HomeAssignment-03

1) explain the mathematical formulation of linear Regression & logistic regression. dicuss how cost functions differ b/w them & the role of regularization.

A) Linear regression

linear regression predicts a continuous output as a linear combination of input features

$$\hat{y} = w_0 + w_1 x_1 + w_2 x_2 + \dots + w_n x_n = w^T x$$

$x = [x_1, x_2 \dots, x_n]^T$ feature vector

$w = [w_0, w_1, \dots, w_n]^T$ - model parameters (weights)

$\hat{y}$ = predicted value

Cost function

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

m = number of training examples

$\hat{y}_i = w^T x_i$

This measures how far predictions are from actual

Logistic regression

logistic regression predicts a probability for a binary outcome (0 or 1):

$$\hat{y} = P(y=1|x) = \sigma(w^T x)$$

where

$$\sigma(z) = \frac{1}{1+e^{-z}}$$

Cost function

$$J(w) = -\frac{1}{m} \sum_{i=1}^{m} [y_i \log(\hat{y}_i) + (1-y_i) \log(1-\hat{y}_i)]$$

this penalizes wrong predictions more heavily when the model is confident but incorrect.

Role of Regularization

Regularization helps reduce overfitting by adding a penalty for large co-efficients in the cost function

a) Ridge Regression

Adds a penalty equal to the square of the coefficients

$$J(w) = Loss + \lambda \sum_{j=1}^{n} w_j^2$$

b) Lasso Regression

Adds a penalty equal to the absolute value of co-efficient

$$J(w) = Loss + \lambda \sum_{j=1}^{n} |w_j|$$

Q) Define the "curse of dimensionality". How does it affect algorithms like K-Nearest Neighbors (KNN)? Suggest two appoaches to reduce its impact in practice.

A) Curse of Dimensionality

The curse of dimensionality refers to the set of Problems that arise when data has too many features Campared to the number of samples

How it Affects Algorithm like KNN

K-Nearest Neighbors depends heavily on distance metrics

1) All points appear equally distant

→ the different between the nearest & farthest points becomes negligible

→ Distance loses it meaning so KNN can't distinguish b/w close & far points efficiently

$$d \to \infty \quad \frac{D_{max} - D_{min}}{D_{min}} \to 0$$

2) overfitting risk increases

* KNN may fit noise rather than real pattern because each sample is isolated in high-dimensional space.

3* computational cost rises

more dimensions -> higher time & memory needed to compute distances for all features.

* Approaches to Reduce its Impact

a) Dimensionality Reduction

-> PCA(principal component Analysis

Projects data to a lower-dimensional space by combining correlated features

b) Feature Normalization/Regularization

* Helps balance the influence of features & reduce noise

③ A data base contains information about whether students pass or fail based on three attributes

Study Hours	Attendence	Internal marks	Result
High	Good	High	Pass
High	Good	Low	pass
Low	Good	High	Pass
Low	poor	Low	Fail
High	Poor	Low	Fail
Low	good	Low	Fail

a) compute the Entropy of the target attribute

b) calculate the information gain for each attribute

c) identify the best attribute to split at the root node

d) draw the first level Decision Tree based on your calculations.

Step 1

Pass = 3 (1, 2, 3)

Fail = 3 (4, 5, 6)

$P(\text{pass}) = \frac{3}{6} = 0.5$, $P(\text{fail}) = 0.5$

Entropy (Result) = $-[0.5 \log_2(0.5) + 0.5 \log_2(0.5)] = 1.0$

Entropy = 1.0

Step 2

a) Study Hours	pass	fail	Total	Entropy
High	2	1	3	$-[\frac{2}{3} \log_2 \frac{2}{3} \log_2 \frac{1}{3}] = 0.918$
Low	1	2	3	$-(\frac{1}{3} \log_2 1/3 + \frac{2}{3} \log_2 \frac{2}{3}] = 0.918$

$\text{Entropy}_{\text{studyhours}} = 3/6 (0.918) + 3/6 (0.918) = 0.918$

IG (Study Hours) = 1.0 - 0.918 = 0.082

b) Attendence	pass	fail	Total	Entropy
Good	3	1	4	$-(\frac{3}{4} \log_2 3/4 + 1/4 \log_2 1/4] = 0.811$
Poor	0	2	2	$-[0 \log_2(0) + 1 \log_2(1)] = 0$

$\text{Entropy}_{\text{Attendence}} = \frac{4}{6}(0.811) + \frac{2}{6}(0) = 0.541$

IG attendence = 1.0 - 0.541 = 0.459

c) Attribute

Internal marks	pass	fail	Total	Entropy
High	2	0	2	$-[1 \log_2(1) + 0] = 0$
Low	1	3	4	$-[\frac{1}{4} \log_2 1/4 + 3/4 \log_2 3/4] = 0.811$

$\text{Entropy}_{\text{Internal Marks}} = 2/6 (0) + \frac{4}{6}(0.811) = 0.541$

IG (Interal Marks) = 1.0 - 0.541 = 0.456

4) Discuss the max-margin principle of Support vector Machines. Explain the concepts of hard margin, Soft margin & the role of kernel functions in handling non-linear data

1) Support vector machine

It's main goal is to find the optimal separating hyperplane that best the margin b/w two classes

Hard margin SVM

* Assumes linearly separable data
* Maximizes distance between Support vectors & hyperplance

Soft Margin Svm

* Allows Some misclassification using penalty terme
* Bances margin maximization with error

object: min $\frac{1}{2}||w||^2 + C \Sigma$ ci

Kernel tricks

maps data into higher-dimensional Space to find alinear baundary

Common Kernels

RBF :- Handles circular/non linear boundaries
Polynomial : captures curved relationhips
Sigmoid : similar to neural networks

5) Campare bagging, Boosting, Random forest, & stracking in terms of methodology, base leavenergy & bias vaiance trade off. provide one react world use case for each

1) Bagging

Methodology :-
Multiple models are trained on different bootstrap Samples of the training data

Base Learners
Typically high-variance, low-bias models like Decision Trees

Bias-variance Trade-off

Reduces Variance without increasing bias much - helps avoid overfitting

Real-world use case

Credit Risk predicition - combining multiple decision trees to stabilize predictions about loan default risk.

Boosting :-

Methodology

Models are built sequentially, where each model is to correct the errors of the previous one.

Base Learners

Usally weak learns, like shallow decision trees

Bias-variance Trade-off

Reduces bias but may increase variance if overfitted

Real-world Use Case

Spam Email Detection - boosting refines predictions to distinguish spam from legitimate mail accurately.

Random forest :-

Methodology :

An improved version of bagging where each tree is trained on a bootstrap sample & uses a random Subset of features at each spilt - increasing model diversity

Base learners

Decision Tress

Bias-variance Trade-off

Greatly reduces variance while maintaing low bias - more stable that bagging alone

Real world use case

Medical Diagnosis - predicting diseases using patient features with high accuracy & robustness.

Stacking

Methodology

Combines different of models & uses a meta-model to learn how to best combine their outputs

Base Learns

Heterogeneous

Bias-Variance Trade-off

Tries to balance both bias & variance by leveraging strengths of diverse methods

Real-world use case

House price prediction stacking regression models improves accurancy over any single model

6) explain the working principle of gaussian Naive Bayes and Gaussian processes. How would model calibration improve the reliability of predicted probabilities?

Gaussian Naive Bayes

* It is uses Bayes rule to predict the class
* It assumes all features are independent
* Each feature follows a normal curve.
* It picks the class with the highest probability

Gaussian Processes

* used for regression
* It assumes data points come from a Gaussian pattern.
* It uses a kernel to find how similar points are.
* Gives both a prediction & how certain it is

Model calibration

* Makes the model's probilities more accounts
* Example: if it says 80% change, it really happens about 8 out of 10 times
* It makes prediction more trustwortly.

7) Differentiate between accuracy, precision, recall, F1. core, Roc and AUC. why are these metrics important for imbalan datasets? Also, briefy discuss the importance of model explainability in modern AI systems.

Accurancy

* Tells how many predictions are correct overall.
* Formula: (correct predictions /total predictions)
* problem: Can be misleading if one class has many more samples

precision

* out of all predicted positives, how many are actually Positions
* focus: Avoiding false positives
* Example: Good for spam detection

Recall:

* out of all actual postives, how many are correctly found
* focus: Avoiding false negatives
* Example: Good for disease detection

F1-Score

* The balance b/w precision & recall
* Formula: 2 x (precision x Recall) | (precision + Recall)
* Good when dat a is imbalance

Roc (Receiver operating charateristic)

* shows how to model performs at different thresholds
* plots true positive Rate Vs False positive Rate,

AUC (Area under curve)

* Measures the overall performance at defferent thesold
* Higher AUC = better model.

why important for imbalanced data

* Accuracy may look high even if the model ignores the smaller ~~dat~~class
* precision, Recall, F1, & AUC give a better picture of real performance.