Machine learning - mid sem

UNIT - I

Machine Learning

Date : 17-Aug-22

Artificial Intelligence

- ✓ Narrow Intelligence – specific task / Weak AI
- ✗ General " – Strong AI (Human Intelligence)
- ✗ Super " – Hypothetical (No Human intervension)

ML

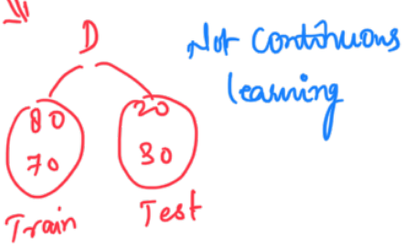
- Supervised
- Unsupervised
- Reinforcement

I/P (Data) → Algo → O/P (Data)

How to map I/p with O/p

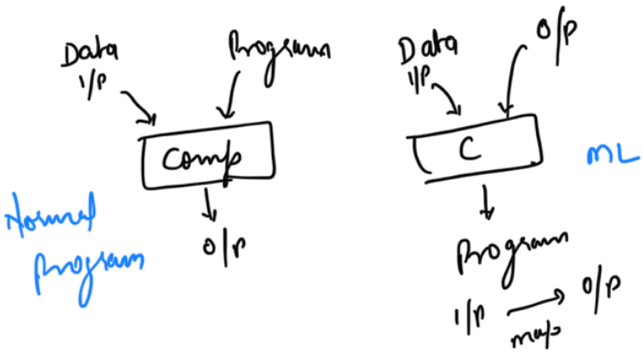
Data – I/P, O/p — cluster, similarity, dissimilarity

Take decision based on rewards.



Not continuous learning

Date : 18-Aug-22



① Performance ↑

② Task (Experience) ↑ wrt. Time

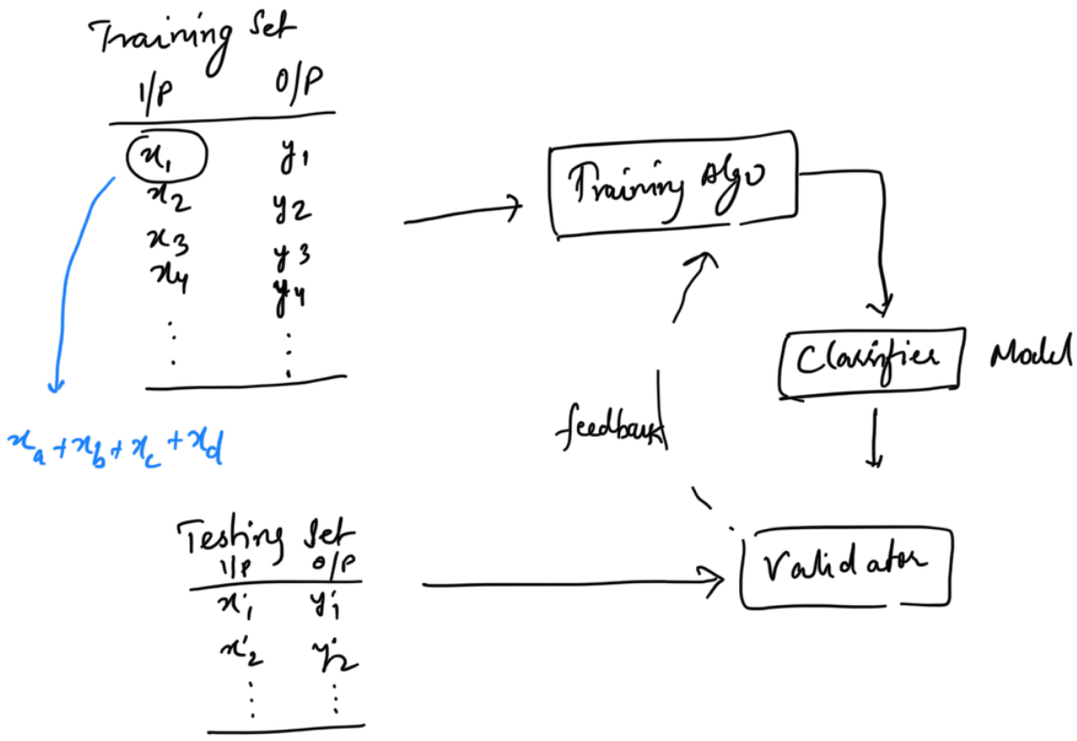
Inductive learning

Eg ① Human
② Machine
③ Ant / Bee / Fungus

Supervised Learning

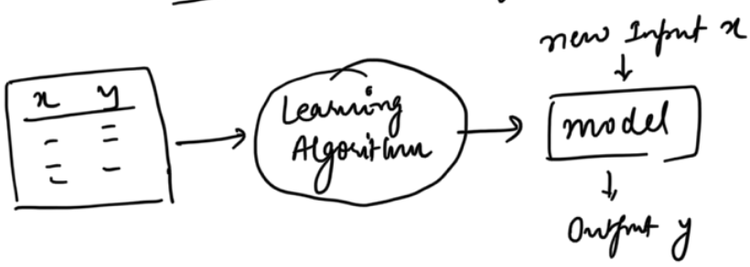
- Here question arises about : Complexity V/s Accuracy V/s Noise

- Assumptions we take in ML
  (i.e biases)
  - How much complexity we require to our model
  - which order we decide the complexities
    i.e x^0 to x^{10} so we can stop till x^7 or x^8.
    (less complex) (More complex)

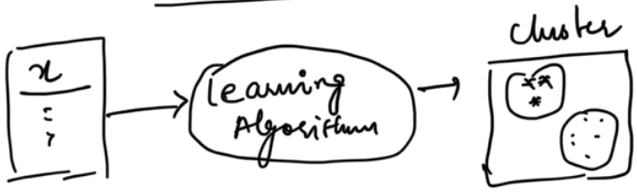


Date: 24-Aug-22

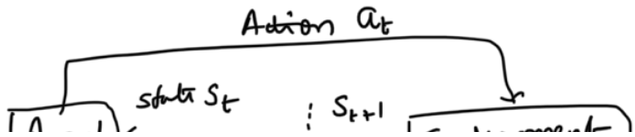
Supervised Learning

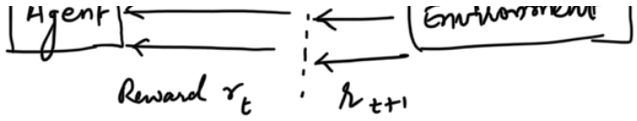


Unsupervised



Reinforment





S.L. # Mathematical Explaination

$\mathcal{X}$

Hypothesis space
- All possible solutions to particular problem.

Biase
- To reduce the hypothesis space.

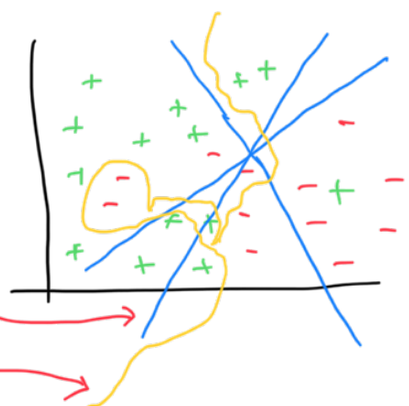
Next class : Biases & Generalization

Date : 25-Aug-22

- Hypothesis h : fn. that approximates f.
- Hypo. space H : set of fn. we allow for approximat f.
- The set of hypothesis that can be produced, can be restricted further by specifying a Language bias.
- Input : Traning set $S \subseteq \mathcal{X}$
- Output : A hypothesis $h \in H$

Baias 1 : Restrict to line
i.e we only draw $y = mx + c$

not - $2 + x^2 + x + 1$



Here, Learning is Refining the Hypothesis Space.

Def Biase : The tendency to prefer one hypothesis over another.

Ex : Occam's Razor
- A classical example of inductive bias.
- The simplest consistant hypothesis about the target function is actually the best.

Next Class : Problems in ML and Generalization

Date - 26-Aug-22

Important Issue in ML

- Generalization

Generalization

components of gen^n error < Bias
Variance : How much models estimated from different training sets differ from each other.

Underfitting and Overfitting

Underfitting < High bias and low variance
High training error and high test error

Overfitting < Low bias and high variance
Low training error and high test error

NOTE : It is good if we are facing underfitting but overfitting is very bad always.

Experimental Evaluation of Learning Algorithms

→ Performance Evaluations:
- Error : AE, MSE etc.
- Accuracy
- Precision / Recall

Absolute Error : $\frac{1}{n} \Sigma |f(x) - \hat{y}|$

MSE : $\frac{1}{n} \Sigma (f(x) - \hat{y})^2$

} Regression

$\frac{1}{n} \Sigma (f(x), y) < \begin{matrix} 0 \\ 1 \end{matrix}$ } Classification

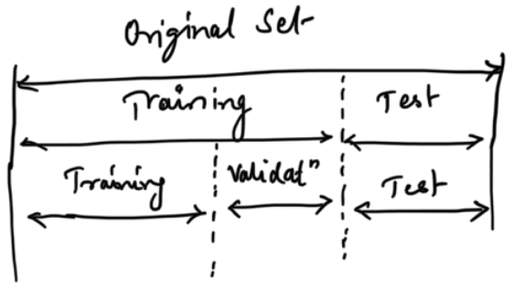
→ Sampling Methods :
- Train / Test sets
- K- Fold cross - validation (when dataset is very less)

Difficulties in evaluating hypotheses with limited data

① Bias in the estimate: The sample error is poor estimator of true error.

② Variance in the estimate: The smaller the test set the greater the expected variance.

Validation Set



Validat" fails to use all the available data.

Solution

K-Fold cross-validation

1) Split the data into K equal subsets
2) Perform K rounds of learning; on each round
   - 1/K of the data is held out as a test set and
   - the remaining examples are used as training data.
3) Compute the average test set score of the K rounds.

So,

Trade-Off

→ In ML, there is always a trade-off b/w
   - complex hypotheses that fit the training data well
   - simpler hypothesis that may generalise better.

→ As the amount of training data increases, the generalization error decreases.

Date: 27-Aug-22

Regression

Single variet LR $y = \beta_0 + \beta_1 x_1 + \epsilon$

(β_0: Population y-intercept, β_1: population slope, ϵ: Random error)

Variance
- Deterministic
- stochastic/probabistic

Assumptions

- The data may not form a perfect line
- $E(\varepsilon_i) = 0$ for $i = 1, 2, 3, \ldots, n$. (Total error)
- $\sigma(\varepsilon_i) = \sigma_\varepsilon$
- The errors are independent
- The ε_i are normally distributed (with mean 0 and standard deviation σ_ε)

Multiple variable LR

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p + \varepsilon$$

How do we "learn" parameters

- To find the values for the coeff. which minimize the objective function we take the partial derivatives of the objective function (SSE) w.r.t the coefficients. Set them to zero and solve

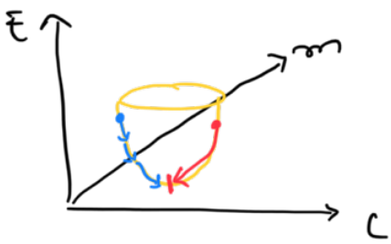
$$\beta_1 = \frac{n\Sigma xy - \Sigma x \Sigma y}{n\Sigma x^2 - (\Sigma x)^2} \qquad \beta_0 = \frac{\Sigma y - \beta_1 \Sigma x}{n}$$

for $y = \beta_0 + \beta_1 x$

Minimize $\sum_{i=1}^{n} [y_i - (b_0 + b_i x_i)]^2$

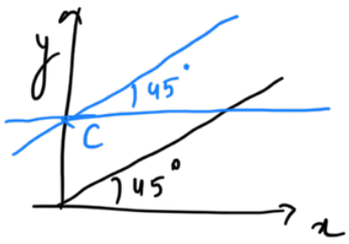
Gradient Descent Algorithm (LMS Algorithm)

least mean slope



Date: 31-Aug

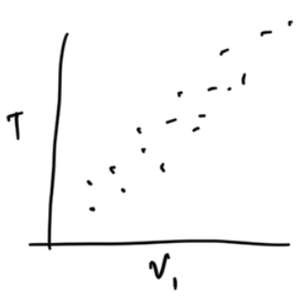
→



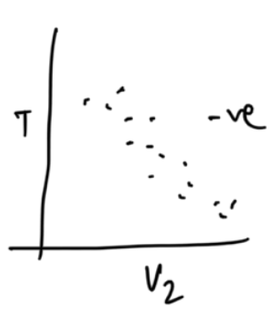
$$y = x$$
$$= 1.x$$
$$= \tan 45^\circ . x = mx$$

$$y = mx + C$$

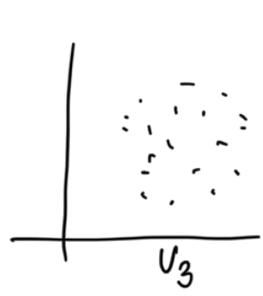
Coefficient of variance (correlation)



+ve



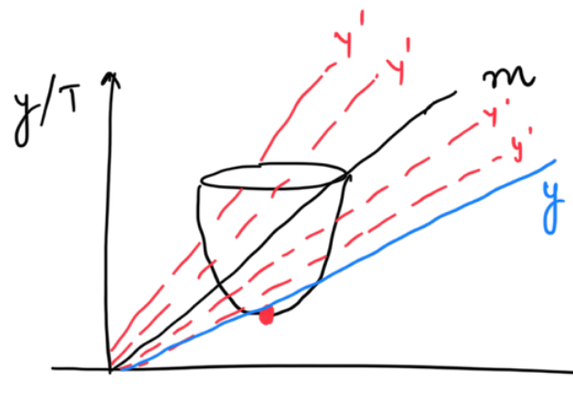
-ve



Zero

Not desire

Desired. Desired



if m = 0
m = +ve or -ve

Coeff. of relation - Pearson's Coefficient $\rho(x, y)$

$$= \text{Cov}(x, y) / (\text{stnd Dev}(x) \times \text{stnd Dev}(y))$$

$$R = r_{xy} = \frac{\Sigma (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\Sigma (x_i - \bar{x})^2}\sqrt{\Sigma (y_i - \bar{y})^2}} \quad \simeq +1 \text{ or } -1$$

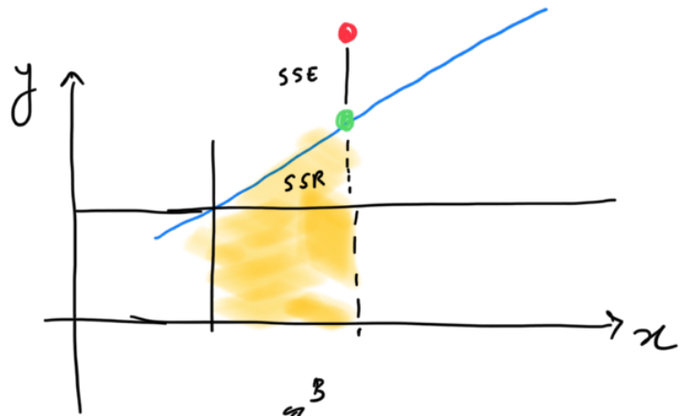
NOTE It is significant or we use only for finding coeff. of relation b/w dependant and independent variable.

Coeff. of Determinant-

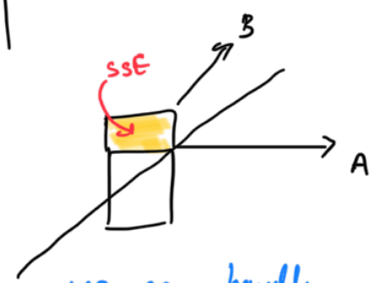
Variance

1) Stochastic variance
   - which we don't know
2) Deterministic variance
   - which we know

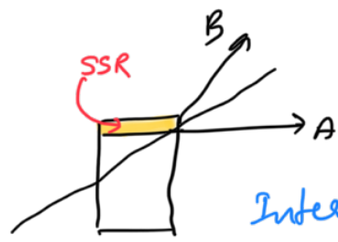
NOTE In LR we try to minimize the stochastic variance.



y 6.1 } Residual error (SSE)
ŷ 5.9 }
5.5



we can handle this easily but not interested.



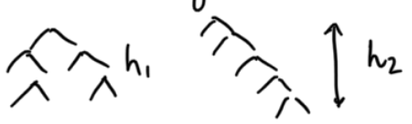
Interested to reduce this residual error.

Date: 1-Sep-22

Decision Tree

Issues

- what kind of data the D.T. will be generated.



- One proposal: prefer the smallest tree that is consistent with data (Bias)
  i.e either less no. of leafs or low height.
- soln: state space search

Choices

- When to stop
  - no more input features
  - All examples are classified the same
  - too few examples to make an informative split
- which test to split on
  - split gives smallest error
  - with multi-valued features
    - split on all values or
    - split value into half

Date: 2-Sep-22

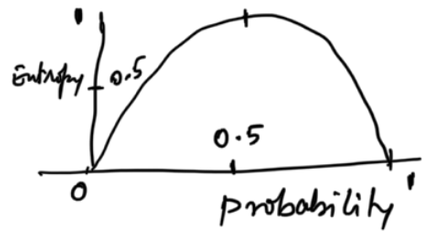
Principled Criterion

(1.) Information gain

- Entropy = Impurity or randomness

Y/N Case

100% Y → E = 0
100% N → E = 0
50% Y & 50% N → E = 1



Gain(S,A) = expected reduction in entropy due to partitioning S on attribute A

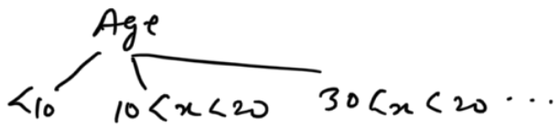
$$Gain(S,A) = Entropy(S) - \sum_{v \in values(v)} |S_v|/|S| \, Entropy(S_v)$$

(2.) GINI index
- measure of node impurity

(3.) chi square

(4.) Reduction in Variance

Continuous Attribute :
- what could be the class / cluster size ?



NOTE Problem in D.T. is that it works on greedy approach.

Random forest -
- Builds multiple decision tree and merges them together.
- More accurate and stable prediction
* Random decision forest correct for decision tree' habit of overfitting to their training set.
- Trained with the "bagging" method.

7-Sep-22

Overfitting

Reasons
1) Dataset set is small
2) Due to noise → Remove the outlier
3) Data has high variance → cross validation
4) The model is too complex

Reason for Underfitting
(4) The model is too simple
(1) to (3) is same as above

NOTES on Overfitting

It happens when model is capture from a limitted setup.

9-Sep-22

K-NN

1 nearest neighbour

Training → Saving the instances

Testing / Predicting → instance x_z given

find (x_1, y_1) nearest to x_z

predicting → y of x_z

KNN

find $\{(x_1, y_1), (x_2, y_2) \cdots (x_k, y_k)\}$

classification → $\{y_1, y_2, y_2 \cdots y_k\}$ ← T / F] y_z

Regression → Mean ↑ or Median

Next :- locally weighted Average ...
class

Error

10-Sep-22

Feature Selection

- Univarient F.S. methods
- Multivarient- "

16-Sep-22

Optimization in KNN

① Increase the accuracy

② Reduce the complexity

Feature Extraction

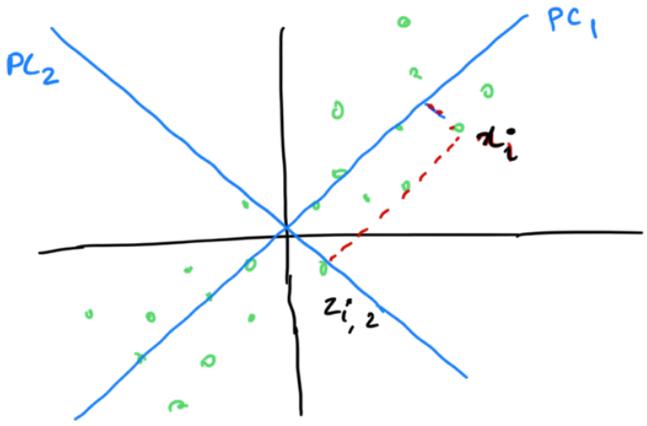
→ find a projection matrix w from N-dim to M-dimentional vectors that keeps

error loss.

$M < N$

→ Expected:
- Uncorrelated, can't be reduced further [corr. $(x_1, x_2 \cdots x_n)$]
- Have large variance or otherwise bear no information [$\sigma^2 (x_1, x_2 \cdots x_n)$]

Principle Components



if 2 variables (or Principle component) are orthogonal, then there is no correlation b/w them.

Problem in PCA projection

- Data variation determines the projection
- class information will be reduced ?

SOLUTION

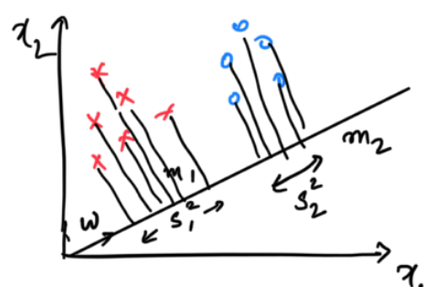
LDA

- Maximize between-class distance.
- Minimize within-class distance

↓

Fisher linear discreminant-

$$J(w) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2}$$



Imp

Recomendation System

17-Sep-22

Types
1) – Popularity RS 20%
2) – Content based RS 30%
3) – Collaborative RS 50% < User based RS / Item based SR
4) – Hybrid (100% used)

} All are based on KNN

#3) collaborative filtering for Rating Prediction

- User based :- Nearest similar User (Neibour)
- Item based :- Nearest Neibour's item (similar)

Ex

	Movies				
	M_1	M_2	M_3	M_4 ---	
U_1	5	4	—	3	1
U_2	4	3	1	2	2
U_3	3	4	2	1	3
U_4	3	4	2	3	2
⋮					

Item based corr. (M_1, M_2, M_3 ---)

User based RS
Corr (U_1, U_2 ...)

Let U_1 & U_4 are correlative ∴ M_3 for U_1 = 2 or 3

Pearson's correlation coefficient

$$sim(u,v) = \frac{\sum_{i \in C} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in C} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in C} (r_{v,i} - \bar{r}_v)^2}}$$

Target User ↑ Neighbour

- Method/process name : Neighborhood formation phase

Now Recommendation Phase

- Use the following formula to compute the rating prediction of item i for target user u

$$p(u,i) = \bar{r}_u + \frac{\sum_{v \in V} sim(u,v) \times (r_{v,i} - \bar{r}_v)}{\sum_{v \in V} |sim(u,v)|}$$

($\bar{r}_u$ ↑ Average)

where V is the set of K similar users, $r_{v,i}$ is the rating of user v given to item i,

Issue
- Lack of scalability.

Enhancement
- Item based CF

Bayesian Learning

21-Sep-22

Probability in learning
- Probability
- Bayesian Estimation
- → Bayesian Probability

Bayes Theorem
- Goal
- Bayes Rule: $P(h|D) = \frac{P(D|h)\,P(h)}{P(D)}$

$P(h)$ = Prior probability of hypothesis, h
$P(D)$ = " " of training data, D
$P(h|D)$ = Prob. of h given D (Posterior density)
$P(D|h)$ = Prob. of D given h (likelihood of D given h)

Example
→ Dilemma

M/F ?

→ Given
Out of 100 women at the movie: 50 – long hair, 50 – short hair
— " — men — " — : 96 – short hair, 4 – long hair

Prior knowledge
→ Out of 2 women in line = 1 – short hair, 1 – long hair
Out of 98 men " " = 96 – S.H, 2 – L.H

To find: $P(\text{man}|\text{long hair}) = ?$

Concepts: Conditional Probability, Marginal ", Joint "

Soln $P(\text{man with LH}) = P(LH) * P(\text{man}|LH)$

$P(LH \text{ and man}) = P(m) * P(LH|M)$

$\because \quad P(m \,\&\, LH) = P(LH \,\&\, m)$

$\therefore$

$= 0.5 *$

$P(m \mid n) =$ ...

Maximum A Posteriori (MAP) Hypothesis

$$P(h|D) = \frac{P(D|h)\,P(h)}{P(D)}$$

The goal of Bayesian learning: the most probable hypothesis given the training data (max. A posteriori hypothesis)

$$h_{MAP} = \arg\max_{h \in H} P(h|D)$$

$$= \arg\max_{h \in H} \frac{P(D|h)\,P(h)}{P(D)}$$

$$= \arg\max_{h \in H} P(D|h)\,P(h)$$

#1 Compute ML hypo (ML - Maximum likelihood)

$$h_{ML} = \arg\min_{h \in H} \sum_{i=1}^{m} (d_i - h(x_i))^2$$

Bayes Optimal

22-Sep-22

NOTE: finding $P(y|x)$ is hard, while it is easy to find $P(x|y)$.

Ex

$P(y|x)$

$P(1 \mid x=(0,2))$

$P(y=0) = \frac{6}{10}$

$P(y=1) = \frac{4}{10}$

$P(x=(0,2) \mid y=1) = \frac{1}{4}$

x_1	x_2	y
0	0	0
0	1	1
1	2	1
0	0	1
2	2	0
1	1	0
0	2	1
2	0	0

$P(x=(0,2) \mid y=0) = 0$

2 1 0
1 0 0

$P(x=(0,2) \mid y=0) * P(y=0) = 0 * \frac{6}{10} = 0$

$P(x=(0,2) \mid y=1) * P(y=1) = \frac{1}{4} * \frac{4}{10} = \frac{1}{10}$

Estimated value of y is 1 given $x=(0,2)$

likelihood, prior

$$P(y \mid x) = \frac{P(x \mid y)\, P(y)}{P(x)}$$

$P(x)$ ← Evidence

Problems

- Prob. of particular $x=\{\}$ set in very large dataset is very complex.

Solution → Naive Bayes

- Assume all x are independent.

$$\therefore P(x=(0,2) \mid y=1) = P(x_1=0 \mid y=1) * P(x_2=2 \mid y=1)$$

$$= \frac{3}{4} * \frac{2}{4} = \frac{3}{8}$$

$$\& \quad P(x=(0,2) \mid y=0) = P(x_1=0 \mid y=0) * P(x_2=2 \mid y=0)$$

$$= \frac{1}{6} * \frac{1}{6} = \frac{1}{36}$$

$\Rightarrow$ 4 . 3 . 2 > 1 1 * 6

Don't forget to multiply the priors probability. i.e $P(y=1)$ and $P(y=0)$.

Problem in Naive Bays

Smoothing is the solution to problem when y^{new} is zero.

* Text recognition is the practical example of Naive Bayes classifier or regression.

Note Even if the variable are not independent Naive Bayes classifier works well.