

Exam: Artificial Intelligence – Algorithms and Application

Module Exam

Winter 2024/2025

Date: 10.04.2025

Important Information



TECHNISCHE
UNIVERSITÄT
DARMSTADT



- Please check your exam copy for completeness.
It covers **19 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 | 35 | |
| 2 | 25 | |
| 3 | 30 | |
| 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:

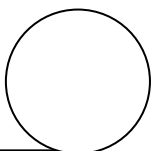
Signature:

First Name Last Name Matr. No.

1 Basic Concepts and Algorithms (35 Points)

1.1 Please provide the **definition of artificial intelligence** that has been defined during the **Dartmouth conference**. (1 P)

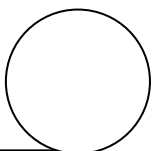
1.2 Please **name** the *three types* into which **machine learning approaches** can be **categorized** based on the **problem type**. (1.5 P)



First Name Last Name Matr. No.

1.3 Please briefly **explain** the *k-fold cross-validation procedure* in the context of a **classification algorithm**. (6 P)

1.4 Is *feature scaling* in **general required** after a *normalization* has been applied?
Please **briefly explain** your **decision**. (2 P)



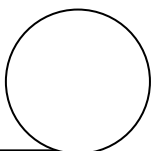
First Name Last Name Matr. No.

1.5 Please briefly **explain**: For *which kind of problems* can you use k-NN algorithms?

What does the *k* stand for? (2 P)

1.6 Please **name** *three other scientific domains of AI* besides **machine learning** and **knowledge reasoning**. (1.5 P)

1.7 Please briefly **explain** the *significance of bias* in a **perceptron model**. Why is it **useful**? (1 P)



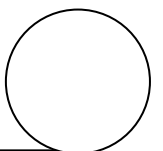
First Name Last Name Matr. No.

1.8 A **delivery drone** needs to **optimize** its **route** in a **crowded city**.

Please **explain**: What *algorithm* would **best suit** this **task**? **Justify** your **answer**. (3 P)

1.9 Please briefly **explain**: What is the **difference** between “*explainability*” and

“*interpretability*” in AI? Please also **provide** an **example** for **each** of the **two**. (4 P)

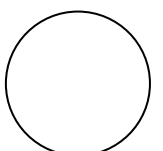


First Name Last Name Matr. No.

1.10 Please **define** the “*no free lunch*” *theorem* in the **context** of **AI**. **Why** is it **important**? (4 P)

1.11 What are *ensemble techniques* in **machine learning**?

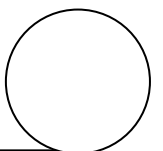
Please provide a **definition** and an **example**. (2 P)



First Name Last Name Matr. No.

1.12 Please briefly **explain**: What is *CRISP-DM*? Please also **name** its **distinct process phases**. (4 P)

1.13 Please briefly **explain**: What is the **difference between** the *Mc Culloch-Pitts Neuron* and the *perceptron*? **How** do they **differ related to learning**? (3 P)



First Name Last Name Matr. No.

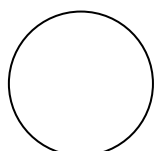
2 Classification and Entropy (25 Points)

A company is developing an *AI-based email spam identifier*. The available **dataset** consists of **email examples** labelled as either **Spam** or **Not Spam**, based on different word occurrences.

Table 1 summarizes the **dataset distribution** for the phrase *“Join me on Whatsapp”* based on a **qualified sample** of the dataset.

| Table 1 | | |
|-------------------------------|--------------|------------------|
| Phrase: “Join me on Whatsapp” | Spam (count) | Not Spam (count) |
| TRUE | 40 | 10 |
| FALSE | 20 | 30 |

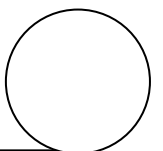
2.1 Please name: What kind of *machine learning problem* is this? (1 P)



First Name Last Name Matr. No.

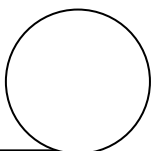
2.2 Please **calculate** the *entropy* of the **dataset** before any **classification** based on the **Spam** and **Not Spam** labels (8 P).

Use the following **formular** for **calculating** *entropy*: $-\sum (p * \log_2 p)$



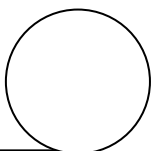
First Name Last Name Matr. No.

2.3 Please **calculate** the *weighted entropy* after **splitting** the **dataset** based on the **presence** of the suspicious **phrase** “*Join me on Whatsapp*”. To compute the **weighted entropy**, **calculate** the **entropy** for **each dataset** and **weight** it **based** on the **number** of **elements** in **each subset**. (6 P)



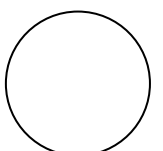
First Name Last Name Matr. No.

2.4 Please **compute** the *information gain* of **splitting** on the **phrase**. Use your values from the previous calculations. Please also **explain**: **Would this be a good split for classification? Why or why not?** (6 P)



First Name Last Name Matr. No.

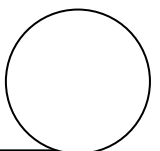
2.5 Please explain: Does **entropy** as a **criterion** to **build** your *decision tree* guarantee the **optimal split** in **each scenario**? **Why** or **why not**? (4 P)



First Name Last Name Matr. No.

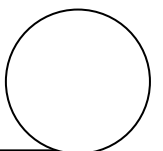
3 Programming with Python (30 Points)

- 3.1 Please write a **Python function** named *load_data* that **accepts** a *filename*, **reads** a *CSV file* into a *Pandas DataFrame*, and **prints** the *first five rows*. (2 P)



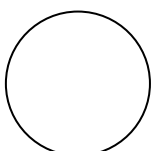
First Name Last Name Matr. No.

- 3.2 Please **write** a **new function** named *handle_missing_values* that does the same as *load_data* but **fills** in **missing values** with the **mean** of their **columns** in the **created DataFrame** before **printing** its *first five rows*. (2 P)



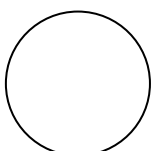
First Name Last Name Matr. No.

3.3 Please **write** a **new function** named *save_clean_data* that does the same as *handle_missing_values* but also **saves** the **cleaned DataFrame** to a **new CSV file** and **adds** a **print statement** when the data is successfully saved. (2 P)



First Name Last Name Matr. No.

3.4 Please **write a short documentation** using **Python documentation** to **explain the *purpose*** and ***parameters*** of your function ***save_clean_data***. (2 P)



First Name Last Name Matr. No.

3.5 Please consider the **following code snippet** that is designed to **predict fraud** in a **car insurance dataset** using a simple **logistic regression** model. The code snippet **contains 6 errors or logical mistakes** that will **result in a runtime error or incorrect results** from your model. Please **identify the 6 errors or logical mistakes** and **explain how to fix them**. (12 P)

```
import pandas as pd
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.preprocessing import StandardScaler

# Load data
data = pd.read_csv('car_insurance.csv')

# Prepare data
features = data[['age_string', 'total_claims']]
labels = data['fraud']

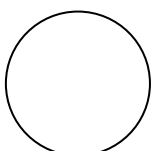
# Split data
X_train, X_test, y_train, y_test = train_test_split(features, labels,
test_size=0.99, random_state=42)

# Scale features
scaler = StandardScaler()
X_train_scaled = X_train
X_test_scaled = scaler.transform(X_train)

# Initialize and train model
model = LogisticRegression()
model.fit(X_train_scaled, y_train)

# Predict and evaluate
predictions = model.predict(X_test_scaled)
accuracy = accuracy_score(y_train, predictions)
print(f' accuracy ')
```

First Name Last Name Matr. No.



First Name Last Name Matr. No.

3.6 Please assume you have a Pandas dataframe that contains the following columns:

`Car_Model`, `Car_Year`, `Failure_Type`, `Repair_Cost`

Please provide **Python code** to **answer** the **following questions** about the dataframe. (10 P)

| Question | Python Code |
|---|-------------|
| How can you extract all records where the Car_Model is 'Panamera' ? | |
| How would you calculate the average Repair_Cost for each Failure_Type ? | |
| How can you sort the DataFrame by the Car_Year in descending order? | |
| How do you find all entries where the Car_Year is before 2015 and the Repair_Cost is greater than \$1000 ? | |
| How can you summarize the total Repair_Cost per car_model ? | |

