**Machine Learning lab Viva Questions**

**Experiment 1: Histograms & Box Plots (California Housing Dataset)**

1. **What is a histogram?**  
   A histogram is a graphical representation of the distribution of numerical data, using bars to show frequency counts.
2. **Why do we use histograms in data analysis?**  
   Histograms help understand the shape, spread, and central tendency of a dataset.
3. **How do you choose the number of bins in a histogram?**  
   The number of bins is typically chosen using rules like Sturgis' rule or the square root rule.
4. **What is a box plot?**  
   A box plot summarizes a dataset using the median, quartiles, and possible outliers.
5. **What information does a box plot provide?**  
   A box plot shows the central tendency, spread, and presence of outliers in the data.
6. **What are whiskers in a box plot?**  
   Whiskers extend to the minimum and maximum values within 1.5 times the IQR from the quartiles.
7. **How can we detect outliers using a box plot?**  
   Data points beyond the whiskers are considered potential outliers.
8. **What is skewness in a histogram?**  
   Skewness indicates whether a distribution is asymmetrical (left or right skewed).
9. **What does a symmetric histogram suggest?**  
   A symmetric histogram suggests that data is normally distributed.
10. **What is kurtosis?**  
    Kurtosis measures the heaviness of the tails in a distribution.
11. **What does a multimodal histogram indicate?**  
    It suggests multiple peaks, meaning the data may contain different subgroups.
12. **Why is the median preferred over the mean in a box plot?**  
    The median is less affected by extreme values or outliers.
13. **How do outliers affect data analysis?**  
    Outliers can distort statistical measures and affect machine learning models.
14. **How can we handle outliers?**  
    Methods include trimming, capping, and using transformations.
15. **What is the interquartile range (IQR)?**  
    IQR is the range between Q1 (25th percentile) and Q3 (75th percentile).
16. **How do you interpret a histogram with a long tail?**  
    A long tail suggests skewness, meaning the data contains extreme values.
17. **What are the advantages of box plots over histograms?**  
    Box plots provide a summary of distribution and highlight outliers.
18. **Can a histogram show categorical data?**  
    No, histograms are used only for numerical data.
19. **What happens when the bin width is too large in a histogram?**  
    It may hide important details in the distribution.
20. **What are alternatives to histograms?**  
    Density plots and violin plots are alternatives.

**Experiment 2: Correlation Matrix & Pair Plot (California Housing Dataset)**

1. **What is a correlation matrix?**  
   A correlation matrix is a table showing correlation coefficients between multiple variables.
2. **Why do we compute a correlation matrix?**  
   It helps in understanding relationships between numerical features.
3. **What is the range of correlation coefficients?**  
   Correlation values range from -1 (strong negative) to +1 (strong positive).
4. **What does a correlation value of 0 indicate?**  
   It suggests no linear relationship between the variables.
5. **What is the difference between positive and negative correlation?**
   * Positive correlation: As one variable increases, the other also increases.
   * Negative correlation: As one variable increases, the other decreases.
6. **How do we interpret a heatmap of a correlation matrix?**  
   Darker/lighter shades indicate stronger/weaker correlations.
7. **What are some real-world applications of correlation matrices?**  
   Used in finance, healthcare, and machine learning feature selection.
8. **What is a pair plot?**  
   A pair plot visualizes pairwise relationships between numerical features.
9. **How does a pair plot help in data analysis?**  
   It reveals relationships, trends, and outliers.
10. **What is multicollinearity?**  
    When two or more variables are highly correlated, causing redundancy in models.
11. **How do we deal with multicollinearity?**  
    By removing one of the correlated features or using PCA.
12. **Why is correlation not causation?**  
    A high correlation does not imply a cause-and-effect relationship.
13. **What is an example of spurious correlation?**  
    Ice cream sales and drowning rates both increase in summer but are unrelated.
14. **Can we compute correlation for categorical data?**  
    No, correlation is only for numerical data.
15. **How do missing values affect correlation?**  
    They can lead to incorrect or undefined correlation values.
16. **What is Pearson’s correlation coefficient?**  
    It measures the linear relationship between two continuous variables.
17. **What is Spearman’s correlation?**  
    It measures the rank correlation, useful for non-linear relationships.
18. **How does outlier presence affect correlation?**  
    Outliers can distort correlation values.
19. **What is the best way to visualize correlations?**  
    Using heatmaps and scatter plots.
20. **How does scaling affect correlation?**  
    Correlation remains unaffected by changes in scale.

**Experiment 3: PCA (Iris Dataset)**

1. **What is Principal Component Analysis (PCA)?**  
   PCA is a dimensionality reduction technique that transforms correlated variables into uncorrelated components.
2. **Why do we use PCA?**  
   To reduce dimensionality while retaining important variance in data.
3. **How does PCA work?**  
   It finds the directions (principal components) that maximize variance in the data.
4. **What are eigenvalues and eigenvectors in PCA?**
   * Eigenvalues measure variance explained by each component.
   * Eigenvectors define the principal component directions.
5. **What is the first principal component (PC1)?**  
   PC1 is the direction that captures the most variance in the dataset.
6. **What is the importance of variance in PCA?**  
   More variance means more information is retained after dimensionality reduction.
7. **How many principal components can PCA produce?**  
   Equal to the number of original features in the dataset.
8. **How do we choose the number of components in PCA?**  
   Using the explained variance ratio or a scree plot.
9. **What happens if we use too few components?**  
   Important information may be lost.
10. **Can PCA be applied to categorical data?**  
    No, PCA is used only for numerical data.
11. **What is the difference between PCA and feature selection?**

* PCA creates new features (principal components).
* Feature selection chooses existing features.

1. **Does PCA always improve model accuracy?**  
   Not necessarily; it depends on the dataset and application.
2. **How is PCA different from LDA?**

* PCA focuses on variance.
* LDA maximizes class separability.

1. **What is dimensionality reduction?**  
   The process of reducing the number of features while retaining meaningful information.
2. **Can PCA be used for visualization?**  
   Yes, reducing dimensions to 2D or 3D helps in visualizing high-dimensional data.
3. **How does standardization affect PCA?**  
   Standardization ensures all features contribute equally by scaling them.
4. **What is the curse of dimensionality?**  
   High-dimensional data makes analysis difficult due to increased complexity.
5. **What are the applications of PCA?**  
   Image compression, noise reduction, and feature extraction.
6. **Does PCA work well for non-linear data?**  
   No, PCA assumes linear relationships in data.
7. **What technique is used when PCA fails for non-linear data?**  
   Kernel PCA can handle non-linearity.

**Experiment 4: Find-S Algorithm (CSV Training Data)**

1. **What is the Find-S algorithm?**  
   Find-S is a specific-to-general algorithm used to find the most specific hypothesis consistent with positive training examples.
2. **What is the assumption of the Find-S algorithm?**  
   It assumes that there is at least one positive example and that the target concept is consistent with the training data.
3. **What is the main drawback of Find-S?**  
   It ignores negative examples and does not handle noisy or inconsistent data well.
4. **What does a hypothesis represent in Find-S?**  
   A hypothesis represents the set of conditions that a target concept satisfies.
5. **What is the initial hypothesis in Find-S?**  
   The most specific hypothesis, represented as a tuple of "Ø" or the most restrictive values.
6. **How does the algorithm update the hypothesis?**  
   It generalizes the hypothesis only when a new positive training example contradicts it.
7. **What happens when Find-S encounters a negative example?**  
   It does nothing, as it only considers positive examples.
8. **Why is Find-S considered a specific-to-general algorithm?**  
   It starts with the most specific hypothesis and generalizes when necessary.
9. **Can Find-S handle noisy data?**  
   No, since it does not revise the hypothesis based on negative examples.
10. **What is a hypothesis space?**  
    The set of all possible hypotheses that can be formed given the attributes.
11. **How does Find-S ensure consistency?**  
    It ensures that the final hypothesis is consistent with all positive training examples.
12. **What is the role of negative examples in learning algorithms?**  
    Negative examples help refine and improve the hypothesis by eliminating incorrect generalizations.
13. **How is Find-S different from the Candidate Elimination Algorithm?**

* Find-S finds the most specific hypothesis.
* Candidate Elimination maintains both a general and a specific boundary.

1. **Why is the Find-S algorithm not always reliable?**  
   It assumes noise-free data and ignores negative examples, which may lead to incorrect generalizations.
2. **What type of learning does Find-S belong to?**  
   It belongs to the category of supervised learning.
3. **What happens if there are conflicting positive examples?**  
   Find-S fails to provide a correct hypothesis since it cannot handle inconsistencies.
4. **Can Find-S learn a disjunctive concept?**  
   No, it learns only the most specific conjunctive hypothesis.
5. **What is the time complexity of Find-S?**  
   It is O(n), where n is the number of training examples.
6. **Why is Find-S not used in real-world applications?**  
   It is too simplistic and does not generalize well to complex datasets.
7. **How can we improve the Find-S algorithm?**  
   By incorporating negative examples and using version space learning like Candidate Elimination.

**Experiment 5: k-NN Classification (Randomly Generated Data)**

1. **What is the k-Nearest Neighbors (k-NN) algorithm?**  
   k-NN is a non-parametric classification algorithm that assigns a class based on the majority class of the k-nearest neighbors.
2. **How does k-NN work?**  
   It calculates the distance between a test point and all training points, then classifies the test point based on the majority vote of its k nearest neighbors.
3. **What are some common distance metrics used in k-NN?**
   * Euclidean distance
   * Manhattan distance
   * Minkowski distance
4. **How does the value of k affect the model?**
   * Small k: More sensitive to noise.
   * Large k: May oversmooth and miss local patterns.
5. **What is the best way to choose k?**  
   Using cross-validation to find the optimal k value.
6. **Is k-NN a supervised or unsupervised algorithm?**  
   It is a supervised learning algorithm.
7. **What is the computational complexity of k-NN?**  
   O(n), since it requires computing the distance to all training points.
8. **What are the advantages of k-NN?**
   * Simple and intuitive
   * Works well with small datasets
   * No need for training
9. **What are the disadvantages of k-NN?**
   * Computationally expensive for large datasets
   * Sensitive to irrelevant features
   * Requires a good choice of k
10. **Can k-NN be used for regression?**  
    Yes, k-NN can predict continuous values by averaging the values of the k nearest neighbors.
11. **What is the difference between k-NN and k-means clustering?**

* k-NN is a classification algorithm.
* k-means is an unsupervised clustering algorithm.

1. **How does feature scaling affect k-NN?**  
   Feature scaling is necessary since k-NN relies on distance calculations.
2. **What happens if two classes have an equal number of nearest neighbors?**  
   The tie is broken randomly or by using weighted voting.
3. **How can we improve k-NN performance?**

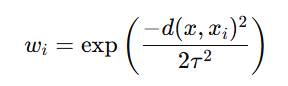
* Use feature selection
* Apply dimensionality reduction (PCA)
* Optimize k using cross-validation

1. **Does k-NN work well with high-dimensional data?**  
   No, k-NN suffers from the curse of dimensionality.
2. **Is k-NN sensitive to noisy data?**  
   Yes, especially when k is small.
3. **Can k-NN handle missing data?**  
   Yes, but imputing missing values is necessary before using k-NN.
4. **What is the role of the decision boundary in k-NN?**  
   The decision boundary is defined based on the voting of neighboring points.
5. **How does k-NN perform in imbalanced datasets?**  
   k-NN can be biased towards the majority class, requiring techniques like weighted voting.
6. **What are some real-world applications of k-NN?**

* Handwriting recognition
* Recommender systems
* Medical diagnosis

**Experiment 6: Locally Weighted Regression (Custom Dataset)**

1. **What is Locally Weighted Regression (LWR)?**  
   LWR is a non-parametric regression algorithm that gives different weights to training examples based on their distance from the query point.
2. **How does LWR differ from ordinary linear regression?**  
   Unlike linear regression, which fits a global model, LWR fits a model locally for each query point.
3. **What is the role of weights in LWR?**  
   The weights determine the importance of each training example when fitting the local model.
4. **What type of kernel functions are used in LWR?**
   * Gaussian Kernel
   * Triangular Kernel
   * Epanechnikov Kernel
5. **How do we compute the weight of a data point?**  
   Using the formula:



where τ is the bandwidth parameter.

1. **What happens when τ (bandwidth) is too small?**  
   The model becomes too sensitive to individual data points and overfits the data.
2. **What happens when τ (bandwidth) is too large?**  
   The model becomes too smooth and underfits the data.
3. **Is LWR a parametric or non-parametric method?**  
   It is a non-parametric method since it does not assume a fixed functional form.
4. **What is the main disadvantage of LWR?**  
   It is computationally expensive since it fits a separate model for each query point.
5. **How does LWR handle non-linearity in data?**  
   By fitting a local linear model at each query point, LWR can capture non-linear relationships.
6. **Can LWR be used for classification tasks?**  
   No, it is primarily used for regression problems.
7. **What is the effect of increasing the number of training points on LWR?**  
   More training points increase computational complexity since a new model must be fitted for each query.
8. **How do we choose the best bandwidth parameter τ?**  
   By using cross-validation to find the value that minimizes error.
9. **Why is LWR considered a memory-based learning algorithm?**  
   Because it requires storing the entire training dataset and performs computations at query time.
10. **How can we optimize LWR for large datasets?**

* Use a k-nearest neighbors approach to limit local points
* Use efficient indexing structures like KD-Trees

1. **What is the curse of dimensionality, and how does it affect LWR?**  
   As the number of features increases, distances become less meaningful, making LWR less effective.
2. **Can LWR work well with sparse data?**  
   No, because finding sufficient nearby points in high-dimensional space is difficult.
3. **Does LWR assume that data is normally distributed?**  
   No, it makes no assumption about the distribution of the data.
4. **What is the computational complexity of LWR?**  
   O(n2) per query, which is higher than traditional linear regression.
5. **What are some real-world applications of LWR?**

* Stock price prediction
* Locally adaptive image processing
* Medical diagnosis

**Experiment 7: Linear & Polynomial Regression (Boston Housing & Auto MPG datasets)**

**Linear Regression**

1. **What is linear regression?**  
   Linear regression is a statistical technique used to model the relationship between a dependent variable and one or more independent variables using a straight-line equation.
2. **Write the equation of simple linear regression.**

y=mx+c

where mmm is the slope and ccc is the intercept.

1. **What are the assumptions of linear regression?**
   * Linearity
   * Independence of errors
   * Homoscedasticity (constant variance of errors)
   * Normal distribution of residuals
2. **What is multiple linear regression?**  
   A regression model that includes multiple independent variables:

y=b0+b1x1+b2x2+...+bnxn

1. **How do we estimate the coefficients in linear regression?**  
   Using the **least squares method**, which minimizes the sum of squared residuals.
2. **What is the cost function used in linear regression?**  
   The **Mean Squared Error (MSE)**:



**How do we evaluate a linear regression model?**

* + R-squared score
  + Mean Absolute Error (MAE)
  + Root Mean Squared Error (RMSE)

1. **What is the significance of the R-squared value?**  
   It represents the proportion of variance explained by the model. A value close to 1 indicates a good fit.
2. **What is multicollinearity in multiple linear regression?**  
   It occurs when independent variables are highly correlated, making coefficient estimation unstable.
3. **How can we detect multicollinearity?**

* Variance Inflation Factor (VIF)
* Correlation matrix

1. **What is the impact of outliers in linear regression?**  
   Outliers can significantly affect the slope and intercept, leading to poor model performance.
2. **How can we handle outliers in linear regression?**

* Removing them using statistical methods
* Using robust regression techniques

1. **What is the difference between simple and multiple linear regression?**

* Simple linear regression has one independent variable.
* Multiple linear regression has two or more independent variables.

1. **What is gradient descent, and why is it used in linear regression?**  
   Gradient descent is an optimization algorithm used to minimize the cost function by updating coefficients iteratively.
2. **What is the difference between Batch Gradient Descent and Stochastic Gradient Descent?**

* **Batch GD** updates weights using the entire dataset.
* **SGD** updates weights after each training example, making it faster but noisier.

1. **How do we test if a linear regression model is statistically significant?**  
   Using the **F-test** and **p-values** for regression coefficients.
2. **What is heteroscedasticity, and why is it a problem in regression?**  
   When the variance of residuals is not constant, it violates regression assumptions and affects model reliability.
3. **How can we check for heteroscedasticity?**

* Residual plots
* Breusch-Pagan test

1. **What is feature scaling, and why is it important in regression?**  
   Scaling ensures numerical stability and speeds up convergence in gradient descent.
2. **What are the limitations of linear regression?**

* Assumes a linear relationship between variables
* Sensitive to outliers
* Struggles with non-linear data

**Polynomial Regression**

1. **What is polynomial regression?**  
   A regression technique that fits a polynomial function to capture non-linearity in data.
2. **What is the equation for polynomial regression of degree 2?**

y=b0+b1x+b2x2

1. **Why do we use polynomial regression instead of linear regression?**  
   When the relationship between variables is non-linear, polynomial regression can capture the curvature.
2. **How do we choose the degree of the polynomial in regression?**

* Using cross-validation
* Checking model performance metrics (e.g., RMSE, R-squared)

1. **What is the risk of choosing a very high-degree polynomial?**  
   Overfitting, where the model fits noise in the training data but performs poorly on new data.
2. **What is overfitting, and how can we prevent it?**  
   Overfitting occurs when a model learns noise instead of patterns. It can be prevented using:

* Regularization (L1, L2)
* Cross-validation
* Reducing model complexity

1. **How do we implement polynomial regression in Python?**  
   Using PolynomialFeatures from sklearn.preprocessing:

from sklearn.preprocessing import PolynomialFeatures

poly = PolynomialFeatures(degree=2)

X\_poly = poly.fit\_transform(X)

1. **What is Ridge Regression, and how does it help in polynomial regression?**  
   Ridge Regression adds an L2 penalty to prevent overfitting by reducing coefficient magnitudes.
2. **What is Lasso Regression?**  
   A regression technique that adds an L1 penalty, leading to some coefficients being exactly zero, effectively performing feature selection.
3. **What is Elastic Net regression?**  
   A combination of L1 (Lasso) and L2 (Ridge) regularization.
4. **How do we evaluate a polynomial regression model?**

* R-squared score
* RMSE (Root Mean Squared Error)
* Cross-validation

1. **What is bias-variance tradeoff in polynomial regression?**

* Low-degree polynomial models have high bias and low variance (underfitting).
* High-degree polynomial models have low bias and high variance (overfitting).

1. **What is a validation curve, and how can it help in polynomial regression?**  
   A validation curve shows model performance across different polynomial degrees, helping choose the optimal degree.
2. **What dataset is used for polynomial regression in this experiment?**  
   The **Auto MPG dataset**, which predicts vehicle fuel efficiency.
3. **What is the difference between logistic regression and polynomial regression?**

* Logistic regression is used for classification.
* Polynomial regression is used for non-linear regression.

1. **Why do we need to transform features for polynomial regression?**  
   To add higher-degree terms and enable the model to learn non-linear relationships.
2. **What happens if we apply linear regression to non-linear data?**  
   The model will underfit and provide poor predictions.
3. **How can we check if a polynomial regression model is better than a linear regression model?**  
   Compare their RMSE, R-squared values, and residual plots.
4. **How does polynomial regression handle extrapolation?**  
   Poorly—predictions outside the training range can be highly inaccurate.
5. **What are the real-world applications of polynomial regression?**

* Predicting stock prices
* Analyzing population growth
* Predicting real estate prices

**Experiment 8: Decision Tree (Breast Cancer Dataset)**

1. **What is a Decision Tree?**  
   A Decision Tree is a supervised learning algorithm used for classification and regression tasks, which splits data into branches based on feature conditions.
2. **What are the components of a Decision Tree?**
   * **Root Node**: The topmost node representing the entire dataset.
   * **Internal Nodes**: Decision points based on features.
   * **Leaves**: Terminal nodes representing class labels or predictions.
3. **What type of problems can be solved using Decision Trees?**
   * Classification problems (e.g., cancer detection)
   * Regression problems (e.g., price prediction)
4. **What is entropy in Decision Trees?**  
   Entropy is a measure of randomness or impurity in data:

****

1. **What is Information Gain?**  
   Information Gain measures how much a feature improves classification:

IG=Entropy(Parent)−∑(Weighted Entropy of Children)

1. **What is Gini Impurity?**  
   Measures impurity in a dataset using the probability of misclassification:

G=1−∑pi2

1. **Which splitting criteria are commonly used in Decision Trees?**
   * **Gini Impurity** (default in CART)
   * **Entropy (Information Gain)**
2. **What is overfitting in Decision Trees?**  
   When the tree becomes too complex, capturing noise instead of patterns.
3. **How can we prevent overfitting in Decision Trees?**
   * Pruning (pre-pruning & post-pruning)
   * Setting depth limits
   * Minimum sample split
4. **What is pruning in Decision Trees?**  
   A technique to remove unnecessary branches, improving generalization.
5. **What are leaf nodes in a Decision Tree?**  
   The final decision outputs, containing predicted class labels.
6. **What are advantages of Decision Trees?**

* Easy to interpret
* Handles both numerical and categorical data
* Requires minimal data preprocessing

1. **What are disadvantages of Decision Trees?**

* Prone to overfitting
* Can be sensitive to noisy data

1. **What is the difference between ID3, C4.5, and CART?**

* **ID3** uses Information Gain for splitting.
* **C4.5** is an improvement over ID3, handling missing values.
* **CART** (Classification and Regression Trees) supports regression and uses Gini Impurity.

1. **How do Decision Trees handle missing values?**

* By assigning the most common class in a split
* Using surrogate splits

1. **How do we evaluate Decision Tree performance?**

* Accuracy
* Precision, Recall, F1-score
* Confusion Matrix

1. **What is the Breast Cancer Dataset used for this experiment?**  
   A dataset containing features of cell nuclei to classify tumors as **malignant** or **benign**.
2. **What are alternative models to Decision Trees?**

* Support Vector Machines
* Random Forest
* Neural Networks

1. **What is the difference between Random Forest and Decision Trees?**

* Decision Tree is a single tree-based model.
* Random Forest is an ensemble of multiple trees, reducing overfitting.

1. **How can Decision Trees be visualized?**

* Using graphviz in Python
* plot\_tree function in sklearn

**Experiment 9: Naive Bayes Classifier (Olivetti Face Dataset)**

1. **What is the Naïve Bayes classifier?**  
   A probabilistic algorithm based on Bayes' Theorem, assuming feature independence.
2. **Write the Bayes' Theorem formula.**

P(A∣B) =P(B∣A) P(A) / P(B)

1. **Why is it called ‘Naïve’ Bayes?**  
   Because it assumes all features are independent, which is rarely true in real-world data.
2. **What are the types of Naïve Bayes classifiers?**
   * **Gaussian**: For continuous data
   * **Multinomial**: For text classification
   * **Bernoulli**: For binary data
3. **What is the Olivetti Face Dataset?**  
   A dataset of 400 grayscale images of 40 individuals, used for face recognition.
4. **What are the applications of Naïve Bayes?**
   * Spam filtering
   * Sentiment analysis
   * Facial recognition
5. **What is the difference between prior and likelihood in Bayes' Theorem?**
   * **Prior**: Initial probability before evidence.
   * **Likelihood**: Probability of evidence given the class.
6. **What is Laplace Smoothing in Naïve Bayes?**  
   A technique to handle zero probabilities by adding a small constant to each count.
7. **What are the advantages of Naïve Bayes?**
   * Works well with small datasets
   * Fast computation
8. **What are the limitations of Naïve Bayes?**

* Assumes independence of features
* Can struggle with correlated features

1. **How is probability calculated in Naïve Bayes for classification?**  
   Using Maximum A Posteriori (MAP) estimation.
2. **What is the difference between Naïve Bayes and Logistic Regression?**

* Naïve Bayes is based on probability, while logistic regression optimizes a decision boundary.

1. **How do we evaluate a Naïve Bayes model?**

* Accuracy
* Precision, Recall, F1-score

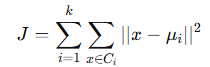
1. **How does Naïve Bayes handle missing values?**  
   It ignores missing values while calculating probabilities.
2. **What is the curse of dimensionality, and how does it affect Naïve Bayes?**  
   High-dimensional data can lead to sparsity, reducing classification accuracy.
3. **Can Naïve Bayes be used for regression?**  
   No, it is strictly a classification algorithm.
4. **What does a high posterior probability indicate in Naïve Bayes?**  
   It indicates strong evidence supporting a particular class.
5. **Why is Naïve Bayes widely used in NLP?**  
   Because it effectively handles high-dimensional text data.
6. **How do we compute probability in Gaussian Naïve Bayes?**  
   Using the normal distribution formula:



1. **Why is Naïve Bayes computationally efficient?**  
   Because it only requires counting and probability calculations.

**Experiment 10: k-Means Clustering (Wisconsin Breast Cancer Dataset)**

1. **What is k-Means Clustering?**  
   k-Means is an unsupervised machine learning algorithm used to partition data into **k** clusters based on feature similarity.
2. **How does k-Means Clustering work?**
   * Select **k** initial cluster centroids.
   * Assign each data point to the nearest centroid.
   * Update centroids by computing the mean of assigned points.
   * Repeat until convergence.
3. **What is the objective function of k-Means?**  
   k-Means minimizes the **within-cluster sum of squares (WCSS)**:



1. **How do we choose the optimal value of k?**  
   Using the **Elbow Method** or **Silhouette Score**.
2. **What are some applications of k-Means clustering?**
   * Customer segmentation
   * Image compression
   * Anomaly detection
3. **What are the assumptions of k-Means Clustering?**
   * Clusters are spherical and of similar sizes.
   * Data points are equally distributed.
4. **What are the limitations of k-Means?**
   * Sensitive to the choice of **k**.
   * May converge to local optima.
   * Struggles with non-spherical clusters.
5. **What is the difference between Hard and Soft Clustering?**
   * **Hard clustering**: Each data point belongs to only one cluster (e.g., k-Means).
   * **Soft clustering**: A data point can belong to multiple clusters with probabilities (e.g., Gaussian Mixture Model).
6. **What is the difference between k-Means and Hierarchical Clustering?**
   * k-Means is faster and requires specifying **k** in advance.
   * Hierarchical clustering creates a hierarchy and does not require **k** beforehand.
7. **What are centroid-based clustering methods?**  
   Methods that assign clusters based on distance to a central point, like k-Means.
8. **Why is k-Means sensitive to the initial choice of centroids?**  
   Poor initialization can lead to suboptimal clusters.
9. **How can we improve centroid initialization in k-Means?**  
   Using **k-Means++**, which spreads initial centroids apart.
10. **How do we handle outliers in k-Means?**
    * Use **k-Medoids** instead of means.
    * Remove extreme values before clustering.
11. **What is cluster inertia in k-Means?**  
    The sum of squared distances between points and their centroids.
12. **How do we evaluate k-Means clustering?**
    * **Elbow Method**: Finds the point where adding more clusters doesn’t significantly reduce inertia.
    * **Silhouette Score**: Measures how similar a point is to its assigned cluster.
13. **What is the Wisconsin Breast Cancer Dataset used for?**  
    It contains tumor cell features to classify them as **malignant** or **benign**.
14. **How does k-Means clustering handle categorical data?**  
    It doesn’t natively work with categorical data; **k-Modes** or **one-hot encoding** is used instead.
15. **What distance metric is used in k-Means?**  
    The **Euclidean distance**:



1. **What happens if k is too small or too large in k-Means?**
   * Too small: Clusters may be too broad.
   * Too large: Clusters may be meaningless and fragmented.
2. **Can k-Means be used for image segmentation?**  
   Yes, it can cluster pixel colors to segment images.