Predictive Analytics in Data Science Q&A Guide

-By Akshara S, Data Science Intern at inGrade

1.What are the key steps in building a predictive model?

Answer: Building a predictive model involves several steps:

- **Define the problem:** Understand what you want to predict (e.g., customer churn, sales).
- **Collect data:** Get the right data from databases, surveys, or files
- **Clean the data**: Remove duplicates, fix missing values, and correct errors.
- **Explore the data (EDA):** Use graphs and statistics to see trends and patterns.
- **Select features:** Choose the most important columns that help prediction.
- **Choose a model:** Pick the right algorithm like Linear Regression or Decision Tree.
- **Split data:** Divide the data into training (for learning) and testing (for checking).
- **Train the model:** Let the model learn from the training data.
- **Evaluate the model:** Test how well it performs using metrics.
- **Improve and deploy:** Tune the model, then put it into use.

Example: If you're trying to predict which customers might leave a company, you'll first look at past customer data like age, complaints, and usage, then build and test a model to see who is likely to leave.

2.How do you choose between linear regression and decision trees?

Answer: The choice depends on the type of data and the problem:

- Use **Linear Regression** when:
  - The relationship between the input and output is mostly a straight line.

    - The data is numerical and doesn't have complex interactions.
    - You need a simple, easy-to-explain model.
- Use **Decision Trees** when:
    - The relationship is not linear (i.e., has curves or rules).
    - The data includes both numbers and categories.
    - You want to understand decisions in an "if-then" format.

Example: Suppose you want to predict house prices:

- If price mainly depends on size, use **Linear Regression**.
- If price also depends on neighborhood, number of rooms, and whether the house has a garage or not, then a **Decision Tree** is better because it can handle more complex rules.

---

3. What is cross-validation, and why is it important in predictive modeling?

Answer: Cross-validation is a method used to test how well a model works on different sets of data. It helps check that the model is not just working well on one specific split of data but performs well in general.

The most common type is **K-Fold Cross-Validation**, where the dataset is divided into **K parts** (say 5 or 10). The model is trained on K-1 parts and tested on the remaining part. This is repeated K times with a different test set each time, and the results are averaged.

Why it's important:

- It reduces the chance of overfitting.
- Gives a better estimate of how the model will perform on unseen data.

Example: If you have 1000 records and use 5-fold cross-validation, your model will be trained and tested 5 times, each time using 800 records to train and 200 to test, but in different combinations. This gives a more accurate performance score.

---

4. What is the difference between training and testing data?

Answer:

- **Training Data:** The data the model learns from. It helps the model understand patterns and relationships.
- **Testing Data:** New data that the model hasn't seen before. It checks how well the model performs in real situations.

Why we split data: If you test a model on the same data it trained on, it might look perfect but fail on new data. So we split the data to see if it truly generalizes.

Example: If you're building a model to predict whether someone will get a loan, you give it 80% of your past data to learn (training data) and the remaining 20% to see how accurately it can predict new cases (testing data).

---

5. What are common evaluation metrics for regression models?

Answer: Regression models are evaluated based on how close their predictions are to the actual values. Common metrics include:

- **Mean Absolute Error (MAE):** Average of the absolute differences between predicted and actual values.

$$MAE = \frac{1}{n}\sum |y_i - \hat{y}_i|$$

- **Mean Squared Error (MSE):** Squares the errors before averaging, giving more weight to larger errors.

$$MSE = \frac{1}{n}\sum (y_i - \hat{y}_i)^2$$

- **Root Mean Squared Error (RMSE):** Square root of MSE; easier to interpret because it's in the same units as the data.

$$RMSE = \sqrt{MSE}$$

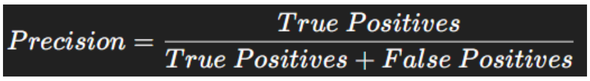
- **R-squared (R^2):** Shows how well the model explains the data variation. Values range from 0 to 1. Higher is better.

Example: If you predict the sales of a shop for each day and the actual vs predicted sales differ by 5 units on average, then MAE = 5. RMSE may be slightly higher if some predictions were way off.

6. What are common evaluation metrics for classification models?

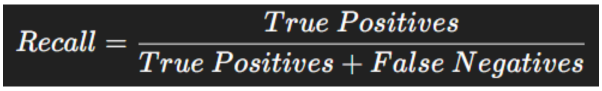
Answer: Classification models are judged by how well they assign items into categories. Key metrics include:

- **Accuracy:** % of correct predictions.
- **Precision:** % of positive predictions that were correct. TP / (TP + FP)



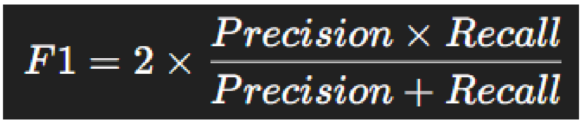
$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

- **Recall (Sensitivity):** % of actual positives that were correctly identified. TP / (TP + FN)



$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

- **F1 Score:** Balance between precision and recall.



$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

- **AUC-ROC:** Measures how well the model can separate the classes.

Example: If you're building a spam email detector:

- **Precision** tells how many predicted spam emails are truly spam.
- **Recall** tells how many actual spam emails were caught.
- A high **F1 score** means the model balances both well.

7. What is feature selection, and why is it important?

Answer: Feature selection is the process of choosing the most relevant features (columns) from the dataset and removing unnecessary or noisy ones.

Why it matters:

- Reduces training time
- Improves model accuracy

- Prevents overfitting
- Makes the model easier to understand

Techniques:

- **Filter methods:** Use statistics like correlation.
- **Wrapper methods:** Try different combinations and check performance (e.g., RFE).
- **Embedded methods:** Built into models (e.g., Lasso Regression).

Example: In predicting a person's salary, useful features might be age, education, and years of experience. A feature like "favorite color" wouldn't help, so it's removed during feature selection.

---

8. How does hyperparameter tuning improve model performance?

Answer: Hyperparameters are settings you define **before** training a model, such as the number of trees in a forest or the depth of a decision tree. They are different from regular parameters, which the model learns automatically.

Tuning hyperparameters helps you find the best combination of settings that give the highest performance.

Common methods:

- **Grid Search:** Try all combinations.
- **Random Search:** Try random combinations.
- **Bayesian Optimization:** Use smart searching to save time.

Example: For a Random Forest, trying different values of n_estimators (number of trees) and max_depth (tree depth) helps find the best-performing model. Without tuning, your model may underperform.

---

9. What is the difference between overfitting and underfitting?

Answer:

- **Overfitting:** The model learns **too much** from the training data, including noise and random details. It works great on training data but poorly on new data.
- **Underfitting:** The model learns **too little**, missing important patterns. It does poorly on both training and test data.

Signs of Overfitting:

- Very high training accuracy
- Low test accuracy

Signs of Underfitting:

- Both training and test accuracies are low

Example: Imagine a student who memorizes the textbook (overfitting) but fails when the questions change. Another student who barely studies at all (underfitting) also fails. A good model is like a student who understands the concept and applies it in new situations.

---

10. What is the difference between parametric and non-parametric models?

Answer:

- **Parametric models** make fixed assumptions about the shape of the data (like straight lines). They learn a limited number of parameters.
  Example: Linear Regression.
- **Non-parametric models** don’t make strong assumptions. They learn more flexible patterns and can adapt better to complex data.
  Example: Decision Trees, k-Nearest Neighbors.

Key Differences:

- Parametric = faster and simpler, but less flexible.
- Non-parametric = more flexible, works better on messy data.

Example: If your data has a clear linear pattern (like income rising with experience), use a parametric model. If the pattern is more complicated or includes conditions, like "if age > 30 and income > 50K", then use a non-parametric model.

---

11. What is multicollinearity, and why is it a problem in regression models?

Answer: Multicollinearity happens when two or more independent variables (features) in a regression model are highly correlated with each other. This makes it hard to know which feature is actually influencing the target variable.

Why it's a problem:

- Coefficients become unstable.
- Model becomes hard to interpret.
- Predictions may be less reliable.

How to detect it:

- Check **correlation matrix**.
- Use **VIF (Variance Inflation Factor)**: If VIF > 5 or 10, there's multicollinearity.

Example: If you're predicting house prices using both house_size in square feet and number_of_rooms, they might be strongly correlated. Using both can confuse the model — so you may drop one.

---

12. What is regularization in machine learning?

Answer: Regularization is a technique to **reduce overfitting** by adding a **penalty term** to the model's loss function. This penalty discourages the model from becoming too complex or using unnecessary features.

Types:

- **L1 Regularization (Lasso):** Adds the sum of absolute coefficients.
- **L2 Regularization (Ridge):** Adds the sum of squared coefficients.
- **ElasticNet:** Combines both L1 and L2.

Example (Lasso): In a salary prediction model, if a feature like "employee badge number" is not useful, Lasso can shrink its weight to 0 and remove it from the model.

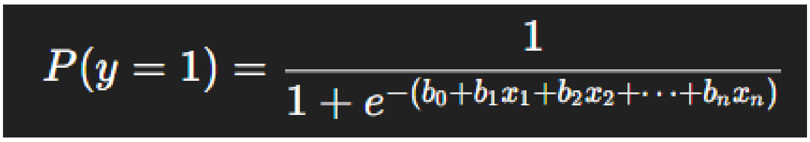
---

13. What is logistic regression, and when should it be used?

Answer: Logistic Regression is used to **predict binary outcomes** (like yes/no, true/false, 0/1). It's a classification algorithm, not a regression one, despite the name.

Instead of predicting exact numbers, it predicts **probabilities**, which are then turned into classes.

Formula:



$$P(y=1) = \frac{1}{1+e^{-(b_0+b_1x_1+b_2x_2+\cdots+b_nx_n)}}$$

Example: If you want to predict whether a customer will buy a product (1 = yes, 0 = no), Logistic Regression can help by estimating the chance of purchase.

---

14. What is the confusion matrix?

Answer: A **confusion matrix** is a table that helps you evaluate the performance of a classification model by comparing **actual vs predicted** results.

	Predicted Yes	**Predicted No**
Actual Yes	True Positive (TP)	False Negative (FN)
Actual No	False Positive (FP)	True Negative (TN)

From this, you can calculate:

- **Accuracy**
- **Precision = TP / (TP + FP)**
- **Recall = TP / (TP + FN)**
- **F1 Score = 2 * (Precision * Recall) / (Precision + Recall)**

Example: If your spam filter predicted 80 emails as spam correctly, missed 10 spam, and wrongly flagged 5 good emails, this table helps track those results and calculate performance.

---

15. What is the difference between supervised and unsupervised learning?

Answer:

- **Supervised Learning:** You have labeled data (you know the correct answers). The model learns from this and makes predictions.
  → Examples: Regression, Classification
- **Unsupervised Learning:** You don't have labeled data. The model finds patterns or groups by itself.
  → Examples: Clustering, Dimensionality Reduction

Example:

- **Supervised:** Predicting exam scores based on hours studied.
- **Unsupervised:** Grouping customers based on shopping habits without knowing their labels.

---

16. What is k-Nearest Neighbors (k-NN) algorithm?

Answer: k-NN is a simple machine learning algorithm used for both **classification and regression**. It looks at the **'k' closest data points** to a new input and decides the output based on them.

- For **classification**, it picks the most common class.
- For **regression**, it averages the values.

Key Concept: Distance matters, commonly used distances are **Euclidean** and **Manhattan**.

Example: To classify a fruit as apple or orange, k-NN checks which fruits (based on color, weight, etc.) are closest. If most of the 5 nearest fruits are apples, it says your fruit is an apple.

---

17. What is the bias-variance tradeoff?

Answer: This tradeoff explains how a model's **error** is made up of:

- **Bias:** Error from wrong assumptions (underfitting).
- **Variance:** Error from too much sensitivity to training data (overfitting).

A good model balances both.

Model Type	Bias	Variance
Simple (e.g., linear)	High	Low
Complex (e.g., deep tree)	Low	High

Example:

- A very simple model might miss key patterns (high bias).
- A very complex model might memorize training data (high variance).

---

18. What is a decision tree, and how does it work?

Answer: A **Decision Tree** is a model that splits data into branches based on features to make a decision. Each node asks a **question**, and the leaves give the **output**.

How it works:

- It chooses the best feature to split based on **Gini Impurity**, **Entropy**, or **Information Gain**.
- Continues splitting until a stopping condition is met (like max depth or no further gain).

Example:
To decide if someone gets a loan:

- Step 1: Is income > 50k?
- Step 2: Is credit score > 700?

- Step 3: Approve or deny

It’s like asking yes/no questions until a final decision is reached.

---

19. What is ensemble learning?

Answer: **Ensemble Learning** combines multiple models to improve prediction performance. The idea is that a group of models can perform better than one alone.

Types:

- **Bagging (Bootstrap Aggregating):** Builds many models from different samples (e.g., Random Forest).
- **Boosting:** Builds models one after another, each learning from the last (e.g., XGBoost, AdaBoost).
- **Stacking:** Combines different model types and uses a final model to blend their outputs.

Example: In a school, if 10 teachers vote on whether a student will pass, their combined opinion (ensemble) is more accurate than just one teacher’s guess.

---

20. What is ROC curve and AUC score in classification?

Answer:

- **ROC (Receiver Operating Characteristic) curve** shows how well a model separates positive and negative classes by plotting:
  - **True Positive Rate (Recall)** vs **False Positive Rate**
- **AUC (Area Under Curve):** Measures the total area under the ROC curve. It ranges from 0 to 1.

Interpretation:

- AUC = 1 → Perfect model
- AUC = 0.5 → No better than guessing
- Higher AUC = Better classification

Example: If a disease prediction model has AUC = 0.92, it means the model is very good at telling sick from healthy patients.

21. What is overfitting in machine learning?

Answer: Overfitting happens when a model learns **too much detail from the training data**, including noise or random fluctuations. It performs well on training data but poorly on new or unseen data.

Symptoms:

- High accuracy on training set.
- Low accuracy on test/validation set.

Example: If you're predicting house prices and your model remembers the exact prices from the training houses, it won't predict well for new houses, that's overfitting.

Solutions:

- Use **simpler models**
- **Regularization**
- **Cross-validation**
- **More data**

22. What is underfitting in machine learning?

Answer: Underfitting occurs when a model is **too simple** and **fails to learn patterns** in the training data. It performs poorly on both training and test data.

Symptoms:

- Low accuracy everywhere.
- Model doesn't capture important features.

Example: Using a straight line (linear model) to predict house prices when the real pattern is curved or complex, the model misses important trends.

Solutions:

- Use a more complex model.
- Add more features.
- Reduce bias.

---

23. What is a residual in regression analysis?

Answer: A **residual** is the **difference between the actual value and the predicted value** by the model.

$$\textbf{Residual}=\textbf{Actual value}-\textbf{Predicted value}$$

Example: If the actual house price is $300,000 and the model predicts $280,000, then:

$$\textbf{Residual}=\textbf{300,000}-\textbf{280,000}=\textbf{20,000}$$

Residuals help check how well the model is doing. Ideally, residuals should be small and randomly scattered.

---

24. What is feature scaling and why is it important?

Answer: **Feature scaling** is the process of **standardizing or normalizing** the range of independent variables so they're on a similar scale.

Why it's important:

- Algorithms like **KNN, SVM, Logistic Regression, Gradient Descent** work better when features are scaled.
- Prevents one feature from dominating due to its larger values.

Methods:

- **Min-Max Scaling:** Rescales between 0 and 1.
- **Standardization:** Mean = 0, Standard Deviation = 1

Example: If age is between 18–60 and salary is between 20,000–150,000, then salary can dominate the model unless we scale them.

---

25. What is cross-validation and how does it work?

Answer: Cross-validation is a technique to **test model performance more reliably** by splitting the data into multiple parts.

How it works:

- Split data into **k folds** (e.g., 5 or 10).
- Train on **k-1 folds**, test on the remaining fold.
- Repeat k times with different folds.
- Take the average score.

Common type: K-Fold Cross Validation

Example: With 5-fold CV, the dataset is split into 5 parts. The model is trained on 4 and tested on 1, and this is repeated 5 times. This helps avoid overfitting to a single test set.

---

26. What is feature engineering in machine learning?

Answer: Feature engineering is the process of **creating new input features** or modifying existing ones to improve model performance.

Includes:

- Creating new features (e.g., from timestamps)
- Encoding categories
- Binning continuous variables
- Extracting keywords from text

Example: From a Date column, we can create:

- Day of the week
- Month

- Is_weekend

These new features might help a sales prediction model understand trends.

---

27. What is a classification problem vs regression problem?

Answer:

- **Classification:** Predicts categories or labels (e.g., spam or not spam).
- **Regression:** Predicts continuous values (e.g., house price).

Task	Output Example	Algorithm Type
Classification	Yes/No, 0/1, A/B	Logistic Regression, Decision Tree
Regression	25.5, 100000, etc.	Linear Regression, Random Forest

Example:

- Predicting if a patient has a disease = **Classification**
- Predicting blood pressure = **Regression**

---

28. What is model evaluation and why is it important?

Answer: Model evaluation helps us understand how well a model performs and whether it's useful in real-world situations.

Why important:

- Avoids deploying bad models.
- Helps compare models.

- Identifies overfitting/underfitting.

Metrics used:

- Accuracy, Precision, Recall, F1 Score (for classification)
- RMSE, MAE, R^2 Score (for regression)

Example: Before using a credit scoring model, we evaluate it using precision and recall, we don’t want it to wrongly reject good customers or accept bad ones.

---

29. What is the difference between MAE, MSE, and RMSE in regression?

Answer:

Metric	**Formula**	**Meaning**
MAE (Mean Absolute Error)	(\frac{1}{n} \sum	y - \hat{y}
MSE (Mean Squared Error)	1n∑(y−y^)2\frac{1}{n} \sum (y - \hat{y})^2n1 ∑(y−y^)2	Penalizes larger errors
RMSE (Root MSE)	MSE\sqrt{MSE}MSE	Same units as target variable

Example:
If MAE = 500, then on average, the model is off by $500.
If RMSE = 1000, it means large errors are present since it squares the errors.

---

30. What is gradient descent in machine learning?

Answer: Gradient Descent is an algorithm used to **find the best model parameters** (like slope and intercept in regression) by minimizing the loss (error) function.

How it works:

- Starts with random weights.
- Calculates error.
- Moves in the direction that reduces error.
- Repeats until minimum error is found.

Key term: Learning rate — how big the steps are.

Example: If you're trying to find the lowest point in a valley (minimum error), gradient descent walks step-by-step downhill until it reaches the bottom.

---

31. What is the difference between bagging and boosting in ensemble learning?

Answer:

- **Bagging (Bootstrap Aggregating):** Trains multiple models independently on different subsets of the data (with replacement), then averages their predictions (for regression) or takes a vote (for classification). It reduces variance and prevents overfitting.
- **Example:**
  In Random Forest, multiple decision trees are trained on different parts of the dataset and their predictions are combined.
- **Boosting:** Builds models sequentially, with each model focusing on the errors made by the previous one. It reduces both bias and variance.

Example: In AdaBoost, each new model gives more weight to the data points that were misclassified by the previous model.

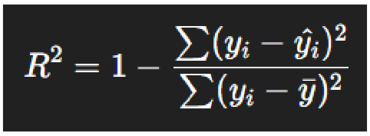
---

32. What is the significance of the R^2 (R-squared) score in regression models?

Answer: The **R^2 score** measures how well the independent variables explain the variation in the dependent variable. It ranges from 0 to 1:

- **0:** No explanation of variance.
- **1:** Perfect explanation of variance.

Formula:



$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

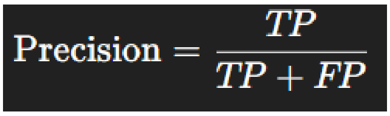
Example: If $R^2 = 0.85$, it means 85% of the variability in the target variable can be explained by the features in the model.

---

33. What is the difference between precision and recall in classification?

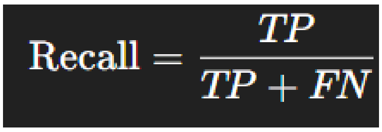
Answer:

- **Precision:** Measures how many of the predicted positive cases are actually positive.



$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall:** Measures how many of the actual positive cases were correctly predicted.



$$\text{Recall} = \frac{TP}{TP + FN}$$

Example:

- **Precision** is important when false positives are costly, like in email spam filtering.
- **Recall** is critical when false negatives are costly, like in cancer detection.

---

34. What is the difference between a hyperparameter and a model parameter?

Answer:

- **Model Parameters** are learned from the data during training (e.g., weights in a neural network, coefficients in linear regression).

- **Hyperparameters** are set before training and control the learning process (e.g., learning rate, number of trees in a Random Forest).

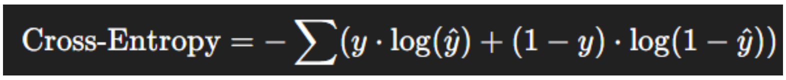
Example:

- **Parameter:** In linear regression, the slope and intercept of the line.
- **Hyperparameter:** In Random Forest, the number of trees in the forest.

35. What is the purpose of cross-entropy loss in classification problems?

Answer: Cross-Entropy Loss is used in **classification problems**, particularly when predicting probabilities. It measures the difference between the predicted probability distribution and the actual distribution (true labels).

Formula:



$$\text{Cross-Entropy} = -\sum(y \cdot \log(\hat{y}) + (1 - y) \cdot \log(1 - \hat{y}))$$

Example: In a binary classification, if the true label is 1 (positive class), and the model predicts 0.9 probability for 1, the cross-entropy will be low because the prediction is close to the true value.

36. What is the difference between the training set, validation set, and test set?

Answer:

- **Training Set:** The data used to train the model.
- **Validation Set:** The data used to tune the model's hyperparameters and assess its performance during training.
- **Test Set:** The data used to evaluate the final performance of the trained model after hyperparameters are set.

Example:

- You train your model on the training set.
- Tune hyperparameters using the validation set.

- Evaluate the final model on the test set.

---

37. What is a random forest and how does it work?

Answer: A **Random Forest** is an ensemble method that builds multiple decision trees and combines their outputs. Each tree is trained on a random subset of data, and the final prediction is made by averaging (regression) or voting (classification) from all trees.

Advantages:

- Reduces overfitting compared to a single decision tree.
- Handles large datasets well.

Example: If you have a dataset of emails, Random Forest can classify each email as spam or not spam by using many decision trees that focus on different aspects (subject, sender, etc.) of the emails.

---

38. What is the purpose of principal component analysis (PCA)?

Answer: **PCA** is a dimensionality reduction technique that transforms high-dimensional data into a smaller set of **uncorrelated variables** called **principal components**. It helps reduce the complexity of the model and retain most of the information.

How it works:

- PCA identifies the directions (principal components) where the data varies the most.
- The first component captures the most variance, the second captures the second-most, and so on.

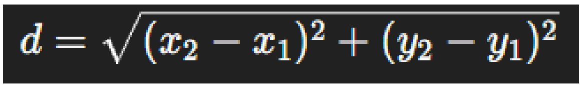
Example: In a dataset with height, weight, and age, PCA might find that the first principal component captures most of the variance, helping reduce the complexity while keeping the important information.

---

39. What is the difference between Euclidean distance and Manhattan distance?

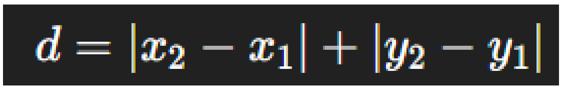
Answer: Both **Euclidean** and **Manhattan distance** are used to measure the distance between two points, but they differ in how they calculate it:

- **Euclidean Distance:** The straight-line distance between two points in a plane (like a direct path).



$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- **Manhattan Distance:** The distance when you move along grid-like paths (only vertical and horizontal moves).



$$d = |x_2 - x_1| + |y_2 - y_1|$$

Example:

- Euclidean: Moving directly from one corner of a square to the opposite corner.
- Manhattan: Moving along the edges of the square, not cutting through the middle.

---

40. What is the purpose of a confusion matrix in classification?

Answer: A **Confusion Matrix** is a table used to evaluate the performance of a classification model. It compares the predicted labels with the actual labels and provides valuable information like:

- **True Positives (TP):** Correctly predicted positive cases.
- **False Positives (FP):** Incorrectly predicted as positive.
- **True Negatives (TN):** Correctly predicted negative cases.
- **False Negatives (FN):** Incorrectly predicted as negative.

From this matrix, you can calculate performance metrics like **accuracy**, **precision**, **recall**, and **F1-score**.

Example: In a medical test for detecting a disease, a confusion matrix helps determine how many sick people were correctly identified, how many healthy people were misdiagnosed, and so on.

42. What is regularization in machine learning and why is it important?

Answer: **Regularization** is a technique used to **penalize large coefficients** in a model, helping prevent overfitting. It adds a penalty term to the loss function based on the size of the coefficients.

Types:

- **L1 Regularization (Lasso):** Adds the absolute value of coefficients.
- **L2 Regularization (Ridge):** Adds the squared value of coefficients.

Example: In linear regression, Lasso or Ridge can reduce the impact of less important features and help improve generalization.

43. What is a decision tree and how does it work?

Answer: A **Decision Tree** is a model that splits the data into different subsets based on the feature values to predict a target variable. Each internal node of the tree represents a test on an attribute, and each branch represents the outcome of the test.

How it works:

- **Splitting:** At each node, the data is split using the best feature (based on metrics like **Gini impurity** or **Information Gain**).
- **Leaf nodes** give the final prediction.

Example: For predicting whether someone will buy a product, a decision tree might split based on Age -> Income -> Interest, and finally classify the outcome.

44. What is feature importance in machine learning?

Answer: Feature Importance is a measure of how much each feature contributes to the predictive power of the model. It helps us understand which features are the most valuable for prediction.

Methods:

- **Random Forest:** Measures how much each feature reduces impurity across all trees.
- **Decision Trees:** Can directly provide feature importance by how frequently they are used in splits.

Example: In a model predicting car prices, mileage and year might have higher importance than color.

---

45. What is the difference between Type I and Type II errors in hypothesis testing?

Answer:

- **Type I Error (False Positive):** Rejecting a true null hypothesis (i.e., saying there's an effect when there isn't).
- **Type II Error (False Negative):** Failing to reject a false null hypothesis (i.e., saying there's no effect when there is).

Example:

- **Type I:** A test says a person has cancer when they don't.
- **Type II:** A test misses detecting cancer in a patient who does have it.

---

46. What is bootstrapping in machine learning?

Answer: Bootstrapping is a resampling technique used to improve model reliability by creating multiple subsets of the data by sampling with replacement.

How it works:

- Randomly sample the data with replacement to create multiple datasets.

- Train the model on each of these datasets and aggregate the results.

Example: In ensemble models like **Bagging** (e.g., Random Forest), bootstrapping helps reduce variance and improves predictions.

---

47. What is the difference between a parametric and non-parametric model?

Answer:

- **Parametric Models:** Assume a specific form for the underlying data distribution (e.g., Linear Regression assumes a linear relationship).
- **Non-Parametric Models:** Make no such assumptions and can model complex data distributions (e.g., Decision Trees, K-Nearest Neighbors).

Example:

- **Parametric:** Linear Regression (assumes data follows a linear pattern).
- **Non-Parametric:** KNN (makes no assumptions about data distribution).

---

48. What is the purpose of the AUC-ROC curve in classification problems?

Answer: The **AUC-ROC curve** helps evaluate the performance of a classification model by plotting the **True Positive Rate (TPR)** vs. the **False Positive Rate (FPR)**.

AUC: Area under the curve, which represents how well the model distinguishes between classes.

- **Higher AUC** means a better model.
- **ROC Curve:** Shows trade-offs between sensitivity and specificity.

Example: A model with an AUC of 0.90 is better than one with an AUC of 0.70 at distinguishing between spam and non-spam emails.

---

49. What is time series forecasting?

Answer: Time Series Forecasting is the process of predicting future values based on past observations. It's used when data points are ordered in time, like stock prices or sales over time.

Common models:

- **ARIMA (AutoRegressive Integrated Moving Average):** Models the relationship between the current value and past values.
- **Exponential Smoothing:** Gives more weight to recent observations.

Example:
Predicting next month's sales based on the past year's monthly sales data.

---

50. What is the significance of the learning rate in machine learning?

Answer: The **learning rate** controls how much the model's weights are adjusted during training. A small learning rate makes gradual improvements, while a large learning rate can make the model jump over the optimal solution.

Example:

- **Small learning rate:** The model learns slowly but steadily.
- **Large learning rate:** The model may skip over the optimal point and fail to converge.
