

AI MASTERY RESOURCES

Data Science & Machine Learning Interview Question Bank

Basic · Intermediate · Advanced

85 original practice questions with preparation cues,
plus portfolio projects for freshers and career transitioners.

2026 Edition

Prepared for professionals building evidence-led careers in data science and ML.

Dr. Nabanita Sinha

How to use this question bank

Use the bank in passes. First answer aloud without notes. Then write a stronger version with assumptions, examples, and one limitation. Finally, practise applying the idea to a business decision or system you have built.

1 • Diagnose the role

Analytics and product data science interviews usually weight SQL, metrics, experimentation, and communication. ML-oriented roles add modeling, evaluation, and production design.

2 • Answer with a frame

For open-ended questions: clarify the goal, define the data and unit of analysis, state assumptions, choose a method, validate it, and explain the decision.

3 • Make trade-offs visible

Say what you would optimise, what could go wrong, what you would do next, and when a simpler solution is preferable.

4 • Practise evidence

Prepare two project stories with a baseline, a measurable result, a failure or limitation, and a clear explanation of what you personally did.

Preparation map

Basic	Definitions, data quality, core metrics, simple SQL, and clear explanations.	Target: accurate and concise.
Intermediate	Inference, experiment design, feature engineering, validation, and production thinking.	Target: structured trade-offs.
Advanced	Causal inference, ML system design, monitoring, fairness, and leadership judgment.	Target: decision-quality depth.

Build clear foundations and answer with confidence.

Aim for clear, technically correct explanations. When you do not know an answer, state your assumptions and reason from first principles.

Statistics & probability

1. What is the difference between a population and a sample?

Prepare: Define the estimand, identify the sampling frame, and explain why a sample should represent the population you want to generalise to.

2. What is mean, median, and mode? When can the median be more useful than the mean?

Prepare: Compare sensitivity to outliers, skewed distributions, and the business meaning of the typical observation.

3. What is variance and standard deviation?

Prepare: Explain spread around the mean, units, and why standard deviation is not a complete description of a distribution.

4. What is conditional probability? Explain Bayes' theorem in plain language.

Prepare: Use $P(A|B) = P(B|A)P(A)/P(B)$ and give a concrete example such as fraud alerts or medical testing.

5. What is the difference between correlation and causation?

Prepare: Mention confounding, selection bias, reverse causality, and why a randomised experiment is stronger evidence for causality.

6. What is a confidence interval?

Prepare: Explain repeated-sampling interpretation, uncertainty around an estimate, and why it is not the probability that a fixed parameter lies in this one interval.

Data, SQL & Python

7. How would you inspect a new dataset before analysing it?

Prepare: Check grain, row counts, types, missingness, duplicates, ranges, distributions, keys, timestamps, and data provenance before making charts.

8. How do you handle missing values?

Prepare: Start by understanding why values are missing; compare deletion, simple imputation, model-based imputation, and a missingness indicator without leaking information.

9. What is the difference between an inner join and a left join?

Prepare: Explain row retention, key cardinality, unmatched records, and how duplicate join keys can silently multiply rows.

10. What is a primary key and why does it matter in analysis?

Prepare: It should identify one entity or event at the intended grain; validate uniqueness before joining or aggregating.

11. What is the difference between a list, tuple, set, and dictionary in Python?

Prepare: Compare mutability, ordering, uniqueness, lookup behavior, and when each structure makes code clearer.

12. How would you make a Python data-cleaning workflow reproducible?

Prepare: Use explicit inputs, deterministic transformations, functions, environment/version notes, tests or assertions, and a saved output or notebook narrative.

Machine learning fundamentals

13. What is supervised learning?

Prepare: The model learns a mapping from labelled examples to a target; distinguish classification and regression and give one use case for each.

14. What is unsupervised learning?

Prepare: There is no labelled target; discuss clustering, dimensionality reduction, anomaly detection, and the need for validation beyond a pleasing visual.

15. What is the difference between a feature and a target?

Prepare: Features are inputs available at prediction time; the target is the outcome to estimate. Check timing and availability to avoid leakage.

16. What is overfitting and underfitting?

Prepare: Overfitting memorises noise and performs poorly on new data; underfitting is too simple. Use learning curves, regularisation, and better features appropriately.

17. What is the bias-variance trade-off?

Prepare: Bias is systematic error from restrictive assumptions; variance is sensitivity to the sample. Model complexity and regularisation balance the two.

18. Why do we split data into train, validation, and test sets?

Prepare: Training fits parameters, validation supports choices, and the untouched test set gives a final estimate. Never tune repeatedly on the test set.

19. What is the difference between linear regression and logistic regression?

Prepare: Linear regression predicts a continuous value; logistic regression models log-odds for a categorical outcome and produces probabilities after a link function.

20. What is feature scaling, and when is it important?

Prepare: Scaling puts numeric features on comparable ranges and matters for distance- or gradient-based methods; tree splits usually do not require it.

Evaluation & communication

21. What is accuracy, and when can it be misleading?

Prepare: On imbalanced data, predicting the majority class can look accurate. Compare precision, recall, specificity, F1, PR-AUC, and business costs.

22. What is the difference between precision and recall?

Prepare: Precision asks how many predicted positives are correct; recall asks how many actual positives were found. Choose based on the cost of false positives and negatives.

23. What is a confusion matrix?

Prepare: It counts true positives, false positives, true negatives, and false negatives; use it to connect model errors to operational consequences.

24. What is cross-validation?

Prepare: Repeatedly train on folds and validate on the held-out fold to estimate generalisation and compare models, while keeping preprocessing inside each fold.

25. How would you explain a model result to a non-technical stakeholder?

Prepare: Lead with the decision, impact, uncertainty, and limitations; use plain language, one visual, and a recommended next action rather than model jargon.

26. What makes a good data science project?

Prepare: A clear problem, trustworthy data, a baseline, defensible evaluation, reproducible code, useful communication, and honest limitations matter more than a flashy algorithm.

Show structured reasoning, good experimental habits, and practical model judgment.

Answer in a structured way: clarify the problem, state assumptions, choose a method, validate it, and connect the result to a decision.

Statistical inference & experimentation

1. What is a null hypothesis and how do you use a p-value?

Prepare: State the null before analysis, define the test statistic, and explain that a p-value measures compatibility with the null—not the probability the null is true.

2. What is statistical power? How can you increase it?

Prepare: Power is the chance of detecting a specified effect when it exists. Increase sample size or effect size, reduce noise, or use a more efficient design.

3. How would you determine the sample size for an A/B test?

Prepare: Specify baseline rate, minimum detectable effect, alpha, desired power, allocation, and practical constraints; account for repeated looks and multiple variants.

4. Why is peeking at an experiment dangerous?

Prepare: Repeated unplanned checks inflate false positives. Use a sequential-testing method or pre-registered stopping rule instead of stopping when a result looks significant.

5. What is Simpson's paradox?

Prepare: An aggregate relationship can reverse within meaningful subgroups because of confounding or different group weights. Always investigate stratification and causal structure.

6. How do you distinguish practical significance from statistical significance?

Prepare: A small effect can be statistically reliable at scale but not worth shipping; define the business threshold and uncertainty around the effect.

SQL, analytics & data quality

7. Write a query to find the second-highest value without relying on a fixed offset.

Prepare: Discuss `DENSE_RANK` or a distinct ordered subquery, ties, null handling, and the intended definition of second-highest.

8. When would you use a window function instead of `GROUP BY`?

Prepare: `GROUP BY` reduces rows; a window function retains row-level detail while calculating ranks, rolling values, or within-group comparisons.

9. How would you calculate a seven-day rolling average?

Prepare: Create a correct date grain, define the window frame, handle missing dates, and make the denominator explicit rather than averaging only observed rows accidentally.

10. How do you investigate a sudden drop in a dashboard metric?

Prepare: Check the metric definition, pipeline freshness, schema changes, filters, denominator, segment mix, instrumentation, and independent source data before proposing a model or business explanation.

11. What is data leakage in an analytics or ML pipeline?

Prepare: Information unavailable at decision time enters features or evaluation. Separate time, fit transformations only on training data, and document feature availability.

Modeling & feature engineering

12. How do L1 and L2 regularisation differ?

Prepare: L1 encourages sparse coefficients and can select features; L2 shrinks correlated coefficients smoothly. Both control complexity but interact differently with scaling.

13. How do you choose a baseline model?

Prepare: Choose a simple, interpretable benchmark aligned to the target and decision, such as a majority class, mean, seasonal naive, or regularised linear model.

14. When would you choose a tree-based model over a linear model?

Prepare: Use trees for nonlinear interactions and mixed scales with less manual feature transformation; use linear models when extrapolation, stability, and interpretability are priorities.

15. How do you tune hyperparameters without overfitting the validation process?

Prepare: Use nested cross-validation or a final untouched test set, keep preprocessing in a pipeline, limit search freedom, and report the search protocol.

16. How would you handle a highly imbalanced classification problem?

Prepare: Start with the base rate and error costs; consider class weights, threshold tuning, stratified splits, resampling inside folds, and PR-AUC or cost-weighted metrics.

17. What is calibration and why does it matter?

Prepare: A calibrated probability means events predicted at 0.7 occur about 70% of the time. It matters for risk ranking, resource allocation, and threshold decisions.

18. How do you select features while avoiding leakage?

Prepare: Use domain timing, training-only statistics, cross-validated selection, regularisation, and a pipeline. Do not select features using the full dataset before splitting.

Time series, unsupervised learning & ML design

19. Why is random train-test splitting often wrong for time series?

Prepare: It allows future information into training and breaks the deployment sequence. Use chronological or rolling-origin validation and explicit forecast horizons.

20. How would you evaluate a forecast?

Prepare: Compare against a seasonal or naive baseline using horizon-appropriate metrics such as MAE, RMSE, or weighted errors, and inspect residuals over time.

21. How do you decide the number of clusters?

Prepare: Use domain usefulness, stability, silhouette or related diagnostics, sensitivity analysis, and whether the segments lead to different actions.

22. What is dimensionality reduction useful for?

Prepare: Compression, visualisation, denoising, or downstream modeling; explain that preserving variance does not necessarily preserve the business signal.

23. Design a churn prediction system. What happens beyond model training?

Prepare: Define the intervention and label window, prevent leakage, choose a useful threshold, connect scores to action, measure uplift or retention, and monitor drift and fairness.

Product, deployment & collaboration

24. How would you define a north-star metric for a product?

Prepare: Explain the concept, state an assumption, and connect it to a practical decision.

25. How do you choose between a model and a rule-based system?

Prepare: Explain the concept, state an assumption, and connect it to a practical decision.

26. What should be logged when a model makes a production prediction?

Prepare: Explain the concept, state an assumption, and connect it to a practical decision.

27. What is the difference between data drift and concept drift?

Prepare: Explain the concept, state an assumption, and connect it to a practical decision.

28. How do you communicate uncertainty when a decision is time-sensitive?

Prepare: Explain the concept, state an assumption, and connect it to a practical decision.

Demonstrate depth in causal reasoning, system design, reliability, and leadership judgment.

Senior answers connect technical choices to risk, reliability, cost, users, and organisational outcomes. Explain trade-offs, not just definitions.

Causal inference & advanced experimentation

1. Design an experiment when users can be exposed to both treatment and control.

Prepare: Define the unit of randomisation, interference risk, contamination, exposure logging, primary metric, guardrails, and an estimand that matches the decision.

2. How would you estimate the effect of a treatment when randomisation is impossible?

Prepare: State assumptions for a quasi-experimental method such as difference-in-differences, matching, regression discontinuity, or instrumental variables, then test sensitivity.

3. What is the difference between intent-to-treat and treatment-on-the-treated?

Prepare: Intent-to-treat preserves randomisation and measures assignment effect; treatment-on-the-treated estimates effect among compliers and needs stronger assumptions.

4. How would you handle multiple metrics and many experiment variants?

Prepare: Pre-specify primary and guardrail metrics, control family-wise or false-discovery error where appropriate, and separate exploratory findings from confirmatory claims.

5. How do heterogeneous treatment effects change an experiment decision?

Prepare: Check pre-specified segments, power and multiplicity, treatment effect uncertainty, fairness, and whether the operational policy can target a segment reliably.

6. What makes a causal graph useful in a data science case?

Prepare: It exposes confounders, mediators, colliders, and required adjustment sets; use it to explain which data can identify the effect and which variables should not be controlled.

ML system design & production

7. Design an end-to-end real-time fraud detection system.

Prepare: Clarify latency, decision point, labels and delay, feature freshness, false-positive cost, human review, fallbacks, monitoring, retraining, and auditability.

8. How would you design a recommendation system with a cold-start problem?

Prepare: Separate candidate generation and ranking, use content or popularity priors, explore safely, collect feedback, and evaluate both offline ranking and online long-term outcomes.

9. How do you prevent training-serving skew?

Prepare: Use shared feature definitions or a feature store, validate schemas and distributions, replay production features in tests, and monitor parity between offline and online transformations.

10. What is the difference between batch and online inference?

Prepare: Batch suits stable decisions and cost efficiency; online inference suits fresh, low-latency decisions. Compare latency, freshness, reliability, cost, and operational complexity.

11. How would you design a model rollback strategy?

Prepare: Version code, data, features, model, and configuration; keep a known-good artifact, define health thresholds, support shadow or canary release, and make rollback observable.

12. How do you build an evaluation set for a high-stakes classifier?

Prepare: Stratify by risk and affected groups, define labels and adjudication, measure label quality, hold out time, include hard negatives, and document representativeness.

13. What is a data contract and how does it protect a model?

Prepare: It specifies schema, semantics, freshness, allowed ranges, and ownership. Enforce it at boundaries so upstream changes fail clearly instead of silently degrading predictions.

14. How would you reduce model cost without harming decision quality?

Prepare: Profile the cost drivers, establish a quality floor, consider distillation, batching, caching, feature reduction, thresholding, or a tiered model path, and validate on business outcomes.

Robustness, fairness & monitoring

15. How would you investigate a model that performs well overall but poorly for one subgroup?

Prepare: Check sample size, label quality, feature coverage, calibration, threshold effects, measurement bias, and intersectional slices before choosing data, model, or policy changes.

16. What is fairness in ML, and why can fairness metrics conflict?

Prepare: Fairness is context-dependent. Base rates, error costs, and definitions such as equal opportunity or predictive parity can make all criteria impossible simultaneously.

17. How do you monitor a model when ground-truth labels arrive months later?

Prepare: Use leading indicators such as input drift, prediction distribution, data quality, business proxies, delayed outcome backfills, and review samples; define escalation thresholds.

18. How would you detect silent failure in a feature pipeline?

Prepare: Monitor freshness, volume, nulls, ranges, cardinality, distribution, join coverage, and feature-to-prediction relationships, with tests against known fixtures.

19. How do adversarial examples or data poisoning affect an ML system?

Prepare: Adversarial inputs manipulate inference; poisoning corrupts training data. Use threat modeling, provenance, robust validation, access controls, monitoring, and incident response.

20. When should a model refuse to make a prediction?

Prepare: Define an abstention or reject option for out-of-distribution, low-confidence, high-impact, or unsupported cases, with a safe fallback and an owner for review.

21. Explain gradient descent and the role of the learning rate.

Prepare: The optimizer uses the gradient to update parameters; too high diverges, too low is slow. Discuss schedules, normalisation, batch size, and validation.

22. What is attention in a transformer?

Prepare: Queries, keys, and values create weighted context across tokens; discuss quadratic sequence cost, masking, positional information, and why attention is useful.

23. How would you evaluate an NLP model when exact-match accuracy is insufficient?

Prepare: Use task metrics plus human or rubric-based quality, calibration, safety, robustness, slices, and a baseline. Tie evaluation to the user decision.

24. What is the difference between offline and online evaluation for ranking systems?

Prepare: Offline replay is fast but limited by logged exposure and counterfactual bias; online tests measure real behavior but need guardrails and sufficient traffic.

25. How do you choose between a foundation model, fine-tuning, retrieval, and a smaller task model?

Prepare: Compare data availability, latency, privacy, change rate, controllability, cost, failure modes, and evaluation evidence—not just benchmark scores.

Leadership, cases & decision-making

26. A stakeholder asks for a model before the problem is defined. What do you do?

Prepare: Clarify the decision, user, intervention, success metric, constraints, and baseline. Explain when analysis, instrumentation, or a rule is the better first step.

27. How would you review a junior data scientist's model before it ships?

Prepare: Check target and leakage, data split, baseline, metrics, slices, reproducibility, error analysis, operational path, documentation, and monitoring plan.

28. How do you prioritise a data science roadmap with limited capacity?

Prepare: Rank opportunities by expected value, evidence, feasibility, risk, reversibility, and learning; make dependencies and stopping criteria explicit.

29. Tell me about a time your analysis contradicted a senior stakeholder's belief.

Prepare: Use a concise situation-action-result story: validate the data, communicate respectfully, show uncertainty, propose a test, and describe the business outcome.

30. When is a model not worth deploying?

Prepare: If the intervention is unclear, value is below operating cost, evidence is weak, risks are unacceptable, or a simpler reliable process performs nearly as well.

31. How would you explain that a model's AUC improved but the product did not?

Prepare: AUC may not reflect threshold economics, calibration, coverage, drift, intervention quality, or the chosen KPI. Trace the full decision funnel.

Projects for freshers and career transition

You do not need previous data science job experience to demonstrate readiness. A small set of finished, well-explained projects can show how you think, work with data, evaluate evidence, and communicate decisions.

What every project should include

Define the business question and decision first. Include the data source and grain, a baseline, reproducible steps, evaluation aligned to the decision, an error or limitation section, and a short README written for a non-technical stakeholder. A finished, modest project is stronger than an unfinished “advanced” notebook.

1. SQL-to-Story Business Dashboard

FRESHERS · FIRST PORTFOLIO PROJECT

Use a public retail, mobility, or subscription dataset to answer five business questions with SQL and a concise dashboard.

Show: Data model and grain notes, joins and window functions, KPI definitions, one anomaly investigation, dashboard screenshots, and a one-page executive summary.

2. Customer Segmentation With an Action Plan

FRESHERS · CAREER TRANSITION

Create behavioural features from transactions or product events, compare a few clustering approaches, and translate segments into different actions.

Show: Feature logic, scaling choices, cluster stability, segment profiles, limitations, and a recommendation for how marketing or product teams would use the segments.

3. Churn Prediction With Leakage Controls

CAREER TRANSITION · APPLIED ML

Predict churn at a clearly defined decision point, then show how the score would support a retention intervention.

Show: Time-aware label design, naive and ML baselines, PR-AUC or cost-based evaluation, threshold analysis, subgroup checks, and a monitoring plan.

4. Demand Forecasting for Operations

FRESHERS · ANALYTICS / DS

Forecast daily or weekly demand for a public time series and recommend staffing, inventory, or capacity actions.

Show: Seasonal-naive baseline, rolling validation, forecast intervals, error by horizon, residual review, and a short business decision memo.

5. Experiment Readout With Synthetic or Public Data

CAREER TRANSITION · PRODUCT DS

Design and analyse an A/B test for a product change, including power, primary and guardrail metrics, and a decision under uncertainty.

Show: Pre-analysis plan, sample-size assumptions, treatment effect with interval, practical significance, segment cautions, and a clear ship/iterate/stop recommendation.

6. End-to-End Model With a Small Production Surface

CAREER TRANSITION · ML-ORIENTED ROLES

Package one useful model behind a simple API or app using a public dataset; prioritise reliability and explanation over complexity.

Show: Repository structure, reproducible environment, input validation, baseline comparison, model card, tests, latency or cost note, and a deployed demo or recorded walkthrough.

A practical 8-week transition plan

Use the plan as a sequence, not a checklist to rush through. Keep one GitHub repository per finished project or a clean portfolio index that makes the evidence easy to review.

Weeks 1–2

Refresh SQL, Python, descriptive statistics, probability, and visualisation. Recreate five queries and explain each result aloud.

Weeks 3–4

Complete the SQL-to-story dashboard. Add a clear metric dictionary, data-quality checks, and an executive summary.

Weeks 5–6

Complete one modeling project. Compare a baseline with one or two stronger models, use leakage-safe validation, and analyse errors by segment.

Week 7

Add an experiment or forecasting project to show a second kind of reasoning beyond classification.

Week 8

Polish two projects: README, screenshots, assumptions, limitations, reproducibility, and a two-minute walkthrough for each.

Project story template

Practise each project in this order. Treat every step as evidence you can explain in an interview:

- 1. Problem and user**
- 2. Data and risks**
- 3. Baseline**
- 4. Method and trade-offs**
- 5. Result and uncertainty**
- 6. What failed**
- 7. Recommendation**
- 8. Next experiment**

This turns portfolio work into strong behavioural and technical interview evidence.

Research basis & source notes

This question bank is an original synthesis created for interview preparation. The sources below informed the topic coverage and preparation guidance; the questions and cues are not copied from any source.

Google Machine Learning Crash Course

<https://developers.google.com/machine-learning/crash-course>

Core concepts, data preparation, classification, numerical data, embeddings, and practical exercises.

Google Rules of Machine Learning

<https://developers.google.com/machine-learning/guides/rules-of-ml>

Production-oriented guidance on metrics, training/serving skew, monitoring, and iteration.

scikit-learn User Guide: Model Selection & Evaluation

https://scikit-learn.org/stable/model_selection.html

Cross-validation, scoring, validation curves, and evaluation design.

scikit-learn: Common Pitfalls & Recommended Practices

https://scikit-learn.org/stable/common_pitfalls.html

Data leakage, preprocessing, pipelines, and reproducible modeling.

pandas User Guide

https://pandas.pydata.org/docs/user_guide/index.html

Data cleaning, indexing, joins, reshaping, groupby, time series, and data handling.

PostgreSQL Documentation: Window Functions

<https://www.postgresql.org/docs/current/tutorial-window.html>

Row-preserving analytics, ranking, running calculations, and window framing.

Kaggle: Data Science Portfolio Tips

<https://www.kaggle.com/code/rtatman/data-science-portfolio-tips>

Portfolio principles: finished work, clear value, communication, reproducibility, and project variety.

Coursera: Machine Learning Interview Prep Guide

<https://www.coursera.org/resources/machine-learning-interview-prep-guide>

A secondary synthesis of common ML interview themes and preparation approaches.

Final reminder

Interviewers are evaluating how you reason, not whether you can recite every definition. Clarify the goal, protect the data, choose an appropriate baseline, quantify uncertainty, and make the next decision obvious.