

Machine Learning - 100 Questions

Pages 12+ Part 4

Nova IMS
Generated: January 20, 2026

LEVEL 1: SUPPORT VECTOR MACHINES DETAILED

Q1. What is a hard margin SVM?

- A) Allows misclassifications
- B) All points correctly classified, only for linearly separable data
- C) Most flexible
- D) Best for all problems

Q2. What is a soft margin SVM?

- A) No misclassifications allowed
- B) Allows some misclassifications with penalty (for non-separable data)
- C) Only for linearly separable
- D) No regularization

Q3. What does the C parameter control in SVM?

- A) Kernel width
- B) Tradeoff between margin size and misclassification penalty
- C) Number of support vectors exactly
- D) Training speed

Q4. What happens with large C value?

- A) Wide margin, more errors tolerated
- B) Narrow margin, fewer errors allowed (risk of overfitting)
- C) No effect
- D) Model cannot train

Q5. What happens with small C value?

- A) Narrow margin, fewer errors
- B) Wide margin, more errors tolerated (more regularization)
- C) Perfect fit always
- D) Cannot classify

Q6. What is the linear kernel?

- A) $K(x, y) = x + y$
- B) $K(x, y) = x^T y$ (dot product, no transformation)
- C) $K(x, y) = \exp(-\gamma ||x-y||^2)$
- D) $K(x, y) = (x^T y + c)^d$

Q7. When should you use linear kernel?

- A) Always for all problems
- B) When data is linearly separable or high-dimensional
- C) Never in practice
- D) Only for images

Q8. What is the polynomial kernel?

- A) $K(x, y) = x^T y$
- B) $K(x, y) = (x^T y + c)^d$ where d is degree
- C) $K(x, y) = \exp(-\gamma ||x-y||^2)$
- D) $K(x, y) = \tanh(x^T y)$

Q9. What does the degree parameter control in polynomial kernel?

- A) Margin size
- B) Polynomial degree (2=quadratic, 3=cubic, etc.)

- C) Training speed
- D) Number of features

Q10. What is the RBF kernel formula?

- A) $K(x, y) = x^T y$
- B) $K(x, y) = \exp(-\gamma ||x-y||^2)$
- C) $K(x, y) = (x^T y)^d$
- D) $K(x, y) = x - y$

Q11. What does gamma (γ) control in RBF kernel?

- A) Margin size
- B) Influence of single training sample (inverse of radius)
- C) Number of support vectors
- D) C parameter

Q12. What happens with large gamma?

- A) Smooth decision boundary
- B) Complex boundary, close points have high influence (overfitting risk)
- C) Linear boundary
- D) No effect

Q13. What happens with small gamma?

- A) Very complex boundary
- B) Smooth decision boundary, far-reaching influence
- C) No decision boundary
- D) Perfect separation always

Q14. What is the sigmoid kernel?

- A) $K(x, y) = x^T y$
- B) $K(x, y) = \tanh(\gamma x^T y + c)$
- C) $K(x, y) = \exp(-x)$
- D) $K(x, y) = 1/(1+e^{-x})$

Q15. Which kernel is most popular for SVM?

- A) Linear kernel
- B) RBF (Gaussian) kernel
- C) Polynomial kernel
- D) Sigmoid kernel

Q16. Why is RBF kernel most popular?

- A) Fastest computation
- B) Handles non-linear patterns well, fewer hyperparameters than polynomial
- C) Always most accurate
- D) Simplest to understand

Q17. What is kernel selection based on?

- A) Random choice
- B) Data characteristics and cross-validation performance
- C) Always use RBF
- D) Personal preference only

Q18. Do SVMs require feature scaling?

- A) No, scale-invariant
- B) Yes, very sensitive to feature scales

- C) Only for classification
- D) Only for regression

Q19. What is an advantage of SVM?

- A) Always interpretable
- B) Effective in high dimensions, memory efficient (only stores support vectors)
- C) Always fastest
- D) Requires no tuning

Q20. What is a limitation of SVM?

- A) Too simple
- B) Slow training $O(n^2 \cdot n^3)$, sensitive to scaling, many hyperparameters
- C) Cannot handle numerical data
- D) Only for binary classification

Q21. Does SVM naturally output probabilities?

- A) Yes, always
- B) No, needs Platt scaling for probability calibration
- C) Only for linear kernel
- D) Only for RBF kernel

Q22. What is SVM good for?

- A) Very large datasets (millions of samples)
- B) Medium-sized datasets with clear margin, text/image classification
- C) Only small datasets (<100 samples)
- D) Time series only

Q23. What is SVR (Support Vector Regression)?

- A) Classification only
- B) SVM adapted for regression (predicts continuous values)
- C) Clustering method
- D) Feature selection

Q24. What is ϵ (epsilon) in SVR?

- A) Learning rate
- B) Width of epsilon-insensitive tube (margin for regression)
- C) C parameter
- D) Kernel parameter

Q25. What is one-class SVM used for?

- A) Binary classification
- B) Anomaly/outlier detection
- C) Multiclass classification
- D) Regression

LEVEL 2: HYPERPARAMETER OPTIMIZATION

Q26. What are hyperparameters?

- A) Learned during training
- B) Set before training (control learning process)
- C) Same as model parameters
- D) Not important

Q27. What are model parameters?

- A) Set before training
- B) Learned from data during training (e.g., weights)
- C) Same as hyperparameters
- D) Always fixed

Q28. Examples of hyperparameters include:

- A) Neural network weights
- B) Learning rate, number of trees, max depth, C, gamma
- C) Linear regression coefficients
- D) Feature values

Q29. What is Grid Search?

- A) Random hyperparameter selection
- B) Exhaustive search through predefined hyperparameter grid
- C) Gradient-based optimization
- D) No search needed

Q30. How does Grid Search work?

- A) Tests one combination
- B) Tests all combinations in Cartesian product of parameter values
- C) Random sampling
- D) Uses gradients

Q31. What is an advantage of Grid Search?

- A) Very fast
- B) Guaranteed to find best combination within grid
- C) Requires no computation
- D) Always finds global optimum

Q32. What is a disadvantage of Grid Search?

- A) Too simple
- B) Exponentially expensive with many hyperparameters
- C) Always inaccurate
- D) Cannot be parallelized

Q33. If testing 10 values each for 3 hyperparameters, how many combinations?

- A) 30 combinations
- B) 1000 combinations (10^3)
- C) 13 combinations
- D) 100 combinations

Q34. What is Random Search?

- A) Exhaustive grid search
- B) Randomly samples from hyperparameter distributions

- C) No search
- D) Only tests one value

Q35. How does Random Search compare to Grid Search?

- A) Always worse
- B) Often finds better solutions with same budget, better for many parameters
- C) Exactly the same
- D) Always slower

Q36. Why is Random Search effective?

- A) Tests fewer combinations
- B) Some hyperparameters more important; random samples these effectively
- C) Uses gradients
- D) Always faster

Q37. How many hyperparameters favor Random Search over Grid?

- A) 1-2 parameters
- B) 4+ parameters
- C) Never
- D) Only for neural networks

Q38. What is Bayesian Optimization?

- A) Random sampling
- B) Uses probabilistic model to guide search toward promising regions
- C) Exhaustive search
- D) No search strategy

Q39. How does Bayesian Optimization work?

- A) Randomly tries values
- B) Builds surrogate model, uses acquisition function to select next point
- C) Tests all combinations
- D) Uses grid only

Q40. What is an advantage of Bayesian Optimization?

- A) Simplest method
- B) Most sample-efficient for expensive model training
- C) Always fastest
- D) Requires no computation

Q41. When is Bayesian Optimization preferred?

- A) Very fast models
- B) Expensive models (each training takes hours)
- C) Only for linear models
- D) Never useful

Q42. What is the acquisition function in Bayesian Optimization?

- A) Loss function
- B) Guides selection of next hyperparameters to evaluate
- C) Activation function
- D) Kernel function

Q43. What libraries provide Bayesian Optimization?

- A) Only scikit-learn
- B) Optuna, Hyperopt, scikit-optimize

- C) Only TensorFlow
- D) No libraries available

Q44. Should hyperparameter tuning use cross-validation?

- A) No, use single train/test split
- B) Yes, always use CV within hyperparameter search
- C) Only for small datasets
- D) Never needed

Q45. What is nested cross-validation?

- A) Single CV loop
- B) Outer loop for evaluation, inner loop for hyperparameter tuning
- C) No nesting
- D) Only for neural networks

Q46. Why use nested CV?

- A) Faster training
- B) Provides unbiased performance estimate (prevents overfitting to validation)
- C) Easier implementation
- D) Not actually useful

Q47. What is the search space in hyperparameter tuning?

- A) Physical space
- B) Range and distribution of hyperparameters to explore
- C) Dataset size
- D) Number of features

Q48. How should learning rates be sampled?

- A) Linearly: [0.1, 0.2, 0.3]
- B) Log scale: [0.001, 0.01, 0.1, 1.0]
- C) Randomly without structure
- D) Always the same value

Q49. What is a good strategy for hyperparameter tuning?

- A) Test everything at once
- B) Start broad, then narrow around promising values (coarse-to-fine)
- C) Only test one value per parameter
- D) Use random values only

Q50. Should you tune all hyperparameters simultaneously?

- A) Yes, always all at once
- B) Start with most important, then fine-tune others
- C) Never tune multiple
- D) Only in pairs

LEVEL 3: MODEL INTERPRETATION

Q51. Why is model interpretation important?

- A) Not actually important
- B) Build trust, debug, ensure fairness, regulatory compliance
- C) Only for documentation
- D) Slows down deployment

Q52. What is feature importance?

- A) Feature values
- B) Measure of each feature's contribution to predictions
- C) Number of features
- D) Feature correlations

Q53. How do tree-based models calculate feature importance?

- A) Random assignment
- B) Based on total reduction in impurity/error from splits using that feature
- C) All features equal
- D) Manual calculation

Q54. What is permutation importance?

- A) Fixed importance scores
- B) Measures performance drop when feature values randomly shuffled
- C) Only for trees
- D) Not useful

Q55. What is an advantage of permutation importance?

- A) Fastest method
- B) Model-agnostic (works for any model)
- C) Only for linear models
- D) Always most accurate

Q56. What is SHAP (SHapley Additive exPlanations)?

- A) A new algorithm
- B) Game theory-based method for explaining individual predictions
- C) Only for neural networks
- D) Feature selection method

Q57. What does SHAP provide?

- A) Only global importance
- B) Both local (per prediction) and global feature importance
- C) Only training metrics
- D) Only model accuracy

Q58. What is a SHAP value?

- A) Model accuracy
- B) Feature's contribution to difference from base value for a prediction
- C) Training loss
- D) Hyperparameter value

Q59. What is LIME (Local Interpretable Model-agnostic Explanations)?

- A) Global explanation only
- B) Explains individual predictions by fitting local linear model

- C) Only for linear models
- D) Not actually useful

Q60. What are partial dependence plots (PDP)?

- A) Training curves
- B) Show marginal effect of features on predictions
- C) Loss functions
- D) Architecture diagrams

Q61. What is a benefit of partial dependence plots?

- A) Very complex
- B) Visualize relationship between features and target
- C) Only for classification
- D) Not interpretable

Q62. What is Individual Conditional Expectation (ICE)?

- A) Same as PDP
- B) Shows effect for each instance individually (disaggregated PDP)
- C) Only for regression
- D) Not useful

Q63. What are surrogate models?

- A) Main prediction models
- B) Interpretable models trained to approximate black-box model
- C) Only for deployment
- D) Not accurate

Q64. Why use surrogate models?

- A) Better accuracy
- B) Understand complex model behavior through simpler approximation
- C) Faster training
- D) Required for all models

Q65. What is the interpretation-accuracy tradeoff?

- A) No tradeoff exists
- B) More interpretable models often less accurate, vice versa
- C) Always same accuracy
- D) Interpretation improves accuracy

LEVEL 4: ML PROJECT WORKFLOW

Q66. What is the first step in an ML project?

- A) Collect data
- B) Define problem and success metrics
- C) Train models
- D) Deploy

Q67. What should success metrics align with?

- A) Model complexity
- B) Business objectives and domain requirements
- C) Training speed
- D) Developer preferences

Q68. What is Exploratory Data Analysis (EDA)?

- A) Final testing
- B) Understanding data: distributions, relationships, quality issues
- C) Model deployment
- D) Hyperparameter tuning

Q69. What should EDA include?

- A) Only mean values
- B) Distributions, correlations, missing patterns, outliers, target balance
- C) Only feature count
- D) Model accuracy

Q70. When should you split data into train/validation/test?

- A) After model training
- B) Early in the project, before any preprocessing
- C) Never split
- D) After feature engineering

Q71. What is a data pipeline?

- A) Random processes
- B) Automated sequence of data preprocessing and transformation steps
- C) Manual data entry
- D) Testing procedure

Q72. What should be included in a pipeline?

- A) Only model training
- B) All preprocessing: scaling, encoding, imputation, feature engineering
- C) Only predictions
- D) Only data loading

Q73. Why use `sklearn.pipeline.Pipeline`?

- A) Not necessary
- B) Prevents data leakage, ensures consistent preprocessing
- C) Slows down training
- D) Only for documentation

Q74. What is the benefit of pipelines?

- A) Slower execution
- B) Reproducibility and proper train/test separation

- C) More complex code
- D) Reduced accuracy

Q75. What comes after baseline model?

- A) Deploy immediately
- B) Iteratively try more complex models and features
- C) Stop project
- D) Collect more data only

Q76. Why start with baseline models?

- A) They're always best
- B) Provides comparison point to measure improvements
- C) Required by law
- D) Fastest deployment

Q77. What should you track during experiments?

- A) Nothing needed
- B) Hyperparameters, metrics, data versions, code versions
- C) Only accuracy
- D) Only training time

Q78. What tools help with experiment tracking?

- A) Excel only
- B) MLflow, Weights & Biases, Neptune.ai
- C) No tools available
- D) Only notebooks

Q79. When should you use the test set?

- A) Throughout development
- B) Only once at the very end for final unbiased evaluation
- C) For hyperparameter tuning
- D) Multiple times

Q80. Why is test set sacred?

- A) It's not important
- B) Using it during development causes overfitting and optimistic estimates
- C) Saves time
- D) Improves accuracy

Q81. What is model card documentation?

- A) Physical card
- B) Document describing model details, intended use, limitations, performance
- C) Not needed
- D) Only accuracy number

Q82. What should model documentation include?

- A) Only accuracy
- B) Data, features, algorithm, performance, limitations, ethical considerations
- C) Only code
- D) Nothing needed

Q83. What is technical debt in ML?

- A) Money owed
- B) Hidden costs from shortcuts and quick fixes accumulating over time

- C) Training cost
- D) No such concept

Q84. What causes ML technical debt?

- A) Good practices
- B) Undocumented pipelines, data dependencies, monitoring gaps
- C) Testing models
- D) Using version control

Q85. How to minimize technical debt?

- A) Skip documentation
- B) Good practices: version control, testing, documentation, monitoring
- C) Rush deployment
- D) Avoid pipelines

LEVEL 5: MODEL DEPLOYMENT & MONITORING

Q86. What is model deployment?

- A) Training model
- B) Making model available for production use and predictions
- C) Data collection
- D) Feature engineering

Q87. What are common deployment patterns?

- A) Only batch processing
- B) Batch, real-time API, edge deployment, embedded systems
- C) Only cloud
- D) Only on-premise

Q88. What is batch prediction?

- A) One prediction at a time
- B) Processing multiple predictions periodically (e.g., nightly)
- C) Real-time only
- D) No predictions

Q89. When is batch prediction appropriate?

- A) When millisecond latency needed
- B) When predictions not needed immediately (e.g., monthly reports)
- C) Always inappropriate
- D) Only for testing

Q90. What is real-time prediction?

- A) Batch processing
- B) On-demand prediction via API with low latency
- C) Weekly predictions
- D) No time constraints

Q91. What is a REST API for ML?

- A) Database
- B) Web service exposing model predictions via HTTP endpoints
- C) Training framework
- D) Data storage

Q92. What frameworks are used for ML APIs?

- A) Only scikit-learn
- B) Flask, FastAPI, TensorFlow Serving
- C) Only Excel
- D) No frameworks exist

Q93. What is model monitoring?

- A) One-time check
- B) Continuous tracking of model performance and data in production
- C) Not needed after deployment
- D) Only during training

Q94. What should be monitored in production?

- A) Nothing after deployment
- B) Prediction distribution, input data distribution, latency, errors, accuracy

- C) Only server uptime
- D) Only cost

Q95. What is data drift?

- A) Data storage issues
- B) Input data distribution changes over time
- C) Random fluctuation
- D) Not a real problem

Q96. What is concept drift?

- A) Data storage issues
- B) Relationship between features and target changes over time
- C) Same as data drift
- D) Not a real problem

Q97. How to handle drift?

- A) Ignore it
- B) Retrain model periodically, set up alerts, use adaptive algorithms
- C) Deploy new features only
- D) Stop using model

Q98. What is A/B testing in ML?

- A) Testing two datasets
- B) Comparing new model against baseline in production with live traffic
- C) Training method
- D) Feature selection

Q99. Why use A/B testing?

- A) Not necessary
- B) Validate real-world performance before full rollout
- C) Slows deployment
- D) Only for documentation

Q100. What are key considerations for ML in production?

- A) Only accuracy matters
- B) Latency, scalability, reliability, security, compliance, cost
- C) Only training speed
- D) Only model size