Q1.1

# What is the difference between static and dynamic variables in Python?

Static variables are class variables shared among all instances, while dynamic variables are instance variables specific to an object. Example: Static variable - `Class.var`, Dynamic variable - `self.var`. Example:   
class MyClass:  
 static\_var = 0  
 def \_\_init\_\_(self, var):  
 self.dynamic\_var = var

# Explain the purpose of 'pop', 'popitem', 'clear()' in a dictionary with suitable examples.

'pop' removes a specified key and returns its value, 'popitem' removes and returns an arbitrary key-value pair, and 'clear()' empties the dictionary. Example:   
d = {'a': 1, 'b': 2, 'c': 3}  
value = d.pop('a')  
key, value = d.popitem()  
d.clear()

# What do you mean by FrozenSet? Explain it with suitable examples.

A FrozenSet is an immutable version of a set in Python. Example:   
fs = frozenset([1, 2, 3])

# Differentiate between mutable and immutable data types in Python and give examples of mutable and immutable data types.

Mutable data types can be changed after creation, while immutable data types cannot. Example: Mutable - `list`, Immutable - `tuple`. Example:   
mutable\_list = [1, 2, 3]  
immutable\_tuple = (1, 2, 3)

# What is \_\_init\_\_? Explain with an example.

\_\_init\_\_ is a special method in Python classes for initializing new objects. Example:   
class Car:  
 def \_\_init\_\_(self, colour):  
 self.colour = colour

# What is docstring in Python? Explain with an example.

A docstring is a string that occurs as the first statement in a module, function, class, or method definition. Example:   
def func():  
 '''This is a docstring.'''  
 pass

# What are unit tests in Python?

Unit tests are used to check the correctness of individual parts of a program. Example: Using `unittest` module to write test cases. Example:   
import unittest  
class TestStringMethods(unittest.TestCase):  
 def test\_upper(self):  
 self.assertEqual('foo'.upper(), 'FOO')

# What is break, continue and pass in Python?

Break exits the loop, continue skips to the next iteration, and pass does nothing. Example:   
for i in range(10):  
 if i == 5:  
 break  
 elif i == 3:  
 continue  
 else:  
 pass

# What is the use of self in Python?

`self` refers to the instance of the class and is used to access variables and methods associated with the instance. Example:   
class Shirt:  
 def \_\_init\_\_(self, size):  
 self.size = size

# What are global, protected and private attributes in Python?

Global attributes are accessible everywhere, protected attributes are intended for use within the class and its subclasses, and private attributes are intended for use only within the class itself. Example:   
class MyClass:  
 public\_var = 1  
 \_protected\_var = 2  
 \_\_private\_var = 3

# What are modules and packages in Python?

A module is a single file containing Python code, while a package is a collection of modules. Example:   
module - math, package - numpy

# What are lists and tuples? What is the key difference between the two?

Lists are mutable sequences, while tuples are immutable sequences. Example:   
list\_example = [1, 2, 3], tuple\_example = (1, 2, 3)

# What is an Interpreted language & dynamically typed language? Write 5 differences between them.

An interpreted language executes code line-by-line, while a dynamically typed language determines data types at runtime. Differences include:   
(1) Compilation, (2) Execution, (3) Type Checking, (4) Performance, (5) Flexibility. Example: Python is both interpreted and dynamically typed.

# What are Dict and List comprehensions?

Dictionary comprehensions create dictionaries, and list comprehensions create lists in a concise way. Example:   
{key: value for key, value in iterable}, [x for x in iterable]. Example:   
dict\_comprehension = {i: i\*\*2 for i in range(10)}  
list\_comprehension = [i\*\*2 for i in range(10)]

# What are decorators in Python? Explain it with an example. Write down its use cases.

Decorators modify the behavior of a function or method. Example:   
@decorator  
def func(): pass. Use cases include logging, authentication, and memoization. Example:   
from functools import wraps  
def my\_decorator(f):  
 @wraps(f)  
 def wrapper(\*args, \*\*kwargs):  
 print('Something is happening before the function is called.')  
 result = f(\*args, \*\*kwargs)  
 print('Something is happening after the function is called.')  
 return result  
 return wrapper

# How is memory managed in Python?

Memory management in Python is handled by the Python Memory Manager and involves reference counting and garbage collection. Example:   
import gc; gc.collect()

# What is lambda in Python? Why is it used?

Lambda is an anonymous function defined with the lambda keyword. Example:   
divide = lambda x, y: x / y

# Explain split() and join() functions in Python?

`split()` divides a string into a list, and `join()` combines a list into a string. Example:   
'Hello World'.split(), ' '.join(['Hello', 'World'])

# What are iterators, iterable & generators in Python?

Iterators are objects that allow traversal through all the elements, iterable is any object capable of returning its members one at a time, and generators yield items one at a time. Example:   
iter([1, 2, 3]), (x for x in range(3))

# What is the difference between xrange and range in Python?

`range` returns a list, while `xrange` returns an xrange object (Python 2). Example:   
range(10), xrange(10)

# Pillars of OOPs.

The pillars of OOPs are Encapsulation, Abstraction, Inheritance, and Polymorphism. Example:   
Class definitions demonstrating these principles

# How will you check if a class is a child of another class?

Use the `issubclass()` function to check if a class is derived from another class. Example:   
issubclass(DerivedClass, BaseClass)

# How does inheritance work in Python? Explain all types of inheritance with an example.

Inheritance in Python allows a class (called the child or subclass) to inherit attributes and methods from another class (called the parent or superclass), promoting code reusability and organization. Python supports several types of inheritance: single inheritance, where a subclass inherits from one superclass; multiple inheritance, where a subclass inherits from more than one superclass; multilevel inheritance, where a subclass inherits from another subclass forming a chain; hierarchical inheritance, where multiple subclasses inherit from one superclass; and hybrid inheritance, a mix of two or more types of inheritance. For example, in single inheritance:

class Animal:

def speak(self):

return "Animal speaks"

class Dog(Animal):

def bark(self):

return "Dog barks"

dog = Dog()

print(dog.speak()) # Animal speaks

print(dog.bark()) # Dog barks

# What is encapsulation? Explain it with an example.

Encapsulation is a fundamental principle of object-oriented programming that involves bundling the data (attributes) and methods (functions) that operate on the data into a single unit, usually a class, and restricting direct access to some of the object's components for better control and protection; for example, in a class `Car`, the internal workings of the engine can be hidden from the user while providing methods like `start()` and `stop()` to interact with it.

# What is polymorphism? Explain it with an example.

Polymorphism allows different classes to be treated as instances of the same class through inheritance and method overriding. Example:   
class Animal:

def speak(self):

pass

class Cat(Animal):

def speak(self):

return 'meow'

class Dog(Animal):

def speak(self):

return 'bark'

Q1.2

Which of the following identifier names are invalid and why?

a) Serial\_no.

b) 1st\_Room

c) Hundred$

d) Total\_Marks

e) total-Marks

f) Total Marks

g) True

h) \_Percentag

1st\_Room - Invalid. Identifiers cannot start with a digit.

total-Marks - Invalid. Hyphens are not allowed in identifiers.

Total Marks - Invalid. Spaces are not allowed in identifiers.

Q20

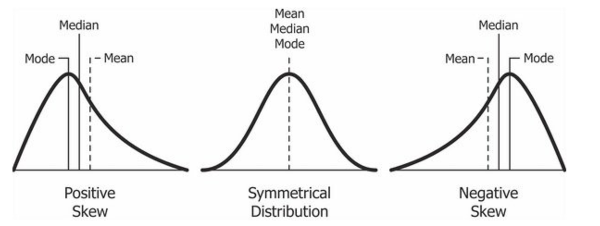
What do you mean by Measure of Central Tendency and Measures of Dispersion. How it can be calculated.

Measures of Central Tendency refer to statistical metrics that identify the center or typical value in a dataset, which includes the mean (average), median (middle value), and mode (most frequent value). Measures of Dispersion, on the other hand, describe the spread or variability within a dataset. Key measures of dispersion include range (difference between the maximum and minimum values), variance (average squared deviation from the mean), and standard deviation (square root of the variance)

Q21

What do you mean by skewness. Explain its types. Use graph to show.

Skewness is a measure of the asymmetry of the probability distribution of a real-valued random variable about its mean. A distribution can be positively skewed (right-skewed), where the tail on the right side of the distribution is longer or fatter than the left side, indicating that a larger number of data points fall on the left side of the mean. Conversely, a distribution can be negatively skewed (left-skewed), where the tail on the left side is longer or fatter than the right side, indicating that more data points are concentrated on the right side of the mean. Symmetrical distributions, such as the normal distribution, have a skewness of zero.



Q22

Explain PROBABILITY MASS FUNCTION (PMF) and PROBABILITY DENSITY FUNCTION (PDF). and what is the difference between them?

A Probability Mass Function (PMF) gives the probability that a discrete random variable is exactly equal to some value. A Probability Density Function (PDF) gives the relative likelihood for a continuous random variable to take on a given value. The main difference is that PMF applies to discrete random variables, whereas PDF applies to continuous random variables, and probabilities for PDFs are found over intervals rather than at specific points.

Q23

What is correlation. Explain its type in details. What are the methods of determining correlation

Correlation measures the strength and direction of the relationship between two variables. There are three main types: positive correlation (both variables increase together), negative correlation (one variable increases as the other decreases), and no correlation (no discernible relationship). Methods to determine correlation include Pearson's correlation coefficient, Spearman's rank correlation, and Kendall's tau coefficient.

Q25

Discuss the 4 Differences Between Correlation and Regression

Correlation measures the strength and direction of a linear relationship between two variables, while regression quantifies the relationship and provides a predictive equation.

Correlation is symmetric (correlation between X and Y is the same as between Y and X, whereas regression is asymmetric, focusing on predicting the dependent variable from the independent variable.

Correlation assesses the degree of association between variables, whereas regression aims to model the relationship and make predictions.

Correlation indicates association but not causation, while regression can help infer causation if other influencing factors are accounted for.

Q26

Find the most likely price at Delhi corresponding to the price of Rs. 70 at Agra from the following data: Coefficient of correlation between the prices of the two places +0.8.

Direct calculation isn't possible without the means and standard deviations of the prices in both locations.

Q28

What is Normal Distribution? What are the four Assumptions of Normal Distribution? Explain in detail.

A normal distribution is a continuous probability distribution characterized by a symmetrical, bell-shaped curve where most observations are around the central peak, and probabilities for values taper off equally in both directions from the mean.

Four assumptions of normal distribution: (1) **Normality**: The data follows a normal distribution. (2) **Independence**: Observations are independent of each other. (3) **Homoscedasticity**: The variance of errors is constant across all levels of the independent variable. (4) **Linearity**: The relationship between the dependent and independent variables is linear. These assumptions are critical for conducting parametric statistical tests that rely on normal distribution properties.

Q29

Write all the characteristics or Properties of the Normal Distribution Curve.

The normal distribution curve is symmetric around the mean, indicating that data near the mean are more frequent in occurrence than data far from the mean. It is unimodal, meaning it has a single peak. The mean, median, and mode of a normal distribution are equal. The area under the curve represents the total probability and is equal to 1. Approximately 68% of the data falls within one standard deviation of the mean, 95% within two standard deviations, and 99.7% within three standard deviations. The tails of the normal distribution asymptotically approach the horizontal axis but never touch it.

Q30

Which of the following options are correct about Normal Distribution Curve.

(a) Within a range 0.6745 of σ on both sides the middle 50% of the observations occur i,e. mean ±0.6745σ

covers 50% area 25% on each side.

(b) Mean ±1S.D. (i,e.μ ± 1σ) covers 68.268% area, 34.134 % area lies on either side of the mean.

(c) Mean ±2S.D. (i,e. μ ± 2σ) covers 95.45% area, 47.725% area lies on either side of the mean.

(d) Mean ±3 S.D. (i,e. μ ±3σ) covers 99.73% area, 49.856% area lies on the either side of the mean.

(e) Only 0.27% area is outside the range μ ±3σ.

The correct options are (b), (c), (d), and (e).

Q31

The mean of a distribution is 60 with a standard deviation of 10. Assuming that the distribution is normal, what percentage of items be (i) between 60 and 72, (ii) between 50 and 60, (iii) beyond 72 and (iv) between 70 and 80?

Use Zscore = (X-mean)/std

1. Z= (72-60)/10 = 1.2, area to left of 1.2 = 0.8849

Percentage between 60 and 72 = (0.8849-0.5) x 100=38.49%

1. Z= (50-60)/10 = -1, area to left of -1 = 0.1587

Percentage between 50 and 60 = (0.5-0.1587) x 100=34.13%

1. Z= (72-60)/10 = 1.2, area to left of 1.2 = 0.8849

Percentage beyond 72 = (1-0.8849) x 100=11.51%

1. Z\_70= (70-60)/10 = 1, area to left of 1 = 0.8413

Z\_80= (80-60)/10 = 2, area to left of 2 = 0.9772

Percentage between 70 and 80 = (0.9772-0.8413) x 100=13.59%

Q32

15000 students sat for an examination. The mean marks was 49 and the distribution of marks had a standard deviation of 6. Assuming that the marks were normally distributed what proportion of students scored (a) more than 55 marks, (b) more than 70 marks

Use Zscore = (X-mean)/std

1. Z= (55-49)/6 = 1, area to left of 1 = 0.8413

Proportion beyond 55= 1-0.8413 = 0.1587

1. Z= (70-49)/6 = 3.5, area to left of 3.5 = 0.9998

Proportion beyond 70= 1-0.9998 = 0.0002

Q33

If the height of 500 students are normally distributed with mean 65 inch and standard deviation 5 inch. How many students have height : a) greater than 70 inch. b) between 60 and 70 inch.

Use Zscore = (X-mean)/std

1. Z= (70-65)/5 = 1, area to left of 1 = 0.8413

Beyond 70= 1-0.8413 = 0.1587

Number of students = 0.1587 x 500 = 79 students

1. Z\_60= (60-65)/5 = -1, area to left of -1 = 0.1587

Z\_70= (70-65)/5 = 1, area to left of 1 = 0.8413

Between 60 and 70 = 0.8413-0.1587 = 0.6826

Number of students = 0.6828 x 500 = 341 students

Q34

What is the statistical hypothesis? Explain the errors in hypothesis testing. b)Explain the Sample. What are Large Samples & Small Samples?

A statistical hypothesis is a specific, testable statement about a population parameter based on sample data. In hypothesis testing, errors can occur in two main forms: Type I and Type II errors. A Type I error occurs when the null hypothesis (H0) is incorrectly rejected when it is actually true, representing a false positive. A Type II error occurs when the null hypothesis is not rejected when it is actually false, representing a false negative. The significance level (alpha) controls the probability of making a Type I error, while the power of the test (1-beta) relates to the probability of avoiding a Type II error.

A sample is a subset of a population used to make inferences about the whole population. Large samples typically consist of 30 or more observations, providing more reliable and generalizable results due to the central limit theorem, which states that the sampling distribution of the sample mean will be approximately normally distributed regardless of the population's distribution. Small samples, on the other hand, consist of fewer than 30 observations and require more stringent statistical methods, such as t-distributions, to account for the greater variability and potential bias in the estimates.

Q35

A random sample of size 25 from a population gives the sample standard derivation to be 9.0. Test the hypothesis that the population standard derivation is 10.5. Hint (Use chi-square distribution).

Null hypothesis (H0) The population standard deviation is 10.5.

Alternative hypothesis (Ha​): The population standard deviation is not 10.5.

Chi statistic = ((n-1)(sample\_std)^2)/population\_std^2

Chi statistic = ((25-1)(9.0)^2)/10.5^2 = 17.63

DoF = 25-1 = 24

For alpha = 0.05, criticals = for 0.025: 13.848, for 0.975: 36.415

17.63 is within the range [13.848,36.415]

So, fail to reject H0, there is not enough evidence to conclude that the population standard deviation is different from 10.5.

Q37

100 students of a PW IOI obtained the following grades in Data Science paper :

Grade :[A, B, C, D, E]

Total Frequency :[15, 17, 30, 22, 16, 100]

Using the χ 2 test , examine the hypothesis that the distribution of grades is uniform.

Null Hypothesis (H0​): The grades are uniformly distributed.

Alternative Hypothesis (Ha​): The grades are not uniformly distributed.

Expected frequency = Total frequency/Number of grades = 100/5 = 20

Chi statistic = summation((O-E)^2/E) = 7.7 after calculation using formula

DoF = 5-1 = 4, alpha =0.05

Critical value for α=0.05 and DoF = 4 is 9.488

(7.7) is less than the critical value (9.488). Therefore, we fail to reject the null hypothesis.

There is not enough evidence to reject the hypothesis that the grades are uniformly distributed.

Q40

Explain how to set up a Flask application to handle form submissions using POST requests.

Define a route that accepts POST requests (`@app.route('/submit', methods=['POST'])`) and retrieve form data using `request.form` within that route function to process and handle the submitted data.

Q42

How can you implement user authentication in a Flask application?

Use extensions like Flask-Login or Flask-JWT. Authenticate users with credentials stored in a database or using OAuth providers.

Q43

Describe the process of connecting a Flask app to a SQLite database using SQLAlchemy.

First initialize SQLAlchemy in the Flask application by creating an instance of `SQLAlchemy`. Then configure the database URI, in the case of SQLite, is a file path. Finally, define models using SQLAlchemy's ORM (Object-Relational Mapping) to interact with the database within your Flask routes and views.

Q44

How would you create a RESTful API endpoint in Flask that returns JSON data?

Define a route using @app.route decorator with a specific URL and HTTP method (e.g., GET, POST). Inside the route function, use Flask's jsonify function to convert your Python data (like a dictionary or list) into JSON format and return it as the response.

Q45

Explain how to use Flask-WTF to create and validate forms in a Flask application.

Define a form class using Flask-WTF's FlaskForm. Inside this class, define form fields using wtforms components such as StringField, BooleanField, etc. Then, in the Flask route, instantiate the form class, pass it to the template, and handle form submission by checking form.validate\_on\_submit() to trigger validation and processing of form data.

Q46

How can you implement file uploads in a Flask application?

Use the `request.files` object to handle incoming file data. Define a route that accepts POST requests and processes the uploaded files using Flask's file handling capabilities.

Q47

Describe the steps to create a Flask blueprint and why you might use one.

Define routes and views in a separate Python file, register the blueprint with your Flask application using app.register\_blueprint(), and then use the blueprint's routes within the application. Blueprints are useful for organizing routes and views into modular components, making the Flask application more maintainable and scalable by separating different parts of functionality into reusable modules.

Q48

How would you deploy a Flask application to a production server using Gunicorn and Nginx?

Configure Gunicorn to serve the Flask app by specifying the app and binding to a socket. Then, set up Nginx as a reverse proxy to forward requests to Gunicorn. Configure Nginx to serve static files and pass other requests to Gunicorn for processing.

Q50

* What is the difference between Series & Dataframes

Series are one-dimensional labeled arrays like a single column in a spreadsheet, while DataFrames are two-dimensional tables with rows and columns that can hold different data types.

* Difference between loc and iloc

In Pandas, ‘.loc’ selects data by labels (like row names) while ‘.iloc’ selects by integer position.

* What is the difference between supervised and unsupervised learning?

Supervised learning trains with labeled data to make predictions, while unsupervised learning finds hidden patterns in unlabeled data.

* Explain the bias-variance tradeoff

The bias-variance tradeoff in machine learning is the balancing act between a model's ability to learn the general trend (bias) and its sensitivity to specific training data (variance).

* What are precision and recall? How are they different from accuracy?

Precision measures the proportion of true positive results among all positive results predicted by the model, while recall measures the proportion of true positive results among all actual positive cases. Accuracy measures the proportion of all correct predictions (both true positives and true negatives) among the total number of predictions.

* What is overfitting and how can it be prevented?

Overfitting occurs when a machine learning model memorizes the training data too well, losing its ability to generalize to unseen data. To prevent this, we can use techniques like regularization or stopping training before the model memorizes noise.

* Explain the concept of cross-validation.

Cross-validation is a technique to evaluate the performance of a model by partitioning the data into subsets, training the model on some subsets, and validating it on the remaining subsets.

* What is the difference between a classification and a regression problem?

Classification predicts discrete labels (e.g., cat vs dog), whereas regression predicts continuous values (e.g., house prices).

* Explain the concept of ensemble learning.

Ensemble learning is a method where multiple models (learners) are combined to solve the same problem, often improving overall performance.

* What is gradient descent and how does it work?

Gradient descent is an optimization algorithm used to minimize the cost function by iteratively adjusting the parameters in the direction of the steepest descent.

* Describe the difference between batch gradient descent and stochastic gradient descent.

Batch gradient descent uses the entire dataset to compute gradients, while stochastic gradient descent uses one sample at a time.

* What is the curse of dimensionality in machine learning?

The curse of dimensionality is when there are too many features in the dataset which results in model training becoming computationally expensive and the model interpretation to become difficult.

* Explain the difference between L1 and L2 regularization.

L1 regularization adds the absolute value of the magnitude of coefficients as a penalty term to the loss function, while L2 regularization adds the squared magnitude of coefficients.

* What is a confusion matrix and how is it used?

A confusion matrix is a table used to evaluate the performance of a classification model by comparing actual vs predicted classifications.

* Define AUC-ROC curve.

The AUC-ROC curve is a performance measurement for classification problems at various threshold settings, plotting true positive rate against false positive rate.

* Explain the k-nearest neighbors algorithm.

The k-nearest neighbors algorithm classifies a data point based on the majority class of its k-nearest neighbors in the feature space.

* Explain the basic concept of a Support Vector Machine (SVM).

An SVM is a supervised learning algorithm used for classification and regression tasks, which finds the optimal hyperplane that maximizes the margin between different classes.

* How does the kernel trick work in SVM?

The kernel trick allows SVMs to efficiently perform non-linear classification by implicitly mapping the input features into high-dimensional spaces.

* What are the different types of kernels used in SVM and when would you use each?

Kernels include linear, polynomial, and radial basis function (RBF); linear is used for linearly separable data, polynomial for polynomial relationships, and RBF for complex relationships.

* What is the hyperplane in SVM and how is it determined?

The hyperplane is the decision boundary that separates different classes; it is determined by maximizing the margin between the closest points of the classes (support vectors).

* What are the pros and cons of using a Support Vector Machine (SVM)?

Pros include effective high-dimensional space handling and robust performance with clear margin of separation; cons include inefficiency with large datasets and sensitivity to the choice of kernel.

* Explain the difference between a hard margin and a soft margin SVM.

A hard margin SVM requires that all data points are correctly classified with no violations, whereas a soft margin SVM allows some misclassifications to handle noisy data better.

* Describe the process of constructing a decision tree.

Constructing a decision tree involves recursively splitting the dataset into subsets based on feature values that result in the maximum information gain.

* Describe the working principle of a decision tree.

A decision tree makes predictions by following the path from the root to a leaf node, where each node represents a feature test and each leaf represents an outcome.

* What is information gain and how is it used in decision trees?

Information gain measures the reduction in entropy or impurity after a dataset is split based on a feature; it is used to determine the best feature to split on.

* Explain Gini impurity and its role in decision trees.

Gini impurity measures the frequency at which a randomly chosen element would be incorrectly labeled; it is used to evaluate splits in the construction of decision trees.

* What are the advantages and disadvantages of decision trees?

Advantages include interpretability and ease of visualization; disadvantages include overfitting and sensitivity to small changes in data.

* How do random forests improve upon decision trees?

Random forests improve upon decision trees by aggregating multiple decision trees trained on different parts of the data, reducing overfitting and increasing generalization.

* How does a random forest algorithm work?

A random forest algorithm works by constructing multiple decision trees during training and outputting the mode or mean prediction of the individual trees.

* What is bootstrapping in the context of random forests?

Bootstrapping involves sampling the dataset with replacement to create multiple subsets for training different decision trees in a random forest.

* Explain the concept of feature importance in random forests.

Feature importance in random forests is a measure of the contribution of each feature to the prediction accuracy, often determined by the reduction in impurity.

* What are the key hyperparameters of a random forest and how do they affect the model?

Key hyperparameters include the number of trees, maximum depth, and minimum samples per leaf; they affect the model's complexity, performance, and risk of overfitting.

* Describe the logistic regression model and its assumptions.

Logistic regression models the probability of a binary outcome using a logistic function, assuming a linear relationship between the input features and the log odds of the outcome.

* How does logistic regression handle binary classification problems?

Logistic regression predicts the probability of the positive class and applies a threshold (usually 0.5) to decide the class label.

* What is the sigmoid function and how is it used in logistic regression?

The sigmoid function maps any real-valued number to a value between 0 and 1, used in logistic regression to model probabilities.

* Explain the concept of the cost function in logistic regression.

The cost function in logistic regression, often the binary cross-entropy loss, measures the difference between the predicted probabilities and the actual labels.

* How can logistic regression be extended to handle multiclass classification?

Logistic regression can handle multiclass classification using strategies like one-vs-rest (OvR) or softmax regression.

* What is the difference between L1 and L2 regularization in logistic regression?

L1 regularization adds the absolute value of the coefficients as a penalty term, leading to sparse models, while L2 regularization adds the squared value of the coefficients.

* What is XGBoost and how does it differ from other boosting algorithms?

XGBoost is an optimized gradient boosting algorithm known for its speed and performance, incorporating regularization to reduce overfitting.

* Explain the concept of boosting in the context of ensemble learning.

Boosting is an ensemble technique that sequentially trains weak learners, each focusing on the errors of the previous one, to improve overall performance.

* How does XGBoost handle missing values?

XGBoost handles missing values by learning the best imputation direction for each feature during training.

* What are the key hyperparameters in XGBoost and how do they affect model performance?

Key hyperparameters include learning rate, number of trees, and maximum depth, affecting the model's learning speed, complexity, and risk of overfitting.

* Describe the process of boosting in XGBoost.

Boosting in XGBoost involves building trees sequentially, where each tree tries to correct the residual errors of the previous trees, using a gradient descent-like procedure.

* What are the advantages and disadvantages of using XGBoost?

Advantages include high performance, speed, and ability to handle missing data; disadvantages include complexity in tuning hyperparameters and risk of overfitting if not properly regularized.