1) **What is supervised learning?**

**Ans**:Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output.

2) **What is unsupervised learning?**

**Ans**:unsupervised learning is a machine learning technique in which models are not supervised using training dataset. Instead, models itself find the hidden patterns and insights from the given data. It can be compared to learning which takes place in the human brain while learning new things.

3) **Difference between supervised and unsupervised machine learning?**

**Ans**:

| Supervised Learning | Unsupervised Learning |
|---|---|
| Learn a mapping from inputs to outputs based on labeled data. | Discover patterns, relationships, or structures in unlabeled data. |
| Labeled dataset (input features with corresponding output labels). | Unlabeled dataset without explicit output labels. |
| Classification, Regression. | Clustering, Dimensionality Reduction, Association. |
| Receives feedback based on the error between predicted and actual outputs. | No explicit feedback mechanism; the algorithm infers structure on its own. |
| Image classification, Spam detection, House price prediction. | Customer segmentation, Topic modeling, Anomaly detection. |
| Metrics like accuracy, precision, recall, F1 score. | Evaluation can be subjective, depending on the specific task (e.g., cluster quality). |

4) **What is clustering?**

**Ans**:Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. It is basically a collection of objects on the basis of similarity and dissimilarity between them.

5) **What is regression?**

**Ans**:Regression is a supervised machine learning technique which is used to predict continuous values. The ultimate goal of the regression algorithm is to plot a best-fit line or a curve between the data.In short it Predicts a number infinitly many possible output.

6) **What is classification?**

**Ans**:Predicts categories small number of possible outputs.

7) **Why do feature scalling needed?**

**Ans**:Feature scaling in machine learning is the process of transforming the numerical features of a dataset to a standardized range, typically through techniques like normalization or standardization. This step is crucial because machine learning algorithms can be sensitive to the magnitude of features, potentially giving undue importance to those with larger scales. By bringing all features to a similar scale, feature scaling ensures fair treatment and prevents dominance based on magnitude, facilitating accurate distance calculations, faster convergence in optimization algorithms like gradient descent, and avoiding saturation issues in activation functions, especially in neural networks. Overall, feature scaling is essential to enhance the robustness and effectiveness of machine learning models.

### 8) What is decision boundary in logistic regression?

Ans:The decision boundary in logistic regression is a crucial concept that separates data points belonging to different classes. It's essentially a line or surface that defines which side of the line a data point falls on, determining its class membership.

### 9) Difference between linear regression vs logistic regression.

Ans:

| Linear Regression | Logistic Regression |
|---|---|
| Model continuous relationships | Predict class membership |
| Continuous value | Probability value (0-1) |
| Linear model | Sigmoid function |
| Mean squared error (MSE) | Cross-entropy, AUC |
| Prediction, analysis, trend prediction | Classification, decision-making |

### 10) What is Overfitting and underfitting and generalization?

**Ans:**

**Overfitting:**Imagine you're training a model to identify cats in images. Overfitting occurs when the model memorizes the training data too closely, capturing even the noise and irrelevant details. This leads to the model performing very well on the specific training images it saw, but failing to generalize its knowledge to new, unseen images. Think of it like studying only past exam questions for a test; you might ace that specific test, but you'll struggle with any new questions.

**Underfitting:**On the other hand, underfitting happens when the model is too simple and cannot capture the underlying patterns in the data. Imagine trying to classify cats with a model that only considers their color. This model would perform poorly on both the training data and new data, because it lacks the necessary complexity to learn the true distinctions between cats and other animals.

**Generalization:**Generalization refers to a model's ability to perform well on data it hasn't seen before during training. A well-generalized model can learn the essential patterns from the training data and apply them to new, unseen examples with good accuracy. This is the ultimate goal of machine learning: to build models that can learn from experience and apply their knowledge to real-world situations.

**11) How to overcome overfitting?**

**12) What is regularization? Explain in brief.**

**Ans**:Regularization is a technique used in machine learning to prevent overfitting and improve the generalization performance of a model. Overfitting occurs when a model fits the training data too closely, capturing noise and fluctuations in the data, which can lead to poor performance on new, unseen data.

The purpose of regularization is to introduce a penalty term into the model's objective function, discouraging overly complex models with large coefficients or weights. This penalty term helps find a balance between fitting the training data well and avoiding unnecessary complexity.

There are two common types of regularization:

1. **L1 Regularization (Lasso):** Adds a penalty term proportional to the absolute values of the model's coefficients to the cost function. It promotes sparsity in the coefficients, effectively performing feature selection by setting some coefficients to zero.
2. **L2 Regularization (Ridge):** Adds a penalty term proportional to the square of the model's coefficients to the cost function. It encourages the model to distribute the weight more evenly among all features, but it rarely leads to exactly zero coefficients.

**13) What is bias and varience trade off?**

**Ans**:The bias-variance tradeoff is a fundamental concept in machine learning that relates to the performance of a model. It refers to the balance between two sources of error that a model might exhibit: bias and variance.

**Bias:**

- ➔ Bias is the error introduced by approximating a real-world problem, which may be complex, by a simplified model.
- ➔ High bias can lead to underfitting, where the model is too simple to capture the underlying patterns in the data.
- ➔ Models with high bias tend to be too generalized and may not perform well on either the training or testing data.

**Variance:**

- ➔ Variance is the error introduced by having too complex of a model that captures noise in the training data as if it were a real pattern.
- ➔ High variance can lead to overfitting, where the model fits the training data very closely but fails to generalize to new, unseen data.
- ➔ Models with high variance perform well on the training data but poorly on testing data.

The tradeoff arises because decreasing bias often increases variance and vice versa. Achieving the right balance is crucial for building models that generalize well to new, unseen data. The goal is to find the sweet spot that minimizes both bias and variance, resulting in a model that performs well on a variety of data.

Regularization techniques, as mentioned in a previous response, are one way to address the bias-variance tradeoff. By penalizing overly complex models, regularization helps prevent overfitting (reducing variance) but may introduce a slight increase in bias.

Understanding and managing the bias-variance tradeoff is essential for building models that generalize well, and it often involves techniques such as cross-validation, choosing appropriate model complexity, and using regularization methods.

14) **K-means automatically adjust the number of cluster-true or false?**

**Ans**:False

15) **What is confusion matrix?**

**Ans**:A confusion matrix is a table that visualizes how well a classification model performs. It breaks down its predictions into four categories:

**True positives:**Correctly classified positive cases.

**True negatives:**Correctly classified negative cases.

**False positives:**Incorrectly classified positive cases (mistakes).

**False negatives:**Incorrectly classified negative cases (missed positives)

16) **What is the sensitivity of an ML model?**

**Ans**:Sensitivity, also known as True Positive Rate, Recall, or Sensitivity, is a performance metric used to evaluate the effectiveness of a machine learning model in correctly identifying positive instances

17) **What is MinMax scalling?**

**Ans**:MinMax scaling is a data normalization technique used in machine learning to scale numerical features to a specific range, typically between 0 and 1. The purpose of MinMax scaling is to bring all the feature values of a dataset into a consistent range, preventing features with larger scales from dominating the learning process in certain algorithms.

18) **What is “kernel trick”?**

**Ans**:The "kernel trick" is a concept in machine learning, particularly in the context of Support Vector Machines (SVMs) and other kernelized algorithms. It allows these algorithms to operate in a higher-dimensional space without explicitly computing the new feature representations, thereby avoiding the computational cost associated with working in high dimensions.

19) **What is activation function?**

**Ans**:An activation function is a mathematical operation applied to the output of each neuron (or node) in a neural network.

20) **What is MSE?**

**Ans**:MSE stands for Mean Squared Error, and it is a commonly used metric to evaluate the performance of a regression model. It measures the average squared difference between the predicted values and the actual values in a dataset.

**21) What is an generative learning algorithm?**

**Ans**:Generative learning algorithms are a special class of machine learning algorithms focused on creating new data that closely resembles the training data. They essentially learn the underlying patterns and distributions within the data and use that knowledge to generate new samples that follow the same patterns.

**22) When the feature space is large,overfitting is more likely-true or false?**

**Ans**:True

23) Is model with higher or lower varience better? Explain?

24) **What is loss function?Give a few examples of loss function.**

**Ans**:A loss function, also known as a cost or objective function, measures how well a machine learning model's predictions match the true values. It quantifies the difference between the predicted and actual outcomes and serves as a guide for the optimization algorithm during training. The goal is to minimize the loss function to improve the model's performance.

Some loss functions are-

- ➔ Mean Squared Error (MSE)
- ➔ Mean Absolute Error (MAE)
- ➔ Binary Cross-Entropy (BCE)
- ➔ Log Loss

25) **Derive the expression of Naive Bayes classifier from conditional probability?**

**Ans:We Know,**

$$P(A|B)=\frac{P(A\cup B)}{P(B)}$$

*and*

$$P(B|A)=\frac{P(B\cup A)}{P(A)}=\frac{P(A\cup B)}{P(A)}$$

$$so\, P(A\cup B)=P(B\cup A)$$

$$P(A|B)P(B)=P(B|A)P(A)$$

$$P(A|B)=\frac{P(A|B)P(A)}{P(B)}$$

26) **What is overfitting in model development ?How can it be avoided?**

**Ans**:Overfitting in model development occurs when a machine learning model learns the training data too well, capturing noise and random fluctuations in addition to the underlying patterns. As a result, the model performs exceptionally well on the training set but fails to generalize effectively to new, unseen data.

Causes of Overfitting:

**Complex Models:** Overly complex models, with too many parameters or features, are more prone to overfitting.

**Insufficient Data**: With a limited amount of data, the model may memorize specific examples instead of learning general patterns.

**Noise in Data**: Noisy or irrelevant features in the training data can mislead the model.

Ways to Avoid Overfitting:

**Cross-Validation:** Use techniques like k-fold cross-validation to evaluate the model's performance on different subsets of the training data.

**Regularization:** Add a penalty term to the model's objective function to discourage large coefficients. Common regularization methods include L1 regularization (Lasso) and L2 regularization (Ridge).

**Feature Selection:** Choose relevant features and eliminate unnecessary ones. This reduces model complexity and the risk of overfitting.

**More Data**: Increase the size of the training dataset to provide the model with a more diverse range of examples.

**Ensemble Methods:** Combine predictions from multiple models (ensemble learning) to reduce overfitting. Examples include Random Forests and Gradient Boosting.

**Early Stopping:** Monitor the model's performance on a validation set during training and stop training when the performance starts to degrade, preventing overfitting.

**Pruning (for Trees):** For decision tree-based models, prune the tree to limit its depth and remove branches that provide little predictive power.

**Simpler Models:** Choose simpler model architectures that are less prone to overfitting. For example, use linear models instead of complex non-linear ones.

**Data Augmentation:** Increase the size of the training dataset by generating new examples through techniques like rotation, flipping, or cropping.

**Hyperparameter Tuning:** Experiment with different hyperparameter values, such as learning rates or regularization strengths, to find the combination that minimizes overfitting.

27) **What is Cross validation? Why is it used?**

**Ans:**Cross-validation is a powerful technique used in machine learning to evaluate the performance of a model on unseen data. It's essentially a way to simulate how well your model would generalize to new situations it hasn't been trained on.

28) **Why do we damentionality reduction?What are its drawbacks?**

Ans:Dimensionality reduction is a technique used in machine learning and data analysis to reduce the number of features or variables in a dataset while preserving its essential characteristics.

The primary motivations for dimensionality reduction include:

➔ **Improved model performance**

➔ **Reduced computational cost**

➔ **Enhanced data visualization**

➔ **Reduced noise and irrelevant features**

Drawbacks are-

**Information loss**

**Interpretation challenges:Interpretation challenges:**

**Choosing the right technique**

## Neural Network

1) **What is Neural network?**

**Ans**:A neural network, in the context of machine learning, is a computational model inspired by the structure and functioning of the human brain. It is composed of interconnected nodes, also known as neurons or artificial neurons, organized in layers. Neural networks are a fundamental component of deep learning, a subfield of machine learning that focuses on models with multiple layers (deep neural networks).

2) **What is feed forward network?**

**Ans**:A feedforward neural network is a type of artificial neural network where information flows in one direction, from input to output, like water flowing through a pipe.

## PCA(Principle component analysis)

1) **What is PCA?**

**Ans**:Principal component analysis (PCA) is a dimensionality-reduction method used to condense a large data set into a smaller, more manageable set. This method serves various purposes in technology and software industries, including the use of facial recognition software and image compression.

2) **What are the limitations of PCA?**

**Ans**:Principal Component Analysis (PCA) is a powerful technique for dimensionality reduction and feature extraction, but it has some limitations:

- **Linearity Assumption:**PCA assumes that the relationships between variables are linear
- **Sensitive to Outliers:**PCA is sensitive to outliers in the data.
- **Preservation of Variances:**While PCA maximally preserves the variance in the original data, it may not effectively preserve local or non-linear structures.
- **Interpretability:**The principal components generated by PCA are linear combinations of the original features, and their interpretability may be limited
- **Orthogonality Assumption:**PCA assumes that principal components are orthogonal
- **Non-Robust to Feature Scaling:**PCA is sensitive to the scale of the features.

- **Memory and Computational Requirements:**PCA requires the computation of covariance or correlation matrices, which can be computationally expensive
- **Limited to Linear Transformations:**
- **Preservation of Individual Data Points:**

3) Write down the steps of PCA.

A) Calculate the mean:

$$\bar{X}_1=(X_{11}+X_{12}+X_{13}+...+X_{1n})/n$$
$$\bar{X}_2=X_{21}+X_{22}+X_{23...}+X_{2n}/n$$

B) Calculating the Co-varience matrix:

$$S=\begin{bmatrix} Cov(X_1,X_1) & Cov(X_1,X_2) \\ Cov(X_2,X_1) & Cov(X_2,X_2) \end{bmatrix}$$

$$Cov(X_1,X_1)=\frac{1}{N-1}\sum_{k=1}^{N}(X_{1k}-\overline{X_1})(X_{1k}-\overline{X_1})$$

$$Cov(X_1,X_2)=\frac{1}{N-1}\sum_{k=1}^{N}(X_{1k}-\overline{X_1})(X_{2k}-\overline{X_2})$$

C) Eigen values of the co-varience matrix:

$$0=det(S-\lambda I)$$

From this equation $\lambda_1,\lambda_2$ value will be found

D) Computation of the eigenvectors:

$$U=\begin{bmatrix} U_1 \\ U_2 \end{bmatrix}$$

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = (S-\lambda I)U$$

From here U1 and U2 value will be found in respect to $\lambda$ .

$$U_1=\begin{bmatrix} 11 \\ 14-\lambda_1 \end{bmatrix}$$

$$\|U_1\|=\sqrt{11^2+(14-\lambda_1^2)}$$

$$e_1=\begin{bmatrix} \frac{11}{\|U_1\|} \\ \frac{14-\lambda_1}{\|U_1\|} \end{bmatrix}$$

E) Computation of first principle components:

Now for every pair of X1 and X2 we have to use the formula-

$$e_1^T*\begin{bmatrix} X_{1k}-\overline{X_1} \\ X_{2k}-\overline{X_2} \end{bmatrix}$$

## SVM & Kernel

### 1) What is SVM and why its use and its drawbacks?

**Ans:**A Support Vector Machine (SVM) is a powerful supervised learning algorithm used in machine learning for tasks like classification, regression, and outlier detection. It's known for its effectiveness in high-dimensional spaces and its ability to find optimal decision boundaries even with limited data.

**What it does:**

**Classification:** Imagine you have data points representing different classes (e.g., apples and oranges). SVM finds the optimal hyperplane (a high-dimensional separation line) that maximizes the margin between these classes. Points closest to the hyperplane, called support vectors, define the margin and influence the decision boundary. New data points are then classified based on which side of the hyperplane they fall on.

**Regression:** Similar to classification, SVM finds a hyperplane that best fits the relationship between features and a continuous target variable. This hyperplane minimizes the error between predictions and actual values.

**Outlier detection:** SVMs can identify data points that significantly deviate from the learned pattern by analyzing their distance from the decision boundary. These outliers can then be investigated further.

**Why it's used:**

**Effective in high-dimensional spaces:** Unlike some algorithms that struggle with many features, SVM can handle high-dimensional data efficiently.

**Finds optimal hyperplanes:** SVM seeks the widest margin between classes, leading to robust and accurate decision boundaries.

**Works well with small datasets:** SVM can perform well even with limited data due to its focus on support vectors.

**Interpretable results:** Support vectors provide insights into the data, making the model's decision-making process more transparent.

**However, SVMs also have some limitations:**

**Computationally expensive:** Training SVMs can be slower than some other algorithms, especially for large datasets.

**Not ideal for complex, non-linear relationships:** While SVMs can handle non-linearity through kernel functions, they may not always be the best choice for highly complex relationships.

**Sensitive to parameter tuning:** Choosing the right kernel and hyperparameters can significantly impact the performance of an SVM.

2) **What is kernel?**

**Ans**:kernel refers to a function that computes the similarity (or distance) between pairs of data points in a high-dimensional space.

3) **What are support vectors?How do support vectors help to find the optimal margin of a model?**

**Ans**:In Support Vector Machines (SVMs), support vectors are the data points closest to the decision boundary, the hyperplane that separates different classes. These points play a crucial role in defining the optimal margin, which is the maximum distance between the hyperplane and any data point of either class.

**Identifying Support Vectors:**

- The support vectors are the data points that lie on or inside the margin. They are the most influential points in determining the position and orientation of the hyperplane. These vectors "support" the decision boundary.

**Margin Calculation:**

- The margin is the distance between the hyperplane and the nearest support vectors. The goal is to maximize this margin. The larger the margin, the better the model's generalization to new, unseen data.

**Optimization Objective:**

- The optimization objective of SVM involves finding the hyperplane parameters (weights and bias) that maximize the margin while satisfying the condition that all data points are correctly classified.

**Soft Margin SVM:**

- In some cases, a perfect separation of classes is not possible, especially when dealing with noisy or overlapping data. In such situations, a "soft margin" is introduced, allowing for some misclassification. The optimization objective is then modified to find a balance between maximizing the margin and minimizing misclassification.

4)

## Probability

1) **In a group of 100 sports car buyers, 40 bought alarm systems, 30 purchased bucket seats, and 20 purchased an alarm system and bucket**

**seats. If a car buyer chosen at random bought an alarm system, what is the probability they also bought bucket seats?**

**Ans**:

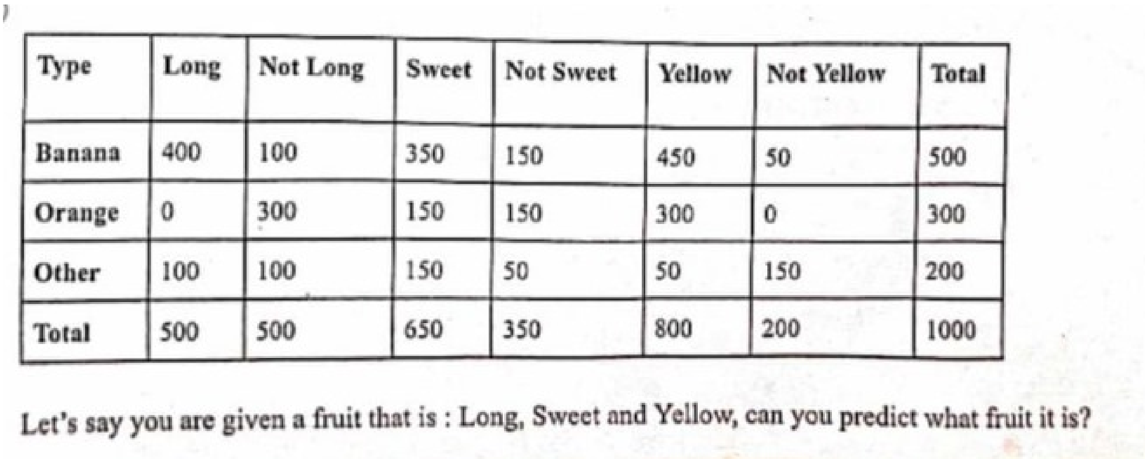
Step 1: Figure out P(A). It's given in the question as 40%, or 0.4.

• Step 2: Figure out P(A∩B). This is the intersection of A and B: both happening together. It's given in the question 20 out of 100 buyers, or 0.2.

• Step 3: Insert your answers into the formula: P(B|A) = P(A∩B) / P(A) = 0.2 / 0.4 = 0.5

•The probability that a buyer bought bucket seats, given that they purchased an alarm system, is 50%.

1)



| Type | Long | Not Long | Sweet | Not Sweet | Yellow | Not Yellow | Total |
|---|---|---|---|---|---|---|---|
| Banana | 400 | 100 | 350 | 150 | 450 | 50 | 500 |
| Orange | 0 | 300 | 150 | 150 | 300 | 0 | 300 |
| Other | 100 | 100 | 150 | 50 | 50 | 150 | 200 |
| Total | 500 | 500 | 650 | 350 | 800 | 200 | 1000 |

Let's say you are given a fruit that is : Long, Sweet and Yellow, can you predict what fruit it is?

**Ans**:

P(banana)=500/1000=1/2

P(orange)=300/1000=3/10

P(other)=200/1000=2/10

P(long|Banana)=400/500=4/5

P(not long|Banana)=100/500=1/5

P(sweet | Banana)=350/500

P(not sweet | Banana)=150/500

P(yellow | Banana) = 450/500=9/10

P(not yellow | Banana)= 50/500

P(Long | orange)=0/300

P(not long | Oranage)=300/300=1

P(yellow | orange) =300/300=1

P(not yellow | orange)=0/300=0

P(sweet | orange) =150/300=1/2

P(not sweet | orange)=150/300=1/2

P(long | other)=100/200=1/2

P(not long | other)= 100/200=1/2

P(sweet | other)=150/200=3/4

P(not sweet | other) = 50/200=1/4
P(yellow | other) = 50/200=1/4
P(not yellow | other) = 150/200=3/4

P(banana |new fruit)=P(banana)*P(long|Banana)*P(sweet | Banana)*P(yellow | Banana)
=1/2*4/5*7/10*1/10=0.028
P(orange | new fruit)=P(orange)*P(Long | orange)*P(yellow | orange)*P(sweet | orange)
=3/10*0*1/2*1=0
P(other | new fruit)=P(other)*P(long | other)*P(sweet | other)*P(yellow | other)
=1/5*1/2*3/4*1/4=00.01875
Since ,P(banana|new fruit)>P(orange | new fruit) >P(other | new fruit)
so, the new fruit belongs to banana class.