DATA SCIENCE ROADMAP

120 concepts to become a data scientist

8 areas, 15 concepts each. Every concept explained in one line, the order to study each area, and two practice questions to test yourself.

How to use it: one area at a time, tick each concept only when you can explain it out loud in under a minute, then answer the two practice questions without notes.

@chalkesahil · chalkesahil.ai

AREA 01 / 08

Python and pandas

Study order: Most of the job is cleaning and shaping data in pandas.

01	**Python basics**	Lists, dicts and functions.
02	**NumPy arrays**	Fast maths on numbers.
03	**Vectorisation**	Avoid slow loops.
04	**pandas DataFrames**	Tables in Python.
05	**Reading data**	CSV, Excel and SQL.
06	**Selecting rows and columns**	loc and iloc.
07	**Filtering**	Boolean masks.
08	**groupby**	Summaries per group.
09	**merge and join**	Combine tables.
10	**Missing values**	Fill or drop.
11	**Dates**	Parse and resample.
12	**apply**	Custom transforms.
13	**pivot_table**	Wide summaries.
14	**Jupyter notebooks**	Code and notes together.
15	**Clean notebook habits**	Restart and run all.

Practice

1. Find average order value per city from an orders table.
2. Fill missing ages in a dataset and justify your choice.

AREA 02 / 08

Statistics

Study order: Interviews test whether you understand the numbers, not just the code.

01	**Mean, median, mode**	Which average to use.
02	**Variance and standard deviation**	Spread of data.
03	**Distributions**	Normal, skewed and long tails.
04	**Percentiles**	What the 90th percentile means.
05	**Probability basics**	Events and chances.
06	**Conditional probability**	Given that something happened.
07	**Bayes' theorem**	Update beliefs with evidence.
08	**Sampling**	Why samples can mislead.
09	**Central limit theorem**	Why averages look normal.
10	**Hypothesis testing**	Is it real or noise?
11	**p-values**	What they mean and don't.
12	**Confidence intervals**	A range, not a guess.
13	**A/B testing**	Fair comparisons.
14	**Correlation vs causation**	The classic trap.
15	**Outliers**	Find and decide what to do.

Practice

1. An A/B test shows a 2% lift. Is it real?
2. Explain p-value to a manager.

AREA 03 / 08

SQL

Study order: Data scientists pull their own data; SQL is non-negotiable.

01	**SELECT and WHERE**	Get the rows you need.
02	**Joins**	Combine tables.
03	**GROUP BY and HAVING**	Aggregates per group.
04	**CASE WHEN**	Labels inside queries.
05	**Subqueries**	Queries inside queries.
06	**CTEs**	Readable steps.
07	**Window functions**	Running totals and ranks.
08	**LAG and LEAD**	Compare with previous rows.
09	**Date functions**	Week, month and cohorts.
10	**NULL handling**	COALESCE.
11	**DISTINCT**	Count unique users right.
12	**Cohort queries**	Retention by signup month.
13	**Funnel queries**	Drop-off at each step.
14	**Query performance**	Filter early.
15	**SQL from Python**	Load results into pandas.

Practice

1. Find 7-day retention for each signup cohort.
2. Rank products by revenue within each category.

AREA 04 / 08

Data visualisation

Study order: A chart that's understood in 5 seconds beats a fancy one.

01	**Exploratory analysis**	Look before you model.
02	**Matplotlib**	The base plotting library.
03	**Seaborn**	Statistical charts quickly.
04	**Histograms**	Shape of a distribution.
05	**Box plots**	Spread and outliers.
06	**Scatter plots**	Relationship between two variables.
07	**Line charts**	Trends over time.
08	**Bar charts**	Compare categories.
09	**Heatmaps**	Correlations at a glance.
10	**Choosing the right chart**	Match chart to question.
11	**Labels and titles**	Charts that explain themselves.
12	**Plotly**	Interactive charts.
13	**Dashboards**	Streamlit or Power BI.
14	**Storytelling**	Lead with the answer.
15	**Presenting to non-technical people**	Simple words.

Practice

1. Show whether delivery time affects ratings.
2. Present one insight to a manager in one minute.

AREA 05 / 08

Machine learning

Study order: Learn the main models well before chasing new ones.

01	**What ML is**	Learning patterns from data.
02	**Supervised vs unsupervised**	Labels vs no labels.
03	**Train/test split**	Check on unseen data.
04	**Linear regression**	Predict a number.
05	**Logistic regression**	Predict yes or no.
06	**Decision trees**	If-else rules learned from data.
07	**Random forests**	Many trees voting.
08	**Gradient boosting**	XGBoost and LightGBM.
09	**KNN**	Nearest neighbours.
10	**K-means**	Group similar points.
11	**Feature engineering**	Better inputs, better models.
12	**Encoding categories**	One-hot and target encoding.
13	**Scaling**	Standardise features.
14	**scikit-learn**	The main ML library.
15	**Pipelines**	Clean, repeatable training.

Practice

1. Predict house prices and explain your features.
2. Group customers into segments without labels.

AREA 06 / 08

Model evaluation

Study order: A model is only as good as how honestly you test it.

01	**Overfitting**	Memorising instead of learning.
02	**Underfitting**	Too simple to learn.
03	**Bias-variance trade-off**	The balance to find.
04	**Cross-validation**	Test on several splits.
05	**Accuracy**	And why it can lie.
06	**Precision and recall**	Which mistakes matter.
07	**F1 score**	Balance of both.
08	**Confusion matrix**	Every type of error.
09	**ROC and AUC**	Ranking quality.
10	**RMSE and MAE**	Error for numbers.
11	**Imbalanced data**	When one class is rare.
12	**Data leakage**	Future info sneaking in.
13	**Hyperparameter tuning**	Grid and random search.
14	**Feature importance**	What drives predictions.
15	**Baseline models**	Always beat a simple guess.

Practice

1. Your fraud model has 99% accuracy. Why might it be useless?
2. How do you detect data leakage?

AREA 07 / 08

Deep learning basics

Study order: Know the basics; most jobs still run on classic ML.

01	**Neural networks**	Layers of simple units.
02	**Neurons and weights**	What gets learned.
03	**Activation functions**	ReLU and sigmoid.
04	**Loss functions**	What the model minimises.
05	**Gradient descent**	How weights update.
06	**Backpropagation**	How errors flow back.
07	**Epochs and batches**	How training runs.
08	**Overfitting in deep learning**	Dropout and early stopping.
09	**PyTorch basics**	The common framework.
10	**CNNs**	Models for images.
11	**RNNs and sequences**	Models for order.
12	**Transformers**	The idea behind modern AI.
13	**Embeddings**	Words and items as numbers.
14	**Pretrained models**	Start from someone else's work.
15	**LLMs for data work**	Use them, check them.

Practice

1. Explain gradient descent with a simple analogy.
2. When would you not use deep learning?

AREA 08 / 08

Projects and interviews

Study order: Projects with a real question beat Kaggle copies.

01	**Project 1: prediction**	A model with a clear business question.
02	**Project 2: analysis**	Insights from a public dataset.
03	**Project 3: deployed model**	Streamlit or FastAPI app.
04	**GitHub**	Clean notebooks with READMEs.
05	**Kaggle**	Practise on real datasets.
06	**Case studies**	Business problems end to end.
07	**Product sense**	Which metric would you track?
08	**ML interview questions**	Bias, variance, metrics.
09	**Stats interview questions**	Tests and probability.
10	**SQL round**	Timed query practice.
11	**Python round**	pandas tasks.
12	**Resume**	Results with numbers.
13	**Explaining a model**	To a non-technical person.
14	**Mock interviews**	Practise out loud.
15	**AI tools**	Speed up code, check every result.

Practice

1. Walk through one project from question to result.
2. How would you measure if a new feature worked?