Analytics & Big Data

Test Exam

Name: _____

Student number: _____

Additional information:

By starting the exam, you confirm that you are capable of taking it. Withdrawal is not possible once the exam has begun.

By participating, you also confirm that you will complete the exam independently, without outside help and without the use of unauthorized aids. Violations will be treated as breaches of the examination regulations and the Honour Code and may result in serious consequences.

The exam consists of four tasks, all of which must be answered. The maximum total score is 60 points. The allotted time is 60 minutes.

Please use the answer booklet for your responses and complete the required information on the cover sheet before starting. Scratch paper may be used for drafts and notes; however, only answers written in the answer booklet will be graded.

You may complete the tasks in any order. Please ensure that it is clearly indicated which task you are answering.

At the end of the exam, all distributed materials must be returned.

For grading:

Question	**1**	**2**	**3**	**4**	**Total**
Possible points					
Points achieved					

Task 1 (15 points)

You are given the following dataset:

Customer_ID	Region	Jan_Sales	Feb_Sales	Mar_Sales
C1	EU	120	140	160
C2	US	90	110	130

a) Explain why this dataset is not in a tidy format.

b) Propose a tidy structure by sketching column names and example rows.

c) Describe the transformation steps required to convert the dataset into tidy format.

Task 2 (15 points)

A restaurant chain wants to analyze its sales performance. Managers are interested in questions such as:

- "What is the total sales amount per restaurant per month?"
- "Which menu items sell best during weekends?"
- "How does average sales amount per customer vary across different restaurants?"
- "What is the sales distribution by menu category (e.g., drinks, main courses, desserts) over time?"

a) Identify facts and dimensions for this scenario.

b) Explain what distinguishes dimensions from facts in the context of data warehouses.

c) Describe the role of granularity in this scenario and give one example.

Task 3 (10 points)

In many machine learning tasks, input data can be transformed or preprocessed before being used by a model.

a) What is feature engineering, and why is it important in machine learning?

b) How important is manual feature engineering when using neural networks? Discuss whether it is more or less important compared to other approaches, and explain why.

Task 4 (20 points)

Study the *Analysis notebook* on the following page.

a) Identify three problematic choices. For each issue, briefly describe the problem, explain how it may affect generalizability, and suggest a specific fix.

b) Discuss the choice of the machine learning model. In your answer, describe its key strengths and limitations, assess its suitability for the prediction task, and suggest one or two alternative models that may be considered.

Analysis notebook

A financial services company is developing a machine learning model to predict whether a transaction is fraudulent.

```
import pandas as pd

df = pd.read_csv("transactions.csv")
df.head()
```

tid	amount	country	time_of_day	prev_transactions	t_flag	account_age_days	device_type	is_fraud
1001	250	DE	evening	5	1	365	mobile	1
1002	40	US	morning	2	0	120	desktop	0
1003	900	CN	night	12	1	30	mobile	1
1004	75	DE	afternoon	3	0	200	tablet	0
1005	600	US	night	8	1	15	mobile	1

The dataset contains the following variables:

Variable name	Description
`tid`	Unique identifier for each transaction
`amount`	Transaction amount in EUR
`country`	Country where the transaction originated
`time_of_day`	Time category (morning, afternoon, evening, night)
`prev_transactions`	Number of previous transactions by the same user
`t_flag`	Whether an IT security employee manually flagged the transaction (0/1)
`account_age_days`	Number of days since the account was created
`device_type`	Type of device used (mobile, desktop, tablet)
`is_fraud`	Target variable (1 = fraud, 0 = not fraud)

The following Python code is used to train and evaluate the model:

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

X = df.drop(columns=["is_fraud"])
y = df["is_fraud"]

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2)

model = LogisticRegression()
model.fit(X_train, y_train)

y_pred = model.predict(X_train)
print("Accuracy:", accuracy_score(y_train, y_pred))
```