Q1. Illustrate with an example the difference between Machine Learning, Deep Learning and Artificial Intelligence.

What is Artificial Intelligence (AI)?

Artificial Intelligence is defined as a field of science and engineering that deals with making intelligent machines or computers to perform human-like activities.

What is Machine Learning?

Machine Learning is defined as the branch of Artificial Intelligence and computer science that focuses on learning and improving the performance of computers/machines through past experience by using algorithms.

AI is used to make intelligent machines/robots, whereas machine learning helps those machines to train for predicting the outcome without human intervention.

What is Deep Learning?

"Deep learning is defined as the subset of machine learning and artificial intelligence that is based on artificial neural networks". In deep learning, **the deep word refers to the number of layers in a neural network.**

Deep learning can be useful to solve many complex problems with more accurate predictions such as **image recognition, voice recognition, product recommendations systems, natural language processing** (NLP), etc.

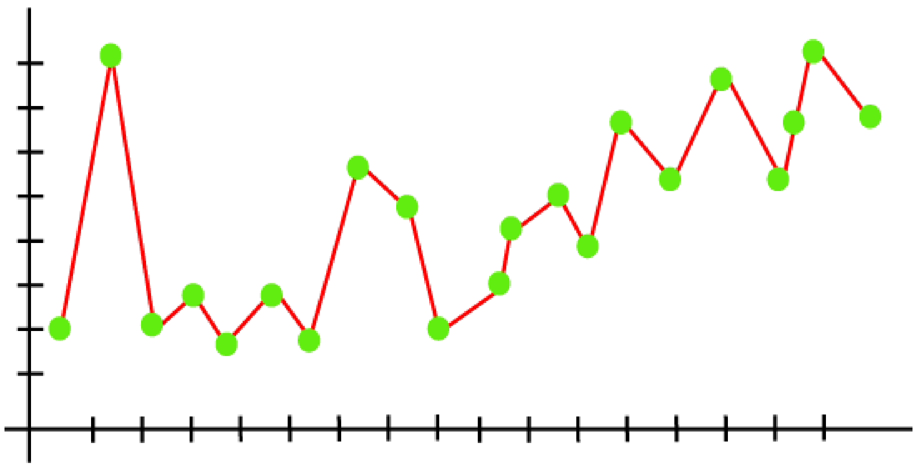
Artificial Intelligence	Machine Learning	Deep Learning
Examples of AI applications include: Google's AI-Powered Predictions, Ridesharing Apps Like Uber and Lyft, Commercial Flights Use an AI Autopilot, etc.	Examples of ML applications include: Virtual Personal Assistants: Siri, Alexa, Google, etc., Email Spam and Malware Filtering.	Examples of DL applications include: Sentiment based news aggregation, Image analysis and caption generation, etc.

Q2. Explain what is overfitting and underfitting. Explain the techniques to avoid overfitting and underfitting.

Ans:- Overfitting

Overfitting occurs when our machine learning model tries to cover all the data points or more than the required data points present in the given dataset. Because of this, the model starts caching noise and inaccurate values present in the dataset, and all these factors reduce the efficiency and accuracy of the model. The overfitted model has **low bias** and **high variance.**

Example: The concept of the overfitting can be understood by the below graph of the linear regression output:



As we can see from the above graph, the model tries to cover all the data points present in the scatter plot. It may look efficient, but in reality, it is not so. Because the goal of the regression model to find the best fit line, but here we have not got any best fit, so, it will generate the prediction errors.

How to avoid the Overfitting in Model

Both overfitting and underfitting cause the degraded performance of the machine learning model. But the main cause is overfitting, so there are some ways by which we can reduce the occurrence of overfitting in our model.

- **Cross-Validation**
- **Training with more data**
- **Removing features**
- **Early stopping the training**

- **Regularization**
- **Ensembling**

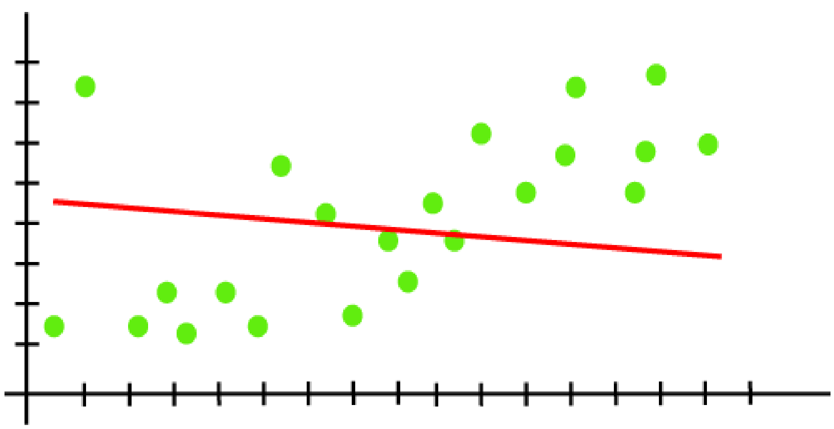
Underfitting

Underfitting occurs when our machine learning model is not able to capture the underlying trend of the data. To avoid the overfitting in the model, the fed of training data can be stopped at an early stage, due to which the model may not learn enough from the training data. As a result, it may fail to find the best fit of the dominant trend in the data.

In the case of underfitting, the model is not able to learn enough from the training data, and hence it reduces the accuracy and produces unreliable predictions.

An underfitted model has high bias and low variance.

Example: We can understand the underfitting using below output of the linear regression model:



As we can see from the above diagram, the model is unable to capture the data points present in the plot.

How to avoid underfitting:

- By increasing the training time of the model.
- By increasing the number of features.

Q3. Explain the terms bias and variance with respect to machine learning. Discuss the biasvariance trade-off in machine learning.

Ans:- **Bias**
The bias is known as the difference between the prediction of the values by the ML model and the correct value. Being high in biasing gives a large error in training as well as testing data. Its recommended that an algorithm should always be low biased to avoid the problem of underfitting.

Variance
The variability of model prediction for a given data point which tells us spread of our data is called the variance of the model.

Bias Variance Tradeoff

If the algorithm is too simple (hypothesis with linear eq.) then it may be on high bias and low variance condition and thus is error-prone. If algorithms fit too complex (hypothesis with high degree eq.) then it may be on high variance and low bias. In the latter condition, the new entries will not perform well. Well, there is something between both of these conditions, known as Trade-off or Bias Variance Trade-off.

Q4. Discuss in detail the application of linear regression for prediction.

Ans:- **Linear regression** is one of the most commonly used techniques in statistics. It is used to quantify the relationship between one or more predictor variables and a response variable. The most basic form of linear is regression is known as simple linear regression, which is used to quantify the relationship between one predictor variable and one response variable.

Businesses often use linear regression to understand the relationship between advertising spending and revenue.

For example, they might fit a simple linear regression model using advertising spending as the predictor variable and revenue as the response variable. The regression model would take the following form:

revenue = $\beta 0$ + $\beta 1$(ad spending)
The coefficient **$\beta 0$** would represent total expected revenue when ad spending is zero.

The coefficient **$\beta 1$** would represent the average change in total revenue when ad spending is increased by one unit (e.g. one dollar).

If $\beta 1$ is negative, it would mean that more ad spending is associated with less revenue.

If β1 is close to zero, it would mean that ad spending has little effect on revenue.

And if β1 is positive, it would mean more ad spending is associated with more revenue.

Depending on the value of β1, a company may decide to either decrease or increase their ad spending.

Q5.Briefly describe any two techniques used to compute the parameters of the linear regression model

Ordinary Least Squares

When we have more than one input we can use Ordinary Least Squares to estimate the values of the coefficients. The Ordinary Least Squares procedure seeks to minimize the sum of the squared residuals. This means that given a regression line through the data we calculate the distance from each data point to the regression line, square it, and sum all of the squared errors together. This is the quantity that ordinary least squares seeks to minimize. This approach treats the data as a matrix and uses linear algebra operations to estimate the optimal values for the coefficients.

Gradient Descent

When there are one or more inputs you can use a process of optimizing the values of the coefficients by iteratively minimizing the error of the model on your training data. This operation is called Gradient Descent and works by starting with zero values for each coefficient. The sum of the squared errors are calculated for each pair of input and output values. A learning rate is used as a scale factor and the coefficients are updated in the direction towards minimizing the error. The process is repeated until a minimum sum squared error is achieved or no further improvement is possible. When using this method, you must select a learning rate (alpha) parameter that determines the size of the improvement step to take on each iteration of the procedure.

Q6.Write a python program that does/has the following:

a. Takes string inputs from the user and store in a list

input:

```
li = list(input("enter some value: ").split(" "))
print(li)
```

Output:

```
enter some value: 88 abd 55 rr kk
['88', 'abd', '55', 'rr', 'kk']
```

b. A user defined function that takes the list in step a as an input parameter and returns the count of strings of length greater than 5 that starts and ends with same character. Example: Input [“121”, “155322”, “abcbba”, “abcda”, “1a1a21”] Output: 2

INPUT:

```
def counter(y):
    count = 0
    for x in y:
        if len(x) > 5:
            if x[0] == x[len(x)-1]:
                count = count+1
    return count
counter(li)
```

OUTPUT:

2

7. Write a python program that does/has the following:

a. Takes key value inputs from the user and store in a dictionary. The value is a list of numbers

Q8. A box contains 3 blue marbles, 4 red, 6 green marbles and 2 yellow marbles. If three marbles are picked at random, what is the probability that they are all blue?

To find : What is the probability that they are all blue?

Solution :

Blue marbles = 3

Red marbles = 4

Green marbles = 6

Yellow marbles = 2

Total marbles = 3+4+6+2=15

Favorable outcome (three marbles picked that are blue)=3

Probability that all three marbles are blue is

$$\text{Probability} = \frac{\text{Favorable outcome}}{\text{Total outcome}}$$

$$\text{Probability} = \frac{3}{15}$$

$$\text{Probability} = \frac{1}{5}$$

Therefore, The probability of getting all blue marbles is $\frac{1}{5}$

Q9.A Bag contains 6 Blue Balls and 4 Red Balls. 3 balls are picked at random. What is the probability that none of them is Red?

Let S be the sample space

then, n(S) = number of ways of drawing 3 balls out of 10

$$= {}^{10}C_3$$

$$= \frac{10!}{3!\,7!} = \frac{10 \times \cancel{9}^{3} \times \cancel{8}^{4} \times 7!}{\cancel{3} \times \cancel{2} \times 7!} = 120$$

let E be the event of not getting a single red ball

i.e., event of getting 3 blue balls out of 6.

then

$$n(E) = {}^{6}C_3 = \frac{6!}{3! \times 3!}$$

$$= \frac{\cancel{6} \times 5 \times 4 \times \cancel{3!}}{\cancel{3} \times \cancel{2} \times \cancel{3!}} = 20$$

Hence the probability of that none of them is red is

$$P(E) = \frac{n(E)}{n(S)} = \frac{20}{120} = \frac{1}{6}$$

Q10. The probability that A speaks truth is 3/5 and that of B speaking truth is 4/7. What is the probability that they agree in stating the same fact?

Given:- probability of A speaks truth is $\frac{3}{5}$ and B speaks truth is $\frac{4}{7}$

Required to find:- probability that A and B are agreeing the same fact

Solution:- $P(A) = \frac{3}{5}$

$P(B) = \frac{4}{7}$

Since A and B are independent Events, so the Probability that A and B agree in stating the same fact is given by

$P(A \cap B)$

$= P(A) \times P(B)$

$= \frac{3}{5} \times \frac{4}{7}$ [Probability of both speaking truth]

$= \frac{12}{35}$

Probability of both speaking false $= [1 - P(A)] \times [1 - P(B)]$

$= [1 - \frac{3}{5}] \times [1 - \frac{4}{7}]$

$= \frac{2}{5} \times \frac{3}{7}$

$= \frac{6}{35}$

Probability both agree the same fact =

Probability of both Speaking truth + probability of both Speaking false

$= \frac{12}{35} + \frac{6}{35}$

$= \boxed{\frac{18}{35}}$

Answer:- Option A

Q11.A decision making system has two algorithms algorithm-I and II. Algorithm-I produces 60% of predictions and algorithm -II produces 40% of the predictions of the total output. Further 2% of the predictions produced by algorithm-I are incorrect whereas 4% produced by algorithm -II are incorrect. If a prediction is drawn at random what is the probability that it is incorrect?

Q12.Explain in detail any five applications of machine learning

Machine learning is a buzzword for today's technology, and it is growing very rapidly day by day. We are using machine learning in our daily life even without knowing it such as Google Maps, Google assistant, Alexa, etc. Below are some most trending real-world applications of Machine Learning:

1. Image Recognition:

Image recognition is one of the most common applications of machine learning. It is used to identify objects, persons, places, digital images, etc. The popular use case of image recognition and face detection is, **Automatic friend tagging suggestion**:

Facebook provides us a feature of auto friend tagging suggestion. Whenever we upload a photo with our Facebook friends, then we automatically get a tagging suggestion with name, and the technology behind this is machine learning's **face detection** and **recognition algorithm**.

It is based on the Facebook project named "**Deep Face**," which is responsible for face recognition and person identification in the picture.

2. Speech Recognition

While using Google, we get an option of "**Search by voice**," it comes under speech recognition, and it's a popular application of machine learning.

Speech recognition is a process of converting voice instructions into text, and it is also known as "**Speech to text**", or "**Computer speech recognition**." At present, machine learning algorithms are widely used by various applications of speech recognition. **Google assistant**, **Siri**, **Cortana**, and **Alexa** are using speech recognition technology to follow the voice instructions.

3. Traffic prediction:

If we want to visit a new place, we take help of Google Maps, which shows us the correct path with the shortest route and predicts the traffic conditions.

It predicts the traffic conditions such as whether traffic is cleared, slow-moving, or heavily congested with the help of two ways:

- **Real Time location** of the vehicle form Google Map app and sensors
- **Average time has taken** on past days at the same time.

Everyone who is using Google Map is helping this app to make it better. It takes information from the user and sends back to its database to improve the performance.

4. Product recommendations:

Machine learning is widely used by various e-commerce and entertainment companies such as **Amazon**, **Netflix**, etc., for product recommendation to the user. Whenever we search for some product on Amazon, then we started getting an advertisement for the same product while internet surfing on the same browser and this is because of machine learning.

Google understands the user interest using various machine learning algorithms and suggests the product as per customer interest.

As similar, when we use Netflix, we find some recommendations for entertainment series, movies, etc., and this is also done with the help of machine learning.

5. Self-driving cars:

One of the most exciting applications of machine learning is self-driving cars. Machine learning plays a significant role in self-driving cars. Tesla, the most popular car manufacturing company is working on self-driving car. It is using unsupervised learning method to train the car models to detect people and objects while driving.

13.Differentiate between Pearson's correlation and Spearman's correlation. Give examples illustrating the use of both the measures.

- Pearson correlation coefficient or Pearson's correlation coefficient or Pearson's r is defined in statistics as the measurement of the strength of the relationship between two variables and their association with each other.
  In simple words, Pearson's correlation coefficient calculates the effect of change in one variable when the other variable changes.

- For example: Up till a certain age, (in most cases) a child's height will keep increasing as his/her age increases. Of course, his/her growth depends upon various factors like genes, location, diet, lifestyle, etc.
- This approach is based on covariance and thus is the best method to measure the relationship between two variables.

The correlation coefficient formula finds out the relation between the variables. It returns the values between -1 and 1. Use the below Pearson coefficient correlation calculator to measure the strength of two variables.
Pearson correlation coefficient formula:

$$r = \frac{N\Sigma xy-(\Sigma x)(\Sigma y)}{\sqrt{[N\Sigma x^2-(\Sigma x)^2][N\Sigma y^2-(\Sigma y)^2]}}$$

Where:
N = the number of pairs of scores
Σxy = the sum of the products of paired scores
Σx = the sum of x scores
Σy = the sum of y scores
Σx2 = the sum of squared x scores
Σy2 = the sum of squared y scores

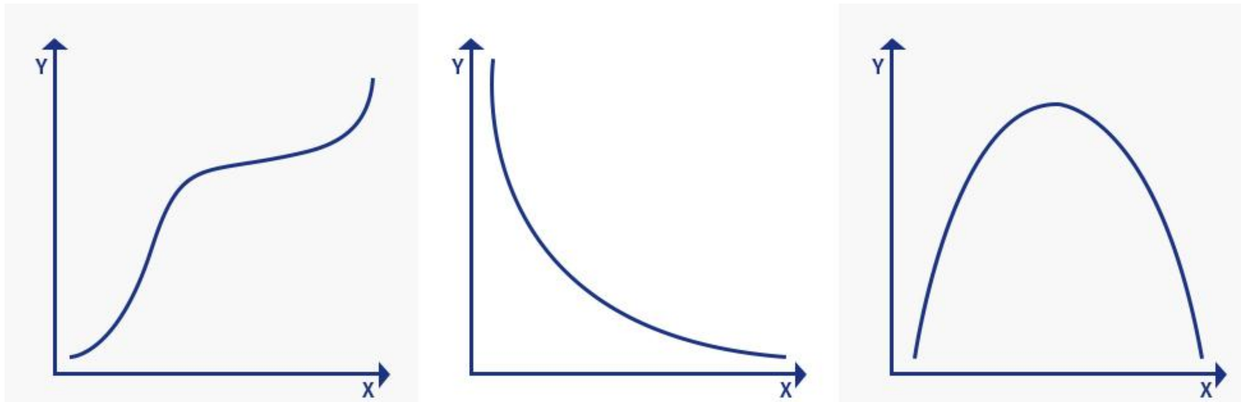
#

- The Spearman's rank coefficient of correlation is a nonparametric measure of rank correlation (statistical dependence of ranking between two variables).
- Named after Charles Spearman, it is often denoted by the Greek letter 'ρ' (rho) and is primarily used for data analysis.
- It measures the strength and direction of the association between two ranked variables. But before we talk about the Spearman correlation coefficient, it is important to understand Pearson's correlation first. A Pearson correlation is a statistical measure of the strength of a linear relationship between paired data.

For the calculation and significance testing of the ranking variable, it requires the following data assumption to hold true:

- Interval or ratio level
- Linearly related
- Bivariant distributed

If your data doesn't meet the above assumptions, then you would need Spearman's Coefficient. It is necessary to know what monotonic function is to understand Spearman correlation coefficient. A monotonic function is one that either never decreases or never increases as it is an independent variable increase. A monotonic function can be explained using the image below:



The image explains three concepts in monotonic function:

1. Monotonically increasing: When the ‘x’ variable increases and the ‘y’ variable never decreases.
2. Monotonically decreasing: When the ‘x’ variable increases but the ‘y’ variable never increases
3. Not monotonic: When the ‘x’ variable increases and the ‘y’ variable sometimes increases and sometimes decreases.

14.Explain in brief any five supervised learning models

=

1-microsoft”s Project InnerEye harnesses computer vision and machine learning to differentiate between tumors and healthy anatomy using 3D radiological images that assist medical experts in radiotherapy and surgical planning. With this AI-based approach, Microsoft aims to produce medicine that is tailored to the unique needs of each patient.

2-Tempus aims to make breakthroughs in cancer research by gathering massive amounts of medical and clinical data to deliver personalized treatments for patients. Analyzing its data library with AI-powered algorithms, Tempus helps with genomic profiling, clinical trial matching, diagnostic biomarking and academic research

3-Path ai's technology employs machine learning to help pathologists make quicker and more accurate diagnoses. The company also offers AI tools for compiling patient info, processing samples and streamlining other tasks for clinical trials and drug development. A partnership network of biopharma groups, labs and clinicians equips PathAI with the resources to provide more effective treatments for patients.

4- In India the game of cricket is not nearly a game but a religion which is followed and loved mustafas as there are multiple format in this game T20 format is most popular one of it is very difficult who is win the game until the last ball is bolde so we pounded why not develop a machine learning model winning a cricket match multiple factor impact the result of match the ground condition past performance records at the venue from of a particular player and team etc. this paper stresses on the key factor impact of the master and suggest regression model which gives the best prediction. Keywords: Naïve Bayes Classification, Euler's Strength Formula, Cricket Prediction, Supervised Learning, KNIME Tool, Cricket prediction, sports analytics, multivariate regression, neural network ,Ensemble technique.

5

To make the lives of diabetes patients more stress-free, Beta Bionics is developing a wearable “bionic” pancreas called iLet. This device — which is still in the investigational stage — constantly monitors blood sugar levels in patients with Type 1 diabetes, so patients don't have to shoulder the burden of tracking their blood glucose levels on a daily basis.

Q15.What is Naïve in Naïve Bayes supervised learning model?

Ans:- Naive Bayes is called naive because it assumes that each input variable is independent. This is a strong assumption and unrealistic for real data; however, the technique is very effective on a large range of complex problems. The thought behind naive Bayes classification is to try to classify the data by maximizing $P(O \mid C_i)P(C_i)$ using Bayes theorem of posterior probability (where O is the Object or tuple in a dataset and "i" is an index of the class).

Q16.Explain the terms loss function and cost function. Differentiate between both

Loss Function:- The loss function is a method of evaluating how well your machine learning algorithm models your featured data set. In other words, loss functions are a measurement of how good your model is in terms of predicting the expected outcome.

Cost Function:- Cost function measures the performance of a machine learning model for a data set. Cost function quantifies the error between predicted and expected values and presents that error in the form of a single real number. Depending on the problem, cost function can be formed in many different ways.

the loss function is to capture the difference between the actual and predicted values for a single record whereas cost functions aggregate the difference for the entire training dataset

Q17.Differentiate between parametric model and non-parametric model.

- In case of parametric models, the assumption related to the functional form is made and linear model is considered. In case of non-parametric models, the assumption about the functional form is not made.
- Parametric models are much easier to fit than non-parametric models because parametric machine learning models only require the estimation of a set of parameters as the model is identified prior as linear model. In case of non-parametric model, one needs to estimate some arbitrary function which is a much difficult task.
- Parametric models often do not match the unknown function we are trying to estimate. The model performance is comparatively lower than the non-parametric models. The estimates done by the parametric models will be farther from being true.

- <u>Parametric models</u> are interpretable unlike the <u>non-parametric models</u>. This essentially means that one can go for parametric models when the goal is to find inference. Instead, one can choose to go for non-parametric models when the goal is to make prediction with higher accuracy and interpretability or inference is not the key ask.

Q18.Define the cost function for logistic regression model?

=

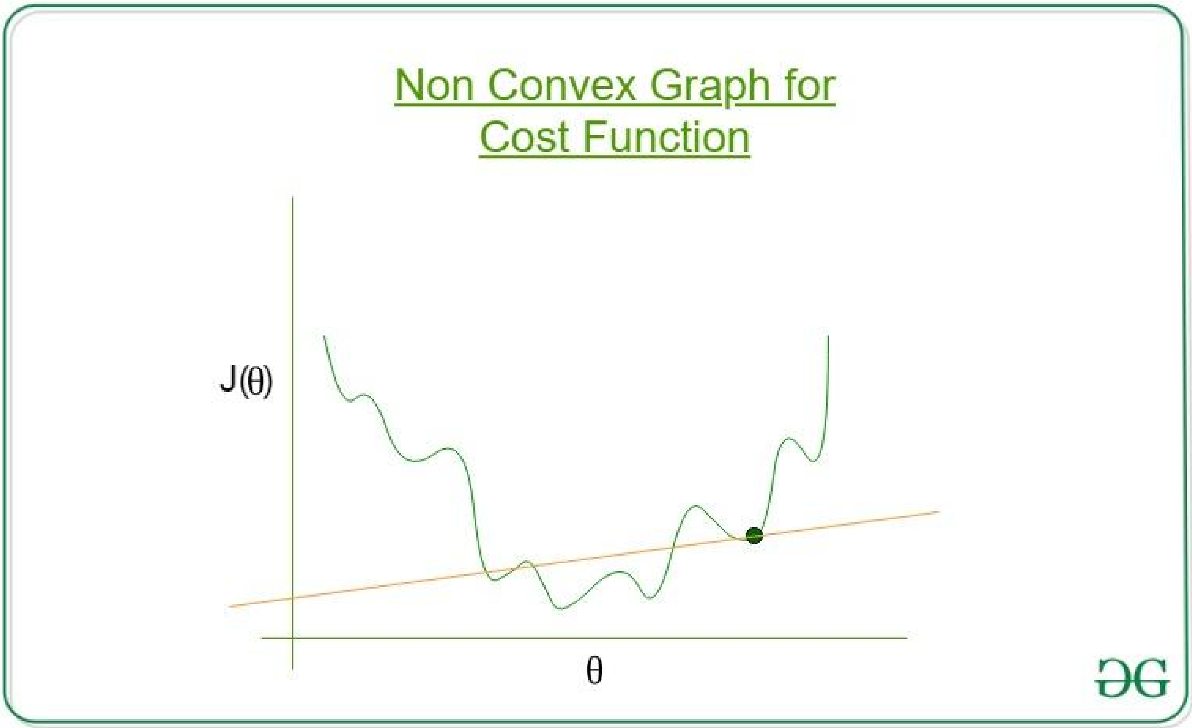
In the case of Linear Regression, the Cost function is –

$$J(\Theta) = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{2}[h_\Theta(x^{(i)}) - y^{(i)}]^2$$

But for Logistic Regression,

$$h_\Theta(x) = g(\Theta^T x)$$

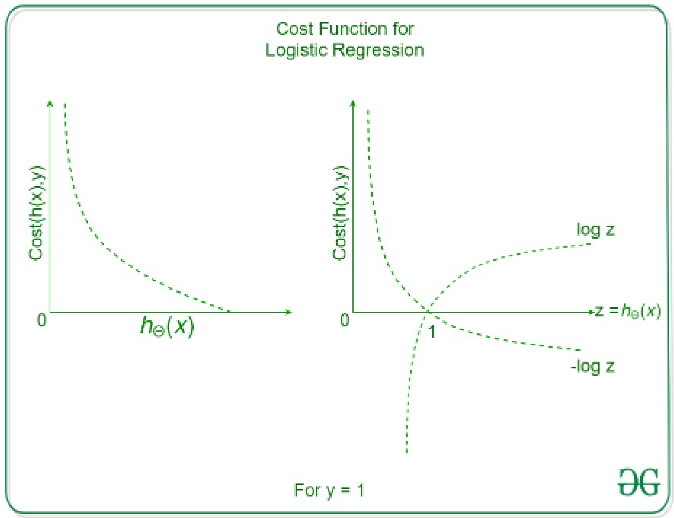
It will result in a non-convex cost function. But this results in cost function with local optima's which is a very big problem for Gradient Descent to compute the global optima.


Non Convex Graph for
Cost Function
J(θ)
θ

So, for Logistic Regression the cost function is

$$\mathrm{Cost}(\mathrm{h}_\Theta(x), y) = \begin{cases} -log(h_\Theta(x)) & if \quad y = 1 \\ -log(1 - h_\Theta(x)) & if \quad y = 0 \end{cases}$$

If y = 1



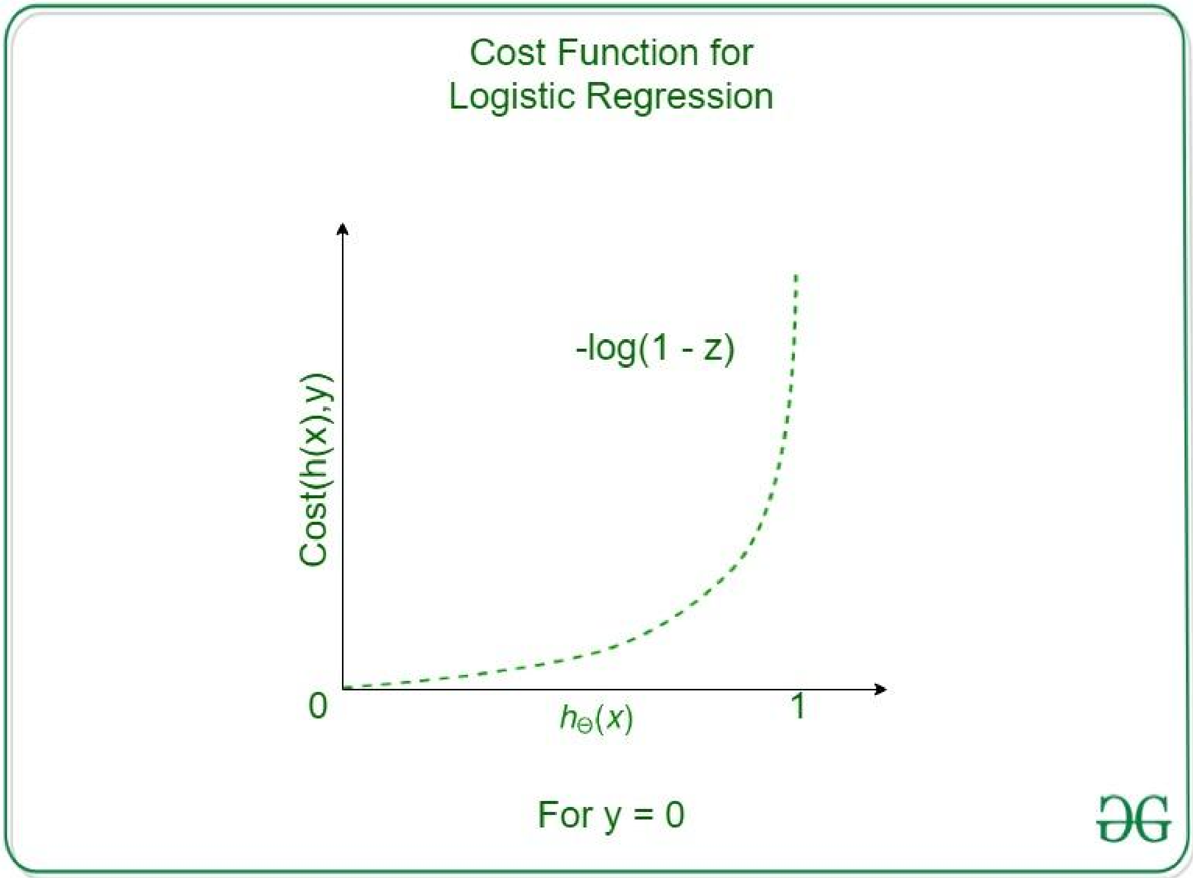
Cost = 0 if y = 1, hθ(x) = 1

But as,

hθ(x) -> 0

Cost -> Infinity

If y = 0



Cost Function for
Logistic Regression
-log(1 - z)
Cost(h(x),y)
0
$h_\Theta(x)$
1
For y = 0

So,

$$\text{Cost}(\text{h}_\Theta(x), y) = \begin{cases} 0 & if \quad h_\Theta(x) = y \\ \infty & if \quad y = 0 \quad and \quad h_\Theta(x) \rightarrow 1 \\ \infty & if \quad y = 1 \quad and \quad h_\Theta(x) \rightarrow 0 \end{cases}$$

$$\text{Cost}(\text{h}_\Theta(x), y) = -ylog(h_\Theta(x)) - (1-y)log(1-h_\Theta(x))$$

$$\text{J}(\Theta) = \frac{-1}{m} \sum_{i=1}^{m} Cost(h_\Theta(x), y)$$

To fit parameter **θ**, J(θ) has to be minimized and for that Gradient Descent is required.

Gradient Descent – Looks similar to that of Linear Regression but the difference lies in the hypothesis $h_\theta(x)$

$$\Theta_j := \Theta_j - \alpha \sum_{i=1}^{m} (h_\Theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

Q19.With respect to gradient descent define the following terms

a. Stochastic gradient descent:- Stochastic gradient descent is an optimization algorithm often used in machine learning applications to find the model parameters that correspond to the best fit between predicted and actual outputs. It's an inexact but powerful technique. Stochastic gradient descent is widely used in machine learning applications.

b. Batch gradient descent :-In Batch Gradient Descent, all the training data is taken into consideration to take a single step. We take the average of the gradients of all the training examples and then use that mean gradient to update our parameters. So that's just one step of gradient descent in one epoch.

Batch Gradient Descent is great for convex or relatively smooth error manifolds. In this case, we

move somewhat directly towards an optimum solution.

c. Mini batch gradient descent:- We use a batch of a fixed number of training examples which is less than the actual dataset and call it a mini-batch. Doing this helps us achieve the advantages of both the former variants we saw. So, after creating the mini-batches of fixed size, we do the following steps in one epoch:

1. Pick a mini-batch
2. Feed it to Neural Network
3. Calculate the mean gradient of the mini-batch
4. Use the mean gradient we calculated in step 3 to update the weights

5. Repeat steps 1–4 for the mini-batches we created

Q20. What is regularization? Explain any two regularization techniques.

Ans:- **Regularization**:- Regularization refers to a set of different techniques that lower the complexity of a neural network model during training, and thus prevent the overfitting.

L2 & L1 regularization

L1 and L2 are the most common types of regularization. These update the general cost function by adding another term known as the regularization term.

Cost function = Loss (say, binary cross entropy) + Regularization term

Due to the addition of this regularization term, the values of weight matrices decrease because it assumes that a neural network with smaller weight matrices leads to simpler models. Therefore, it will also reduce overfitting to quite an extent.

However, this regularization term differs in L1 and L2.

In L2, we have:

$$Cost\,function = Loss + \frac{\lambda}{2m} * \sum \|w\|^2$$

Here, lambda is the regularization parameter. It is the hyperparameter whose value is optimized for better results. L2 regularization is also known as *weight decay* as it forces the weights to decay towards zero (but not exactly zero).

In L1, we have:

$$Cost\,function = Loss + \frac{\lambda}{2m} * \sum \|w\|$$

In this, we penalize the absolute value of the weights. Unlike L2, the weights may be reduced to zero here. Hence, it is very useful when we are trying to compress our model. Otherwise, we usually prefer L2 over it.