

# Oral Exam: Machine Learning (2ST129)

Måns Magnusson, Department of Statistics, Uppsala University

## Oral Exam Instructions

The oral exam will start at the scheduled time, and you need to be there when it begins. We, as examiners, will then take a random sample of three students and inform you of this order. We will continuously inform you of the two students in the queue so you can prepare and wait for your turn. This means that if you are not next in line, you can go to the toilet, etc., and do not need to be immediately prepared. If you do not show up when called, you will fail the exam.

During the exam, you will get one random question from the list of questions below and will have to answer it orally or with the help of a whiteboard and pen to show that you know the material. If you fail the first question, a new random question will be drawn (however, not the previous question). If the second question also is failed, you will fail the exam. Otherwise, you will have passed the exam. If you have done the reading assignment and the turn-in assignments, this should not be hard. However, I recommend reviewing the questions before the exam to be prepared.

The oral exam will have two additional re-examination opportunities, one within 3-14 days and one on the final day of the course, after the presentations. If all three oral exams are failed, the student will need to retake the course in full next year.

## Oral Exam Questions – Block 1 through 5

### Assignment 1

1. Explain the difference between gradient descent, stochastic gradient descent, and mini-batch gradient descent. When might you prefer one over another?
2. Derive and explain the gradient of the negative log-likelihood (NLL) for logistic regression where

$$p_i = \frac{1}{1 + e^{-x_i^T \theta}}$$

and

$$\mathcal{L}(\theta) = \prod_i p_i^{y_i} (1 - p_i)^{(1-y_i)}.$$

3. What does the learning rate ( $\eta$ ) control in gradient descent, and what happens if it is too large or too small?
4. Give at least two examples of differences between *traditional regression methods* and the *pure prediction algorithms* according to Efron (2020).

5. In many machine learning applications, we use Stochastic Gradient Descent, even though other optimization algorithms that use second-order derivatives are better. Why is this the case?
6. Describe the concept of model complexity/capacity.
7. Explain the main difference between Ridge regression and LASSO, and give one practical implication.
8. What is the difference between training error and generalization error? Why is it important to distinguish between them?
9. Explain the bias–variance trade-off in supervised learning.
10. What is the purpose of cross-validation, and how does  $k$ -fold cross-validation work?
11. Describe how regularization affects the objective function in regression models.
12. Explain the probabilistic interpretation of L2 (ridge) regularization.
13. What are the main differences between linear regression and logistic regression in terms of model output and objective function?
14. What is the role of the objective (or loss) function in supervised learning?
15. How does stochastic gradient descent (SGD) differ in convergence behavior compared to full-batch gradient descent?
16. How does regularization help prevent overfitting in machine learning models?
17. What is the main difference in the effects of L1 and L2 penalties on model coefficients?
18. When using cross-validation to choose a regularization parameter ( $\lambda$ ), what does the optimal  $\lambda$  represent?

## Assignment 2

19. Explain how a decision tree decides where to split the data. What criterion is used and what is its goal?
20. What is the main idea behind bagging (bootstrap aggregating), and why does it improve model performance?
21. Describe how a random forest differs from bagging. What additional randomness does it introduce, and why?
22. What is the fundamental idea behind boosting, and how does it differ conceptually from bagging?
23. In ensemble methods like bagging and boosting, what happens to bias and variance as more models are added?
24. What is “overfitting” in a decision tree, and how can it be prevented?
25. Why is randomness crucial in ensemble methods like random forests?
26. What does the parameter  $m$  (number of features sampled at each split) control in a random forest?
27. What role does the learning rate play in boosting algorithms?

## Assignment 3

28. Describe the output layer in a (feed-forward) neural network. What does it do?
29. Describe the input layer in a (feed-forward) neural network. What does it do?
30. What is the benefit of using (negative) log-likelihood as loss functions?
31. Describe weight decay regularization.
32. What is the role of activation functions in a neural network, and why are ReLU (or its variants) often preferred?

33. What is backpropagation, and how does it enable a neural network to learn?
34. Explain how batch normalization works and why it improves training stability and generalization.
35. Describe how early stopping can be used as a regularization technique in neural networks.
36. What is transfer learning, and when is it especially useful?
37. What does dropout do during training, and how does it help prevent overfitting?
38. What are the main components of a feed-forward neural network architecture?
39. Why do we typically use softmax activation in the output layer for multi-class classification?
40. What is the purpose of using validation accuracy and loss curves during training?
41. How does transfer learning benefit from freezing early layers in a pre-trained model?
42. What is fine-tuning in transfer learning, and how does it differ from feature extraction?
43. What loss functions are typically used for regression and classification tasks in neural networks, and why?
44. Why does using too many hidden layers or neurons not always improve performance?
45. What are frozen layers in transfer learning, and why might we unfreeze some of them later?
46. What evaluation metrics besides accuracy can be useful for assessing a neural network's performance, and when?

#### Assignment 4

47. What advantages do convolutional layers offer compared to fully connected layers when processing image data?
48. Explain the purpose of pooling layers in CNNs and how max pooling works.
49. How does padding affect the output of a convolutional layer, and why might we use it?
50. Explain what a convolutional kernel (filter) is and how it is learned during training.
51. Why do deeper CNNs generally perform better than shallower ones, and what are the risks of increasing depth too much?
52. What is the difference between stride and pooling in CNNs?
53. What is transfer learning, and how can pretrained CNNs like VGG16 improve performance on small datasets?
54. When performing hyperparameter tuning, why is random search often preferred over grid search?
55. Describe data augmentation and how it can improve CNN generalization.
56. What does “feature map” mean in the context of CNNs, and what does it represent?

#### Assignment 5

57. How do word embeddings differ from one-hot encoding, and why are embeddings useful in NLP?
58. Explain the role of the query, key, and value matrices in the transformer attention mechanism.
59. Why do transformers use positional encoding, and how does it work?

60. What is multi-head attention, and why is it beneficial compared to single-head attention?
61. What is BERT, and how does it differ from the original transformer model?
62. What problem does the attention mechanism solve compared to traditional RNNs?
63. What does “self-attention” mean in the transformer architecture?
64. What are word embeddings like Word2Vec or GloVe missing that contextual embeddings like BERT provide?
65. What is the main architectural difference between the transformer encoder and decoder?
66. How does fine-tuning differ from pretraining in transformer models?
67. What is tokenization in NLP models, and why do models like BERT use WordPiece tokenization?
68. How does GPT differ architecturally and conceptually from BERT?