CS 4/5780

ML Setup, Recap:

1. Data: $D = (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$

   x's : Input instances $x \in \mathbb{R}^d$

   y's : Corresponding output $y \in C$

Binary classification $C = \{0, 1\}$ or $\{-1, 1\}$, $C = \{1, 2, \dots, K\}$ multi class, $C = \mathbb{R}$ regression

2. $\mathcal{H}$: Set of models or hypotheses

   each model $h \in \mathcal{H}$, given an input instance x, outputs $h(x)$

3. ℓ: loss function measures performance of a model

Eg: 0-1 loss : $\ell(h(x), y) = \delta\{h(x) \neq y\}$

Squared loss : $\ell(h(x), y) = (h(x) - y)$

absolute loss : $\ell(h(x), y) = |h(x) - y|$

Goal of supervised learning:

Given data D find model h such that loss $\ell(h(x), y)$ on future instances (x, y) is small.

Asking for model that minimizes loss $\ell(h(x), y)$ for all possible future (x, y) is too much.

Why?

Future instances generated from some mechanism (often represented by some) distribution P written as

$$(x, y) \sim P$$

Formal Goal of supervised learning:

Given Data D find model h that minimizes

Risk or population loss : $\mathbb{E}_{(x,y)\sim P}\, \ell(h(x), y)$

generalization loss

Learning Algo. is the procedure that tries to attain above goal.

What is a good proxy for

$\mathbb{E}_{(x,y)\sim P}\left[\ell(h(x), y)\right]$? (and why?)

Test loss : For evaluation $\frac{1}{|D_{Test}|} \sum_{(x,y)\in D_{Test}} \ell(h(x), y)$

Split Data :

$$D = \left[\ \underbrace{D_{train}}_{\text{Training}}\ \middle|\ \underbrace{D_{val}?}_{\text{validation}}\ \middle|\ \underbrace{D_{Test}}_{\text{test}}\ \right]$$

(typical Algo : $\operatorname{argmin}_{h\in\mathcal{H}} \frac{1}{|D_{Train}|} \sum_{(x,y)\in D_{Train}} \ell(h(x), y)$)

No free Lunch theorem:

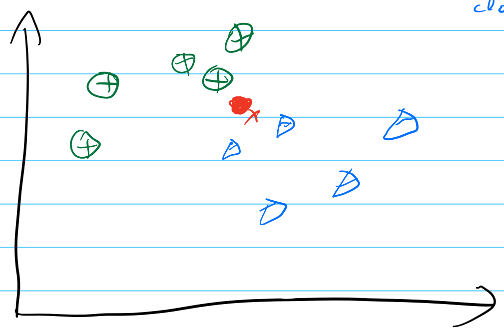
"Every ML Algorithm makes assumptions."

CS 4/5780

K-Nearest Neighbors Classifier

Assumption: Similar points are likely to share same label

Classification Rule: For a test point x, assign the most common label amongst the k most similar training instances
closest



Formally: Given $D = \{(x_1,y_1),(x_2,y_2),\ldots,(x_n,y_n)\}$ and test point x

Find $S_x \subseteq D$ s.t. $|S_x| = k$ and $\forall (x',y') \in D \setminus S_x$,

$$\text{dist}(x',x) \geq \max_{(x'',y'') \in S_x} \text{dist}(x'',x) \qquad h(x) = \text{MODE}(y' : (x',y') \in S_x)$$

Pro-tip: In case of a tie, reduce k by 1 and repeat

Common distances: Minkowski distance or ℓ_p distances

$$\text{dist}(x,x') = \left(\sum_{i=1}^{d} |x[i] - x'[i]|^p\right)^{1/p}$$

$p = 2$ is Euclidean distance

Quiz: What distances do $p=1$, $p \to \infty$ and $p \to 0$ correspond to.

Bayes Optimal Classifier

If we knew P how well could we do?

Bayes Error : $\min_{\text{All possible } h} \mathbb{E}_{(x,y)\sim D}\, \ell(h(x),y)$

If you knew $P(Y|X)$, Given point $x \in \mathbb{R}$, optimal classifier:

$$h_{opt}(x) = \underset{y \in C}{\text{argmax}} \; P(Y=y|X=x)$$

$$\text{Bayes Error}(x) = 1 - \max_{y \in C} P(Y=y|X=x)$$

This is the Best we can do!

Quiz: Coin has probability p of heads.

1. If tossed twice, what is the probability q, of two different outcomes?

2. Conclude that $q \leq 2(1-p)$

1-NN classifier: Simplest case $k=1$

Risk of 1-NN $\leq$ 2 Bayes ERROR

Formal proof is involved, see Cover & Hart '67

Intuition:

1. Say P was a discrete distribution on a finite set of points. Then, as $n \to \infty$ every test point has already occured in D_{Train}. (Say we pick any one of previous ocurrances as the nearest neighbor)

2. Risk of 1-NN Classifier is now given by the quiz Question. Why?

We are asking the question, what is the probability that, label y of a new test instance x matches that of a randomly chosen training point $x_i \in D_{Train}$ such that, $x_i = x$. Its label y_i is drawn independently from $P(Y|X=x)$

Hence, Risk of 1-NN $\leq 2\left(1 - \max_{y \in \{0,1\}} P(Y=y|X=x)\right)$

Claim: Given x, let $\hat{x} \in D_{TRAIN}$ be the 1-NN of x in D_T
as $n \to \infty$, $dist(x, \hat{x}) \to 0$, $\hat{x} \to x$

RAIN

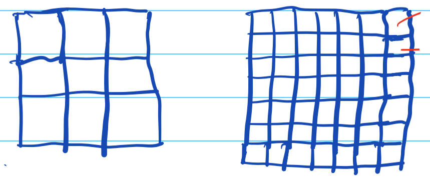
Risk of 1-NN $\leq$ 2 Bayes ERROR

k-NN For general $k > 1$

a. Larger k, as $n \to \infty$,
Intuition: For point x, we predict as majority of k draws from $P(Y|X=x)$

b. But, if k grows too fast, the more we rely on farther points to predict label for x

The Curse of Dimensionality:



1. Each sub-cube has same area

2. if we randomly pick a few we will pick more cubes near surface!

1 Assume points are drawn uniformly at random from a unit hypercube $[0,1]^d$

2 Hypercube of volume k/n within unit hypercube is expected to contain k out of n points

3 Length ℓ of such cube is given by $\ell^d = k/n$

What does this mean?

$k = 10$
$n = 1000$ →

d	2	10	100	1000
ℓ	0.1	0.63	0.955	0.9954

$\ell = \left(\frac{k}{n}\right)^{1/d}$