.

Nati Srebro has improved to γ^2 , which is tight, + extended to hinge-loss.

- n So, at qualitative level, can have theory of similarity function that doesn't require implicit spaces.

Learning with Multiple Similarity Functions

• Let K_1, \dots, K_r be similarity functions s. t. some (unknown) convex combination of them is (ϵ, γ) -good.

Algorithm

w Draw $S=\{y_1, \dots, y_d\}$, $S'=\{z_1, \dots, z_d\}$, $d=O((1/\gamma^2) \ln(1/\delta^2))$.

w Use to "triangulate" data:

$$F(x) = [K_1(x,y_1), \dots, K_r(x,y_d), K_1(x,z_1), \dots, K_r(x,z_d)]$$

Guarantee: The induced distribution $F(P)$ in \mathbb{R}^{2dr} has a separator of error $\leq \epsilon + \delta$ at margin at least $\frac{\gamma}{4\sqrt{r}}$.

Sample complexity for Perceptron or SVM is roughly r/γ^2 .