Machine Learning - 100 Answers

Pages 12+ Part 1: With Explanations

Nova IMS
Generated: January 16, 2026

LEVEL 1: CATEGORICAL ENCODING BASICS

Q1. Why do machine learning algorithms require numerical inputs?

ANSWER: B

Explanation: Machine learning algorithms perform mathematical operations (addition, multiplication, matrix operations) that require numerical inputs. Categorical text values cannot be directly used in these calculations.

Q2. What is label encoding?

ANSWER: B

Explanation: Label encoding assigns sequential integers (0, 1, 2, 3...) to each unique category, converting categorical text into numbers that algorithms can process.

Q3. What problem does label encoding create for nominal variables?

ANSWER: B

Explanation: Label encoding creates an artificial ordinal relationship (0 < 1 < 2) where none exists for nominal variables. For example, encoding [Red, Blue, Green] as [0, 1, 2] incorrectly implies Red < Blue < Green.

Q4. When is label encoding appropriate?

ANSWER: B

Explanation: Label encoding is appropriate for ordinal variables (which have natural order like Small < Medium < Large) or tree-based models (which aren't affected by the numerical ordering).

Q5. What is one-hot encoding?

ANSWER: B

Explanation: One-hot encoding creates a separate binary (0/1) column for each category. For example, Color=[Red, Blue, Green] becomes three columns: is_Red, is_Blue, is_Green.

Q6. What is the main advantage of one-hot encoding?

ANSWER: B

Explanation: One-hot encoding treats each category independently without implying any order, making it ideal for nominal (unordered) categorical variables like color, country, or product type.

Q7. What is the main disadvantage of one-hot encoding?

ANSWER: B

Explanation: One-hot encoding creates one new column per category. With high-cardinality features (e.g., 1000 unique cities), this creates 1000 new columns, dramatically increasing dimensionality.

Q8. For how many categories is one-hot encoding typically appropriate?

ANSWER: B

Explanation: One-hot encoding is practical when you have relatively few categories (typically <15-20). With more categories, the dimensionality explosion becomes problematic.

Q9. What is the "dummy variable trap" in one-hot encoding?

ANSWER: B

Explanation: The dummy variable trap occurs when all dummy variables are included, creating perfect multicollinearity (one column can be predicted from others). For k categories, you only need k-1 dummy variables.

Q10. How do you avoid the dummy variable trap?

ANSWER: B

Explanation: Drop one dummy variable (typically the first or last) because with k-1 dummy variables, the kth category is implicitly represented when all k-1 are zero.

Q11. What is target encoding (mean encoding)?

ANSWER: B

Explanation: Target encoding replaces each category with the mean (or other aggregate) of the target variable for that category. For example, City='NYC' might become 75,000 (average salary in NYC).

Q12. What is the main advantage of target encoding?

ANSWER: B

Explanation: Target encoding captures the relationship between categories and the target variable, often providing high predictive power because it directly encodes target information.

Q13. What is the main risk of target encoding?

ANSWER: B

Explanation: Target encoding can cause severe target leakage and overfitting because it uses target variable information during encoding. The encoding itself contains information about the target, leading to unrealistically good training performance.

Q14. What is frequency encoding?

ANSWER: B

Explanation: Frequency encoding replaces each category with how often it appears in the dataset. Common categories get higher values, rare ones get lower values.

Q15. What is binary encoding used for?

ANSWER: B

Explanation: Binary encoding converts categories to binary numbers, then splits the binary digits into separate columns. This reduces dimensionality compared to one-hot (log■(n) columns vs n columns).

LEVEL 2: FEATURE ENGINEERING & TRANSFORMATION

Q16. What is feature engineering?

ANSWER: B

Explanation: Feature engineering creates new features from existing data using domain knowledge and creativity. It's about extracting more information from available data to improve model performance.

Q17. What can be extracted from date/time features?

ANSWER: B

Explanation: From date/time features, you can extract year, month, day, day_of_week (Mon-Sun), hour, minute, is_weekend, is_holiday, season, and time-based cyclical features (sin/cos transformations).

Q18. What features can be extracted from text data?

ANSWER: B

Explanation: Text data can yield character count, word count, average word length, sentiment scores, TF-IDF vectors, n-grams, part-of-speech tags, and named entity counts.

Q19. What are ratio features in feature engineering?

ANSWER: B

Explanation: Ratio features create meaningful relationships like debt-to-income ratio, price-per-square-foot, clicks-per-impression, or revenue-per-customer that often have stronger predictive power than raw features.

Q20. What are aggregation features?

ANSWER: B

Explanation: Aggregation features compute statistics (mean, median, std, min, max, count) grouped by categories. For example, average_purchase_by_customer or total_sales_by_region.

Q21. What are interaction features?

ANSWER: B

Explanation: Interaction features capture relationships between features by multiplying or combining them. For example, bedrooms × bathrooms or combining age and income for age-income segments.

Q22. What is polynomial feature transformation?

ANSWER: B

Explanation: Polynomial features create combinations like x^2, x^3, x■×x■. For 2 features and degree 2, you get: x■, x■, x■^{2}, x■^{2}, x■×x■.

Q23. What is a disadvantage of polynomial features?

ANSWER: B

Explanation: Polynomial features dramatically increase dimensionality. For n features and degree d, you can get O(n■) features, causing curse of dimensionality and overfitting with high degrees.

Q24. What is binning (discretization)?

ANSWER: B

Explanation: Binning converts continuous variables into categorical ranges (bins). For example, age → [child, teen, adult, senior] or income → [low, medium, high].

Q25. When is log transformation useful?

ANSWER: B

Explanation: Log transformation is useful for right-skewed distributions (long right tail) because it compresses large values more than small values, reducing the impact of extreme outliers.

Q26. What does log transformation do to multiplicative relationships?

ANSWER: B

Explanation: Log transformation converts multiplicative relationships to additive ones (log(a×b) = log(a) + log(b)), which linear models can handle more easily.

Q27. What is Box-Cox transformation?

ANSWER: B

Explanation: Box-Cox transformation automatically finds the optimal power transformation (λ) to make data more normal. It's a family of transformations: $y(\lambda) = (y\lambda - 1)/\lambda$.

Q28. What limitation does Box-Cox have?

ANSWER: B

Explanation: Box-Cox requires strictly positive values (y > 0) because it involves power transformations that are undefined for zero or negative values.

Q29. What is Yeo-Johnson transformation?

ANSWER: B

Explanation: Yeo-Johnson is similar to Box-Cox but extended to handle zero and negative values, making it more flexible for real-world data that may include zero or negative values.

Q30. What does automated feature engineering using Featuretools do?

ANSWER: B

Explanation: Featuretools performs automated deep feature synthesis by automatically generating features through aggregations, transformations, and combinations across related tables.

Q31. What role does domain knowledge play in feature engineering?

ANSWER: B

Explanation: Domain knowledge is essential for identifying meaningful feature interactions, understanding what ratios make sense, recognizing important time patterns, and avoiding spurious features.

Q32. What is the BMI formula as an example of feature engineering?

ANSWER: B

Explanation: BMI (Body Mass Index) = weight(kg) / height²(m²) is a classic feature engineering example, combining two measurements into a medically meaningful metric.

Q33. What are lagged features in time series?

ANSWER: B

Explanation: Lagged features use previous time point values as features. For example, sales_lag_1 (yesterday's sales), temperature_lag_7 (temperature 7 days ago) to capture temporal patterns.

Q34. What are rolling window features?

ANSWER: B

Explanation: Rolling window features compute statistics over moving time windows. For example, 7-day moving average of sales, 30-day rolling standard deviation of stock prices.

Q35. Why is feature engineering considered both art and science?

ANSWER: B

Explanation: Feature engineering requires both creativity (art) to imagine useful features and technical skill (science) to implement them correctly, understand their statistical properties, and validate their utility.

LEVEL 3: FEATURE SELECTION METHODS

Q36. Why is feature selection important?

ANSWER: B

Explanation: Feature selection reduces dimensionality, decreases overfitting (fewer parameters to learn), improves model performance, reduces training time, and improves interpretability by focusing on important features.

Q37. What are the three main categories of feature selection methods?

ANSWER: B

Explanation: The three categories are: Filter methods (statistical tests, model-independent), Wrapper methods (use model performance, model-dependent), and Embedded methods (built into model training).

Q38. What are filter methods in feature selection?

ANSWER: B

Explanation: Filter methods use statistical tests independent of any ML model. They evaluate features based on statistical properties like correlation, variance, or information gain.

Q39. What does variance threshold do?

ANSWER: B

Explanation: Variance threshold removes features with near-constant values (low variance). If a feature has the same value for most samples, it provides little information for distinguishing between samples.

Q40. Why remove low-variance features?

ANSWER: B

Explanation: Near-constant features (like a gender column that's 99% male) provide minimal discriminative information and can cause numerical instability in some algorithms.

Q41. What does correlation analysis identify?

ANSWER: B

Explanation: Correlation analysis identifies highly correlated features (multicollinearity). When two features are highly correlated (>0.9), they provide redundant information, and one can be removed.

Q42. What is the chi-squared test used for in feature selection?

ANSWER: B

Explanation: Chi-squared test measures dependence between categorical features and categorical target. Higher chi-squared scores indicate stronger relationships and more predictive features.

Q43. What is the ANOVA F-test used for?

ANSWER: B

Explanation: ANOVA F-test measures whether means of a numerical feature differ significantly across categorical target classes. Higher F-scores indicate features that differentiate classes well.

Q44. What does mutual information measure?

ANSWER: B

Explanation: Mutual information measures how much information one variable provides about another, capturing both linear and non-linear relationships (unlike correlation which only captures linear).

Q45. What is an advantage of filter methods?

ANSWER: B

Explanation: Filter methods are fast (no model training), model-agnostic (work with any algorithm), and have no overfitting risk since they don't use the model.

Q46. What is a disadvantage of filter methods?

ANSWER: B

Explanation: Filter methods evaluate features individually (univariate), ignoring feature interactions and combinations that might be predictive together even if individually weak.

Q47. What are wrapper methods?

ANSWER: B

Explanation: Wrapper methods use actual model performance to evaluate feature subsets. They search through different feature combinations and select the best performing subset.

Q48. What is forward selection?

ANSWER: B

Explanation: Forward selection starts with an empty feature set and iteratively adds the feature that improves model performance the most, until no improvement or stopping criterion is met.

Q49. What is backward elimination?

ANSWER: B

Explanation: Backward elimination starts with all features and iteratively removes the feature whose removal hurts performance the least, until performance degrades significantly.

Q50. What is Recursive Feature Elimination (RFE)?

ANSWER: B

Explanation: RFE (Recursive Feature Elimination) trains a model, ranks features by importance, removes the least important features, and repeats recursively until desired number of features remains.

Q51. What is an advantage of wrapper methods?

ANSWER: B

Explanation: Wrapper methods consider feature interactions because they evaluate actual model performance with different feature combinations, capturing synergistic effects.

Q52. What is a disadvantage of wrapper methods?

ANSWER: B

Explanation: Wrapper methods require training many models (one per feature subset evaluated), making them computationally expensive. They also risk overfitting to the validation set.

Q53. What are embedded methods?

ANSWER: B

Explanation: Embedded methods perform feature selection as part of the model training process itself, integrated into the learning algorithm.

Q54. How does Lasso (L1) perform feature selection?

ANSWER: B

Explanation: Lasso (L1 regularization) adds a penalty proportional to the absolute value of coefficients. As the penalty increases, some coefficients shrink exactly to zero, effectively removing those features.

Q55. How do tree-based models provide feature importance?

ANSWER: B

Explanation: Tree-based models calculate feature importance based on how much each feature decreases impurity (Gini or entropy) when used for splits, weighted by the number of samples affected.

LEVEL 4: DIMENSIONALITY REDUCTION & TRAIN/TEST SPLITTING

Q56. What is the difference between feature selection and feature extraction?

ANSWER: B

Explanation: Feature selection keeps original features (selects subset), while feature extraction creates new features (linear combinations like PCA). Selection preserves interpretability, extraction may perform better.

Q57. What does PCA (Principal Component Analysis) do?

ANSWER: B

Explanation: PCA creates new uncorrelated features (principal components) that are linear combinations of original features, ordered by how much variance they explain.

Q58. Are PCA components the same as original features?

ANSWER: B

Explanation: No, PCA components are linear combinations (weighted sums) of original features. PC1 might be 0.5×feature1 + 0.3×feature2 + 0.2×feature3, making them harder to interpret.

Q59. What does LDA (Linear Discriminant Analysis) maximize?

ANSWER: B

Explanation: LDA (Linear Discriminant Analysis) finds linear combinations that maximize separation between classes. Unlike PCA (maximizes variance), LDA is supervised and maximizes class separability.

Q60. What is t-SNE used for?

ANSWER: B

Explanation: t-SNE is a non-linear technique for visualizing high-dimensional data in 2D or 3D. It preserves local structure (nearby points stay nearby) but is primarily for visualization, not modeling.

Q61. What is UMAP?

ANSWER: B

Explanation: UMAP (Uniform Manifold Approximation and Projection) is faster than t-SNE and better preserves global structure while maintaining local relationships. It's more suitable for large datasets.

Q62. What are autoencoders?

ANSWER: B

Explanation: Autoencoders are neural networks trained to compress data (encoder) and reconstruct it (decoder). The compressed representation (bottleneck layer) serves as reduced-dimension features.

Q63. Why split data into training and test sets?

ANSWER: B

Explanation: Splitting data into train/test sets allows us to evaluate model performance on unseen data, providing an estimate of how it will generalize to new real-world data.

Q64. What is the typical train/test split ratio?

ANSWER: B

Explanation: The most common ratios are 80/20 (80% train, 20% test) or 70/30. For smaller datasets, 60/20/20 (train/validation/test) is common.

Q65. What is stratified sampling?

ANSWER: B

Explanation: Stratified sampling ensures that the class distribution (proportion of each class) is maintained in both training and test sets, preventing biased splits.

Q66. When is stratified sampling essential?

ANSWER: B

Explanation: Stratified sampling is essential for imbalanced datasets to ensure the minority class is adequately represented in both train and test sets, preventing test sets with no minority class samples.

Q67. What is a train/validation/test split?

ANSWER: B

Explanation: A three-way split: training (for learning parameters), validation (for hyperparameter tuning), and test (for final unbiased evaluation). For example, 60% train, 20% validation, 20% test.

Q68. What is the validation set used for?

ANSWER: B

Explanation: The validation set is used for hyperparameter tuning and model selection. You can evaluate many times on validation data to optimize hyperparameters without biasing the final test evaluation.

Q69. What is the test set used for?

ANSWER: B

Explanation: The test set should be used only once at the very end for final unbiased performance evaluation. Using it multiple times causes overfitting to the test set.

Q70. What is time series splitting?

ANSWER: B

Explanation: Time series splitting maintains temporal order: training data comes before validation, which comes before test. Random splitting would cause look-ahead bias (using future to predict past).

LEVEL 5: CROSS-VALIDATION TECHNIQUES

Q71. What is cross-validation?

ANSWER: B

Explanation: Cross-validation creates multiple train/test splits to get a more robust and reliable estimate of model performance, reducing the impact of any single lucky or unlucky split.

Q72. What is K-Fold cross-validation?

ANSWER: B

Explanation: K-Fold divides data into K equal-sized folds. For each iteration, train on K-1 folds and validate on the remaining fold. Repeat K times, averaging the K performance scores.

Q73. What are typical values for K in K-Fold CV?

ANSWER: B

Explanation: K=5 or K=10 are most common, balancing computational cost with reliable estimates. K=5 is faster but K=10 gives slightly more reliable estimates.

Q74. How is the final cross-validation score calculated?

ANSWER: B

Explanation: Average the performance metric (accuracy, RMSE, etc.) across all K folds. This average is a more robust estimate than a single train/test split.

Q75. What is stratified K-Fold?

ANSWER: B

Explanation: Stratified K-Fold ensures each fold maintains the same class distribution as the original dataset. Each fold has the same proportion of classes.

Q76. When is stratified K-Fold essential?

ANSWER: B

Explanation: For imbalanced data, stratified K-Fold prevents folds with no samples from the minority class, ensuring every fold can properly evaluate performance on all classes.

Q77. What is Leave-One-Out Cross-Validation (LOOCV)?

ANSWER: B

Explanation: LOOCV (Leave-One-Out Cross-Validation) is K-Fold where K=n (number of samples). Each sample serves as a one-sample test set once, with all others as training.

Q78. When is LOOCV appropriate?

ANSWER: B

Explanation: LOOCV is appropriate only for very small datasets (<100 samples) where you need to maximize training data. It's computationally expensive (n model trainings).

Q79. What is repeated K-Fold CV?

ANSWER: B

Explanation: Repeated K-Fold runs K-Fold CV multiple times with different random shuffles. For example, 5-fold repeated 10 times = 50 train/test splits, giving even more robust estimates.

Q80. What is time series cross-validation?

ANSWER: B

Explanation: Time series CV uses forward-chaining: train on [1:100], test on [101:150]; train on [1:150], test on [151:200]; etc. The training window progressively grows, respecting temporal order.

Q81. Why can't you use standard K-Fold for time series?

ANSWER: B

Explanation: Standard K-Fold randomly assigns samples to folds, which would allow future data points to appear in training when predicting past, creating unrealistic look-ahead bias.

Q82. What is Group K-Fold?

ANSWER: B

Explanation: Group K-Fold ensures all samples from the same group (e.g., all scans from one patient) stay together in the same fold, preventing leakage between related samples.

Q83. When is Group K-Fold needed?

ANSWER: B

Explanation: When samples are not independent (e.g., multiple measurements from same patient, photos of same person), splitting groups prevents information leakage from one fold to another.

Q84. What is nested cross-validation?

ANSWER: B

Explanation: Nested CV has two loops: outer loop for unbiased performance estimation, inner loop for hyperparameter tuning. This prevents hyperparameter optimization from biasing performance estimates.

Q85. Why is nested CV considered the gold standard?

ANSWER: B

Explanation: Nested CV provides truly unbiased performance estimates because hyperparameter tuning is done completely separately for each outer fold, preventing optimization from contaminating the evaluation.

LEVEL 6: CLASSIFICATION METRICS - CONFUSION MATRIX

Q86. What does a confusion matrix show?

ANSWER: B

Explanation: A confusion matrix displays actual classes on one axis and predicted classes on the other, showing counts for True Positives, False Positives, True Negatives, and False Negatives.

Q87. What are True Positives (TP)?

ANSWER: B

Explanation: True Positives (TP) are cases correctly predicted as positive - the model predicted positive AND the actual class is positive.

Q88. What are False Positives (FP)?

ANSWER: B

Explanation: False Positives (FP) are negative cases incorrectly predicted as positive - the model predicted positive BUT the actual class is negative. This is a Type I error.

Q89. What are False Negatives (FN)?

ANSWER: C

Explanation: False Negatives (FN) are positive cases incorrectly predicted as negative - the model predicted negative BUT the actual class is positive. This is a Type II error.

Q90. What are True Negatives (TN)?

ANSWER: D

Explanation: True Negatives (TN) are cases correctly predicted as negative - the model predicted negative AND the actual class is negative.

Q91. What is a Type I Error?

ANSWER: B

Explanation: Type I Error is a False Positive - rejecting a true null hypothesis, or incorrectly predicting positive when it's actually negative (false alarm).

Q92. What is a Type II Error?

ANSWER: B

Explanation: Type II Error is a False Negative - failing to reject a false null hypothesis, or incorrectly predicting negative when it's actually positive (missed detection).

Q93. What is accuracy in classification?

ANSWER: B

Explanation: Accuracy = (TP + TN) / (TP + TN + FP + FN), the proportion of all predictions that are correct (both positive and negative).

Q94. Why is accuracy misleading for imbalanced datasets?

ANSWER: B

Explanation: With imbalanced classes (e.g., 95% negative), a model can achieve 95% accuracy by always predicting negative, without learning anything useful about the minority positive class.

Q95. If 95% of emails are not spam, what accuracy can you get by always predicting "not spam"?

ANSWER: B

Explanation: By always predicting 'not spam' (the majority class), you'd achieve 95% accuracy without any learning. This model is useless for catching actual spam (the important 5%).

Q96. What is precision in classification?

ANSWER: B

Explanation: Precision = TP / (TP + FP), which answers: 'Of all cases predicted as positive, what proportion were actually positive?' It measures the accuracy of positive predictions.

Q97. What question does precision answer?

ANSWER: B

Explanation: Precision answers: 'When the model predicts positive, how often is it right?' This is critical when false positives are costly.

Q98. What is recall (sensitivity)?

ANSWER: B

Explanation: Recall = TP / (TP + FN), which answers: 'Of all actual positive cases, what proportion did we correctly identify?' It measures completeness of positive detection.

Q99. What question does recall answer?

ANSWER: B

Explanation: Recall answers: 'Of all actual positive cases, how many did we catch?' This is critical when missing positives (false negatives) is costly.

Q100. Which metric focuses on minimizing false positives?

ANSWER: B

Explanation: Precision focuses on minimizing false positives by being selective about positive predictions. High precision means few false alarms when predicting positive.