AI Research Labs

Deep Learning Foundations · Practice test 8 · 25 questions

# Generalisation and Data

**How to use this test.** Answer all 25 questions without notes, then mark yourself with the answer key at the back. Each question has exactly one best answer. Score 75% or more and you've mastered the material; below that, reread the lessons for the questions you missed.

## 8.1 Fitting Is Not Learning

Q1 interpret Polynomials of degree 1, 4 and 15 are fitted to 20 noisy samples of $\sin 2\pi x$. Which has the lowest error on those 20 points?

A. Degree 1
B. Degree 15
C. All three are equal
D. Degree 4

Q2 concept What does "generalisation" mean for a trained network?

A. Reaching the lowest possible error on the training data
B. Working equally well on any kind of data whatsoever
C. Using as few parameters as the problem allows
D. Performing well on new data from the same distribution

Q3 interpret A large network reaches near-zero training error on a dataset whose labels were shuffled at random. What does this show?

A. Training error is a reliable measure of what was learned
B. It has discovered a hidden pattern in the shuffled labels
C. It can memorise data that has no pattern to learn
D. The labels can't have been shuffled properly after all

Q4 apply How should you decide whether your model generalises?

A. Train it for more epochs and see if loss falls
B. Measure its error on data it was not trained on
C. Count how many parameters the network has
D. Check that the training loss is close to zero

Q5 concept Between the training points, why does the degree-15 fit swing so wildly?

A. High-degree polynomials can only represent periodic functions well
B. It has too few parameters to follow the sine curve smoothly
C. The training points were placed too close together on the axis
D. It was fitted to the noise, and nothing constrains it between points

## 8.2 Training, Validation and Test Sets

Q6 concept What is the validation set used for?

- A Choosing the architecture, learning rate and regularisation
- B Computing the gradients that update the network's weights
- C Storing extra copies of the training data for later use
- D Giving the one final, unbiased estimate of performance

Q7 apply You try 30 configurations and report the best test-set score. What's wrong?

- A Thirty configurations is too few to find a good model
- B Nothing: the test set is meant for comparing models
- C You should have reported the average of all 30 scores
- D You tuned on the test set, so the score is optimistic

Q8 apply A medical dataset has several scans per patient. How should you split it?

- A At random by scan, so every set gets a mix of scans
- B By image size, so each set has similar resolutions
- C By date of scan, with the oldest scans in the test set
- D By patient, so each patient's scans are all in one set

Q9 concept You're forecasting next month's sales from past months. How should you split the data?

- A Use k-fold cross-validation with random folds
- B Train on the past and test on the most recent period
- C Shuffle the months and split them at random 80/20
- D Test on the oldest data and train on the newest

Q10 calculate With 5-fold cross-validation, how many times is each training example used for validation?

- A Once
- B Five times
- C Never
- D Four times

## 8.3 Underfitting, Overfitting and Learning Curves

Q11 interpret Training error 18%, validation error 19%, while similar models reach 5%. What's the diagnosis?

- A A good fit, since the errors are close together
- B Underfitting: both errors are high
- C Data leakage between the two sets
- D Overfitting: the validation error is higher

Q12 `interpret` Training error 1%, validation error 15%. What's the diagnosis?

- A The learning rate is far too low
- B Underfitting: the model is too small
- C A good fit: training error is low
- D Overfitting: there is a large gap

Q13 `concept` On a typical learning curve, what does the validation cost do as training continues?

- A It rises steadily from the very first epoch
- B It falls, reaches a minimum, then rises
- C It falls steadily for as long as training goes
- D It stays flat while the training cost falls

Q14 `concept` In the bias–variance decomposition, which term no model can reduce?

- A The model's training error
- B The noise variance $\sigma^2$
- C The model's variance term
- D The squared bias term

Q15 `apply` A degree-15 polynomial changes drastically when refitted on a fresh sample of 20 points. Which term is large?

- A Noise
- B Training error
- C Variance
- D Bias

## 8.4 Preparing the Data

Q16 `calculate` Training data for a feature range from 20 to 70. What is the min–max scaled value of 45?

- A 25
- B 0.9
- C 0.64
- D 0.5

Q17 `apply` You standardise using the mean and standard deviation of the whole dataset before splitting. What's the problem?

- A Information from the test data leaks into training
- B The mean must be computed on the test set instead
- C Nothing: scaling statistics don't affect evaluation
- D Standardisation should only be applied to the targets

Q18 concept Why do unscaled inputs in the thousands cause trouble for sigmoid hidden units?

- A Large inputs saturate the units and kill their gradients
- B Unscaled inputs increase the number of network parameters
- C Large inputs make the network's outputs turn negative
- D Sigmoid units cannot accept any input larger than one

Q19 apply A feature "colour" takes values red, green and blue. How should you feed it to the network?

- A One-hot encode it as three binary inputs
- B Standardise the colour names before training
- C Code it as 1, 2, 3 in a single numeric input
- D Drop it, since networks can only use numbers

Q20 interpret 99% of transactions are legitimate. Your fraud model has 99% accuracy. What should you check next?

- A The number of hidden layers in the network
- B Its recall on fraud, via the confusion matrix
- C Nothing: 99% accuracy is excellent performance
- D Its training loss, to make sure it's still falling

## 8.5 Hyperparameters and a Training Recipe

Q21 interpret The training cost oscillates wildly and occasionally jumps up. What should you change first?

- A Lower the learning rate
- B Remove the validation set
- C Add more hidden layers
- D Raise the learning rate

Q22 apply Your new network can't drive the training loss near zero even on 100 examples. What does this most likely indicate?

- A The test set is too small
- B The network is overfitting
- C You need more training data
- D A bug in the code or data

Q23 concept Why search learning rates like 0.001, 0.01, 0.1, 1 rather than 0.1, 0.2, 0.3, 0.4?

- A Optimisers reject any learning rate above 0.4
- B Its effect depends on the order of magnitude
- C Smaller learning rates always train faster
- D Learning rates must always be powers of ten

Q24 apply Training and validation errors are both high and close. Which change fits the recipe?

A Stopping training at an earlier epoch

B Testing on the test set more often

C Stronger dropout and more weight decay

D A bigger network or longer training

Q25 concept With more than two or three hyperparameters, why is random search usually preferred to grid search?

A It needs no validation set to compare runs

B It is the only method that works with Adam

C It always finds the exact global optimum

D It explores more distinct values of each one

# Answer key

Each answer comes with the reasoning, including why the most tempting wrong option is wrong.

Q1 **B** · lesson 8.1

The most flexible model fits the training points most closely, noise included, so degree 15 has the lowest training error, and yet the worst predictions between the points. Degree 4 generalises best; degree 1 underfits.

Q2 **D** · lesson 8.1

Generalisation is about unseen inputs drawn from the same distribution as the training data. Low training error is only a proxy; working on any distribution is far more than generalisation promises.

Q3 **C** · lesson 8.1

Random labels contain nothing to generalise, so fitting them is pure memorisation. That's why low training error alone proves nothing: you must test on held-out data.

Q4 **B** · lesson 8.1

Generalisation can only be measured on held-out data. The training loss can be near zero for a memorising model, parameter counts don't tell you, and more epochs only show the training loss.

Q5 **D** · lesson 8.1

With 16 parameters and 20 noisy points, the fit bends to pass near each point, noise included, and is free to swing between them. It has too many parameters, not too few.

Q6 **A** · lesson 8.2

Validation data drive every decision other than the weights themselves, and detect overfitting. Gradients come from the training set; the final unbiased estimate comes from the test set, used once.

Q7 **D** · lesson 8.2

Picking the best of 30 test scores turns the test set into a validation set, and the winning score is biased upwards. Compare configurations on validation data and look at the test set once, at the end.

Q8 **D** · lesson 8.2

Scans of the same patient are strongly related; splitting them across sets lets the model recognise the patient rather than the condition, inflating the test score. Group the split by patient.

Q9 **B** · lesson 8.2

In use, the model always predicts the future from the past, so the test must mirror that. A random split lets the model learn from months after the ones it's tested on, which is leakage.

Q10 **A** · lesson 8.2

Each of the 5 folds takes one turn as the validation fold, so every example is validated exactly once and used for training in the other 4 rounds.

Q11 **B** · lesson 8.3

Both errors are high and close together: the model lacks capacity or hasn't trained enough. Overfitting shows a large gap; a small gap alone doesn't mean a good fit if both errors are poor.

Q12 **D** · lesson 8.3

Low training error with much higher validation error is the overfitting signature; the 14-point gap measures it. An underfitting model would have high training error too.

Q13 **B** · lesson 8.3

Early on, both costs fall as the network learns the real pattern; later it fits noise, so training cost keeps falling but validation cost rises. The minimum is where to stop.

Q14 **B** · lesson 8.3

$\sigma^2$ is randomness in the targets themselves; no model can predict it. Bias falls with more capacity, variance falls with more data or regularisation, and training error isn't a term of the decomposition.

Q15 **C** · lesson 8.3

Variance measures how much the fitted model changes with the training sample: high-capacity models have high variance. High bias would show as consistently missing the pattern; noise is fixed by the data.

Q16 **D** · lesson 8.4

$(45 - 20)/(70 - 20) = 25/50 = 0.5$. 0.64 divides by 70 instead of the range, 0.9 divides by 50 without subtracting the minimum, and 25 forgets to divide.

Q17 **A** · lesson 8.4

The test rows influence the statistics used to transform the training data, a subtle form of leakage. Compute them on the training set only and apply the same transformation to validation and test data.

Q18 **A** · lesson 8.4

Large inputs give large $|z|$, where $\sigma' \approx 0$, so learning stalls; the loss surface is also badly conditioned. Sigmoids accept any input, and scaling doesn't change the parameter count.

Q19 **A** · lesson 8.4

One-hot encoding gives each colour its own input without implying an order. Coding 1, 2, 3 tells the network blue is "three times" red and green sits between them, which is meaningless.

Q20 **B** · lesson 8.4

Predicting "legitimate" for everything already scores 99%. The confusion matrix, precision and recall show whether it catches any fraud. Consider re-weighting the loss or over-sampling fraud cases.

Q21 **A** · lesson 8.5

Oscillating or jumping cost is the signature of steps that overshoot: reduce $\eta$. Raising it makes things worse, and architecture or validation changes don't address unstable steps.

Q22 **D** · lesson 8.5

Memorising 100 examples is easy for a reasonably sized network; failing suggests wrong labels, a broken gradient or a mistaken setting. Overfitting would mean succeeding too well, and more data won't fix a bug.

Q23 **B** · lesson 8.5

Moving from 0.001 to 0.01 matters as much as 0.1 to 1, so a log scale covers the useful range evenly. There's no rule that rates be powers of ten, and smaller rates are usually slower.

Q24 **D** · lesson 8.5

Both errors high means underfitting, so add capacity, train longer, or raise $\eta$ (and check normalisation). Dropout, weight decay and early stopping fight overfitting; repeated test-set use is leakage.

Q25 **D** · lesson 8.5

A grid's cost grows exponentially and repeats the same few values of each hyperparameter; random sampling tries many different values for the same budget. It still needs validation data and guarantees nothing.