Exam Neural Networks

Wojtek Kowalczyk
w.j.kowalczyk@liacs.leidenuniv.nl

01.06.2018

It is a closed book exam: you are not allowed to use any notes, books, calculators, smartphones, etc. For each question you will get some points; additionally you will get 10 points for free. The number of points attached to each question reflects the (subjective) level of question's difficulty. In total you may get 100 points. The final grade for the exam is the total number of points you receive divided by 10.

The exam consists of a number of "multiple questions - single choice answer" questions. It means that for each question you should select exactly one answer. For every correct choice you get some points; for an incorrect choice the same number of points is subtracted; in case you don't select anything you get 0 points. Therefore, if you are not confident about the correctness of your answer, consider not answering the question at all.

Mark your choices by crossing the selected option. In case you want to "undo" your choice put a circle around the cross. For example, on the left side the option **b** is selected; on the right side nothing is select – the selection of **b** is "undone":

a) bla bla
b) ble ble
c) bli ble
a) bla bla
b) ble ble
c) bli ble

If you think that your marking is no longer readable, put your final choice(s) on the left margin (e.g., by writing "a" if you want to select "a"). Finally, you are free to add to your answers your comments (in a free space). We don't know yet if and how we will process such comments, but it may help us to adjust the exam grade (up or down – depending on the comment).

Before starting answering the questions, fill in the following entries:

Name:

Student number:

Study type (ICT, Astronomy, …):

5 pts	**NetTalk**
Question	In 1987 Sejnowski and Rosenberg published a paper on their Text2Speech system called NetTalk. What is the closest estimate of the number of weights used by their system?
Answer options	a) 100 b) 1000 c) 10000 d) 100000 The NetTalk Network had 7*29 input 80 hidden and 26 output nodes (see slide 27 from NN1). Therefore the number of weights is 7*29*80+80*26=18320 (assuming no biases). So the correct answer is c.

3 pts	**Deep Networks**
Question	What was the type of the first neural network, successfully applied to a practical problem, which could be called “deep” (i.e., having more than 3 hidden layers)?
Answer options	a) A multi-layer perceptron b) A convolutional network (LeNet) c) A recurrent network d) An autoencoder e) A deep belief network f) An LSTM network

9 pts	**Bayes’ Rule**
Question	Suppose that you have to develop, with help of the Bayes’ Rule, a simple system that decides if an input image of a fruit is a banana or an apple. The system should use only one binary feature of input images: the color of the fruit, which may be either yellow or green, and no other colors (or fruits) are possible. As a training set a random sample of 1000 bananas and 1000 apples is given. It turns out that in this set 800 bananas are yellow and 600 apples are green. Additionally, let us assume that in reality apples are 3 times more frequent than bananas (so the training set is biased) - this fact should also be taken into account when building the system. What probability estimate of P(Apple|Yellow) should be produced by your system?
Answer options	a) 0.2 b) 0.3 c) 0.4 d) 0.6 e) Something else f) Apples are 3 times more frequent than bananas, so P(Apple)=0.75, P(Banana)=0.25. Moreover, P(Yellow|Banana)=800/1000=0.8 and P(Yellow|Apple)=400/1000=0.4. Therefore P(Apple|Yellow)=P(Yellow|Apple)*P(Apple)/NF = 0.4*0.75/NF=3/NF and P(Banana|Yellow)=P(Yellow|Banana)*P(Banana)/NF=0.8*0.25/NF=2/NF, where NF is a normalization factor that has to satisfy 3/NF+2/NF=1, so NF=5 and P(Apple|Yellow)=3/5=0.6.

3 pts	**Discriminant function**
Question	Let us suppose that $f(x)$ is a discriminant function for sets A and B. Which of the following functions is not necessarily a discriminant for A and B?
Answer options	a) $–f(x)$ b) $f(x)^2$ c) $\exp(f(x))$ d) $\mathrm{sigmoid}(f(x))$ e) $\exp(-f(x))$ Functions $\exp(\cdot)$, $\mathrm{sigmoid}(\cdot)$, $-(\cdot)$ are strictly monotonic, while $(\cdot)^2$ is not, so $f(x)^2$ might not discriminate between A and B.

3 pts	**Multi-class linear separability**
Question	Which definition of the *multi-class linear separability* concept (restricted here to 3 classes) is correct: Sets A, B, C are linearly separable if:
Answer options	a) Any two of them linearly separable. b) Any one of them is linearly separable from the union of the remaining two sets. c) There are 3 linear functions f_A, f_B, f_C such that: for all $x \in A$ $f_A(x)$ is bigger than $f_B(x)$ and $f_C(x)$, and for all $x \in B$ $f_B(x)$ is bigger than $f_A(x)$ and $f_C(x)$, and for all $x \in C$ $f_C(x)$ is bigger than $f_A(x)$ and $f_B(x)$. d) None of the above definitions is correct. The answer (c) is a rephrasing of the original definition from slide 39 of NN3_LinearModels. The figure on slide 41 demonstrates that the answer (b) is incorrect. The answer (a) is also incorrect: for example, three parallel stripes are separable according to (a) but are not separable according to (c)

5 pts	**Overfitting and Regularization**
Question	One way of fighting overfitting is L^2-regularization: punishing the network for having too big (squared) weights. The regularization parameter λ controls the weight of "punishment". Let us suppose that for a given training set you've trained a network and found the optimal value of λ that minimizes overfitting. Next, you were given a much bigger training set (from the same distribution) and you've retrained the network and optimized λ again. What should you expect on the new value of λ:
Answer options	a) The new λ should be more or less the same as the previous one. b) The new λ should be bigger than the previous one. **c) The new λ should be smaller than the previous one.** d) It's impossible to say if the new λ should be smaller or bigger than the previous one. When the size of the training set is growing then the risk of overfitting is smaller therefore we expect that the new optimal value of λ should be smaller. However, as noticed by some students with statistical background, when working with relatively small data sets and models with relatively few parameters, it is impossible to predict if the optimal value of the new λ would be smaller or bigger. Therefore we decided to consider answer (d) to be partially correct and give 3 points for it.

10 pts	**Backpropagation**
Question	Let us consider the XOR-network: a simple network with two hidden nodes and one output node that is supposed to be trained on the XOR problem. This network, when initialized at random and trained with the backpropagation algorithm converges to a solution of the XOR problem. Now suppose that all weights of this network are initialized to the same value (for example, 1.0) and the backpropagation algorithm is applied again. Which statement is true:
Answer options	a) The algorithm will converge to a correct solution. **b) The algorithm will never converge to a correct solution.** c) The algorithm will converge provided the learning rate is set to a right value. d) The algorithm will converge provided the learning rate is set to a right value and the training cases are shown to the network in a right order. By initializing all weights to the same value we force the two hidden nodes to return the same value, so they could be replaced by a single node, and we know that such a network is not able to solve the XOR problem.

2 pts	**Binary classification problems**
Question	Which setup (activation function of the output layer and the loss function) is recommended for solving binary classification problems with MLP?
Answer options	a) Linear and Sum of Squared Errors b) Linear and Cross-entropy c) Sigmoid and Sum of Squared Errors d) Sigmoid and Cross-entropy e) Softmax and Sum of Squared Errors

2 pts	**Multi-class classification problems**
Question	Which setup (activation function of the output layer and the loss function) is recommended for solving multiclass classification problems with MLP?
Answer options	a) Sigmoid and Sum of Squared Errors b) Sigmoid and Cross-entropy c) Softmax and Sum of Squared Errors d) Softmax and Cross-entropy e) Linear and Cross-entropy

2 pts	**Regression problems**
Question	Which setup (activation function of the output layer and the loss function) is recommended for solving regression problems with MLP?
Answer options	a) Sigmoid and Sum of Squared Errors b) Linear and Sum of Squared Errors c) Softmax and Sum of Squared Errors d) Softmax and Cross-entropy e) Linear and Cross-entropy

4 pts	**SGD with Nesterov momentum**
Question	Suppose that while searching for a minimum of a function $f(x)$, after k iterations of the gradient descent algorithm with Nesterov momentum the algorithm reached a point x_k. What update rule should be applied to find x_{k+1}? Here α denotes the learning rate, β denotes the momentum rate, $g(x)$ denotes the gradient of the function being optimized (i.e., the vector of partial derivatives of f at x), and $d(x)$ denotes the direction of the last step, i.e., $d(x_k) = x_k - x_{k-1}$
Answer options	*a)* $x_{k+1} = x_k + \alpha g(x_k) + \beta d(x_k)$ *b)* $x_{k+1} = x_k - \alpha g(x_k) + \beta d(x_k)$ *c)* $x_{k+1} = x_k + \alpha g(x_k) - \beta d(x_k)$ *d)* $x_{k+1} = x_k - \alpha g(x_k) - \beta d(x_k)$ ***e) none of the above*** **Nesterov momentum involves finding the derivative of f at x_k $\beta d(x_k)$ which was not provided as a possible answer.**

5 pts	**SGD with momentum**
Question	Suppose that while searching for a minimum of a function $f(x)$, after k iterations of the gradient descent algorithm with momentum the algorithm reached a point x_k. What update rule should be applied to find x_{k+1}? Here α denotes the learning rate, β denotes the momentum rate, $g(x)$ denotes the gradient of the function being optimized (i.e., the vector of partial derivatives of f at x), and $d(x)$ denotes the direction of the last step, i.e., $d(x_k) = x_k - x_{k-1}$
Answer options	*a)* $x_{k+1} = x_k + \alpha g(x_k) + \beta d(x_k)$ ***b)*** $\boldsymbol{x_{k+1} = x_k - \alpha g(x_k) + \beta d(x_k)}$ *c)* $x_{k+1} = x_k + \alpha g(x_k) - \beta d(x_k)$ *d)* $x_{k+1} = x_k - \alpha g(x_k) - \beta d(x_k)$ *e) none of the above*

3*3 pts	**LeNet5 (3 questions)**
Question	The first convolutional layer of LeNet5 consists of 6 feature maps, each of size 28x28. Each map is determined by a convolutional filter of size 5x5 which is applied to the input layer of size 32x32, with padding=0 and stride=1. Moreover, each filter uses a bias term.
Answer options	a) How many trainable parameters define this convolutional layer? a. 6*28*28 b. 6*28*28+6 c. 6*25 **d. 6*26** e. None from the above b) How many connections exist between the input and the first convolutional layer? a. 6*25*28 b. 6*26*28 c. 6*25*28*28 **d. 6*26*28*28** e. None from the above

	c) How many connections would exist if the input and the first convolutional layer were fully connected? a. 32*32*6*28*28 b. 32*32*6*28*28+1 c. 32*32*6*28*28+28*28 **d. 32*32*6*28*28+6*28*28** e. None from the above

8 pts	**Vanilla Recurrent Neural Networks**
Question	During the course we discussed a simple recurrent network that was used for language modelling. The network used 8000 input nodes to represent 8000 words (using “one-hot” encoding), 8000 output nodes to represent probability distribution over “the next word in the sentence”, and 100 hidden nodes. No bias parameters were used. What is the total number of trainable weights used by this network?
Answer options	a) 2*100*8000 b) 3*100*8000 c) 2*100*8000 + 100 **d) 2*100*8000 + 100*100** e) None of the above

3 pts	**LSTM Networks**
Question	What is the biggest advantage of LSTM networks over Vanilla Recurrent Networks?
Answer options	a) Faster convergence? b) Ability to remember the most recent states? c) Ability to remember the remote states? **d) Ability of learning which remote and recent information is relevant for the given task and using this information to generate output** e) None of the above

3 pts	**Contrastive Divergence Algorithm**
Question	How many multiplications are needed by the Contrastive Divergence algorithm to update weights for a single input vector x, assuming that three steps of Gibbs sampling are used? Assume that the network has M input nodes, N hidden nodes, and no biases.
Answer options	a) 5*M*N b) 6*M*N c) 7*M*N **d) 10*M*N** **e) None of the above** Both d) and e) are correct answers – depending on what we mean by “update”. We need to “go up” 4 times and “go down” 3 times; additionally we have to compute the products of “input-hidden” states at the beginning and at the end. So 9*M*N multiplications are required – therefore answer e) is correct. However, in the very last step we multiply all delta’s by a learning rate, so the answer 10*M*N is also correct. See slide 9 from NN12_RBMs.

3 pts	**RBM Networks**
Question	What function is optimized when training an RBM network?
Answer options	a) Cross Entropy b) Contrastive Divergence **c) LogLikelihood** d) Mean Squared Error e) None of the above

5 pts	**RBM networks for recommender systems**
Question	During the course we discussed an application of an RBM network to the Netflix Challenge, where the task was to predict a rating a user could give to a movie. There were X users, Y movies, Z possible ratings, and the network was using T hidden nodes. How many trainable parameters had this network? For simplicity, we ignore the biases.
Answer options	a) X*Y*Z*T b) X*Y*Z **c) Y*Z*T** d) X*Z*T e) None of the above

3 pts	**Batch Normalization**
Question	What is batch normalization?
Answer options	a) The process of normalizing each batch of data by dividing it by its mean and standard deviation. b) The process of scaling and shifting each batch of data before the batch is used for training. c) The process of finding an optimal transformation of each batch that is executed before the network is trained. **d) None of the above.** Batch normalization is a complex process which involves finding optimal values of scaling and shifting parameters (for each layer) that is taking place **during the training process**. Therefore answers a, b, c are incorrect.

3 pts	**Modern CNNs**
Question	What is the most successful convolutional network that was discussed during the course?
Answer options	a) LeNet5 b) ImageNet c) AlexNet **d) ResNet** e) GoogleNet