Faculty of Architecture and the Built Environment

*Final Exam*

**Course**: Machine Learning for the Built Environment (GEO5017) – 5 EC

**Location**: ROOM NUMBER

**Time**: XX/XX/XXXX, HH:MM - HH:MM

**Instructor(s)**: NAMES OF ALL TEACHERS



1. This exam is closed-book, but **you're allowed to bring a cheatsheet with no more than 3× A4 papers** (you can choose to print on both sides, and thus no more than 6 printed pages). **You must answer the questions alone** without any help from anybody else.
2. **Electronic devices are forbidden**, including cell phones, iPads, headphones, laptops, and calculators.
3. This exam consists of two parts (starting on the next page):
   (a) 20 multiple-choice questions (40 points). Each question has a single correct answer.
   (b) 6 open questions (60 points).
4. The total number of points is 110 (including 10 bonus points), and the minimum number of points to pass this exam is 55. Your grade is determined by dividing the total number of obtained points by 10.
5. **Write your answers on the printed exam papers**. The provided space is more than sufficient for complete answers. **Attention: any text on additional papers will not be considered for grading; irrelevant text in answers may lead to deductions.**.
6. If you think a question is ambiguous, mark what you think is the best answer. As always, we will consider written regrade requests if your interpretation of a question differed from what we intended.
7. Before starting,
   (a) check if all pages (XX pages in total) and questions are present. If not, ask the teacher.
   (b) fill out your **student ID** and **full name**.
   (c) read the instruction at the beginning of each part before you answer the questions.
8. **You have 2.5 hours**. Schedule your time so you can attempt to answer all the questions.
9. The exam questions are confidential and breaching this confidentiality agreement can be penalized.

**Student ID**: ____________________ **Name**: ________________________________________

# Part 1 Multiple choice questions *[40 points]*

- *Every question has 4 choices and only 1 correct answer*
- *Every question counts for 2 points and the grading is based on*
  - *Answer is correct: 2 points*
  - *No answer is provided or 'not sure' is indicated: 0 point*
  - *Answer is wrong: -1 point (to discourage random guessing)*

1. "A computer program is said to learn from experience $E$ with respect to some task $T$ and some performance measure $P$ if its performance on $T$, as measured by $P$, improves with experience $E$". Suppose we feed a learning algorithm a lot of historical weather data and have it learned to predict the weather. What would be a reasonable choice for $P$?

   A. The process of the algorithm examining a large amount of historical weather data.

   B. The weather prediction task.

   C. The probability of it correctly predicting a future date's weather.

   D. None of these.

   **Answer:** ______________

2. One of the problems below can be best addressed using a supervised learning algorithm, and the others with an unsupervised learning algorithm. Which of the following would you apply supervised learning to? (In each case, assume some appropriate dataset is available for your algorithm to learn from.)

   A. Examine a large collection of emails that are known to be spam emails, to discover if there are sub-types of spam emails.

   B. Take a collection of 1000 essays written on the Dutch Economy and find a way to automatically group these essays into a small number of groups of essays that are somehow "similar" or "related".

   C. Given historical data of children's ages and heights, predict children's height as a function of their age.

   D. Given genetic (DNA) data of 100,000 patients, find the common patterns in their genes that may relate to a certain disease.

   **Answer:** ______________

3. In the following statements about logistic regression, which one is NOT correct?

   A. Logistic regression is a machine learning model for the task of classification.

   B. Logistic regression minimizes the number of erroneously classified objects of the training set.

   C. Logistic regression assumes the posterior probability of the class $y$ is a logistic sigmoid of a linear function of the input data samples $\mathbf{x}$.

   D. Logistic regression is more robust to the data distribution of the input $\mathbf{x}$ compared to the standard linear classification (i.e., using least squares as its loss function).

   **Answer:** ______________

4. Many machine learning models learn a discriminant function that directly maps from the input data $\mathbf{x}$ to the output class $y$. Such functions will generate one or several decision boundaries which partition the input feature space into discrete regions. Which one of the following classifiers is ONLY able to generate a linear decision boundary?

A. A SVM classifier with a polynomial kernel of degree 2.
B. A random forest classifier with 20 decision tree estimators.
C. A neural network with 4 layers of perceptrons.
D. A standard linear classifier.

**Answer:** ____________

5. Which of the following is FALSE about sigmoid activation?
   A. Sigmoid function smooth approximation of the step function.
   B. Sigmoid function is zero-mean non-linear function.
   C. Sigmoid functions can cause vanishing gradients problem.
   D. Sigmoid function brings non-linearity to the network.

**Answer:** ____________

6. Which of the following is FALSE about CNN?
   A. Convolutional filter operations are usually followed by activation functions
   B. Having more layers always leads to better results.
   C. Fully connected layers use extracted features to classify the data.
   D. Shallow networks can help with overfitting in the case of small training data.

**Answer:** ____________

7. Which of the following statements about MLPs (Multi-Layer Perceptrons) in machine learning is true?
   A. MLPs are a type of deep neural networks that consist of only one layer.
   B. MLPs can only be used for classification tasks, but not for regression tasks.
   C. MLPs use backpropagation to update the weights of the connections between neurons during training.
   D. MLPs are only useful when working with large datasets with millions of examples.

**Answer:** ____________

8. What is the main difference between a fully connected neural network and a convolutional neural network?
   A. The fully connected neural network is used for image processing, while the convolutional neural network is used for natural language processing.
   B. The fully connected neural network processes the entire input at once, while the convolutional neural network processes local regions of the input.
   C. The fully connected neural network only has one hidden layer, while the convolutional neural network can have multiple hidden layers.
   D. The fully connected neural network does not use activation functions, while the convolutional neural network uses non-linear activation functions.

**Answer:** ____________

## Part 2 Open questions *[60 points + 10 bonus points]*

- *No point will be awarded without a valid explanation*
- *Keep your answers as concise/short as possible. The space following each question is more than sufficient to provide a complete answer.*

9. The following figure demonstrates four sets of (2D) feature points with different distributions (i.e., the groundtruth clusters). Please suggest a proper clustering method for each feature distribution, and justify your choice of the method. *[10 points]*


**Solution:** Correct answer must contain (but can be organized differently):

- (a) DBSCAN. Data is not linearly separable. *[2 points]*
- (b) DBSCAN. Data is not linearly separable. *[2 points]*
- (c) DBSCAN. Data is not linearly separable. *[3 points]*
- (d) k-means, hierarchical, or DBSCAN. *[3 points]*

10. If the model used for regression is

$$y = a + b(x-1)^2,$$

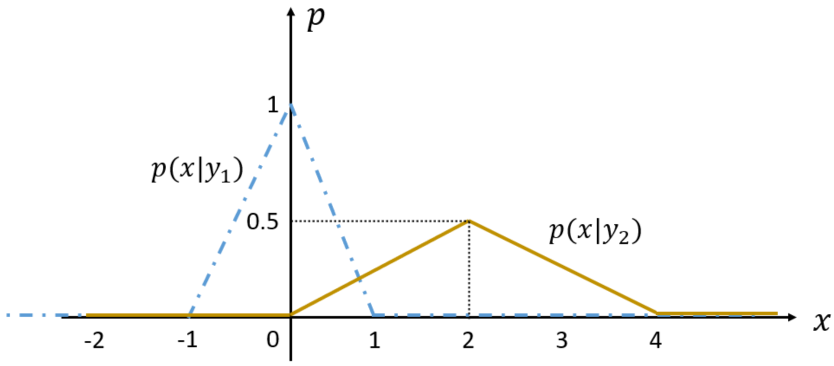
is it a multivariate linear regression problem? If your answer is "No", then what type of linear regression is it? Please explain your answer. *[10 points]*

**Solution:**

- No. *[3 points]*
- It is "polynomial" regression. *[3 points]*
- Reason: it has a single variable and in a polynomial expression. *[4 points]*

11. Two class conditional probability density functions are given in the following figure. The first class $y_1$ is represented by a dashed blue line and the second class $y_2$ is represented with a solid yellow line. Two classes have equal prior probability:

$$P(y_1) = P(y_2) = \frac{1}{2}$$




(a) Use the Bayes' rule (so you can derive the class posterior probabilities), please answer to which class each of the following objects should be assigned? *[6 points]*

- $x_1 = -0.5$.
- $x_2 = 0.5$.
- $x_3 = 3$.

(b) What is the decision boundary of the Bayes classifier with such input data distribution? *[4 points]*

**Solution:**

- (a) Applying Bayes rule:

$$P(y_i|\mathbf{x}) = \frac{p(\mathbf{x}|y_i)P(y_i)}{p(\mathbf{x})}$$

Because $P(y_1) = P(y_2)$, you only need to consider the inequality between $p(\mathbf{x}|y_1)$ and $p(\mathbf{x}|y_2)$ for classifying a data point $\mathbf{x}$. In this question, $x$ is a scalar. For $x_1 = -0.5$, it is assigned to class $y_1$ *[2 points]*; For $x_2 = 0.5$, it is assigned to class $y_1$ *[2 points]*; For $x_3 = 3$, it is assigned to class $y_2$ *[2 points]*.

- (b) Applying the Bayes rule and the equal class prior as mentioned above, We want to derive the value of $x$ where

$$p(x|y_1) = p(x|y_2)$$

Solving this equation you will get:

$$x = 0.8$$

This is the decision boundary of the classifier. *[4 points]*

12. You want to train an SVM model on your point cloud dataset for the task of object classification.

(a) Describe the high-level objective of the SVM classifier (i.e., you can use figures and mathematical formulations to illustrate the objective.) *[6 points]*

(b) Describe why you need a train-test split when running your machine learning experiment, and how you will split the dataset. *[4 points]*

**Solution:**

- (a) Correct answer must elaborate on that SVM is designed for finding the max-margin decision boundary. *[6 points]*
- (b) Correct answer must point out that training and testing on the same set gives an optimistic estimate of the classifier performance. *[4 points]*

13. Consider a dataset of color images of Houses and Offices. You are tasked with building a neural network to classify these images. Check below a Convolutional Neural Network (CNN) architecture using PyTorch to accomplish this task. For this task, you will have a dataset that consists of in total 25,000 labeled color images. Each image is $144 \times 144$ pixels in size.

   (a) What should be the value of *in_channels* in the constructor of the CNN class? (instead of "?")

   (b) What should be the value of $fc1$'s input size? (instead of "??") What the aim of the view function and what should be second argument? (instead of "???")

   (c) Based on the architecture of the CNN and goal of the task, what should be the output activation function? (instead of '????")

   (d) What should be the shape of the output tensor? (print statement in the main function)

   (e) What is the number of the trainable parameters in the conv3, pool3?

```python
# Import the necessary packages
import torch
import torch.nn as nn

class CNN(nn.Module):
        def __init__(self, in_channels=?):
                super(CNN, self).__init__()
                self.conv1 = nn.Conv2d(in_channels=in_channels, out_channels=32,
                        kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
                self.pool1 = nn.MaxPool2d(kernel_size=(2, 2), stride=(2, 2))
                self.conv2 = nn.Conv2d(in_channels = 32, out_channels = 64,
                        kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
                self.pool2 = nn.MaxPool2d(kernel_size=(2, 2), stride=(2, 2))
                self.conv3 = nn.Conv2d(in_channels = 64, out_channels= 128,
                        kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
                self.pool3 = nn.MaxPool2d(kernel_size=(2, 2), stride=(2, 2))
                self.fc1 = nn.Linear(128 * ?? * ??, 512)
                self.fc2 = nn.Linear(512, 1)
                self.relu = nn.ReLU()
                self.out_act = ????

        def forward(self, x):
                x = self.conv1(x)
                x = self.relu(x)
                x = self.pool1(x)
                x = self.conv2(x)
                x = self.relu(x)
                x = self.pool2(x)
                x = self.conv3(x)
                x = self.relu(x)
                x = self.pool3(x)
                x = x.view(-1,???)
                x = self.fc1(x)
                x = self.relu(x)
                x = self.fc2(x)
                x = self.out_act(x)
                return x

if __name__ == "__main__":
        model = CNN()
        # Create a tensor of size 10x3x144x144
        x = torch.randn(10, 3, 144, 144)
        # Pass the tensor to the model
        print(model(x).shape)
```

**Solution:**

The solution to this question is intentionally removed. Please do run the code, test, play with the parameters, and make sure you really understand how to answer these and similar questions.