# Answers

June 23, 2025

## 1 Exam Preparation Q & A

---

### 1.1 Multiple Linear Regression

**Q1.** What is the main assumption behind multiple linear regression?

**A.** The expected response is a *linear* combination of the predictors (after any chosen transforms), and the errors are i.i.d., homoscedastic, normal, and uncorrelated, with no perfect multicollinearity.

**Q2.** How is the loss function in multiple linear regression defined?

**A.** Ordinary-Least-Squares minimises the mean-squared error $\mathcal{L}(\beta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{x}_i^\top \beta)^2$.

**Q3.** When should multiple linear regression be avoided?

**A.** When the relationship is highly non-linear, residual variance is non-constant, predictors are strongly collinear, influential outliers dominate, or $p \gg n$.

**Q4.** How do you interpret the coefficients of a linear regression model?

**A.** Each $\beta_j$ is the expected change in $y$ for a one-unit increase in $x_j$ **holding all other predictors constant**.

---

### 1.2 Polynomial & Spline Regression

**Q5.** Why use polynomial regression over linear regression?

**A.** It captures smooth curvature while keeping a model that is linear in the parameters.

**Q6.** What is the danger of using high-degree polynomials?

**A.** High variance: wild oscillations between data points, poor extrapolation, and multicollinearity.

**Q7.** How do splines improve over high-degree polynomials?

**A.** They fit *piece-wise* low-degree polynomials joined smoothly at knots, so flexibility is local and stable.

---

### 1.3 Ridge & Lasso Regression

**Q8.** What is the purpose of regularisation?

**A.** Add a penalty to shrink coefficients, reducing overfitting and multicollinearity.

**Q9.** How does Ridge differ from Lasso regression?

**A.** Ridge ($L_2$) shrinks all coefficients; Lasso ($L_1$) can set some exactly to zero, giving sparsity.

**Q10.** Which regularisation technique can be used for feature selection?

**A.** Lasso (or Elastic Net).

**Q11.** What does the **alpha** hyper-parameter control in Ridge and Lasso?

**A.** Penalty strength: larger $\alpha$ $\Rightarrow$ stronger shrinkage / more zeros; $\alpha = 0$ recovers OLS.

---

### 1.4 Logistic Regression

**Q12.** What type of problem does logistic regression solve?

**A.** Classification (binary, with soft-max extension to multi-class).

**Q13.** What is the output of a logistic regression model?

**A.** The estimated probability $\hat{p} = \sigma(\mathbf{x}^\top \beta) = \frac{1}{1+e^{-\mathbf{x}^\top \beta}}$.

**Q14.** How does F1-score help with imbalanced datasets?

**A.** $F_1 = 2\,\frac{PR}{P+R}$ balances precision $P$ and recall $R$, so majority-class accuracy doesn't dominate.

**Q15.** Scenario where recall > precision is more important and why?

**A.** Medical screening: missing a sick patient (low recall) is costlier than extra false positives (lower precision).

---

### 1.5 Convolutional Neural Networks (CNNs)

**Q17.** What kind of data are CNNs best suited for?

**A.** Grid-like data with local spatial correlations (images, video frames, spectrograms, 1-D/3-D signals).

**Q18.** Name two common layers used in CNNs.

**A.** Convolutional layers and pooling (max or average) layers.

**Q19.** Name parameters used for defining a CNN.

**A.** Number & size of filters, kernel size, stride, padding, number of conv–pool blocks, learning rate, weight decay, batch size.

**Q20.** What is the role of filters in CNNs?

**A.** Learn local patterns (edges $\rightarrow$ textures $\rightarrow$ objects); stacking filters builds hierarchical features.

---

### 1.6 Recurrent Neural Networks (RNNs)

**Q21.** What makes RNNs different from feed-forward networks?

**A.** They pass a hidden state from one time-step to the next, giving memory of prior inputs.

**Q22.** What is a limitation of typical RNNs?

**A.** Vanishing/exploding gradients, so they struggle with long-range dependencies (mitigated by LSTM/GRU).

**Q23.** What data types are RNNs commonly used for?

**A.** Sequential data: text, speech, sensor or financial time-series, DNA.

**Q24.** Give three examples of sequential problems.

**A.** Language modelling, machine translation, stock-price prediction.

---

### 1.7 Ensemble Methods

**Q25.** What is the core idea behind ensemble methods?

**A.** Combine multiple diverse learners so their uncorrelated errors cancel out.

**Q26.** What are the main types of ensemble methods?

**A.** Bagging (e.g. Random Forest), Boosting (Ada/Gradient/XGBoost), Stacking/Voting.

**Q27.** What is stacking in ensemble learning?

**A.** Base models are trained; their out-of-fold predictions feed a meta-learner that blends them.

**Q28.** What are the advantages of using ensemble models?

**A.** Higher accuracy, lower variance, greater robustness, better feature-importance stability.

---

### 1.8 Random Forest

**Q29.** How does a Random Forest differ from a decision tree?

**A.** It builds many trees on bootstrap samples with random feature splits and averages/votes the results.

**Q30.** What is the role of **max_features** in Random Forest?

**A.** Sets how many predictors each split may test; smaller values decorrelate trees (reduce variance).

**Q31.** What is an out-of-bag (OOB) sample?

**A.** Data not included in a tree's bootstrap draw (~ ); used for internal error and feature-importance estimates.

**Q32.** What does increasing **n_estimators** typically do?

**A.** Reduces variance and OOB/test error until a plateau; after that it mostly increases compute cost.

---

### 1.9 Boosting

**Q33.** How does boosting work conceptually?

**A.** Learners are added sequentially, each focusing more on the mistakes of its predecessors.

**Q34.** Impact of learners on the final prediction in AdaBoost?

**A.** Each weak learner gets weight $\alpha_m = \ln(\frac{1-\text{err}_m}{\text{err}_m})$; better learners thus influence more.

**Q35.** What is the role of **learning_rate** in boosting?

**A.** Scales each learner's contribution; small values slow learning and act as regularisation.

**Q36.** Why is boosting more prone to overfitting than bagging?

**A.** It keeps fitting hard/noisy points, especially if base learners are too complex.

---

### 1.10 DBSCAN

**Q37.** What kind of clustering does DBSCAN perform?

**A.** Density-based clustering.

**Q38.** What does the **eps** parameter control in DBSCAN?

**A.** Neighbourhood radius within which points are considered neighbours of a *core* point.

**Q39.** How does DBSCAN handle outliers?

**A.** Points not density-reachable from any core point are labelled *noise*.

**Q40.** What type of datasets is DBSCAN ideal for?

**A.** Arbitrary-shaped clusters, noise, unknown $k$, low/medium-dimensional numeric data.

---

### 1.11 UMAP

**Q41.** What is the primary purpose of UMAP?

**A.** Non-linear dimensionality reduction / visualisation that preserves local and some global structure.

**Q42.** What does **n_neighbors** control in UMAP

**A.** Trade-off between local detail (small values) and global structure (large values).

**Q43.** Common application of UMAP?

**A.** Visualising single-cell RNA-seq, image or word embeddings; producing embeddings for clustering.

**Q44.** Is UMAP deterministic?

**A.** Not strictly; fixing the random seed gives repeatable results.

---

### 1.12 Multidimensional Scaling (MDS)

**Q45.** What is the main idea of MDS?

**A.** Place points in low-D so that pairwise Euclidean distances approximate a given dissimilarity matrix.

**Q46.** Difference between metric and non-metric MDS?

**A.** Metric preserves exact distances; non-metric preserves only their rank order via a monotone transform.

**Q47.** Example when metric MDS is recommended.

**A.** When dissimilarities are true Euclidean distances, e.g. great-circle distances between cities.

**Q48.** Computational speed of MDS vs. UMAP/t-SNE?

**A.** Classical MDS needs an $O(n^3)$ eigendecomposition, so it is much slower on large $n$.

---

### 1.13 Support-Vector Machines (SVM)

**Q49.** What does the SVM algorithm aim to maximise?

**A.** The margin (minimum distance) between classes.

**Q50.** What is a support vector?

**A.** A training point lying on or inside the margin that defines the decision boundary.

**Q51.** What does the kernel trick allow in SVMs?

**A.** Computes inner products in high/infinite-D feature space without explicit mapping, enabling non-linear separation.

**Q52.** What does the hyper-parameter **C** control?

**A.** Soft-margin penalty: small $C$ $\Rightarrow$ wider margin, higher bias; large $C$ $\Rightarrow$ narrow margin, lower bias but risk of overfitting.