Machine Learning Algorithms – Interview Preparation Handbook

This handbook covers 10 essential Machine Learning algorithms you need for interviews. For each algorithm, we cover: - What it is (simple explanation) - How it works (step-by-step intuition) - Math behind it (easy to remember equations) - Interview Q&A; (mock questions you may face) Follow one algorithm per day for maximum retention.

■ Linear Regression

What it is: Supervised algorithm for predicting continuous values (e.g., house prices).

How it works: Assumes linear relationship between input features and output. Fits a straight line (or hyperplane).

Math intuition:
Equation: y = w1*x1 + w2*x2 + ... + wn*xn + b
Loss: Mean Squared Error (MSE) = (1/N)Σ(y - ■)^2
Optimization: Gradient Descent to update weights.

Interview Q&A;:
- What assumptions does Linear Regression make?
- How do you handle multicollinearity?
- Difference between Linear and Logistic Regression?

■ Logistic Regression

What it is: Used for classification problems (yes/no, spam/not spam).

How it works: Predicts probability using sigmoid function. Threshold decides the class.

Math intuition:
Sigmoid: P(y=1|x) = 1 / (1 + e^(-(wx+b)))
Loss: Log Loss = -(1/N)Σ(y log(■) + (1-y) log(1-■))

Interview Q&A;:
- Why can’t we use Linear Regression for classification?
- Explain the sigmoid function.
- How to handle imbalanced data?

■ Decision Trees

What it is: Tree-like structure for both classification and regression.

How it works: Splits data into subsets using conditions. Chooses best split based on purity.

Math intuition:
Entropy = -Σ p log2(p)
Info Gain = Entropy(parent) - Σ (Nj/N)*Entropy(child_j)

Interview Q&A;:
- Difference between Gini Index and Entropy?
- How to avoid overfitting in Decision Trees?
- Advantages/Disadvantages of Decision Trees?

■ Random Forest

What it is: Ensemble of decision trees (bagging technique).

How it works: Builds multiple trees on random subsets of data + features, aggregates results.

Math intuition:
Reduces variance by averaging predictions.

Interview Q&A;:
- Why does Random Forest reduce overfitting compared to a single tree?
- Difference between Bagging and Boosting?
- How to measure feature importance?

■ Gradient Boosting (XGBoost, LightGBM, CatBoost)

What it is: Ensemble boosting method for strong predictive power.

How it works: Builds trees sequentially, each correcting previous errors.

Math intuition:
Prediction: $y = \Sigma$ Tree_m(x)
Optimized using Gradient Descent on residuals.

Interview Q&A;:
- Difference between Bagging and Boosting?
- Why is XGBoost faster than other boosting algorithms?
- How do you tune hyperparameters?

■ K-Nearest Neighbors (KNN)

What it is: Lazy learning algorithm for classification/regression.

How it works: Predicts based on closest K neighbors (majority vote or average).

Math intuition:
Distance: Euclidean = sqrt(Σ(xi - yi)^2)

Interview Q&A;:
- What is the effect of K on bias/variance?
- How to choose distance metric?
- Is KNN sensitive to outliers?

■ Support Vector Machine (SVM)

What it is: Classification algorithm that finds best separating hyperplane.

How it works: Maximizes margin between classes. Can use kernel trick for non-linear boundaries.

Math intuition:
Optimization: min ||w||^2 subject to yi(w·xi+b) ≥ 1

Interview Q&A;:
- What are support vectors?
- Explain kernel trick.
- Difference between hard margin and soft margin?

■ Naive Bayes

What it is: Probabilistic classifier using Bayes Theorem with independence assumption.

How it works: Class with highest probability chosen based on conditional probabilities.

Math intuition:
P(y|x) = (P(x|y) * P(y)) / P(x)

Interview Q&A;:
- Why is it called 'naive'?
- Where is Naive Bayes commonly used?
- Limitation of Naive Bayes?

■ K-Means Clustering

What it is: Unsupervised learning algorithm for grouping data.

How it works: Iteratively assigns points to nearest centroid, updates centroids until convergence.

Math intuition:
Objective: minimize WCSS = $\Sigma \Sigma ||x - \mu||^2$

Interview Q&A;:
- How do you choose K?
- Limitations of K-Means?
- Difference between K-Means and Hierarchical clustering?

■ Principal Component Analysis (PCA)

What it is: Dimensionality reduction technique.

How it works: Transforms features into new axes (principal components) capturing maximum variance.

Math intuition:
Steps: Compute covariance matrix → Eigenvectors & Eigenvalues → Sort → Project data.

Interview Q&A;:
- Difference between PCA and Feature Selection?
- What do eigenvectors/eigenvalues represent in PCA?
- When would you use PCA?