Cardiff School of Computer Science and Informatics

Coursework Assessment Pro-forma

Module Code: CM2105
Module Title: Data Processing and Visualisation
Lecturer: Dr Hantao Liu
Assessment Title: Coursework
Assessment Number: 1
Date Set: Tuesday 1 November 2022
Submission Date and Time: by Monday 05 December 2022 at 9:30am
Feedback return date: Monday 09 January 2023

Extenuating Circumstances submission deadline will be 2 weeks after the submission date above
Extenuating Circumstances marks and feedback return will be 2 weeks after the feedback return date above

This assignment is worth 100% of the total marks available for this module. If coursework is submitted late (and where there are no extenuating circumstances):

1 If the assessment is submitted no later than 24 hours after the deadline, the mark for the assessment will be capped at the minimum pass mark;
2 If the assessment is submitted more than 24 hours after the deadline, a mark of 0 will be given for the assessment.

Extensions to the coursework submission date can ***only*** be requested using the Extenuating Circumstances procedure. Only students with approved extenuating circumstances may use the extenuating circumstances submission deadline. Any coursework submitted after the initial submission deadline without *approved* extenuating circumstances will be treated as late.

More information on the extenuating circumstances procedure can be found on the Intranet: https://intranet.cardiff.ac.uk/students/study/exams-and-assessment/extenuating-circumstances

By submitting this assignment you are accepting the terms of the following declaration:

I hereby declare that my submission (or my contribution to it in the case of group submissions) is all my own work, that it has not previously been submitted for assessment and that I have not knowingly allowed it to be copied by another student. I understand that deceiving or attempting to deceive examiners by passing off the work of another writer, as one's own is plagiarism. I also understand that plagiarising another's work or knowingly allowing another student to plagiarise from my work is against the University regulations and that doing so will result in loss of marks and possible disciplinary proceedings[1].

[1] https://intranet.cardiff.ac.uk/students/study/exams-and-assessment/academic-integrity/cheating-and-academic-misconduct

Assignment

For this coursework you must write a Python program (your code should be contained within a Jupyter Notebook (.ipynb) file) that analyses and visualises the given data.

Q1. Part1:

The QS World University Rankings are a ranking of the world's top universities published annually since 2004. Along with Academic Ranking of World Universities and THE World University Rankings, the QS World University Rankings is widely recognised and cited as one of the three main world university rankings. Universities are evaluated according to the following six metrics:

– **Academic Reputation (AR)**
– **Employer Reputation (ER)**
– **Faculty/Student Ratio (FR)**
– **Citations per Faculty (CF)**
– **International Faculty (IF)**
– **International Student Ratio (IR)**

The Microsoft Excel file named "**2018-QS-World-University-Rankings-Top200.xlsx**" (available on Learning Central and see below illustration of a sample of the file) contains the data of the top 200 universities in the world.

Table 1: Illustration of a sample of the data file.

Institution Name	Location	Rank	Academic Reputation	Employer Reputation	Faculty Student	Citations per Faculty	International Faculty	International Students	Overall Score
MASSACHUSETTS INSTITUTE OF TECHNOLOGY (MIT)	United States	1	100	100	100	99.9	100	96.1	100
STANFORD UNIVERSITY	United States	2	100	100	100	99.4	99.6	72.7	98.7
HARVARD UNIVERSITY	United States	3	100	100	98.3	99.9	96.5	75.2	98.4
CALIFORNIA INSTITUTE OF TECHNOLOGY (CALTECH)	United States	4	99.5	85.4	100	100	93.4	89.2	97.7
UNIVERSITY OF CAMBRIDGE	United Kingdom	5	100	100	100	78.3	97.4	97.7	95.6
…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…

1) [**cell1 – 3 marks**] **Download the file "CW_your student number 2022.ipynb" from Learning Central**, and upload it to your Jupyter Notebook. Change the title of the file using your student number (e.g., CW_1234567.ipynb). Write code to read the given data (i.e., "2018-QS-World-University-Rankings-Top200.xlsx") into your programme.
   – **Display a tabular data structure** in your programme, showing the **top ten institutions in the world** (observe the index values of the data file, and the first row must represent the column names as illustrated in Table 1).

   Write code to **create ONE horizontal bar graph** of the QS-UK-rankings that illustrates all UK universities in the top 200 in the QS-World-University-Rankings. The following information is required to be included and accessible in the visualisation: the "National Rank" (i.e., 1, 2, 3…); the "International Rank" (i.e., 5, 6, 7…); the "Institution Name"; the "Overall Score"; and the distinction between world's Top50 and other institutions in the UK. Add appropriate title, horizontal axis label and vertical axis label to the bar graph.
   – **Display the visualisation** in your programme. Note: display a single plot ONLY.

2) [**cell2 – 3 marks**] Write code to analyse the data contained in the variable called "Academic Reputation" in the given data (i.e., "2018-QS-World-University-Rankings-Top200.xlsx"). Write code to **create ONE vertical bar graph** that illustrates the **mean measure** of "Academic Reputation" for each "Location" (note "Location" data is available in the given data file, and **you should exclude any "Location" containing less than two institutions**). Add appropriate title, horizontal axis label and vertical axis label to the bar graph. The following information is required to be included and accessible in the visualisation: show bars (note each bar represents a unique "Location") in the graph in descending order (i.e., from left to right: highest to lowest bars); add error bars to the bar graph, showing the 95% confidence interval; a bar should be highlighted/indicated if its mean "Academic Reputation" is statistically significantly higher than that of the adjacent bar (i.e., the bar on the right).
   – **Display the visualisation** in your programme. Note: display a single plot ONLY.

– **Print ONE short paragraph** (up to 4 sentences) that summarises how the data analysis is performed to reveal the statistical difference in the above context (i.e., the difference between two adjacent bars). The following information is required in the description: the name(s) of chosen test(s) and appropriate interpretation of test results.

3) [**cell3 – 4 marks**] Given the following scenario "the data contained in the variables called "Employer Reputation (ER)", "International Faculty (IF)", and "International Student Ratio (IR)" for all UK institutions are missing", estimate the "Overall Score (OS)" for **UK institutions** using linear regression. First, use the data of the **non-UK institutions** contained in the given data (i.e., "2018-QS-World-University-Rankings-Top200.xlsx") to build **as many suitable linear regression models as possible**. Each model must contain **at least two predictor variables** (note the target variable must be "Overall Score (OS)"). Second, once the models are built, apply these models to the data (with missing variables as mentioned above) of the UK institutions to estimate their "Overall Score (OS)", and evaluate the performance of these models using **Mean Squared Error (MSE)** – average of the squares of the errors. Note, MSE measures the error of prediction between the true and predicted "Overall Score (OS)".

– **Print all suitable linear regression models** (i.e., **linear equations**) in the programme. Note, print equations ONLY, and **use above-mentioned acronyms** for variable names.
– **Construct and display a tabular data structure** (i.e., a DataFrame) to illustrate the performance of all models on the data of UK institutions. The following information is required to be included in the tabular data structure: the model equation; and the MSE.
– **Print ONE sentence**, stating your conclusion and justification on which model should be used for predicting the "Overall Score" of UK institutions.
[Note, quantitative results in the cell output should be shown as rounded values with TWO decimal places.]

Q1. Part2:

You are given a set of grayscale images, i.e., "m1.png" – "m10.png" in **data file "model.zip"**. Shown below are examples of images, i.e., "m1.png" and "m5.png". Each image is an array of integers; and the value (i.e., in the range [0, 255]) of each integer is the intensity at a pixel location. [**Note: use code "*img=imread('m1.png')*255*" to read an image into your programme**] For each image, a **true overall image quality score** is given (see data file "Q_scores.xlsx").

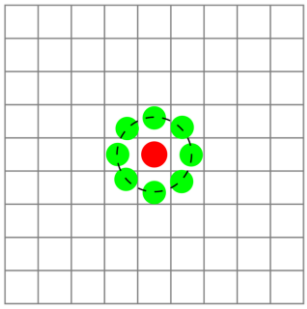
m1.png

m5.png

4) [**cell4 – 3 marks**] Based on "**m1.png**", write code to create a histogram for the local binary patterns (LBP), using the following process:

- Exam a local cell of 3×3 pixels, which is centred on the current pixel in question (as shown below in the form of a red dot).
- For each pixel in a cell, compare the pixel to each of its 8 neighbours (on its top-left, top-middle, top-right, etc.). Follow the pixels along a circle clockwise (note, use the top-left as the starting point) as shown below in the form of a sequence of green dots.
- As shown below, where the centre pixel's value is greater than the neighbour's value, write "0". Otherwise, write "1". This gives an 8-digit binary number, which is then converted to decimal.
- All pixels (excluding the borders) of the image in question must each produces an LBP number. Construct a histogram of the frequency of each "number" occurring (i.e., each combination of which pixels are smaller and which are greater than the centre).

– **Display the resulting histogram**. The following information is required: there are 256 different possible local binary patterns (LBP), and so the histogram will graphically display 256 numbers (i.e., 256 bins) showing the distribution of LBP amongst those values. Add appropriate title, horizontal axis label and vertical axis label to the graph.



40	70	40
80	30	60
10	50	10

1	1	1
1		1
0	1	0

Pattern= 11110101

LBP=$1x2^0+ 1x2^1+ 1x2^2+ 1x2^3+ 0x2^4+ 1x2^5+ 0x2^6+ 1x2^7=245$

5) [**cell5 – 4 marks**] The histogram of LBP can be seen as a 256-dimensional feature vector that represents the characteristics of an image. Based on the LBP histogram, write code to build an automated image quality assessor (IQA), where the **input** contains **two images** (one is the reference image, i.e., "**m1.png**" (note the **reference** should **always be "m1.png"** which represents perfect quality); and one is the test image, i.e., any image contained in the data file "model.zip"); and the **output** should be **a single value** that represents the **estimated quality of the test image**. For example, if the test image is "m1.png", the output should be "0" meaning the test image is estimated to be of perfect quality. The IQA algorithm should involve the following key components: (1) the calculation and normalisation of the LBP histogram; and (2) the calculation of a similarity measure (i.e., a numerical similarity score) between the reference-image LBP histogram and test-image LBP histogram, based on Euclidean distance.

After the IQA algorithm is formulated and programmed, correlation analysis should be performed to evaluate how well the algorithm can predict the real image quality. This is to write code to analyse the **linear correlation** between the "**algorithm generated image quality**" and "**true overall image quality** (available via data file "Q_scores.xlsx)" for the entire image dataset (available via data file "model.zip"). [Note: quantitative results should be shown as rounded values with TWO decimal places.]
– **Display a scatter plot** in your programme. The following information is required to be included and accessible in the visualisation: the horizontal axis represents the variable "true overall image quality"; the vertical axis represents the variable "algorithm generated image quality"; use a legend to indicate the Pearson linear correlation coefficient; add appropriate title, horizontal axis label and vertical axis label to the visualisation. Note: display a single plot ONLY.
– **Print ONE sentence** to state your interpretation of the correlation coefficient in the above context. [Note, quantitative results in the cell output should be shown as rounded values with TWO decimal places.]

6) [**cell6 – 3 marks**] The task is to develop an alternative image quality assessor (IQA), where the **input** contains the **test image ONLY**, i.e., any image contained in the data file "model.zip"; and the **output** should be **a single value** that represents the estimated quality of the test image. Write code to build an IQA algorithm to predict the "overall image quality (note: use IQ-p as the name of target variable)" from some simple image statistics. The options for image statistics are: the "**mean pixel intensity** of an **entire image** (note: use AP as the name of predictor variable)", and the "**medium pixel intensity** of an **entire image** (note: use MP as the name of predictor variable)". The model should be expressed in the following equation:

$$IQ\text{-}p = \alpha \times AP + \beta \times MP \quad (1)$$

where α and β must satisfy the following rules: (1) α + β =1; (2) α and β ranges from "0" to "1" in steps of "0.1". The **performance of the model** must be quantified by the **linear correlation** between the "**algorithm generated image quality**" and "**true overall image quality** (available via data file "Q_scores.xlsx)".

– Search the parameter space of α and β to find **the best combination** for the model (see equation (1) above). **Visualise a single plot** (**you are free to choose the format of visualisation**) that illustrates the **results of your analysis**. The following information is required to be included and accessible in the visualisation: **all possible combinations of α and β values** and the **corresponding performance of the IQA model**; add appropriate title, horizontal axis label and vertical axis label to the visualisation. Note, display a single plot ONLY.
[Note, quantitative results in the cell output should be shown as rounded values with TWO decimal places.]

Learning Outcomes Assessed

This assignment assesses the Learning outcomes 1-4 as stated in the module description.

Criteria for assessment

Credit will be awarded against the following criteria.

Your CODE and RESULTS should be contained within a Jupyter Notebook that analyses and visualises given data (should be obtained via Learning Central: CM2105 Data Processing and Visualisation). This coursework assesses the intended learning outcomes of 1, 2, 3, 4:

1. **Use Python to extract, manipulate, store, and analyse information from a range of sources;**
2. **Understand statistical methods to apply to data;**
3. **Understand static visualisations of data;**
4. **Create static visualisations of data.**

THE PENALTY FOR **UNEXECUTED CODE FOR EACH CELL** IS **AN AWARD OF ZERO MARKS**
Before you submit your Jupyter Notebook file, MAKE SURE you perform the following steps:
(1) Go to "Kernel", and perform "**Restart & Clear Output**";
(2) Go to "Cell", and perform "**Run All**";
(3) Carefully check the results/outputs of each cell, as they are the contents that will be marked.
Note: When marking your Jupyter Notebook submission, the module assessors will first perform steps (1) and (2), then start marking the results/outputs of all cells. It is your responsibility to make sure the **code** is **error free**.

Note: The submission is limited to 6 code cells in your Jupyter Notebook file. If you submit more than 6 code cells, then ONLY THE FIRST SIX CELLS of the submission will be marked as to the stated requirement. Extra submissions will be ignored.

Note: Regarding the layout/organisation of files/folders for marking. Your submitted jupyter notebook (.ipynb) will be placed in the same folder along with "2018-QS-World-University-Rankings-Top200.xlsx", Q_scores.xlsx and Model folder (unzipped folder containing the images). MAKE SURE you use **relative file path** rather than **absolute file path in your code**. [For reference, the absolute path is the full path to some place on your computer. The relative path is the path to some file with respect to the current working directory.]

The maximum mark for the coursework is **20** [equivalent to **100%** of the total marks available for this module]. The mark obtainable for a question or part of a question is shown in brackets alongside the question.

Credit will be awarded according to the correct functioning of the required components of the code, and the account of credit will be awarded according to the following indicators:

- **Fail**: the output of the code cell **does not adequately address** the stated requirement for the Part.
- **3rd**: the output of the code cell **minimally addresses** the stated requirement for the Part; for example, where multiple instances are required, at least one appropriate instance is provided.
- **2.2**: the output of the code cell **partially addresses** the stated requirement for the Part; for example, where multiple instances are required, most instances are appropriately provided.
- **2.1**: the output of the code cell **fully addresses** the stated requirement for the Part, but has weaknesses in terms of the **weakness indicators** below.
- **1st**: the output of the code cell **fully addresses** the stated requirement for the Part, as well as meeting the **excellence indicators** below.

Weakness indicator: Results are not presented in a structured and accessible manner. Little insight and understanding.
Excellent indicator: Results are presented in a structured and accessible manner. Has developed considerable insight and understanding.

An indication of the level of attainment against the appropriate award is given below.

Undergraduate
1st (70-100%)
2.1 (60-69%)
2.2 (50-59%)
3rd (40-49)
Fail (0-39%)

Feedback and suggestion for future learning

Feedback on your coursework will address the above criteria. Feedback and marks will be returned via Learning Central by the date stated on the front page of this document. If you have any questions relating to your individual solutions talk to the lecturer.

Feedback from this assignment will be useful for e.g., CM3203: One Semester Individual Project.

Submission Instructions

Your coursework – your **code and results** should be contained within **an executed Jupyter Notebook named "CW_your student number.ipynb (e.g., CW_1234567)"** – should be submitted via Learning Central by 9:30am on the submission date.

Description		Type	Name
Q1	**Compulsory**	One Jupyter Notebook (.ipynb) file	CW_[student number].ipynb

Any code submitted will be run on a system equivalent to those available in the Windows laboratory OR University provided Windows laptop and must be submitted as stipulated in the instructions above.

Any deviation from the submission instructions above (including the number and types of files submitted) may result in a mark of zero for the assessment or question part.

Staff reserve the right to invite students to a meeting to discuss coursework submissions

Support for assessment

Questions about the assessment can be asked at the beginning of the lectures in Weeks 6-10].

Support for the programming elements of the assessment will be available in the lab classes in Weeks 6-10, or in the daily drop-in lab sessions.