Date: / /
Page

Machine learning

(main)

Supervised ML

- Regression
  - linear
  - Ridge & lasso
  - polynomial Reg.
  - Multilinear
  - Non-linear
  - OLS
  - Timeseries forecasting.
- classification
  - logistic
  - KNN
  - DT
  - Naive Byes
  - RF
  - XGB
  - GBM
  - SVM
  - ANN
  - catboost, adaboost

Unsupervised ML

- clustering
  - kmeans
  - DBscan
  - hierarchical
- apriori
- Recommdation
  - Content
  - Memory
    - User based
    - item based
- Dimentionality Reduction
  - PCA
  - tSNE
  - ICA
  - SVD
  - Auto Encoding.

Supervised ML →

We will start with Regression → linear

linear Regression

- Simple linear Regression (SLR)
- Multiple linear Regression (MLR)
- polynomial Regression

- Simple linear Regression (SLR) :- when for continuous output we have only one feature then it is called SLR

for Ex.

CGPA	package
6.0	2 LPA
7.2	3 LPA
8.9	6 LPA.

Now find its loss function.

$$\text{loss} = \frac{[(10-8)^2+(14-11)^2+(18-14)^2+(22-17)^2+(26-20)^2]}{5}$$

$$= \frac{4+9+16+25+36}{5} = \frac{90}{5} = 18$$

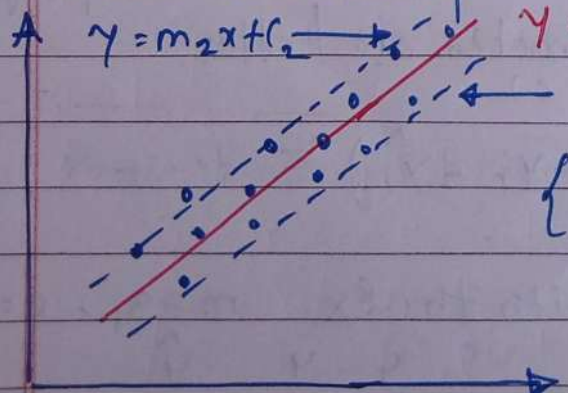
{ reason to take square :: so that positive and negative value may not cancel each other }

low loss value ⟶ High Accuracy
high loss value ⟶ low Accuracy

We can improve the model by some optimization technique called as "gradient descent" where repeate the process iteratively unhill we get bat parameter (m, c) for which model will give minimum loss function. Called as 'global minimum" in 'gradient descent"

How we can use gradient Descent for linear regression for optimization.

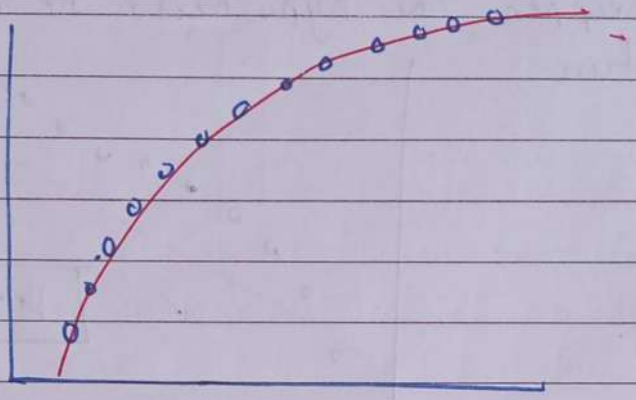
→ optimization refers to determining bat parameter for model such thal loss function of the model decreases as result of which model can predict more accurately.



{ m and c are the parameters }

here we can find model 3 is best fit since the loss function is least and thus this model is optimum this where we used gradient descent to optimize model

- Bias-Variance tradeoff :- if the algorithm is too simple (hypothesis with linear eq^) then it may be on high bias and low variance and thus it is error prone.
- if error fit too complex (hypothesis with high degree equation) then it may be on high variance and low bias, in the latter condition the new entries will not perform well. there is something between both of these condition known as tradeoff or bias-variance tradeoff



4) Effect of Regularization on loss function

loss function :-

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- it measures how far an estimated value from its true value
- if we are training on different models LR, DT, RF to know which model perform better and which parameters are better loss function is useful.

Elastic Net :- (L1+L2) : it is combination of both regularization technique.

$$L_1 \text{ Reg} = \frac{\sum (y_i - \hat{y}_i)^2}{n} + \underbrace{\lambda(|m| + |c|)}_{\text{penalty which is imposed on parameter}}$$

hyperparameter

$L_4 = \frac{1}{2}(L_3 + T_3) + \frac{1}{2}(Y_4)$

$= \frac{1}{2}(2.375 + 0.5625) + \frac{1}{2}(4)$

$= \frac{2.9375}{2} + 2$

$= 1.4687 + 2 = 3.4687$

$T_4 = \frac{1}{2}(T_3) + \frac{1}{2}(L_4 - L_3)$

$= \frac{1}{2}(0.5625) + \frac{1}{2}(3.4687 - 2.375)$

$= 0.28125 + 0.5(1.0937)$

$= 0.28125 + 0.54685$

$= 0.8281$

$L_5 = \frac{1}{2}(L_4 + T_4) + \frac{1}{2}(Y_5)$

$= \frac{1}{2}(3.4687 + 0.8281) + \frac{1}{2}(5)$

$= 2.1484 + 2.5$

$= 4.6484$

$T_5 = \frac{1}{2}T_4 + \frac{1}{2}(L_5 - L_4)$

$= \frac{1}{2}(0.8281) + \frac{1}{2}(4.6484 - 2.4687)$

$= 0.41405 + 0.5(1.1797)$

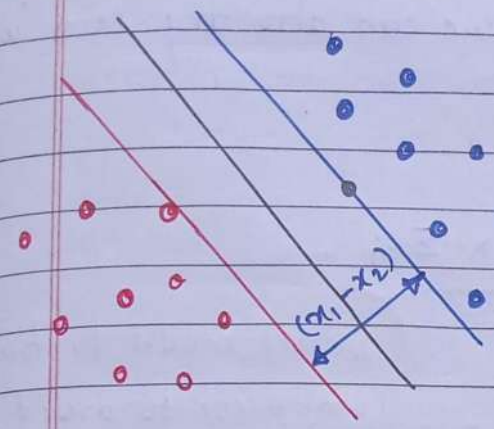
$= 0.41405 + 0.58985$

$= 1.0039$

T_1	T_2	T_3	T_4	T_5	} Trend is increased.
0	0.25	0.56	0.82	1.0039	

	Actual	Predicted
Simple Exponential	4.063	5
Double Exponential	4.6484	5

$w^T x + b = label$



Equation of point or blue support vector & its output value any negative value.

$w^T x + b = 1 \Rightarrow$ this is equation of point or red support vector and its output value could be any positive value

to get margine let substract one from another.

$$w^T x_1 + b = 1$$
$$(-)\ w^T x_2 + b = -1$$
$$w^T (x_1 - x_2) = 2$$

$$w^T (x_1 - x_2) = 2$$

divide both side by $\|w\|$

$$\frac{w^T (x_1 - x_2)}{\|w\|} = \frac{2}{\|w\|}$$

$$(x_1 - x_2) = \frac{2}{\|w\|}$$ ← this is nothing but magnitude of vector.

and

$$y_i = \begin{cases} -1 & w^T x_1 + b \leq -1 \quad (\text{label}) \\ 1 & w^T x_1 + b \geq 1 \end{cases}$$

So $\max \left(\frac{2}{\|w\|} \right)$ such that.

$$y_i = \begin{cases} -1 & w^T x_1 + b \leq -1 \\ 1 & w^T x_1 + b \geq 1 \end{cases}$$

Date: / /
Page

Decision Tree for regression.

Let understand with Example

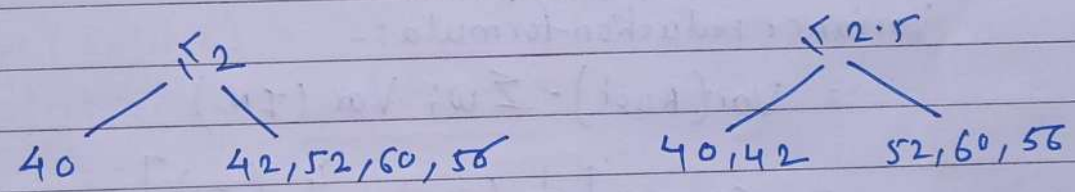
Exp	Gap	Salary (K)
2	Yes	40
2.5	Yes	42
3	No	52
4	No	60
4.5	Yes	56

$\bar{y} = 50$ ← Average.

Let take experience at root node

{Note:- Since exp is continuous data or arrange it in ascending order

- Now for comparison we will take two node example



- Now to decide which split is suitable we used one Concept called "variance reduction."

$$\text{Variance} = \frac{1}{n}\sum_{i=1}^{n}(y-\bar{y})^2 \leftarrow (\text{MSE formula})$$

where $\bar{y}$ = Average output.

- Now we have to calculate variance at each node.
- 1st we will calculate variance at root.

$$\text{Variance (Root)} = \frac{1}{5}\left[(40-50)^2+(42-50)^2+(52-50)^2 + (60-50)^2+(56-50)^2\right]$$

$$= \frac{1}{5}\left[100+64+4+100+36\right]$$

$$= 60.8$$

prediction = initial leaf + learning rate x (current leaf

- let take learning rate is 0.8 which is very large but it for illustrative purpose however 0.1 is most common.

Now predict for each person, person 1

$$0.7 + (0.8 \times 1.4) = 1.8$$

$$\text{probability} = \frac{e^{1.8}}{1+e^{1.8}} = 0.9$$

Similar we do for all the person.

like popcorn	Age	favorail Color	love Movie	Residual	predicted	Residual.
yes	12	Blue	yes	0.3	0.9	0.1
yes	87	green	yes	0.3	0.5	0.5
No	44	blue	No	-0.7	0.5	-0.5
yes	19	Red	No	-0.7	0.1	-0.1
No	32	green	yes	0.3	0.9	0.1
No	14	blue	yes	0.3	0.9	0.1

this new predicted probabilities are worst than before & that's why we build lot of tree and not just one

- And now just like before we calculate new raiduals since Residual: observed - predicted.

for person 1st : (1-0.9) = 0.1
2nd : (1-0.5) = 0.5
⋮ ⋮ ⋮
6th : (1-0.9) = 0.1

} put in above table

Date : / /
Page

if $\lambda = 0$ mean w/o regularization = -10.5
if $\lambda = 1$ with regularization factor = -5.25

if $\lambda > 0$ then it will reduce the amount that this individual observation adds to overall prediction.

thus λ (lambda) regularization parameter will reduce the prediction sensitivity to this individual observation

output of leaf ② : $\frac{6.5 + 7.5}{2+0} = 7$

when $\lambda = 0$ o/p of value is simply avg of residual of leaf.

output of leaf ⑧ : -7.5

Since we have build new tree we can make new prediction
like gradient boosting we have to use learning rate here also, default value is 0.3. rest is same

∴ initial leaf + learning rate × DT.

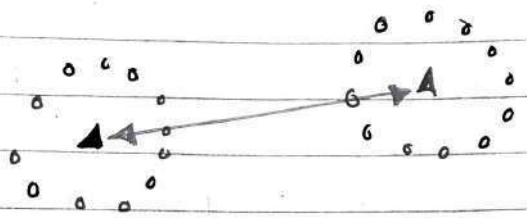
prediction for Dosage 10

$$0.5 + 0.3(-10.5) = -2.65$$

Now we can see new residual is smaller than before this means we are taking small step in right direction

Dosage = 20 = 2.6
Dosage = 22.5 = -1.75 } smaller than before.

4) Centroid linkage:- it is linkage method in which the distance between the centroid of the cluster is calculated.



- These components are orthogonal i.e the correlation between a pair of variable is zero.
- The importance of each component decreases when going to 1 to n it means the 1PC has most importance and n PC will have the least importance

Steps for PCA Algorithm :-

1. Getting the dataset :- firstly we need to take the input dataset and divide it into two subparts X and Y where X is training set and Y is the validation set.

2) Representing data into structure :- Now we will represent our dataset into a structure such as we will represent the two dimensional matrix of independent variable X. Here each row corresponds to the data items and the column corresponds to the feature. The number of columns is the dimension of the dataset.

3) standardizing the data :- in this step we will standardize our dataset. such as particular column, the feature with high variance are more important compared to the features with lower variance. if the importance of the feature is ~~dep~~ independent of the variance of the feature, then we will divide each data item in a column with the std deviation of the column. Here we will name the matrix as Z.