

Section 1: Explain the problem



MYP Design Cycle – Criterion A

Inquiry & Analysis

Project 2 – Intelligent Systems & Machine Learning

Course: Solving Complex Problems Through Programming

Time for section A : 2 Weeks

Important:

You are not coding yet. You are **designing** a machine learning system.

Your solution must:

- Belong to ONE of the five pathways
- Use Python
- Include a machine learning component written from scratch
- Include structured data
- Involve non-trivial algorithmic processing

You may NOT use:

scikit-learn

pandas

numpy

matplotlib

Any ML framework

You must implement your own:

- Feature representation
- Model structure
- Prediction function
- Evaluation logic

Your system must clearly separate:

Input → Feature Engineering → Model → Output → Evaluation

Sections

There are four sections to this part of inquiry and analysis. Each section has multiple tasks.

There is a rubric checklist at the end of this document (on the last document tab) to help you write a high-level inquiry and analysis. There are also command term definitions to help you.

Section 1, Task 1

Task 1 — Identify Your Pathway

There are two text sections for task 1

1. **State** your pathway
2. **Describe** your chosen problem. Each problem must be approved by your awesome teacher.

Please erase this text and type your answers to the above two instructions in this text box.

Hint:

Bad:

“This predicts something.”

Better:

“This predicts whether a student will struggle in a course based on attendance, homework completion rate, and past assessment performance. The variables interact and are not linearly independent.”

Then answer:

1. **State** the real-world decision your system helps make.
2. **State** if this a classification problem (category output) or regression problem (numeric output).
3. **Outline** why this problem is non-trivial.
4. **Explain** what makes the decision complex.

Please erase this text and type your answers to the four above instructions in this text box.

Section 1, Task 2

Task 2 — Describe the hypothetical target user

There is one text section for task 2

Answer clearly (make an educated, best-guess):

1. **State** who might use this system.
2. **Outline** what decisions they currently make without data support.
3. **State** any errors or inefficiencies occur now.
4. **Outline** measurable improvements your system may provide.

Please erase this text and type your answers to the four above instructions in this text box.

Hint: Be concrete and data-driven.

Example:

Instead of “helps coaches,” write:

“Helps coaches predict player fatigue based on training load, sleep hours, and match intensity, reducing injury risk.”

Section 1, Task 3

Task 3 — Justify the Need for a Machine Learning System

There is one text section for task 3

1. **Describe** why this problem can't be solved with simple IF statements.
2. **Explain** why probability or weighted decision-making is appropriate.

Please erase this text and type your answers to the two above instructions in this text box.

Hint:

Use the terms:

Feature

Weight

Bias

Training data

Prediction

Model

Section 2: Research and data modeling

SECTION 2 — RESEARCH AND DATA MODELING

There are two tasks associated with this section. You can find them on the document tabs.

Section 2, Task 1

Task 1 — Define the Data Structure

There are two text sections for section 2, task 1.

Your system requires structured data.

Task 1:

1. **List** the features (minimum 5, maximum 15).
2. **State** the target variable.
3. **State** the data type of each feature (int, float, categorical, boolean).
4. **Describe** how categorical data be encoded numerically.

Please erase this text and type your answers to the four above instructions in this text box.

Create a sample dataset of at least 10 rows.

Represent it as a Python-style list of dictionaries or list of lists. It's ok to paste this in from Visual Studio Code.

Example format:

```
data = [  
{"sleep": 7, "sports": 2, "prev_absences": 1, "risk": 0},  
...  
]
```

Task 2:

1. **Explain** why your features are relevant predictors.
2. **Justify** relationships logically.

Please erase this text and type your answers to the two above instructions in this text box.

Section 2, Task 2

Task 2 — Mathematical Model Design

There is one text section for section 2, task 2.

You must choose ONE model type:

Linear regression

Binary perceptron

Logistic-style classifier (sigmoid implemented manually)

Weighted scoring model with normalization

Task 2:

1. **Define** mathematically:

$$z = w_1x_1 + w_2x_2 + \dots + b$$

2. **Outline** z .
3. **Outline** what the weights represent.
4. **Outline** what does the bias represent.
5. **Outline** what the output transformation is.
6. **Explain** why your chosen model fits your problem.

Please erase this text and type your answers to the six above instructions in this text box.

--

Section 3: Algorithmic design

SECTION 3 — ALGORITHMIC DESIGN

Section 3, Task 1

Task 1 — Write the Prediction Function (Pseudocode)

There is one text section for section 3, task 1.

Task 1:

1. **Construct** Python-style pseudocode for a function named `predict()`:

It must:

`predict(features, weights, bias)`

Compute weighted sum

Apply activation

Return output

Not use any libraries (except random and maybe math and maybe pandas)

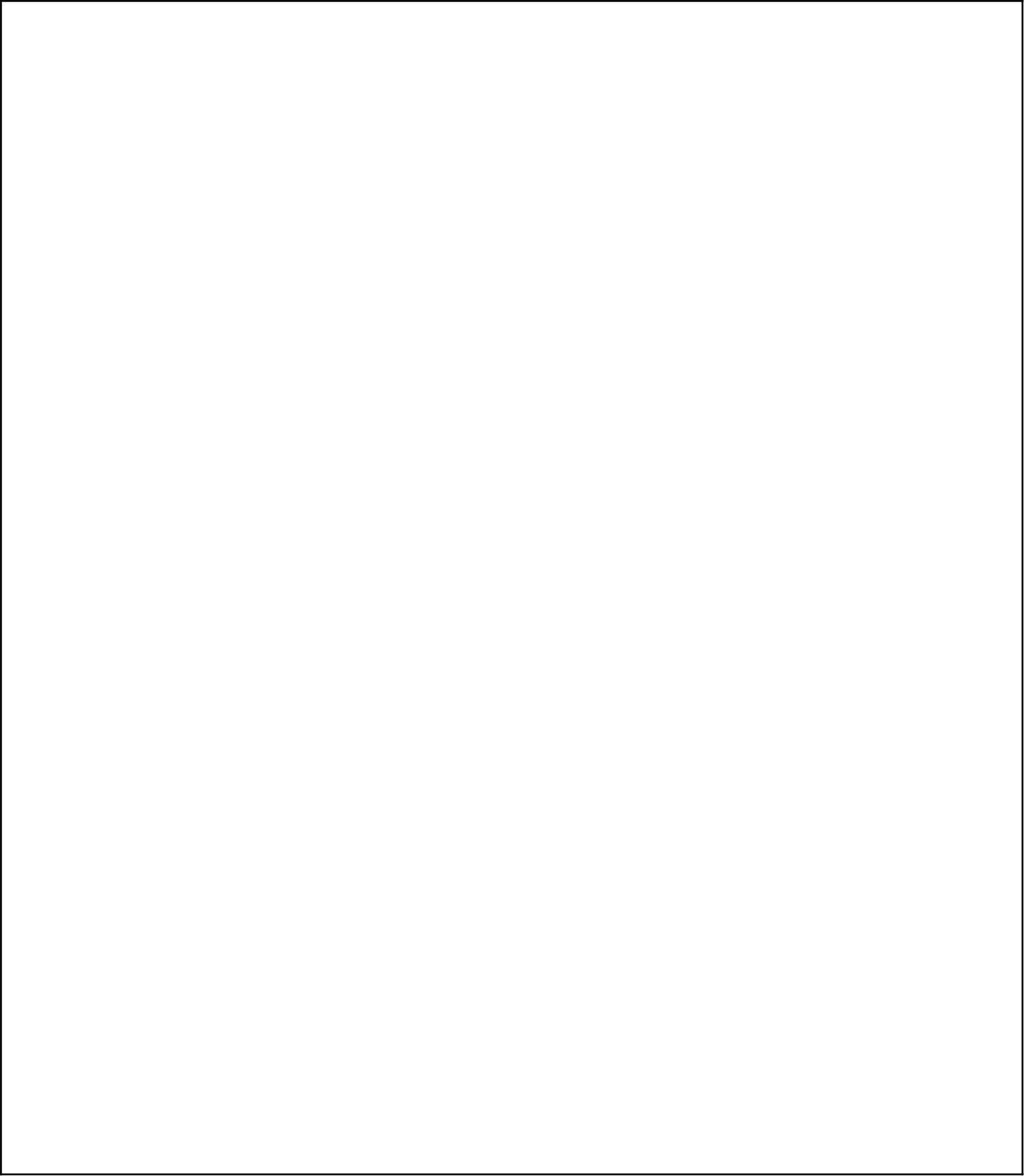
Be explicit about loops and indexing.

Example structure:

```
def predict(x, weights, bias):  
    z = 0  
    for i in range(len(x)):  
        z += x[i] * weights[i]  
    z += bias  
    return activation(z)
```

Hint: Do not copy this. Write your own version based on your model.

Please erase this text and type your answers to the one above instruction in this text box.



Section 3, Task 2

Task 2 — Evaluation Strategy

There is one text section for section 3, task 2.

How will you evaluate accuracy?

For classification:

- Accuracy percentage
- Confusion matrix
- Precision/recall

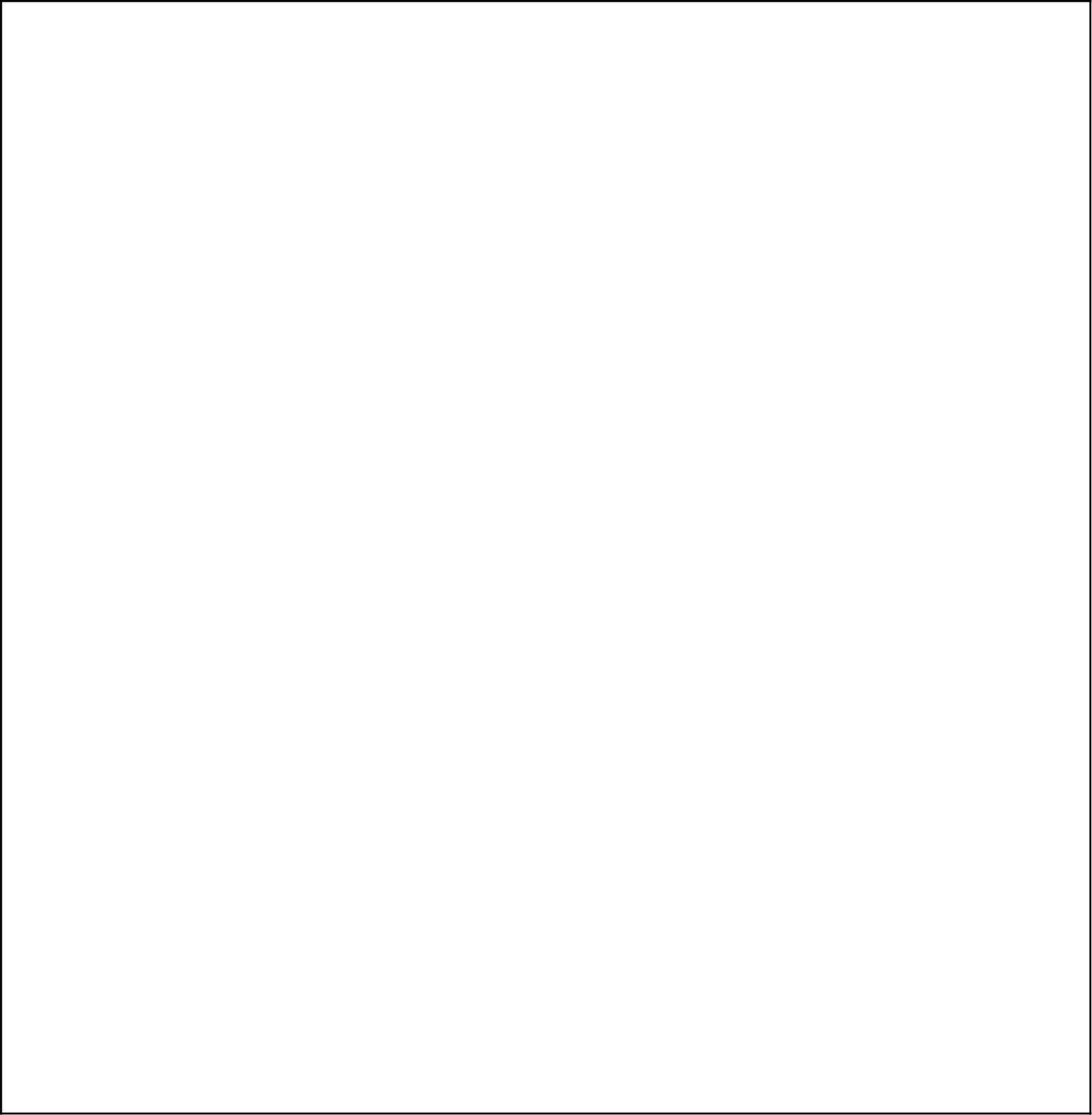
For regression:

- Mean absolute error
- Mean squared error

Task 2:

1. **Define** mathematically how your evaluation metric works.
2. **Explain** why your metric matches your problem.

Please erase this text and type your answers to the two above instructions in this text box.



Section 4 — System Requirements

Section 4 — System Requirements

Task 9 — Success Criteria (Measurable)

Write at least 8 measurable success criteria.

Each must be testable.

Bad:

“The model works well.”

Better:

“The model predicts outcomes with at least 70% accuracy on unseen test data.”

Include criteria for:

Correct mathematical implementation

Correct feature encoding

No external ML libraries

Evaluation metric functioning

Clear user interface

Logical data storage

Readable code structure

Documented limitations

End Section 4

Section 4, Task 1

Task 1 — Technical Requirements

There are two text sections for section 4, task 1.

1. **State** if your program requires persistent storage.
2. **State** if your program requires multiple pages.
3. **State** if your program requires user-uploaded data.
4. **State** if your program requires admin controls.
5. **State** if your program requires manual data entry.

Please erase this text and type your answers to the five above instructions in this text box.

1. **Construct** a simple architecture diagram (hand drawn is acceptable, but I prefer Lucidchart):

Your chart must have: User → Form → Python Processing → Model → Result → Storage

Input

Processing

Output

Persistent Storage

Data Structures

Please erase this text and construct a diagram according to the above instructions.

--

Rubric Checklist & command terms

RUBRIC CHECKLIST (Student Self-Assessment)

To reach 7–8 in Criterion A, you must:

- ☐ Clearly explain and justify a complex problem
- ☐ Demonstrate why machine learning is necessary
- ☐ Construct a detailed and logically structured dataset
- ☐ Define a mathematically valid model
- ☐ Provide complete algorithmic pseudocode
- ☐ Justify evaluation metrics
- ☐ Define measurable success criteria
- ☐ Use precise technical vocabulary correctly

Lower markbands typically show:

- **Vague** problem definitions
- Features chosen **without** justification
- No mathematical explanation
- Missing evaluation method
- No explanation of why rules alone fail

Command terms:

• **Construct**

Create or develop something in a structured way. This may involve building code, diagrams, datasets, or logical procedures.

• **Define**

Give the precise meaning of a term, concept, or expression.

• **Demonstrate**

Show evidence of understanding through clear reasoning, accurate application, or correct implementation.

• **Describe**

Provide a detailed account of something. Include relevant characteristics, features, or processes.

• **Explain**

Give reasons or make relationships clear. Show how or why something works using logical reasoning.

- **Identify**

Select or name something clearly and specifically.

- **Include**

Ensure specific required elements are present.

- **Justify**

Give valid reasons or evidence to support a decision or conclusion. Demonstrate why your choice is appropriate.

- **List**

Provide a sequence of brief items without explanation.

- **Outline**

Give a brief summary of the main points. Focus on structure and key ideas rather than deep explanation.

- **State**

Give a clear, concise answer without explanation. A short, direct response is sufficient.

- **Use**

Apply a concept, method, or term correctly in context.