Exam: Artificial Intelligence – Algorithms and Application

Module Exam

Summer 2021 Date: 03.09.2021

Important Information

TECHNISCHE UNIVERSITÄT DARMSTADT

WIRTSCHAFTS INFORMATIK

- Please check your exam copy for completeness.
  It covers **18 pages** (cover sheet included).
- Fill out the cover sheet immediately after receiving the exam.
- Use only the examination paper to solve the tasks. If you do not have enough space, you can receive additional paper during the examination. Additional papers must also be marked with your name and matriculation number.
- Please leave a **correction margin of 3 cm**.
- You have a total of **90 minutes** to complete the exam.
- Except for a **non-programmable calculator, no other aids** are allowed in the exam.

We wish you much success!

Please fill out clearly in block letters.

First Name ..... Last Name ..... Seat No. .....

Matr. No. ..... Course of Study ..... ❑ Master ❑ Diplom

Repeater:
❑ yes ❑ no

Section	Max. Points	Achieved Points
1	32	
2	34	
3	24	
Sum	90	

Exam Review („Klausureinsicht“):

(do not fill out before the review)

I have reviewed the corrected exam:

☐ There are no complaints about the correction.

☐ Complaints about the correction exist (see additional sheet).

Date: ……..

Signature: ……....

First Name..... Last Name..... Matr. No.....

1 Basic Concepts and Algorithms (32 Points)

1.1 Many people have tried to define the concept “artificial intelligence”. The most popular one is from McCarthy. Please give **his definition** of **Artificial Intelligence** that we have discussed in the lecture. (1 P)

1.2 What is the **potential problem** of **his definition**? (1 P)

1.3 Please **briefly explain** the **central statements** of the ***Moravec’s Paradox***.
Please **also provide an example** of your own regarding today’s Artificial Intelligence. (4 P)

1.4 One very popular search problem in Artificial Intelligence is the ***TSP problem***. Please **explain** the **main concept** and **solution objective** behind these kinds of problems. (2 P)

First Name..... Last Name..... Matr. No.....

1.5 Please **provide** the **pseudo code** of ***genetic algorithms***. Please also **explain** the algorithm briefly using an **illustration** of the underlying process. (3 + 3 = 6 P)

1.6 In the lecture you learned about the two concepts of an agent program and agent function. Please **briefly explain** the **difference** between an **agent *program*** and an **agent *function***.

Also, is a ***hand-held calculator*** an **agent** that chooses the **action** of **displaying "8"** when a user has inserted the **percept sequence "4 + 4 ="**? Please **explain**. (2 P)

1.7 What is the **difference** between the steps **"*mutation*"** and **"*crossover*"** in the context of **genetic algorithms**? Please **explain**. (2 P)

1.8 Please **explain** the difference between ***feedback*** and ***feedforward*** **ANNs**. How do the **connections among neurons** in the **same layers** look like in each case? (1 + 1 = 2 P)

1.9 Please **explain** (2 + 2 = 4 P):

- Why is **biased data** a **challenge** for **AI Design**?
- What is the **difference** between **"*Aggregation bias*"** and **"*Measurement bias*"**?

1.10 Please **name two examples** from **real-life AI applications** for the **design recommendation "*G3 - Time services based on context*"** that we discussed in the context of AI design in the lecture. (1 P)

1.11 Please **explain**: Why is the **research** of **Kate Crawford relevant** for **today's real-life applications** of **Artificial Intelligence**? Please also **provide an example**. (1 P)

1.12 Please **explain**:
What is a **"*knowledge base*"** and **"*expert*"** in the context of an **expert system**? (1 + 1 = 2 P)

1.13 What are the **most common *data sources*** in **AI projects** that we discussed in the lecture? Please **name** and **explain** them briefly. (4 P)

First Name..... Last Name..... Matr. No.....

2 Application of Markov Theory (34 Points)

Consider the following **game** where **two** intelligent agents **play against each other**:

> *Both, Agent A and agent B have a start capital of 2 monetary units (MU) each. In each game round, each agent pays 1 MU to participate in the game. The game ends when one of the two agents has no money left to pay the minimum stake. The deposited amount of 2 MU is divided between the two agents depending on the result of a roll of a fair die according to the following rules: Agent A gets the entire stake (the 2 MU) if the roll of the fair die is 6, he gets his stake back (1 MU) if the roll is 5 and he gets nothing if the roll is 1, 2, 3 or 4. The remainder of the joint bet is paid to Agent B.*

2.1 Please **describe** the **course of the game** using a **Markov chain**. In particular, please **determine** the ***state space***, ***transition matrix***, and ***transition graph*** of the Markov chain.
Please **explain** your **modelling decisions**. (8 P)

2.2 Please **calculate** the **probability** with which **agent A** has a **capital** of **3 MU** after **2 rounds** of play. (5 P)

2.3 Please **calculate** the **probability** with which **agent A** has a **capital** of **less** than **3 MU** after **2 rounds** of play. (3 P)

2.4 What is the **probability** of playing **at least 3 rounds**? (5 P)

2.5 On average, **after how many rounds** of play will the **game be over**? (7 P)

2.6 The game **disadvantages *player A***. Please **explain why**.
Also, what do you think a **fair game might look like**? (6 P)

First Name..... Last Name..... Matr. No.....

3 Data Preprocessing with Python (24 Points)

In your current project for a big German sport car company, you plan to investigate unknown car failures. You aim to find correlations between different parts to identify the root cause. During your first analysis, you face the following dataset that you received from the R&D department:

PartID	Height (cm)	Height (inches)	Weight
1	68	26,77	120
2	74	29,13	130
3	76	29,92	141
…	…	…	…

3.1 Firstly, you try to **load the dataset** in your IDE. However, the upload **does not work** and **returns** the following **errors**.

Python Code

```
connection = sqlite3.connect("failures.db")
cursor = connection.cursor()
sql = "SELECT * FROM failure_report"
cursor.execute(sql)

Traceback (most recent call last):
  File "C:\Users\dominik\.spyder-py3\temp.py", line 7, in <module>
  connection = sqlite3.connect("failures.db")
  NameError: name 'sqlite3' is not defined
```

Please **explain** the (a) ***reason for the errors*** and (b) ***how to solve them***. By doing so, please **provide code** in Python that solves this issue. (1 P)

First Name..... Last Name..... Matr. No.....

3.2 In a first step, you plan to **cluster similar parts**. For that purpose you define the following **`distance()` function** based on the **Euclidian distance**:

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}$$

For example, you should be able to use this function on input like:

```
point_1 = np.array((1, 2))
point_2 = np.array((3, 4))
distance(point_1, point_2)
```

Please **implement** this function in **Python**. Write down the function in **code** and **comment** your code to explain it. (4 P)

3.3 Please **compute** the **results** of the following **code** (rounded to three decimal places):

```
sim_12_inches = distance([68, 120], [74, 130])
```

Please **do the same** for the **following similarities**. (6P)

(**Note** that, e.g., `sim_23_inches` means that you should compute the distance from the point with *PartID=2* to the point with *PartID=3* based on their respective *Height (inches)* and *Weight*.)

Similarity	Result
sim_23_inches	
sim_13_inches	
sim_12_cm	
sim_23_cm	
sim_13_cm	

3.4 Compare the results of **part 2** using inches and centimeters. Please **discuss** and **explain** the results. **How** do you **handle such behavior** in data preprocessing ***in general***? (2P)

First Name..... Last Name..... Matr. No.....

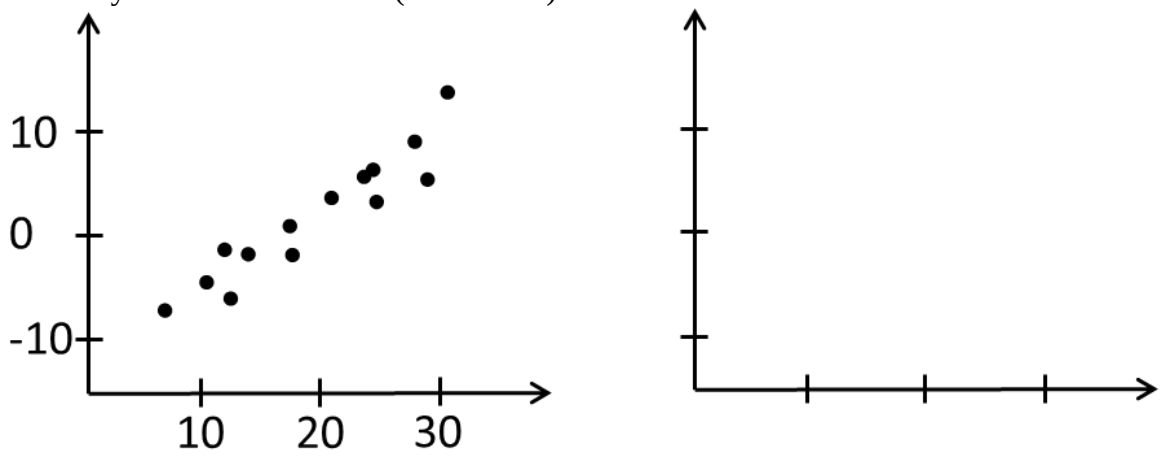
3.5 To handle this problem, you decide to write the following **rescaling function**:

Python Code

```
def scale(data):
    num_rows, num_cols = shape(data)
    means = [mean(get_column(data, j))
                  for j in range(num_cols)]
    stdevs = [standard_deviation(get_column(data, j))
                  for j in range(num_cols)]
    return means, stdevs

def rescale(data):
    means, stdevs = scale(data)
    def rescaled(i,j):
        if stdevs[j] > 0 :
            return(data[i][j] - means[j]) / stdevs[j]
        else:
            return data[i][j]
    num_rows, num_cols = shape(data)
    return make_matrix(num_rows, num_cols, rescaled)
```

Then, you apply the `rescale()` function on the following dataset.
Please **illustrate how** the `rescale()` function **changes** the **data** by **painting a new plot**. For this purpose, please **explain** what the **two functions do** and **how they work** and **how** they **influence the data**. (4 + 4 = 8 P)



First Name..... Last Name..... Matr. No.....

3.6 To rescale data can be necessary, but it is not always useful. Please **provide one example when rescaling <u>is</u> useful** and **one example** for a case where **rescaling <u>is not</u> useful**. In addition, please **explain** how you would **decide** in the **above case**. (3 P)