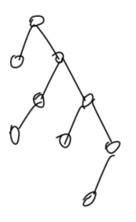
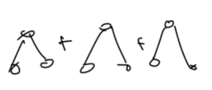
Week 3



$\uparrow$ var $\longrightarrow$ $\downarrow$ var

Bagging

RF



$\uparrow$ bias $\longrightarrow$ $\downarrow$ bias

Boosting

GBRTs

Adv

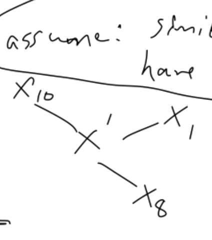
1. Interpretable
2. Few assumptions

"SimpleShot"

KNN (Cover & Hart, 1967)

$\boxed{\text{KNN algorithm}}$

assume: similar inputs (x) have similar outputs (y)



Input: $\{(x_i, y_i)\}_{i=1}^{n}$, x'

$dist(\cdot, \cdot)$

k (hyperparam)

$S_{x'} = \{\}$ // indices of NNs

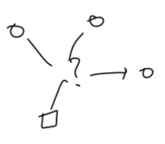
for $i = 1 : k$ do

$j^* = \operatorname{argmin}_{j \in \{1, \dots, n\} \setminus S_{x'}} dist(x_j, x')$

$S_{x'} \leftarrow S_{x'} \cup \{j^*\}$

end

Return mode($\{y_{j^*} : j^* \in S_{x'}\}$)



Quiz: How does k affect KNN?

$k = n$

$k = 1$

Distance Functions

Minkowski Distance

$$dist(x, z) = \left(\sum_{r=1}^{d} |x_r - z_r|^p \right)^{1/p}$$

Quiz: Can you name special cases of $p = 1, 2, \infty$

$\boxed{\text{1-NN convergence}}$

Bayes Optimal Classifier

$p(y|x)$

x' $\quad y^* = h_{opt}(x') = \operatorname{argmax}_y p(y|x')$

$\epsilon_{BayesOpt} = 1 - P(h_{opt}(x')|x') = 1 - P(y^*|x')$

lower bound

upper bound: ~~constant~~ classifier

Theorem (Cover & Hart, 1967)

As $n \to \infty$, the 1-NN error is no more than $2 \times \epsilon_{BayesOpt}$

Proof.

x_t : test point

x_{NN} : NN of x_t

$n \to \infty$

$dist(x_{NN}, x_t) \to 0$

$x_{NN} \to x_t$

Cases (where label of x_{NN} is not label of x_t)

1. label of x_t is y^*
   label of x_{NN} is $\neg y^*$
2. label of x_t is $\neg y^*$
   label of x_{NN} is y^*

$$\epsilon_{NN} = \underbrace{p(y^*|x_t)}_{\leq 1}(1 - p(y^*|x_{NN})) + \underbrace{p(y^*|x_{NN})}_{\leq 1}(1 - p(y^*|x_t))$$

$$\leq (1 - p(y^*|x_{NN})) + (1 - p(y^*|x_t))$$

$$= 2(1 - p(y^*|x_t))$$

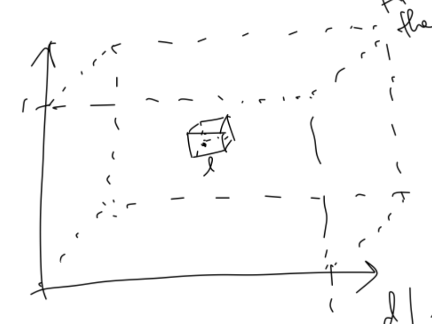
$p(y^*|x_t) = p(y^*|x_{NN})$

$= 2\epsilon_{BayesOPT}$ $\square$

$\boxed{\text{Curse of Dimensionality}}$

$x_i \in [0,1]^d \quad \forall i$

How much space will the KNNs take up inside the unit cube?



$k = 10$

$l^d \approx \frac{k}{n}$

$l \approx \left(\frac{k}{n}\right)^{1/d}$

d	l
2	0.1
10	0.63
100	0.955
1000	0.9954

How many datapoints would I need to make l small?

$l = 0.1$ $\quad n = \frac{k}{l^d} = k 10^d$

$d > 100$

$n >$ # of electrons in universe!!!! !!!!

Data with low dimensional Structure

e.g. natural images (digits, faces, etc...)

$d = 18M$

< 50

(eye color, hair-color, hair type, face shape)

Summary

- k-NN: interpretable, easy-to-implement flexible
- $n \to \infty$
  KNN slows
- $d \gg 1$
  KNN breaks down