UNSW COMPUTING

UNSW DATA9001
Assignment 3

Due: Friday 8nd of August 18:00 Sydney time

Intro

This assignment builds on and extends skills and knowledge acquired in the Computer Science part of DATA9001 (week 8 and 9). Please read each question carefully, answer all of them, report results in a way that is easy to follow. Email any questions or comments to Hao Xue (hao.xue1@unsw.edu.au)

Note:

- Total value: 15 marks (15% of the final mark). Submission is due on **Friday of Week 10 (8 August), 6:00 pm**.
- Please submit your answers to assignment questions in PDF format using the following name: **`A3-z1234567-FirstName-Surname-Report.pdf`**
- Please also submit the Python code that you developed as part of the assignment. The PDF report and code should be submitted in **`A3-z1234567-FirstName-LastName.zip`** file.
- The submission link is on the Moodle site under Assessments Hub.
- **Please sign the Plagiarism Declaration at the bottom of this page. Assignments without signed Plagiarism Declaration will not be accepted.**
- Format: 14 pages max. Do not include a separate title page. At least 11-point font should be used, with adequate margins for comments. Any extra pages will not be marked. Please use clear section titles such as Problem 1, Problem 2, and Problem 3 in the submitted report PDF.
- Please make sure you place your full name and zid on the header of all pages.
- Late assignments will not be accepted unless extension is Special Consideration is granted by UNSW.

Plagiarism Declaration:

I declare that this assessment item is my own work, except where acknowledged, and has not been submitted for academic credit elsewhere. I acknowledge that the assessor of this item may, for the purpose of assessing this item reproduce this assessment item and provide a copy to another member of the University; and/or communicate a copy of this assessment item to a plagiarism checking service (which may then retain a copy of the assessment item on its database for the purpose of future plagiarism checking). I certify that I have read and understood the University Rules in respect of Student Academic Misconduct.

Name: _____ StudentNo:_____

Signature:_____ Date:_____

Problem1 – Theoretical [4 marks]

A two-class model was trained and then tested with a dataset of 100 instances. The test set contained 60 instances in positive class P, and 40 instances in negative class N. As a result of testing, the following counts were obtained:

- 45 instances of P were classified correctly,
- 15 instances of P were classified into N,
- 25 instances of N were classified correctly,
- 15 instances of N were classified into P.

i) [1 mark] Construct contingency table (also called confusion matrix)

ii) [1.5 marks] Calculate the following macro metrics: (show your calculations)

a. Precision

b. Recall

c. F1

iii) [1.5 marks] Calculate the following micro metrices:

a. Precision

b. Recall

c. F1

Note: For this problem, Problem1 -Theoretical, you can write your solution on a paper and scan your solutions.

Problem2 – Practical Part 1 [5 marks]

Background:

A large food-processing company wants to **identify dry-bean cultivars** from computer-vision measurements so that each batch is routed to the correct milling process.
You will evaluate a **k-Nearest Neighbour (KNN)** classifier and investigate how a different hyper-parameter changes performance.

Dataset Description:

Seven different types of dry beans were used in this research, taking into account the features such as form, shape, type, and structure by the market situation. A computer vision system was developed to distinguish seven different registered varieties of dry beans with similar features in order to obtain uniform seed classification.
You are provided with many samples. Each sample includes the following features:

Variable Name	Role	Type	Description
Area (A)	Feature	Numerical	The area of a bean zone and the number of pixels within its boundaries.
Perimeter (P)	Feature	Numerical	Bean circumference is defined as the length of its border.
Major axis length (L)	Feature	Numerical	The distance between the ends of the longest line that can be drawn from a bean.
Minor axis length (l)	Feature	Numerical	The longest line that can be drawn from the bean while standing perpendicular to the main axis.
Aspect ratio (K)	Feature	Numerical	Defines the relationship between L and l.
Eccentricity (Ec)	Feature	Numerical	Eccentricity of the ellipse having the same moments as the region.
Convex area (C)	Feature	Numerical	Number of pixels in the smallest convex polygon that can contain the area of a bean seed.
Equivalent diameter (Ed)	Feature	Numerical	The diameter of a circle having the same area as a bean seed area.
Extent (Ex)	Feature	Numerical	The ratio of the pixels in the bounding box to the bean area.
Solidity (S)	Feature	Numerical	Also known as convexity. The ratio of the pixels in the convex shell to those found in beans.
Roundness (R)	Feature	Numerical	Calculated with the following formula: (4piA)/(P^2)
Compactness (CO)	Feature	Numerical	Measures the roundness of an object: Ed/L

ShapeFactor1-4(SF1-4)	Feature	Numerical	Shape Factors
Class	Target	Categorical	Seker, Barbunya, Bombay, Cali, Dermosan, Horoz and Sira

Task Description:

You are required to achieve following steps:

1. **[1 mark] Data Splitting**: Divide the dataset into a training set and a test set based on a proper ratio.
2. **[1 mark] Data Preprocessing**: Perform necessary data preprocessing steps. Normalized or standardized the data if required.
3. **[1 mark] Model Implementation**: Implement the k-Nearest Neighbour (KNN) classifier using the provided features to predict the classes of beans.
4. **[2 marks] Model Evaluation**: Evaluate the model on the test set and report the classification accuracy, confusion matrix, precision, recall, and F1-score. Change another parameter that you like, compare the new results with old results and analyze them.

Default Parameters:

- k-Nearest Neighbour (KNN): n_neighbors = (zid) mod 5 + 2
  example: If your zid is z5114514, then your default n_neighbors = 6

Provided for this Assignment:

- Dry_Bean_Dataset.csv: The dataset of bean samples in CSV format.

Deliverables:

- Code Implementation: Submit a Python script or Jupyter Notebook containing your implementation.
- Report: A brief report summarizing your approach, the results of your model, and any observations or conclusions.

Problem3 – Practical Part 2 [6 marks]

Background:

This assignment is inspired by a real-life scenario. A Portuguese retail bank wants to discover which customer attributes and marketing-contact details best predict whether a client will subscribe to a **term-deposit** product.
For this assignment, you will be given a dataset of bank records. Each evaluation consists of several features. You are required to evaluate either the Naive Bayes **or** Decision Tree method to classify has the client subscribed a term deposit? (binary: "yes","no").

Dataset Description:

You are provided with many car samples. Each sample includes the following features:

Variable Name	Role	Type	Description	Value
Age	Feature	Numeric	Client age	Non Negative Numbers
Marital	Feature	Categorical	Marital Status	"married","divorced","single"
Education	Feature	Categorical	Education Status	"unknown","secondary","primary","tertiary"
Default	Feature	Categorical	Has credit in default?	"yes","no"
Balance	Feature	Numerical	Average yearly balance, in euros	Non Negative Numbers
Housing	Feature	Categorical	has housing loan?	"yes","no"
Contact	Feature	Categorical	contact communication type	"unknown","telephone","cellular"
Poutcome	Feature	Categorical	outcome of the previous marketing campaign	"unknown","other","failure","success"
y	Target	Categorical	has the client subscribed a term deposit?	"yes","no"

Task Description:

You are required to achieve following steps:

1. **[1 mark] Data Splitting**: Divide the dataset into a training set and a test set based on a proper ratio.

2. **[1 mark] Data Preprocessing**: Perform necessary data preprocessing steps. It's ok to transfer the numerical variable to reasonable categorical type. Normalized or standardized the data if required.
3. **[1 mark] Model Selection and parameter tuning**: Choose either Naive Bayes or Decision Tree for your model. Select the parameters that you believe are optimal for your chosen model.
4. **[1 mark] Model Training**: Train the model on the training set.
5. **[2 marks] Model Evaluation**: Evaluate the model on the test set and report the classification accuracy, confusion matrix, precision, recall, and F1-score.

Provided for this Assignment:

- bank.csv: The dataset of car evaluations in CSV format.

Deliverables:

- Code Implementation: Submit a Python script or Jupyter Notebook containing your implementation.
- Report: A brief report summarizing your approach, the results of your model, how you handled the imbalanced data, and any observations or conclusions.