SUPERVISED LEARNING - TEST 1 (20 to 30 mins)

15-Question Test on Linear Regression, Gradient Descent, Bias & Variance

Topics:

 Linear Regression:

 Gradient Descent:

 Bias & Variance:



Section 1: Linear Regression

Q1. What is Linear Regression?   
Answer: Linear regression is fundamental statistical method used to model the relationship between a dependent variable (what you're trying to predict) and one or more independent variables (the factors influencing the prediction). The idea is simple: you fit a straight line through your data points in a way that minimizes the distance between the actual values and the predicted values.

Q2. Write the formula for simple linear regression. Answer: y = mx + b

where:

* y is the dependent variable,
* mm is the slope of the line,
* x is the independent variable, and
* b is the intercept.

Q3. What is the purpose of the cost function in linear regression?

Answer:

Q4. How do you interpret the coefficients in a multiple linear regression model?

Answer: The **cost function** measures how well the model’s predictions align with the actual data. Its primary role is to quantify the difference between predicted values and actual values, helping to guide the model toward the best-fit line.

Q5. What are the assumptions of Linear Regression?

Answer: **Linearity** – The relationship between the independent variables and the dependent variable must be linear. If the data follows a curved pattern, linear regression may not be the best choice.

**Independence** – Observations should be independent of one another. This means that the errors (residuals) should not be correlated, which is especially important in time-series data.



Section 2: Gradient Descent

Q6. What is Gradient Descent?

Answer: Gradient Descent is an optimization algorithm used to minimize a function by iteratively adjusting its parameters. In the context of **linear regression**, it helps find the best-fit line by minimizing the **cost function** (typically Mean Squared Error)

Q7. Write the formula for parameter update in Gradient Descent.

Answer:

Q8. What is the role of the learning rate in Gradient Descent?

Answer: The **learning rate** (α\alpha) in Gradient Descent controls how big of a step the algorithm takes toward minimizing the cost function in each iteration. It plays a critical role in determining the efficiency and accuracy of the optimization process.



Q9. What is the primary purpose of regularization in machine learning models?

A) To increase the complexity of the model to fit the training data better.

**B) To minimize the training error without regard to generalization.**

C) To prevent overfitting by penalizing large coefficients in the model.

D) To ensure that all features are included in the final model regardless of their importance.

Q10. What happens if the learning rate is too small or too large?

Answer: The learning rate (

𝛼

) in Gradient Descent controls the size of steps the algorithm takes toward minimizing the cost function. Choosing an appropriate learning rate is crucial, as an improper value can lead to inefficient training or failure to converge.

**Effects of a Learning Rate That’s Too Small**

**Slow convergence** – The model takes very tiny steps toward the optimal solution, requiring a large number of iterations.

**Risk of getting stuck** – The algorithm might stop improving if it reaches a local minimum too slowly.

**Computational inefficiency** – Training takes longer, which can be costly for complex models.

**Effects of a Learning Rate That’s Too Large**

**Overshooting** – Instead of moving toward the minimum, the algorithm jumps over it repeatedly.

**Divergence** – The cost function may actually increase instead of decrease, preventing convergence.

**Unstable training** – The model oscillates wildly between values and fails to settle on an optimal solution.



Section 3: Bias & Variance

Q11. Define Bias and Variance in the context of machine learning models.

Answer: Bias and variance are key concepts in machine learning that help explain how well a model generalizes to new data. They represent two types of errors that can occur when training a model:

### Bias

* Refers to the error introduced by assumptions made in the learning algorithm.
* High bias means the model is **too simplistic**, ignoring underlying patterns in the data (leading to **underfitting**).
* Example: A linear model trying to fit highly non-linear data will have high bias.

### Variance

* Measures how much the model's predictions fluctuate for different training datasets.
* High variance means the model is **too complex**, capturing noise instead of general patterns (leading to **overfitting**).
* Example: A deep neural network memorizing training data instead of learning true relationships will have high variance.

Q12. What is the Bias-Variance tradeoff?

Answer: The **Bias-Variance Tradeoff** is a fundamental concept in machine learning that describes the balance between two sources of error that affect model performance: **bias** and **variance**.

### Understanding the Tradeoff

1. **High Bias, Low Variance (Underfitting)**
   * The model makes strong assumptions about the data.
   * It oversimplifies patterns, leading to poor accuracy.
   * Example: A linear model used for complex, non-linear data.
2. **Low Bias, High Variance (Overfitting)**
   * The model learns too much from training data, including noise.
   * It performs well on training data but poorly on unseen data.
   * Example: A deep neural network memorizing training data instead of generalizing

Q13. How does increasing the complexity of a model affect bias and variance?

Answer: **Effects of Higher Complexity**

* **Bias ↓ (Lower Bias)** – A more complex model can better capture intricate patterns in the data, reducing the error caused by overly simplistic assumptions.
* **Variance ↑ (Higher Variance)** – More complexity means the model becomes highly sensitive to training data, which can lead to **overfitting**—where the model memorizes noise instead of learning general trends.

Q14. What is underfitting and overfitting in machine learning? Answer: **Underfitting** and **overfitting** describe two key issues in machine learning that affect a model's ability to generalize to unseen data.

### Underfitting (High Bias, Low Variance)

* Occurs when the model is too simple to capture the underlying patterns in the data.
* The model performs poorly on both training and test data.
* Example: Using **linear regression** to fit highly non-linear data, missing key trends.
* Solution: Increase model complexity, add more relevant features, or reduce bias.

### Overfitting (Low Bias, High Variance)

* Happens when the model learns too much from the training data—including noise.
* The model performs well on training data but poorly on new, unseen data.
* Example: A deep neural network memorizing training examples instead of generalizing patterns.

Q15. How can you reduce overfitting in a model?

Answer: Reducing **overfitting** is crucial for ensuring a machine learning model generalizes well to unseen data. Here are several effective strategies:

### 1. Regularization

* **L1 Regularization (Lasso)**: Encourages sparsity by pushing some coefficients to zero.
* **L2 Regularization (Ridge)**: Penalizes large weights, preventing excessive reliance on specific features.

### 2. Cross-Validation

* Use **k-fold cross-validation** to evaluate model performance on different subsets of data, ensuring it doesn't rely too heavily on any single portion.

### 3. Reduce Model Complexity

* Use **simpler models** when appropriate (e.g., avoid overly deep neural networks if the problem doesn’t require them).
* **Feature selection** helps remove irrelevant or redundant features that contribute to overfitting.

### 4. Increase Training Data

* More data can help the model learn true patterns instead of memorizing noise.
* Data **augmentation** can artificially expand datasets in cases like image processing.

### 5. Early Stopping

* Monitor model performance during training and stop when validation loss starts increasing, preventing excessive memorization.

### 6. Dropout (for Neural Networks)

* Randomly **drop units** in the network during training to make the model more robust and prevent dependence on specific features.

### 7. Ensemble Methods

* Combining multiple models (e.g., **bagging, boosting**) helps smooth out individual model biases and reduces overfitting.