Linear Regression Interview – In-Depth Revision Guide

This document provides detailed, interview-ready explanations for core Linear Regression concepts, combining theory, intuition, and practical considerations.

1. What is Linear Regression?

Linear regression is a supervised learning algorithm used to predict a continuous target variable by modeling the linear relationship between input features and the output. It assumes that the expected value of the target is a linear combination of the input features. Mathematically, it is expressed as $y = w■ + w■x■ + w■x■ + \ldots + w■x■ + \varepsilon$, where w are learnable parameters and ε is the error term. It is widely used because it is simple, interpretable, fast to train, and works well when the relationship is approximately linear.

2. What cost function is used in Linear Regression and why?

Linear regression typically uses Mean Squared Error (MSE) as the cost function, defined as the average of $(y - ■)^2$ over all samples. MSE is convex, differentiable, and strongly penalizes large errors, which helps in finding a stable global minimum. Its convex nature guarantees that optimization methods like gradient descent converge to a unique solution.

3. How are parameters learned? (Optimization)

Parameters are learned using Gradient Descent or its variants. The idea is to iteratively update the weights in the opposite direction of the gradient of the loss function. This means moving downhill on the error surface to reduce prediction error. The update rule is $w = w - \alpha \cdot \partial J/\partial w$, where α is the learning rate. Because the MSE loss for linear regression is convex, gradient descent is guaranteed to reach the global minimum.

4. Assumptions of Linear Regression

Linear regression relies on several key assumptions: 1) Linearity – the relationship between features and target is linear. 2) Independence – errors are independent of each other. 3) Homoscedasticity – constant variance of errors across all levels of the input. 4) Normality of errors – residuals are normally distributed, especially important for confidence intervals. 5) No multicollinearity – features should not be highly correlated with each other. Violating these assumptions can lead to biased, unstable, or unreliable models.

5. What is overfitting and how do you prevent it?

Overfitting occurs when a model learns noise and patterns specific to the training data instead of the underlying relationship, leading to poor performance on unseen data. In linear regression, this often happens when there are too many features relative to the amount of data. It can be prevented by using regularization (Ridge or Lasso), reducing feature count, applying cross-validation, and collecting more data. The goal is to improve generalization, not just training accuracy.

6. Ridge vs Lasso Regression

Both Ridge and Lasso are regularization techniques used to prevent overfitting. Ridge regression uses L2 regularization ($\lambda\sum w^2$), which shrinks coefficients but does not set them exactly to zero. It is useful when most features are relevant and multicollinearity is present. Lasso regression uses L1 regularization ($\lambda\sum |w|$), which can force some coefficients to exactly zero, performing automatic feature selection. Ridge is preferred for stability, while Lasso is preferred for sparsity and interpretability.

7. What is multicollinearity and why is it a problem?

Multicollinearity occurs when two or more input features are highly correlated with each other. This makes coefficient estimates unstable and highly sensitive to small changes in data. It does not significantly harm prediction accuracy but severely impacts interpretability and reliability of feature importance. It can be detected using correlation matrices or Variance Inflation Factor (VIF) and handled by removing features, using Ridge regression, or applying PCA.

8. Bias–Variance Tradeoff

Bias is error caused by overly simplistic assumptions that lead to underfitting. Variance is error caused by excessive sensitivity to training data that leads to overfitting. The bias–variance tradeoff refers to balancing these two to achieve good generalization. In linear regression, adding regularization increases bias slightly but significantly reduces variance, which often improves performance on unseen data.

9. Why is feature scaling important?

Feature scaling is important because gradient descent converges much faster when all features are on similar scales. Without scaling, the cost surface becomes elongated, causing slow zig-zag convergence. Scaling ensures that each feature contributes proportionately to the loss function. Common methods include standardization (Z-score normalization) and Min–Max scaling.

10. When should you NOT use Linear Regression?

Linear regression is not suitable for classification tasks because it does not output probabilities. It is not used for clustering because it is a supervised algorithm. It performs poorly on highly non-linear relationships because it cannot capture complex patterns. In such cases, logistic regression, tree-based models, kernel methods, or neural networks are better choices.

Bonus: Interview Power Statements

• Ridge handles multicollinearity well by distributing weight across correlated features. • Lasso performs embedded feature selection by zeroing weak coefficients. • Regularization increases bias slightly but reduces variance significantly. • Linear regression is fast, interpretable, and easy to monitor in production. • Multicollinearity hurts interpretability more than accuracy.