Answer the following questions:

1. What is the role of an optimizer in a neural network?
2. How does learning rate impact the performance of a model? Explain briefly.
3. What causes a function to lack the characteristics of a convex curve? Is the squared error function a good choice for logistic regression? Justify your answer.
4. **Determine how information and probability relate to one another. How can the logistics loss function be derived using information theory?**
5. Write down the differences between linear regression and logistic regression. What are the steps involved in converting a linear regression to a logistic regression?
6. What is the role of bias in a logistic regression algorithm? Explain Briefly.
7. Draw the pipeline of a logistic regression algorithm. For each step, describe how it works?
8. Draw the pipeline of a Logistic Regression. Explain the working procedure of every step
9. Why is it necessary for the loss function to be convex? Is squared error function a good choice? Justify your answer.
10. Write down your ideas about information theory. How can the logistics loss function be derived using information theory.
11. Describe the backward propagation of a logistic regression where the activation function is a "Sigmoid" activation function and the loss function is the "Cross Entropy Loss" function
12. Write down the algorithm to perform a logistic regression. How can you vectorize the algorithm?

1. Role of an Optimizer in a Neural Network:

An optimizer in a neural network is responsible for adjusting the weights and biases of the network during the training process to minimize the error or loss function. The optimizer uses optimization algorithms, such as gradient descent, to find the optimal values for the model parameters, thereby improving the model's performance.

2. Impact of Learning Rate on Model Performance: (Quize-1 e eta nei)

The learning rate is a hyperparameter that determines the step size at which the optimizer adjusts the model parameters. A too high learning rate may cause the model to overshoot the minimum, leading to divergence or oscillation, while a too low learning rate may result in slow convergence or getting stuck in local minima. Finding an appropriate learning rate is crucial for achieving optimal training results.

3.Characteristics of Convex Curves and Choice of Squared Error for Logistic Regression: (Quize-1 e eta nei)

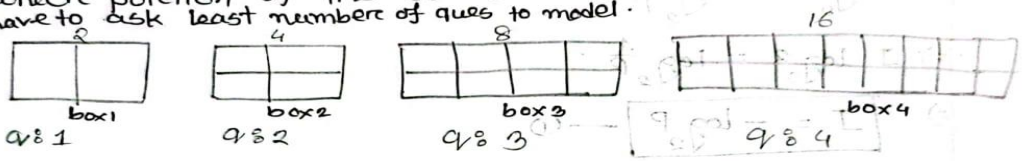
Convexity in a function implies that any line segment connecting two points on the curve lies above the curve itself. Functions with multiple minima or non-convex shapes can lack this property. The squared error function is not a good choice for logistic regression because the logistic regression model involves a non-linear

transformation (sigmoid function) to map inputs to probabilities, leading to a non-convex optimization problem. The use of squared error can result in non-convexity and convergence issues.

4. Information and Probability in Logistics Loss Function:

Determine how information and probability relate to one another. How can the logistics loss function be derived using information theory?

Suppose there some boxes with balls & have to guess/predict which portion of the boxes have the ball. To predict ball have to ask least number of ques to model.



Probability:

$P = \frac{1}{2} = \frac{1}{2^1}$ $\quad P = \frac{1}{4} = \frac{1}{2^2}$ $\quad P = \frac{1}{8} = \frac{1}{2^3}$ $\quad P = \frac{1}{16} = \frac{1}{2^4}$ $\quad \cdots \frac{1}{2^n}$

Here to find ball at least 1 ques need to get/predict the ans for box 1. 2, 3, 4 for box2, box3, box4 respectively. Strategy is half the box in every possible portion.

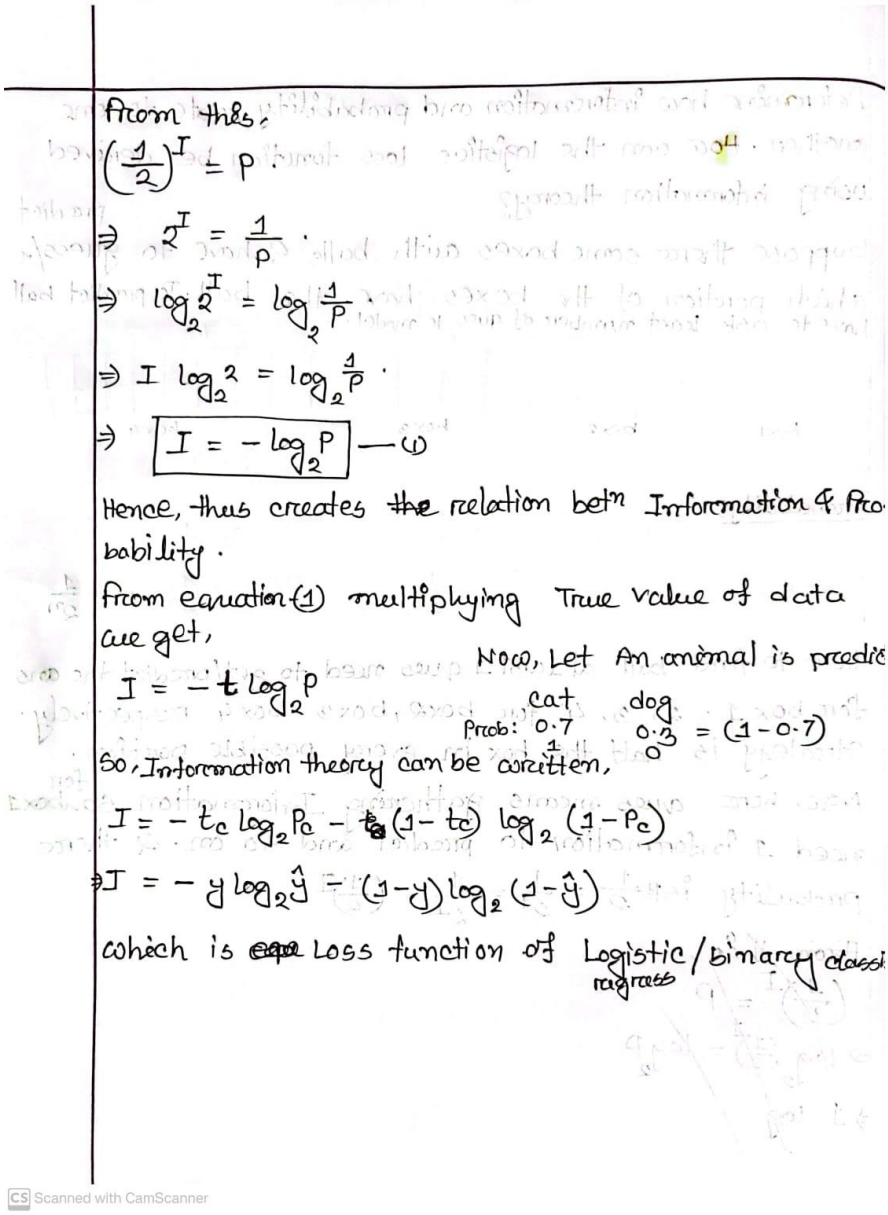
Now, here ques means gathering Information so for box1 need 1 information to predict and so on. & there probality is $P = \frac{1}{2} = \frac{1}{2^1} = \frac{1}{2^I} = \left(\frac{1}{2}\right)^I$

From this,

$\left(\frac{1}{2}\right)^I = P$

$\Rightarrow \log_2 \left(\frac{1}{2}\right)^I = \log_2 P$

$\Rightarrow I \log$



Information theory measures the amount of surprise or uncertainty associated with an event. Cross-entropy, a concept from information theory, is used to derive the logistic loss function. The logistic loss is essentially the negative log-likelihood of the true labels given the predicted probabilities. Minimizing this loss is equivalent to maximizing the likelihood of the observed data, aligning with the principles of information theory.

5. Differences Between Linear Regression and Logistic Regression:

- Linear Regression predicts continuous output, while Logistic Regression predicts the probability of an event.

- The output of linear regression is unbounded, while logistic regression output is constrained between 0 and 1.

- Linear regression uses the least squares method, while logistic regression uses the maximum likelihood estimation.

- The decision boundary in logistic regression is a sigmoid function.

Converting linear regression to logistic regression involves applying a logistic (sigmoid) function to the linear combination of inputs, transforming the output to a probability scale.

6. Role of Bias in Logistic Regression:

The bias term in logistic regression allows for the model to make predictions even when all input features are zero. It represents the log-odds of the probability of the positive class when all input features are zero. Including a bias term helps the logistic regression model to better fit the data and capture the intercept or baseline probability of the event being predicted.

FROM SLIDE:

The bias value allows the activation function to be shifted to the left or right, to better fit the data.

● Changes to the weights alter the steepness of the sigmoid curve, whilst the bias offsets it, shifting the entire curve so it fits better.

● Bias only influences the output values, it doesn't interact with the actual input data. That's why it is called bias.

● You can think of the bias as a measure of how easy it is to get a node to fire.

○ For a node with a large bias, the output will tend to be intrinsically high, with small positive weights and inputs producing large positive outputs (near to 1).

○ Biases can be also negative, leading to sigmoid outputs near to 0.

○ If the bias is very small (or 0), the output will be decided by the values of weights and inputs alone.

7. Logistic Regression Algorithm Pipeline:

1. Input Data:

   - Input features (X) and target labels (y).

2. Initialization:

   - Initialize weights (W) and bias (b) with small random values.

3. Linear Transformation:

   - Calculate the linear combination of input features and weights: (z = XW + b).

4. Activation Function (Sigmoid):

   - Apply the sigmoid activation function to the linear output: (a = sigma(z)), where (sigma(z) = frac{1}{1 + e^{-z}}).

5. Prediction:

   - Interpret the sigmoid output as the probability of belonging to the positive class.

6. Loss Calculation:

   - Calculate the logistic loss using the cross-entropy loss function:

(L(y, a) = -y log(a) - (1-y) log(1-a)).

7. Gradient Descent:

   - Compute the gradient of the loss with respect to the weights and bias.

8. Parameter Update:

   - Update weights and bias using the gradient and the chosen learning rate.

9. Iteration:

   - Repeat steps 3-8 until convergence or a set number of iterations.

8. Logistic Regression Pipeline (Detailed Explanation):

- Input Data:Features (X) and target labels (y).

- Initialization: Initialize weights (W) and bias (b).

- Linear Transformation: Calculate \(z = XW + b\).

- Activation Function (Sigmoid): Apply the sigmoid function to \(z\), yielding \(a = \sigma(z)\).

- Prediction: Interpret \(a\) as the probability of belonging to the positive class.

- Loss Calculation: Compute the logistic loss using the cross-entropy loss function.

- Gradient Descent: Compute the gradient of the loss with respect to weights and bias.

- Parameter Update: Update weights and bias using the gradient and learning rate.

- Iteration: Repeat steps until convergence.

9. Necessity of Convex Loss Function: (Quize-1 e eta nei)

- Convexity ensures the existence of a global minimum, making optimization more reliable.

10. Is squared error function a good choice? Justify your answer.

- The squared error function is not ideal for logistic regression because the sigmoid activation introduces non-linearity, leading to a non-convex optimization problem. Gradient-based optimization algorithms may get stuck in local minima.

FROM SLIDE

The squared error function (commonly used function for linear regression) is not very suitable for logistic regression.

○ In case of logistic regression, the hypothesis / prediction is non-linear (sigmoid function), which makes the square error function to be non-convex.

○ On the other hand, logarithmic function is a convex function for which there is no local optima, so gradient descent works well.

● If you are doing binary classification, squared error function generally also penalize examples that are correctly classified but are still near the decision boundary, thus creating a "margin."

● Gradient descent waste a lot of time getting predictions very close to {0, 1}

$$L(\hat{y}^{(i)}, y^{(i)}) = \frac{1}{2}(\hat{y}^{(i)} - y^{(i)})^2$$

- We can see an extra (1/2) in the right side of the equation. Does it matter?

● It is because when you take the derivative of the cost function, that is used in updating the parameters during gradient descent, that 2 in the power get cancelled with the (1/2) multiplier.

● These techniques are or somewhat similar are widely used in math in order "To make the derivations mathematically more convenient".

11. Information Theory and Logistic Loss Function:

- Information theory measures uncertainty. Cross-entropy is a measure of surprise or information gain.

- Logistic loss, derived from cross-entropy, quantifies the difference between predicted probabilities and actual labels.

- Minimizing logistic loss aligns with maximizing the likelihood of observed data, connecting logistic regression to information theory.

12. Backward Propagation in Logistic Regression: (Quize-1 e eta nei)

- Loss Gradient: Compute the gradient of the loss with respect to the predicted probabilities.

- Sigmoid Backward:** Compute the gradient of the sigmoid activation function.

- Chain Rule: Combine gradients to compute the overall gradient of the loss with respect to the weighted inputs.

-Parameter Update: Use gradients to update weights and bias using gradient descent.

13. Logistic Regression Algorithm: (Quize-1 e eta nei)

-Input: Features (X), labels (y), learning rate, and number of iterations.

- Initialize: Weights (W) and bias (b).

- For each iteration:

  - Linear Transformation: \(z = XW + b\).

  - Sigmoid Activation: \(a = \sigma(z)\).

  - Loss Calculation: \(L = -y \log(a) - (1-y) \log(1-a)\).

  - Gradient Calculation: Compute gradients with respect to weights and bias.

  - Parameter Update: Update weights and bias using the gradients and learning rate.

- Vectorization: Instead of looping through individual samples, perform operations on entire matrices. This enhances computational efficiency, often implemented using libraries like NumPy.

14.What are the steps involved in converting a linear regression to a logistic regression?

Converting a linear regression model to a logistic regression model involves several steps to adapt the model for classification tasks. Here are the key steps:

1.**Sigmoid Activation Function:**

- Replace the linear activation function in the output layer with the sigmoid activation function (logistic function):

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

- This function squashes the output between 0 and 1, transforming the continuous output into a probability.

2. **Set a Threshold:**

- Choose a threshold (commonly 0.5) to convert probabilities into binary predictions. If \(a \geq 0.5\), predict class 1; otherwise, predict class 0.

3. **Change the Output Range:**

- Since logistic regression is used for binary classification, modify the output interpretation. Instead of predicting a continuous value, interpret the output as the probability of belonging to the positive class.

4. **Modify the Loss Function:**

- Change the loss function from Mean Squared Error (MSE) to the binary cross-entropy loss (log loss) for logistic regression:

[L(y, a) = -y log(a) - (1-y) log(1-a)]

- Here, (y) is the true label (0 or 1) and (a) is the predicted probability.

5. **Gradient Descent Update:**

- Adjust the gradient descent update rule accordingly to accommodate the new loss function and activation function.

1. regression Vs Classification.

2. Supervise vs unsupervise data.

3.Structured vs Unstructured Data.

1. Regression vs Classification:

- **Output Type:**

- Regression: Predicts a continuous output or numerical value.

- Example: Predicting the temperature (a continuous value) based on features like time of day, humidity, and wind speed.

- Classification: Predicts a discrete category or label.

- Example: Classifying emails as either spam or not spam based on features like sender, subject, and content.

- **Objective:**
  - Regression: Minimizes the difference between predicted and actual values.
  - Classification: Assigns inputs to predefined categories or classes.
- **Output Interpretation:**
  - Regression: The output represents a quantity that can be interpreted on a scale.
  - Classification: The output represents a class or category.
- Examples of Algorithms:
  - Regression; Linear Regression, Polynomial Regression.
  - Classification: Logistic Regression, Decision Trees, Support Vector Machines.

2. Supervised vs Unsupervised Data:

- **Labeled vs Unlabeled:**
  - Supervised: Training data includes labeled examples with corresponding outputs.
  - Example: A dataset of labeled images with annotations indicating whether they contain cats or dogs.
  -Unsupervised: No explicit labels are provided, and the algorithm discovers patterns or relationships.
  - Example: Clustering a dataset of customer purchase history to identify groups with similar buying behavior.

- **Guidance:**
  - Supervised: The algorithm is guided by the labeled output during training.
  -Unsupervised: The algorithm explores the data structure without explicit guidance.
- **Objective:**
  - Supervised: Predict or classify based on known relationships.
  - Unsupervised: Discover hidden patterns or relationships within the data.
- **Examples of Algorithms:**
  - Supervised: Linear Regression, Classification models, Neural Networks.

-Unsupervised: K-Means Clustering, Hierarchical Clustering, Principal Component Analysis (PCA).

3. Structured vs Unstructured Data:

- Organization:

- Structured: Data is organized and follows a clear, predefined structure.

- Example: Relational databases with tables, rows, and columns.

- Unstructured: Data lacks a predefined structure and organization.

- Example: Text documents, images, audio recordings.

- Searchability:

- Structured: Easily searchable and accessible due to a clear schema.

- Unstructured: Requires more advanced methods (e.g., natural language processing for text) for effective search and retrieval.

- Examples:

- Structured: Excel spreadsheets, SQL databases.

- Unstructured: Social media posts, images, videos, audio recordings.

- Flexibility:

- Structured More rigid and less adaptable to changes in data types.

-Unstructured: Allows for more flexibility as data types and formats can vary widely.

These differences highlight the diverse nature of data and the various approaches used in machine learning based on the characteristics of the data at hand.