Machine Learning 2019

Final Exam

25 March 2019, 15:15–18:00

The questions start on the next page. **Please do not open the exam booklet until the scheduled starting time.**

Feel free to write on the exam booklet, and take it with you afterwards. The exam will also be made available on Canvas later today, together with the correct answers. We recommend that you copy your final answers onto the booklet, so you can check how you did.

To get a passing grade, you will need to get approximately 25 of the 40 questions correct. The true pass mark will be decided after the results have been analysed: if any questions are deemed to have been too difficult, they will become bonus questions.

Rules:

- You are allowed to use a calculator or graphical calculator.
- You are *not* allowed to use your phone or smartphone.
- The exam is closed-book.
- You are allowed to use the formula sheet provided through Canvas.
- The formula sheet should not have any writing on it, except in the cheat sheet box. Anything inside the box should be written by hand.

Addendum: questions 9 and 21 contained mistakes. These will be considered bonus questions. You still get the point if you chose on of the answers that were correct, but the question will not count towards the total.

1 Questions

1. What separates **offline learning** from **reinforcement learning**?

   A In reinforcement learning the training labels are reinforced through boosting.
   B Offline learning can be done without connection to the internet. Reinforcement learning requires reinforcement from a separate server.
   C In reinforcement learning, the learner takes actions and receives feedback from the environment. In offline learning we learn from a fixed dataset.
   D Reinforcement learning uses backpropagation to approximate the gradient, whereas offline learning uses symbolic computation.

2. The most important rule in machine learning is "never judge your performance on the training data." If we break this rule, what can happen as a consequence?

   A The loss surface no longer provides an informative gradient.
   B We get cost imbalance.
   C We end up choosing a model that overfits the training data.
   D We commit multiple testing.

3. The squared error loss function looks like this: $\sum_i (y_i - t_i)^2$, where the sum is over all instances, y_i is the model output for instance i and t_i is the training label. Which is **not** a reason for squaring the difference between the two?

   A It ensures that negative and positive differences don't cancel out in the sum.
   B It ensures that large errors count very heavily towards the total loss.
   C When used in classification, it ensures that points near the decision boundary weigh most heavily.
   D It is a consequence of assuming normally distributed errors, and deriving the maximum likelihood solution.

4. We have a classifier c and a test set. Which is **true**?
   A To compute the *precision* for c on the test set, we must define how to turn it into a ranking classifier.
   B To compute the *false positive rate* for c on the test set, we must define how to turn it into a ranking classifier.
   C To compute the *confusion matrix* for c on the test set, we must define how to turn it into a ranking classifier.
   D To compute the *area under the curve* for c on the test set, we must define how to turn it into a ranking classifier.

5. Testing too many times on the test set increases the chance of random effects influencing your choice of model. What is the solution suggested in the lectures?
   A To withhold the test set until a hypothesis is established, and use a train/validation split on the remainder to evaluate model choice and hyperparameters.
   B To perform cross-validation on the training data, so that all instances are used as training data at least once.
   C To use bootstrap sampling to gauge the variance of the model.
   D To use a boosted ensemble, to reduce the variance of the model, and with it, the probability of random effects.

6. Different features in our data may have wildly different scales: a person's age may fall in the range from 0 to 100, while their savings can fall in the range from 0 to 100 000. For many machine learning algorithms, we need to modify the data so that all features have roughly the same scale. Which is **not** a method to achieve this?
   A Imputation
   B Standardization
   C Normalization
   D Principal Component Analysis

7. Sophie and Emma are doing a machine learning project together, and training the single-feature regression model $y = w_1x^2 + w_2x + b$. Sophie says this is a non-linear model, because it learns a parabola not a line. Emma says it is a linear model, but on the features x and x^2, derived from the original single feature.
   A Sophie is right. Emma is wrong.
   B Emma is right. Sophie is wrong.
   C Both are right.
   D Both are wrong.

8. You finish this exam and hand it in. You say to your fellow students: "The probability that I've passed this exam is 60%." Which is **true**?

   A This is **not** a frequentist use of the word probability, because it uses a percentage instead of a frequency.
   B This is **not** a Bayesian use of the word probability because it expresses a belief, not a result of repeated experiments.
   C This is a frequentist use of the word probability.
   D This is a Bayesian use of the word probability.

9. (Bonus question) We have a dataset with a number of categoric features, each of which takes one of two values. In naive Bayes, probability estimates can go to zero if we see a feature take a value that it doesn't take in the training data. We can solve this by Laplace smoothing, which we can interpret as adding pseudo-observations. Which is **true**?

   A The number of pseudo-observations we need to add is the number of classes times two.
   B The number of pseudo-observations we need to add is the number of classes, times two to the power of the number of features.
   C After adding the pseudo-observations, we must change the denominator for the probability estimate to ensure that all probabilities still add up to ~~zero~~ one.
   D After adding the pseudo-observations, we must change the numerator for the probability estimate to ensure that all probabilities still add up to ~~zero~~ one.

10. We have two discrete random variables: A with outcomes $1, 2, 3$ and B with outcomes a, b, c. We are given the joint probability $p(A, B)$ in a table, with the outcomes of A enumerated along the rows (vertically), and the outcomes of B enumerated along the columns (horizontally). How do we compute the probability $p(A = 1 \mid B = a)$?

    A We find the probability in the first column and the first row.
    B We find the probability in the first column and the first row, and divide it by the sum over the first column.
    C We find the probability in the first column and the first row, and divide it by the sum over the first row.
    D We sum the probabilities over the first column and the first row.

11. The maximum margin objective that leads to the support vector machine, can be developed in two different ways. Which is **false**?

    A We can rewrite the objective so that it can be trained with basic gradient descent, but then we can't use the kernel trick.

    B We can rewrite the objective so that we can use the kernel trick, but then we can't train it with basic gradient descent.

    C To be able to use basic gradient descent, we must rewrite the objective function using Lagrange multipliers.

    D To be able to use the kernel trick, we must rewrite the objective function using Lagrange multipliers.

12. L1, L2, and Dropout are forms of *regularization*. Which is **true**?

    A L1 and L2 work by randomly disabling hidden nodes.

    B Dropout works by adding a term to the loss function that represents the complexity of the model.

    C L1 is a sparsity-enforcing regularizer. It makes it more likely that weights become exactly 0.

    D L2 is a sparsity-enforcing regularizer. It makes it more likely that weights become exactly 0.

13. A convolutional layer is defined by four parameters: *stride*, *padding*, *kernel size* and the number of *output channels*. What should we do if we want the resolution of the output to be the same as the input?

    A Don't use any padding, and make the stride as big as the number of output channels.

    B Make the kernel size the same as the image resolution, and set padding and stride to 0.

    C Set the kernel size to 1 by 1, use a stride of 1 and set padding to 3.

    D Set the stride to 1, and the padding to half the kernel size minus one.

14. The probability density function of the univariate normal distribution is $N(x \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(\mu - x)^2\right]$. What is the function of the part in green?

    A It ensures that the inflection points of the density curve hit the values $-\sigma$ and $+\sigma$

    B It ensures that the distribution has a definite *scale*: a particular range of values that are much more likely than other outcomes.

    C It ensures that the mean μ has a higher probability than any other outcome.

    D It ensures that the area under the total probability density curve sums to 1.

15. The Expectation-Maximization algorithm is used to approximate the maximum likelihood fit for a probability model. For which of the following models does it make sense to use it?

    A A univariate normal distribution
    B A multivariate normal distribution
    C A Gaussian mixture model
    D A generator neural network

16. The slides mention four types of GANs (generative adversarial networks). One of them trains two generators: to map from domain A to B and from B to A. It then adds a term to the loss function that ensures that mapping from A to B and back again, results in as little change as possible. Which GAN is this?

    A Vanilla GAN
    B Conditional GAN
    C CycleGAN
    D StyleGAN

17. What is the **cold start** problem?

    A The situation where a neural network is initialized so that its sigmoid activations are saturated, and it has no gradient to start learning with.
    B The situation in sequence-to-label learning where there is a long distance between the start of the sequence and the label, so that the model only learns from the end of the sequence.
    C The situation where we can sample from a generator network but we don't know which instance from the data to compare it to, because we don't have a mapping from the data to the latent space.
    D The situation where a new item (user, movie, etc.) is added to a recommender system, and we have no ratings to build an embedding from.

18. What is **mode collapse**?

    A When the loss surface of a generator network flattens out, so the gradient becomes zero.
    B When a network has a sampling step in the middle, so we cannot backpropagate down to the input.
    C When the distribution in the latent space is similar to a hypersphere, so we should interpolate along an arc instead of a line.
    D When a generator network outputs the mean of the data, instead of providing different samples with variation between samples.

19. The *variational* autoencoder adapts the regular autoencoder in a number of ways. Which is **not** one of them?

    A It adds a sampling step in the middle.

    B It makes the outputs of the encoder and decoder parameters of probability distributions.

    C It adds a loss term to ensure that the latent space is laid out like a standard normal distribution.

    D It adds a discriminator that learns to separate generated examples from those in the dataset.

20. In a tree model (a decision or regression tree), when does it make sense to split on the same feature twice for the same instance?

    A This never makes sense.

    B When the feature is categoric, but not when it is numeric.

    C When the feature is numeric, but not when it is categoric.

    D When we are training a regression tree, but not when we are training a decision tree.

21. (Bonus question) Which is **true**?

    A Bagging reduces variance. To reduce bias we can use boosting.

    B Boosting reduces variance. To reduce ~~variance~~ bias, we can use bagging.

    C It is not possible to reduce bias through ensembling: hypothesis boosting has been proven to be impossible.

    D Ensembling methods reduce neither bias nor variance. They just allow us to estimate our bias and variance more accurately.

22. Why is the Markov assumption like the Naive Bayes assumption?

    A They both translate a probability model to a differentiable network, so that we can use backpropagation to approximate a solution.

    B They both use approximations to conditional probability which are not actually valid probabilities, but which work in practice.

    C They both make assumptions which we know to be untrue, but which simplify the model a lot.

    D They both enable us to convert numeric data to categoric data.

23. In the context of recurrent neural networks (RNNs), what is unrolling?

A A process that turns an RNN into a generator by providing it with a randomly sampled input.
B A process that turns an RNN into a non-recurrent network by making a copy of the network for each timestep in the input sequence.
C A process that samples a random sequence by sequentially sampling from the probabilities predicted by the network, and feeding it back the resulting sample.
D A process that eliminates recurrent connections, by decaying particular weights of the network.

24. We are building a recommender system for movies, based on matrix factorization. We decide to withhold some of the users and movies as a test set. Will this work?
A No, different users may have different average ratings. Discarding some users may change the distribution.
B No, the model trains embeddings for the users and the movies. We can't make predictions for users and movies that we haven't seen during training.
C No, more obscure movies tend to get higher ratings on average, because only people who like them are aware of them. Withholding these movies will change the data distribution.
D Yes, but only if we make sure that users in the test set joined *after* users in the training set.

25. Why is gradient descent difficult to apply in a reinforcement learning setting?
A The loss surface is flat in most places, so the gradient is zero almost everywhere.
B The backpropagation algorithm doesn't apply if the output of a model is a probability distribution.
C There is a non-differentiable step between the model input and the model output.
D There is a non-differentiable step between the model output and the reward.

We want to train the following model:

$$y_i = -v{x_i}^2 + wx_i + b$$

with parameters w and b, where x_i and y_i are scalars. the dataset provides target values t_i. We derive the gradient of the loss with respect to b

as follows:

$$
\begin{aligned}
\frac{\partial \frac{1}{2}\sum_i (y_i - t_i)^2}{\partial b} &= \frac{1}{2}\frac{\partial \sum_i (y_i - t_i)^2}{\partial b} && (1)\\
&= \frac{1}{2}\sum_i \frac{\partial (y_i - t_i)^2}{\partial b} && (2)\\
&= \frac{1}{2}\sum_i \frac{\partial (y_i - t_i)^2}{\partial (y_i - t_i)}\frac{\partial (y_i - t_i)}{\partial b} && (3)\\
&= \frac{1}{2}\sum_i 2(y_i - t_i)\frac{\partial (y_i - t_i)}{\partial b} && (4)
\end{aligned}
$$

26. To get from line (2) to line (3), we use the

A Chain rule
B Product rule
C Constant factor rule
D Sum rule

27. To get from line (1) to line (2), we use the

A Chain rule
B Product rule
C Constant factor rule
D Sum rule

28. Fill in the definition of y_i and work out the derivative with respect to b. Which is the correct result?

A $\frac{1}{2}\sum_i \left(-v{x_i}^2 + wx_i + b - t_i\right)$
B $\sum_i \left(-v{x_i}^2 + wx_i + b - t_i\right)$
C $\frac{1}{2}\sum_i x_i \left(-v{x_i}^2 + wx_i + b - t_i\right)$
D $\sum_i x_i \left(-v{x_i}^2 + wx_i + b - t_i\right)$

We have the following training set:

	x_1	x_2	label
a	0	1	Neg
b	2	2	Neg
c	1	4	Neg
d	2	5	Neg
e	3	6	Pos
f	6	8	Pos
g	5	3	Pos
h	8	7	Pos

For the following questions, it helps to draw the data and the classification boundary in feature space.
We use a linear classifier defined by

$$c(x_1, x_2) = \begin{cases} \text{Pos} & \text{if } 0 \cdot x_1 + x_2 - 2 > 0 \\ \text{Neg} & \text{otherwise.} \end{cases}$$

29. If we turn c into a *ranking* classifier, how does it rank the points, from most Negative to most Positive ?

A a b c d e f g h
B a b g c d e h f
C a b g c d h e f
D a c b d e g f h

30. How many ranking errors does the classifier make?

A None
B 1
C 2
D 3

31. If we draw a coverage matrix (as done in the slides), what proportion of the cells will be red?

A $\frac{3}{15}$ **B** $\frac{6}{15}$ **C** $\frac{1}{8}$ **D** $\frac{6}{16}$

Here are two distributions, p and q, on the members of a set $X = \{a, b, c, d\}$. We will use binary entropy (i.e. computed with base-2 logarithms). Note that in the computation of the entropy $0 \cdot \log_2(0) = 0$, but otherwise, $\log_2 0$ is undefined as usual.

	p	q
a	¼	0
b	¼	¼
c	¼	¼
d	¼	½

32. What are their entropies?

A $H(p) = 2$, $H(q) = 1$
B $H(p) = 2$, $H(q) = 1.5$
C $H(p) = 1$, $H(q) = 1$
D $H(p) = 1$, $H(q) = 1.5$

33. What is the cross-entropy $H(q, p)$?

A $H(q, p) = 1$
B $H(q, p) = 1.5$
C $H(q, p) = 2$
D $H(q, p) = 2.5$

34. If you try to compute $H(p, q)$, you'll notice that something goes wrong. What does this mean?

A As $q(a)$ goes to zero, a's codelength with code q goes to infinity. This makes the expected codelength under the uniform distribution p infinite as well.
B Because p is uniform, its expected codelength is always optimal under *any* distribution.
C A standard (graphical) calculator does not have the precision to compute the answer. In a python notebook, we would not have a problem.
D Because $q(a) = 0$ the expected codelength with code q, under distribution q is not defined. This means that the resulting cross-entropy also becomes undefined.

We will use the backpropagation algorithm to find the derivative of the function

$$f(x) = \sin\left(\sin(x)\cos(x)\right)$$

with respect to x.

First, we break the function up into modules:

$$\begin{aligned} f &= \sin(c) \\ c &= \ldots \\ b &= \ldots \\ a &= \sin(x) \end{aligned}$$

35. What should be in the place of the dots?

A $b = \cos(x)$, $c = ab$
B $b = \cos(x)$, $c = \sin(x)\cos(x)$
C $b = \cos(c)$, $c = ab$
D $b = \cos(c)$, $c = \sin(x)\cos(x)$

36. In terms of the *local* derivatives. Which is the correct expression for the gradient $\frac{\partial f}{\partial x}$?

A $\sin(c)\,[b\cos(x) + a\sin(x)]$
B $\sin(c)\,[b\cos(x) - a\sin(x)]$
C $\cos(c)\,[b\cos(x) + a\sin(x)]$
D $\cos(c)\,[b\cos(x) - a\sin(x)]$

Consider the following task. The aim is to predict the class y from the binary features x_1, x_2, x_3 and x_4.

x_1	x_2	x_3	x_4	y
B	A	A	A	Yes
A	A	B	A	No
B	A	A	A	Yes
A	A	B	A	No
B	B	A	A	Yes
B	B	A	A	Yes
B	A	A	B	Yes
A	B	B	B	No
A	A	B	B	No
A	B	A	B	Yes
B	B	B	B	No
A	B	B	B	No

37. In standard decision tree learning (as explained in the lectures), without pruning. Which would be the first feature chosen for a split?

A x_1 **B** x_2 **C** x_3 **D** x_4

38. If we remove that feature from the data, which would be chosen instead?
A x_1 **B** x_2 **C** x_3 **D** x_4

The EM and VAE algorithms are both based on the following decomposition.

$$
\begin{aligned}
L(q, \theta) + KL(q, p) &= \mathbb{E}_q \ln \frac{p(\mathbf{x}, \mathbf{z} \mid \theta)}{q(z \mid \mathbf{x})} - \mathbb{E}_q \ln \frac{p(\mathbf{z} \mid \mathbf{x}, \theta)}{q(\mathbf{z} \mid \mathbf{x})} \\
&= \mathbb{E}_q \ln p(\mathbf{x}, \mathbf{z} \mid \theta) - \mathbb{E}_q \ln q(z \mid \mathbf{x}) - \langle a \rangle + \mathbb{E}_q \ln q(z \mid \mathbf{x}) \\
&= \mathbb{E}_q \ln p(\mathbf{x}, \mathbf{z} \mid \theta) - \mathbb{E}_q \ln p(\mathbf{z} \mid \mathbf{x}, \theta) \\
&= \mathbb{E}_q \ln \frac{p(\mathbf{x}, \mathbf{z} \mid \theta)}{p(\mathbf{z} \mid \mathbf{x}, \theta)} \\
&= \mathbb{E}_q \ln \frac{p(\mathbf{z} \mid \mathbf{x}, \theta) p(\mathbf{x} \mid \theta)}{p(\mathbf{z} \mid \mathbf{x}, \theta)} \\
&= \mathbb{E}_q \ln p(\mathbf{x} \mid \theta) \\
&= \langle b \rangle
\end{aligned}
$$

39. What should be in place of $\langle a \rangle$ and $\langle b \rangle$?
A $\langle a \rangle : \mathbb{E}_q \ln p(\mathbf{z} \mid \mathbf{x}, \theta), \quad \langle b \rangle : \ln p(x \mid \theta)$
B $\langle a \rangle : \mathbb{E}_q \ln p(\mathbf{z} \mid \mathbf{x}, \theta), \quad \langle b \rangle : \mathbb{E}_q \ln p(x)$
C $\langle a \rangle : \mathbb{E}_q \ln p(\mathbf{z}, \mathbf{x} \mid \theta), \quad \langle b \rangle : \ln p(x \mid \theta)$
D $\langle a \rangle : \mathbb{E}_q \ln p(\mathbf{z}, \mathbf{x} \mid \theta), \quad \langle b \rangle : \mathbb{E}_q \ln p(x)$

40. How is this derivation used in the EM algorithm?
A To maximize $\ln p(\mathbf{x} \mid \theta)$, we iterate between choosing θ to maximize $L(q, \theta)$ and then choosing q to minimize $KL(q, p)$.
B To maximize $\ln p(\mathbf{x} \mid \theta)$, we treat $L(q, \theta)$ as a lower bound and optimize its parameters by backpropagation.
C To maximize $L(q, \theta) + KL(q, p)$ we rewrite it to $\ln p(\mathbf{x} \mid \theta)$ and use random search to find the optimal θ.
D To maximize $L(q, \theta) + KL(q, p)$ we rewrite it to $\ln p(\mathbf{x} \mid \theta)$ and apply the kernel trick to find the optimal θ.

Thank you for your effort. Please check that you've put your name and student number on the answer sheet.

The student evaluations will be open on VUNet after this exam. Please take a few minutes to fill them in. Your feedback is crucial for us.