$\alpha_1 + \alpha_3 = \frac{1}{8}$ $\boxed{\alpha_2 = \frac{1}{8}}$ → same hyperplane as α_1

α_2 & α_1 are related to −ve hyperplane

∴ we can make $\boxed{\alpha_1 = 0}$

∴ $\boxed{\alpha_3 = \frac{1}{8}}$ (+ve hyperplane)

$$\bar{w} = \sum_{i=1}^{3} \alpha_i y_i \bar{x}_i = 0 + \frac{1}{8}(-1)$$

$$= \alpha_1 y_1 \bar{x}_1 + \alpha_2 y_2 \bar{x}_2 + \alpha_3 y_3 \bar{x}_3$$

$$= 0 + \frac{1}{8}(-1)[-1 \quad 2] + \frac{1}{8}(1)[-1 \quad 2]$$

$$= \frac{1}{8}[1 \quad -2] + \frac{1}{8}[-1 \quad -2]$$

$$= \frac{1}{8}[0 \quad -4] = [0 \quad -\tfrac{1}{2}]$$

w_1, w_2

$$\Rightarrow \boxed{\bar{w} = [0 \quad -\tfrac{1}{2}]}$$

$$w_0 = \frac{1}{S}\sum_{i=1}^{S}(y_i - \bar{w}\bar{x}_i)$$

→ x_2 & x_3 are closest

analysis is for support vector 1 & 2

$$= \frac{1}{2}\left(y_2 - \bar{w}\bar{x}_2 + y_3 - \bar{w}\bar{x}_3\right)$$

$$= \frac{1}{2}\left(-1 - [0 \quad -\tfrac{1}{2}]\cdot[-1 \quad 2] + 1 - [0 \quad -\tfrac{1}{2}][1 \quad -2]\right)$$

$$= \frac{1}{2}(-1 + 1 + 1 - 1) = 0 \Rightarrow \boxed{w_0 = 0}$$

$$w_1 x_1 + w_2 x_2 + w_0 = 0$$

$$\bar{w} = [\underset{w_1}{0} \quad \underset{w_2}{-\tfrac{1}{2}}]$$

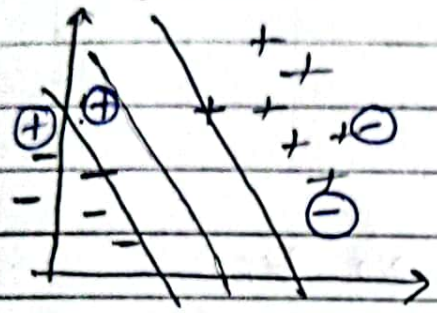
$$\Rightarrow 0x_1 + -\frac{1}{2}x_2 + 0 = 0$$

$$\Rightarrow \boxed{x_2 = 0}$$ final eqn

More margin SVM → rigid to outliers

Flexibility is required → soft margin SVM

Soft Margin SVM



All +ve should be on one side & all -ve sides on other

not rigid on this card -

① Hard margin SVM is very rigid for 2 constraints of SVM:-
(i) Decision boundary should be as far as from the nearest data point.
(ii) Avoid misclassification.

② But the real world problems are complex and often datasets are affected by noise & outliers.
[data will not be completely linearly separable]

③ But dataset is almost linearly separable, (Alternative in margin, Balance b/w margin & misclassification.)

$$\underbrace{\min_{\bar{w}, w_0} \frac{1}{2} ||w||^2}_{\text{distance error}} + \underbrace{C \sum_{i=1}^{m} \xi_i}_{\text{Classification error}}$$

Hyperparameter

$\xi_i \Rightarrow$ Slack variable

subject to constraint $\xi_i \geq 0 \ \forall x_i$

correctly classified $\xi_i = 0$

misclassification $\xi_i > 0$

$y_i(\bar{w} \cdot \bar{x}_i + w_0) \geq 1 - \xi_i$

hyperparameter → you control it

C is more ⇒ more importance to misclassification error

C is less ⇒ distance error importance ⇒ margin error importance

13.03.2024
Wednesday.

Lagrangian dual of this is similar to Hard margin SVM
only change $\alpha >= 0$ & $\alpha < C$

Soft margin SVM cost function:-

$$\min_{\bar{w}, w_0} \frac{1}{2}||\bar{w}||^2 + C\sum_{i=1}^{m} \xi_i$$

$$y_i(\bar{w}\bar{x}_i + w_0) \geq 1 - \xi_i \quad , \quad \xi_i \geq 0 \ \forall i$$

Lagrangian Optimization will be

$$L(\bar{w}, w_0, \xi, \alpha, \beta) = \frac{1}{2}||\bar{w}||^2 + C\sum_{i=1}^{m} \xi_i - \sum_{i=1}^{m} \alpha_i [y_i(\bar{w}.\bar{x}_i + w_0) - 1 + \xi_i] - \sum_{i=1}^{m} \beta_i \xi_i$$

α & β are independent Lagrangian coefficient.

$$\frac{dL}{dw} = 0 \Rightarrow \bar{w} = \sum_{i=1}^{m} \alpha_i y_i x_i$$

$$\frac{dL}{dw_0} = 0 \Rightarrow \sum_{i=1}^{m} \alpha_i y_i = 0$$

$$\frac{dL}{d\xi} = 0 \Rightarrow C - \alpha_i - \beta_i = 0 \Rightarrow \alpha_i + \beta_i = C$$

This can be substituted back to get the dual formulation.

$$\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j x_i x_j \qquad \underbrace{0 \leq \alpha_i \leq C}$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0$$

Only change from Hard margin

$\alpha_i + \beta_i = C$ (new)

Find (Same process as of Hard margin SVM)

$$\frac{dL}{d\alpha_1} = 0, \ \frac{dL}{d\alpha_2} = 0, \ \ldots, \ \frac{dL}{d\alpha_m} = 0$$

m variables : $\{\alpha_1, \alpha_2, \ldots, \alpha_m\}$

m eqns

Find $\{\alpha_1, \alpha_2, \ldots, \alpha_m\} \longrightarrow$ Find $\bar{w} = \sum_{i=1}^{m} \alpha_i y_i x_i$

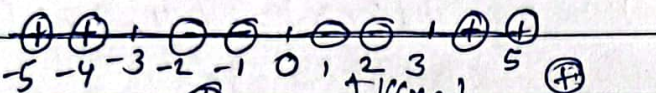
& $w_0 = \frac{1}{S}\sum_{i=1}^{S}(y_i - \bar{w}x_i)$

only check for cond^n $0 <= \alpha_i <= C$
& take parameter C as input.

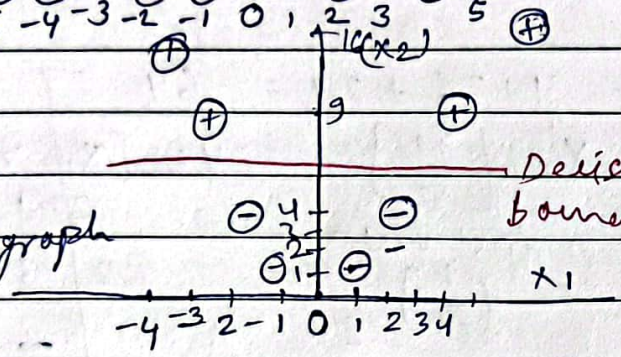
NON LINEAR SVM :-

x_1	y	$x_2 = x_1^2$
1	-1	1
2	-1	4
-1	-1	1
-2	-1	4
4	1	16
5	1	25
-4	1	16
-5	1	25

1D graph

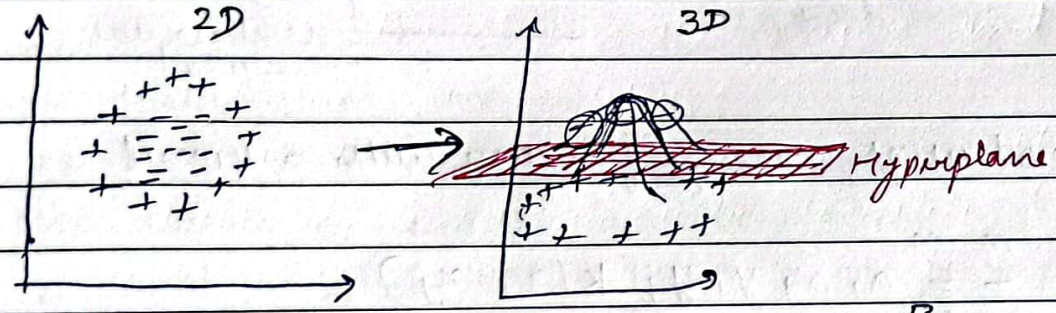


Hyperplane for 1D = point
But a single point can't do separate data.



2D graph

~~Now~~ Linearly separable in 2D



Mapping functions are used to map the data from lower dimension to higher dimension
[Because there is chance of linear separation]

2 D to 3D Mapping function

$\phi(x,y) = (x^2, y^2, \sqrt{2}xy)$

Data point (2D) (2,3) $\longrightarrow$ 3D $(4, 9, 6\sqrt{2})$

This would change lagrangian dual function

$$L(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \phi(x_i) \phi(x_j)$$

↳ 3D vector

Because this transformation process is costly dummies,
⇒ computation time is high in transforming data into higher dimensions

Solution: Use the Kernel.

KERNEL :- K is representating kernel function. $K(x,y)$

$$\boxed{K(x,y) = \phi(x) \cdot \phi(y)}$$

02.04.2024
Tuesday

DATE
PAGE

$x_1 \; x_2 \; x_1 \; x_2$

Consider 2 data points (1,2) & (2,3). Apply polynomial kernel $K(x,y) = (x^T y)^2$ & show that it is equivalent to mapping function $\phi(x_1^2, x_2^2, \sqrt{2}\, x_1 x_2)$

Sol: $\phi(1,2) = (1, 4, 2\sqrt{2})$ $\qquad$ $\phi(2,3) = (4, 9, 6\sqrt{2})$

$x_1 \; x_2$

$\phi(1,2) \cdot \phi(2,3) = 4 + 36 + 24 = 64$

$K(x,y) = (x^T y)^2$ $\qquad$ $x^T = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ $\qquad$ $y = [x_1 \;\; x_2]$

2nd matrix

first matrix

$= (2+6)^2$ $\qquad$ $= \begin{bmatrix} 1 \\ 2 \end{bmatrix}_{1\times 2}$ $\qquad$ $[2 \;\; 3]_{2\times 1}$

$= |8|^2 \longrightarrow$ determinant

$= 64$

The two values are equal

Kernel Trick

Replacing the dot product of mapping function with a kernel function.

$$\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

Linear Kernel Function $\qquad$ Polynomial Kernel Function $\qquad$ Gaussian Kernel Function (RBF) Famous

Radial Basis Function

Ensemble Learning

↳ group decision making

Ensemble is a process of combining more than one weak learner (models) to produce a strong learner that performs better than ~~the~~ a single weak learner

Let there are 'n' machine learning models

Prob. of error by any model be p. $\qquad$ (Accuracy $= 1-p$)

Prob. of that 'k' learners will make error

$${}^{n}C_{k}\, p^{k} (1-p)^{n-k}$$

voting type answer

DATE
PAGE

Q) suppose we have 5 people models and prob. of error by any model is 0.2 or accuracy 0.8, prob. that group decision is wrong or k learners make error where $k >= 3$.

Sol: $P(k>=3) = P(k=3) + P(k=4) + P(k=5)$ [majority wrong]

$= {}^5C_3(0.2)^3(0.8)^2 + {}^5C_4(0.2)^4(0.8)^1 + {}^5C_5(0.2)^5(0.8)^0$

$= \left[\frac{5\times4^2}{2} \times 0.8^2 + 5\times0.2\times0.8 + 1\times(0.2)^2\right](0.2)^3$

$(0.2)^3(0.64 + 0.8 + 0.04) = (0.2)^3(1.48)$

$= 0.057 = 0.01184$

↳ Better than single model.

There are different ways of diversity — ↳ ↑ing accuracy of model.

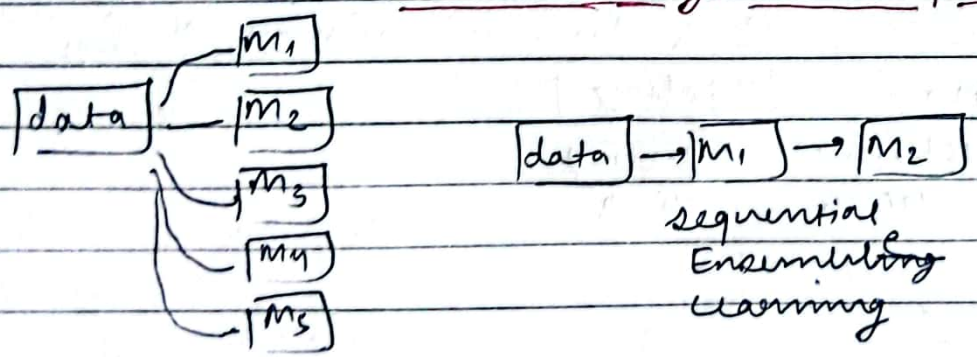
(1) Use different ML models.

(2) Use different population (datasets).

(3) Use different hyperparameters.

(4) Use different features.

Ensembling Technique



Classification on the basis of type of learning

- Homogenous → Same type of models
- Hetrogenous → Different type of models

4 Ensemble Techniques:-

① Voting
② Bagging
} Parallel Ensemble Learning

③ Stacking
④ Boosting
}

03.04.2024 Wednesday

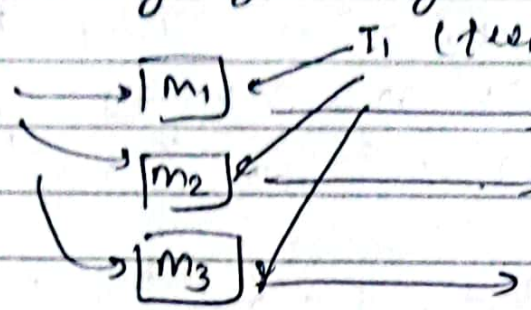
Creates diff. training dataset
↳ Deleting key + nearest - ve classes & replicate other entries (randomly)
Bootstrapping → computing performance
K fold cross data validation ↳

PAGE

1) Voting :-

(Parallel + heterogeneous) D

↳ Based on majority voting.



⊙ majority = 1 (classification model)

⊙ Regression model
↳ take average as an output
$\frac{1+0+1}{3}$

Same dataset, generally different models.

$\forall M_i$ where $i \in N \Rightarrow$ Accuracy$(M_i) > 50\%$
Otherwise, Accuracy (Ensemble model) will be low.

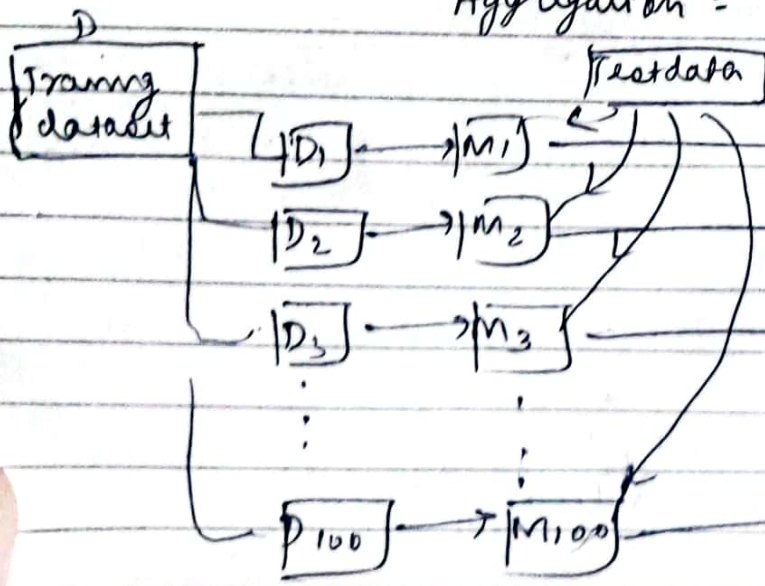
2) Bagging :-

↳ Bootstrap ~~Recordsbing~~ Resampling
It is process of creating many samples S_i $(1 \le i \le m)$ using a method called Random sampling with replacement.
(Duplicate data can be there in a dataset)
↳ some data may not be used in any.

[Training]

→ Bootstrap Aggregation = Bagging [Testing]

Bootstraping = for training
Aggregation = testing



$M_1 == M_2 == \dots == M_{100}$
Models same (Homogeneous)
Parallel
Dataset different.

(i) for classification do majority voting

(ii) for regression, final o/p = avg. o/p all given by M_i

$\frac{X_1 + X_2 + \dots X_{100}}{100}$

$\frac{\sum_{i=1}^{100} (o/p)\, M_i}{\sum 1}$

special type of bagging } → Random forest.

DATE
PAGE

Types of Bagging –

(1) Pasting :- Row sampling without replacement.

(2) Random sampling :- Column sampling only

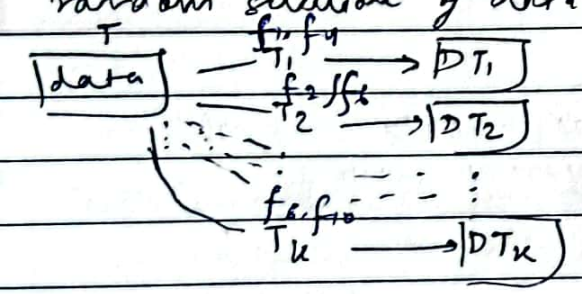
f1	f2	...	f100

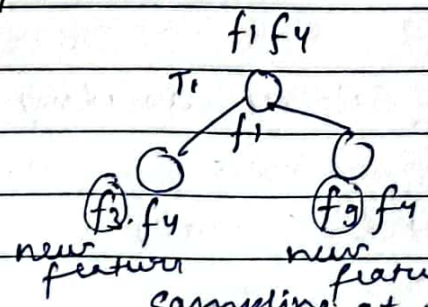
$\underbrace{f_1 \cdots f_{100}}_{\text{for } M_1 \text{ training}}$ $\underbrace{f_{21} \text{ to } f_{20}}_{\text{for } M_2 \text{ training}}$ ... } same voting

(3) Random patches :- Both row & column sampling.

Random Forest :- It is a variant of bagging which uses decision tree for learning. (more randomness than Bagging)

* Forest as it is always a group of trees and random word come from random selection of datasubsets for each tree.





sampling takes place at tree level

only in Random forest

$f_1, f_2, f_3, \ldots f_{10}$

sampling at each level
* (only in Random forest)

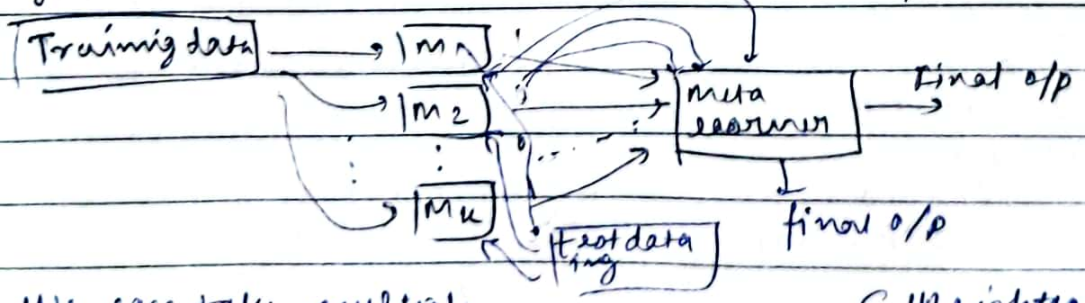
Out of Bag Evaluation :-

Nearly 37% of data items will never get selected in any bag (Reason) During the random sampling of data items with replacement, it is possible that some rows will never be selected in any subset.
These rows are called out of Bag ~~Evaluation~~ data
(Subset)

(3) Stacking :-

(Parallel + Heterogeneous)

It uses the prediction that are generated by ~~the~~ learners at 1st level as features to train a meta learners and make final predictions.



US SO
1 0

But ans = 1
since US are more accurate
(Imbalanced data)
meta learner learning

We can take multiple levels of base learner

Benefit { Weightage assigning if accuracy is more

09.04.2024 Tuesday

Homogeneous Ensemble Classification — Heterogeneous Ensemble Classifier

Easy Ensemble : Balan

Meta Classifier : Weighted Voting

(4) Boosting — एक model weightage के साथ सबसे Predicted अच्छे model को देता है।

Q) For a multiclass classification with m class ($C_1, C_2, \ldots, C_m$) Those are m instances having $k_1, k_2, k_3, \ldots k_m$ instances. How many unique experiments you can design using k fold cross validation.

Sol: or

Training instances x_1, x_2, x_3, x_4 → +1 class

y_1, y_2, y_3, y_4 → −1 class

Determine how many unique experimental setups using 2 fold cross validations.

Q) k fold cross validations does not ensure unique experimental setup

[Turn evaluate - Keel dataset repo. 5 fold cv]

Easy ensemble : Balance Cascading → To solve class Imbalance problem [Majority : Boys]

90 Boys 10 Girls

Undersampling of majority

10B1 10B2 ... 10B9 — 10G → 9 models → voting

No deletion of Boy

Reducing error due to variance

Performance fluctuation — Reason of ensemble

Component Classifier should be different & accurate
↳ diversity in classifier

Naive Baye's Classifier
↳ Soft Classifier

Majority Voting — Prediction combining N(C-1) N(C-2)

Soft Voting — combining the score of (C-1) classes & C-2 classes [Addition of Prob.]

Balance Cascade + Easy ensemble [class Imbalance problem]

90 B 10 G

Random 10B

① Assign equal wt. $\frac{1}{m}$
② Performance → misclassified weighted error
ε, α ③ Revised wt.

Balance cascade → sequential ensemble of classifier
[misclassified sample by classifier-1

Boosting training (weighted) [Ada Boost]
There are 10 instances 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
(2, 3, 4 misclassified)

Objective: minimize error

$\frac{1}{10}$ each → wt (misclassified error)

$\Rightarrow \varepsilon = \frac{3}{10}$ weighted error

$$\boxed{\text{wt. of this classified} = \alpha = \frac{1}{2}\ln\left(\frac{1-\varepsilon}{\varepsilon}\right) = \frac{1}{2}}$$

$= 0.4236$

$e^{-\alpha}$ $\alpha_1 = 0.4236$ → multiply these to correct instances

α_i = weight of classifier i

e^{α} to incorrect one

$e^{-\alpha} \times wt$

$e^{-0.4236} = 0.65 \times \frac{1}{10}$
$= 0.065$

$\hookrightarrow 1.52 = e^{0.4236} \times wt$
0.152
0.152×3
0.456

Rescale the weights
total wt. $= 0.65 \times 7$
$= \cancel{4.55}\ 0.455$

Total wt $= \cancel{5.006}\ 0.911$

Scale the wt.

1	2	3	4	5	6	7	8	9	10
$\frac{0.065}{0.911}$	$\frac{0.152}{0.911}$								

Sum = 1
After scaling

↳ for next iteration

[जिसको wt. ज्यादा दिया, उसको ज्यादा learn करते हैं]

Power generation of turbine
wt. square least square

Wind speed & Power generation relation
hill top पर ज्यादा नहीं
hill bottom

10.04.2024 Wednesday

DATE
PAGE

~~There are 2 implementation of SVMs~~ variant

Q) ~~Standard~~ Lib SVM → standard implementation of SVM
WSVM → Weighted SVM & Weighted extreme learning machines
Weighted Naive Bayes

Create a AdaBoost ~~SVM~~ Ensemble of WSVM for diabetes detection] PIMA diabetes assignment

Q) Learn the derivation of weight assigned to component classifier, & using wikipedia (Ada Boost Algo)
Sequential Algo

BOOSTING

The general principle of this model is to use multiple ~~are~~ weak learners sequentially and error exhibited by a weak learner is used to give importance to data instances which are further used to train the next weak learners.

Decision Stump → (Special) decision tree of ~~two~~ split (level)
used for Ada Boost

Ada Boost (Adaptive Boosting)

Step.1: Assign uniform wt. 1/N to all the data instances
→ weight

$$wt(d_j) = 1/N$$

Step.2: Repeat for each weak learner

(a) Train the weak learner H_i with a random sample

(b) compute the error ε_i of H_i on current training dataset

ε_i = sum of the wt. of wrong prediction of H_i

(c) Compute the wt. of weak learners.

$$\boxed{\alpha_i = \frac{1}{2} \ln\left(\frac{1-\varepsilon_i}{\varepsilon_i}\right)}$$

(d) Update the weight of all data instances.

$$\boxed{\begin{aligned} wt(d_j)_{i+1} &= wt(d_j)_i * e^{-\alpha_i} \\ wt(d_j)_{i+1} &= wt(d_j)_i * e^{\alpha_i} \end{aligned}}$$

[for correct classification]
[for incorrect classification]

↳ wt. of data instance j in i^{th} iteration (old wt)

(e) Normalize the weights of data instances

K Mean Clustering
Elbow ~~method~~ algo → to find K

Normalizing → Create Basket

0.21	0.21/x	0.12	0 – 0.12
0.012	0.012/x	0.11	0.13 – 0.23
0.42		0.11	0.24 – 0.34
		0.25	0.35 – 0.59
Sum = x		0.25	+0.60 – 0.85
		0.11	0.86 – 0.97
		0.03	0.98 – 1

DATE
PAGE

0 – 1

Prob
Randomly take.

0.xx
No.

Higher wt.
→ Bigger basket
P(solution) more

(f) Perform the unsampling.

'1' 37 times '1' 12

Step-3 : $H_F(d_j) = \text{sign}\left(\sum_{i=1}^{n} \alpha_i H_i(d_j)\right)$

$\alpha_1 H(d_j) + \alpha_2 H_2(d_j) + \alpha_3 H_3(d_j)$ –

d_j → m_1, m_2, m_3

m_1	m_2	m_3
α_1	α_2	α_3
Yes	No	Yes
+1	–1	+1

$H_F(d_j)$ → + Yes, – No

Q) Decision tree

Gender	Age	Income	Illness
M	41	40K	Yes
M	54	30K	No
F	42	25K	No
F	40	60K	Yes
M	46	50K	Yes
F	41	50K	No

~~Step-1~~ Wt Assign	Predicted		New wt	Normalize New wt / Total wt
1/6	Yes	✓	0.118	0.125
1/6	Yes	✗	0.235	0.249
1/6	No	✓	0.118	0.125
1/6	No	✗	0.235	0.249
1/6	Yes	✓	0.118	0.125
1/6	No	✓	0.118	0.125

$\Sigma = 0.942$

$\epsilon_1 = \frac{1}{6} + \frac{1}{6} = \frac{1}{3}$

$\alpha_1 = \frac{1}{2} \ln\left(\frac{1-\epsilon_1}{\epsilon_1}\right)$

Gender = M → Yes
Gender = Female → No

0.31 } Gender more important to correctly classify data point
0.34

0 – 0.125
0.125 – 0.374
0.375 – 0.478
0.479 – 0.727
0.728 – 0.852
0.853 – 1

creating baskets

15.04.2024 monday

Elbow method : Value of K in K means clustering

Run K means multiple times.

optimal K

CLUSTERING ALGO. → finding communities in social n/w graphs / customer classification

Clustering is a technique of partitioning a collection of unlabelled objects with many attributes into meaningful disjoint groups.

idea is that :

Characteristics of objects within the cluster are similar.

while differ from

Proximity Measure :-

distance measure

Minkowski Distance

(x_i, x_j)

$$\text{Distance}(x_i, x_j) = \left(\sum_{s=1}^{n} |x_{is} - x_{js}|^p \right)^{1/p}$$

s is number of features.

Euclidean distance

$p = 2$

$$\text{Distance}(x_i, x_j) = \sqrt{\sum_{i=1}^{s} (x_{is} - x_{js})^2}$$

manhattan dist.

$p = 1$

$$\text{Dist.}(x_i, x_j) = \sum_{s=1}^{n} |x_{is} - x_{js}|$$

Types of Clustering Algorithm :-

(1) Partitional clustering algo (K means Algo)

(2) Hierarchical clustering algo

(3) Density based clustering

Q) SNO X Y

SNO	X	Y
1	2	4
2	4	0
3	6	8
4	10	4
5	12	4

If K = 2 find clusters

Steps Algo

(i) Find K = no of clusters

(ii) Find K random centroids in dataset (centroids centroid)

(iii) Find dist. of all points to each centroid

(iv) Get minimum & put in that cluster

(v) Get new centroids

minimize sum of square error (SSE)

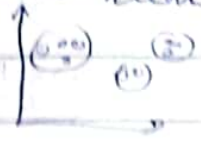
minimum

$$SSE = \sum_{i=1}^{K} \text{dist}(C_i, x)^2$$

C_i: centroid of ith cluster

x is a data point in that cluster

K is total no of clusters

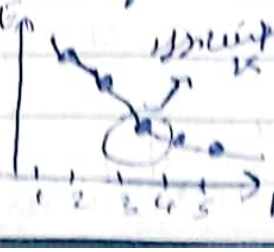


(2) Hierarchical clustering Algo

(i) Agglomerative method

(ii) Divisive method.

Elbow point where there is max. change in slope.



K	SSE
1	
2	
3	
4	
5	
6	

Apply KM

Natural & human
nm & neuron.

DATE
PAGE

The hierarchy relationship is carried in the form of a dendrogram. A dendrogram is a tree like diagram that records the sequence of merge or split

SNO	X	Y	C-1 (4,0)	C-2 (12,4)	C0 cluster centroids	(3,2)	(14,8)	C1 cluster
1	2	4	4.47	10	C-1	2.23	12.64	C-1
2	4	0	0	8.94	C-1	2.23	12.80	C-1
3	6	8	8.24	7.211	C-2	6.70	8	C-1
4	10	4	7.211	2	C-2	7.28	5.65	C-2
5	12	4	8.94	0	C-2	9.21	4.47	C-2

New cluster centroids
C-1 } → (3,2)
C-2 } → ~~(14,8)~~ (9.33, 5.33)

SNO.	X	Y	(3,2)	(9.33, 5.33)	C1 cluster centroids
1	2	4	2.23	7.44	C-1
2	4	0	2.23	7.53	C-1
3	6	8	6.70	4.709	C-2
4	10	4	7.28	1.48	C-2
5	12	4	9.21	2.98	C-2

Same as C0 cluster centroids

∴ cluster centroids = (3,2) & (9.33, 5.33)