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 2 November 2021
Submission Date and Time: Monday 06 December 2021 at 9:30am
Return Date: Monday 10 January 2022

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.

Your submission must include the official Coursework Submission Cover sheet, which can be found here:

https://docs.cs.cf.ac.uk/downloads/coursework/Coversheet.pdf

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
Cover sheet	**Compulsory**	One PDF (.pdf) file	student number.pdf
Q1	**Compulsory**	One Jupyter Notebook (.ipynb) file	CW_student number.ipynb

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

---

Assignment

Q1. Part1:
The US annual "County Health Rankings" provide information of how health is influenced by where people live, learn, work and play. They provide a starting point for change in communities. A data analytics team attempts to estimate the premature death (i.e., "**Years of Potential Life Lost Rate (YPLLR)**" based on number of deaths under age 75) in Florida. Other variables that they believe offer some insight on the premature death include:
– Teen births, i.e., "**Teen Birth Rate (TBR)**" (Teen births/females ages 15-19 * 1,000);
– Violent crime, i.e., "**Violent Crime Rate (VCR)**" (violent crimes/population * 100,000) ;
– Adult smoking, i.e., "**Percentage Smokers (PS)**" (Percentage of adults that reported currently smoking).
The text file named "**2017Health.txt**" (available on Learning Central) contains the data. Shown below in Table 1 is the form of the data.

Table 1: Example data

State	County	Years of Potential Life Lost Rate	Teen Birth Rate	Violent Crime Rate	Percentage Smokers
Florida	Alachua	6633	19	579	16
Florida	Baker	8270	58	360	19
Florida	Bay	9168	50	508	18
Florida	Bradford	10346	61	461	18
Florida	Brevard	7722	25	518	16
Florida	Broward	5737	23	441	15
Florida	Calhoun	6415	59	130	19
Florida	Charlotte	7353	30	219	14
…	…	…	…	…	…

1) [**cell1 – 10 mark**] **Download the file "CW-your student number 2021.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., "2017Health.txt") into your programme. Write code to analyse the data contained in the variables called "Teen Birth Rate", "Violent Crime Rate", and "Percentage Smokers".
   – Write code to construct ONE tabular data structure (i.e., a DataFrame) that shows the following statistics: the "mean", "minimum", "maximum" and "standard deviation" of the variables called "Teen Birth Rate", "Violent Crime Rate", and "Percentage Smokers", respectively. **Display the tabular data structure** in your programme. Note: display ONE tabular data structure ONLY.
   – **Print the "95% confidence interval"** of the variable called "Percentage Smokers".
   – **Print ONE sentence**, stating your interpretation of the "95% confidence interval" of the variable called "Percentage Smokers".
   [Note: quantitative results should be shown as rounded values with TWO decimal places.]

2) [**cell2 – 10 marks**] Write code to analyse the data contained in the variable called "Years of Potential Life Lost Rate". Write code to plot a bar graph (note, visualise a single plot) that illustrates the difference in the "Years of Potential Life Lost Rate" between North Florida, Central Florida and South Florida. The following information is required to be included in the visualisation: the mean measure of "Years of Potential Life Lost Rate" in each region in question (i.e., North Florida, Central Florida or South Florida); and the **error bars** that indicate the 95% confidence interval.
   [North Florida: use the measures of the following counties: Duval, Alachua, Leon, Flagler, Marion; Central Florida: use the measures of the following counties: Orange, Polk, Hillsborough, Pinellas, Brevard; South Florida: use the measures of the following counties: Miami-Dade, Broward, Lee, Palm Beach, Sarasota.]
   – **Display the visualisation** in your programme. Note: display a single plot ONLY.

3) [**cell3 – 10 marks**] Based on the following two predictor variables: "Teen Birth Rate (TBR)" and "Percentage Smokers (PS)", write code to build ONE linear regression model to estimate the "Years of Potential Life Lost Rate (YPLLR)" in Florida.
   – **Print the resulting linear equation** (i.e., regression model) in the programme.
   [Note: print ONE equation ONLY; quantitative results should be shown as rounded values with TWO decimal places.]

Other regression models could be built using the given data. Based on the error of prediction (note, in this question part you must use the absolute error (or absolute difference) between the measured "Years of Potential Life Lost Rate" and the predicted "Years of Potential Life Lost Rate"), compare the following two linear regression models:

Model A: YPLLR = 60.6 × TBR + 5297.06
Model B: YPLLR = 1.36 × VCR + 7254.3

and advise the data analytics team which model should be used. Write code to perform appropriate statistical data analysis to reveal the statistical difference in the performance (i.e., quantified by the mean absolute error (MAE)) between these two models.
– **Print ONE short paragraph** that describes the data analysis and results. The following information is required in the description: the name(s) of chosen test(s) and results of test(s).
– **Print ONE sentence**, stating your conclusion and justification on the observed difference in performance between these two models (i.e., Model A and B), in terms of predicting the "Years of Potential Life Lost Rate".

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 e.g., "*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").

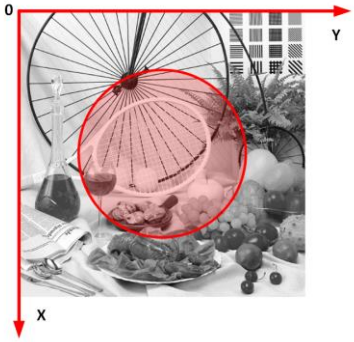
m1.png

m5.png

4) [**cell4 – 10 marks**] Based on "**m1.png**", write code to create a circle that represents a circular region-of-interest (ROI) on the current image, as shown below in the form of a partially-opaque red circular ROI. The coordinates of the centre point of the ROI are [256, 256]. The Radius of the circle is 150 pixels. Write code to analyse the pixels that belong to the ROI including the edge of the circle (i.e., referred to as "inside-ROI") and pixels outside the ROI (i.e., referred to as "outside-ROI"). Write code to visualise a histogram of pixel intensity values. The histogram is a graph showing the number of pixels at each different intensity value.

– **Display the histograms** of inside-ROI and outside-ROI, respectively. The following information is required: there are 256 different possible intensities in a grayscale image, and so the histogram will graphically display 256 numbers (i.e., 256 bins) showing the distribution of pixels amongst those values.
– Calculate a similarity measure (i.e., a numerical score) between the inside-ROI histogram and outside-ROI histogram, based on Euclidean distance. **Print the similarity score** and **ONE sentence** to state your interpretation of the similarity score.

[Note: quantitative results should be shown as rounded values with TWO decimal places.]



5) [**cell5 – 10 marks**] Write code to create a circular region-of-interest (ROI), as illustrated in Q1. Part2: 4), on each image contained in the data file "model.zip". The coordinates of the centre point of the ROI are [256, 256]. The Radius of the circle is 150 pixels. Write code to analyse the linear correlation between the "average pixel intensity of inside-ROI" and "overall image quality". The "average pixel intensity of inside-ROI" is quantified as the intensity value averaged over all pixel locations that belong to the ROI including the edge of the circle.
– **Print the Pearson linear correlation coefficient**.
– **Print ONE sentence** to state your interpretation of the correlation coefficient.
– Conduct your correlation analysis multiple times using different sizes (i.e., radius=50, 100, 150, 200, 250 pixels) of the circular ROI. Each time the same size of ROI is applied to all images contained in the data file to produce the correlation coefficient between the "average pixel intensity of inside-ROI" and "overall image quality". **Visualise a single plot** to illustrate the results of your correlation analysis.
[Note: quantitative results should be shown as rounded values with TWO decimal places.]

6) [**cell6 – 10 marks**] Based on the data file "model.zip", write code to build a linear regression model to predict the "overall image quality (note: use IQ as the name of target variable)" from the "average pixel intensity of an entire image (note: use API as the name of predictor variable)".
– **Print the resulting linear equation** (i.e., regression model). Note, print ONE equation ONLY.
– **Print the mean squared error (MSE)** – the average of the squares of the errors – of the model.
– After building the linear regression model (i.e., best-fit line) for "IQ" and "API", write code to evaluate the outliers. As a rough rule of thumb, we can flag any data point that is located far away from (above or below) the best-fit line as an outlier. **Construct and display a tabular data structure** (i.e., a DataFrame) to illustrate the five worst outliers (i.e., "Names of Images"). The following information is required to be included in the tabular data structure: the API; true IQ; and predicted IQ for each outlier.
– Outliers could be removed from a dataset, and a linear regression model could be re-built based on the remaining data and its performance could be evaluated (based on the remaining data) using MSE. From the five worst outliers, remove ONLY TWO outliers so that the re-built linear regression model gives the minimum MSE. **Visualise a single plot** that illustrates the results of your analysis and indicates the names of the two outliers that should be removed. Note, display ONE visualisation ONLY.
[Note: quantitative results 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 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**.

The maximum mark for the coursework is **60** [equivalent to **100%** of the total marks available for this module]. A mark breakdown in terms of the 60 mark scale (rounded to 0.5 marks) is shown below.

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.

Notebook cell	Maximum mark	1st	2.1	2.2	3rd	Fail
cell 1	10	>=7	>=6	>=5	>=4	<4
cell 2	10	>=7	>=6	>=5	>=4	<4
cell 3	10	>=7	>=6	>=5	>=4	<4
cell 4	10	>=7	>=6	>=5	>=4	<4
cell 5	10	>=7	>=6	>=5	>=4	<4
cell 6	10	>=7	>=6	>=5	>=4	<4

- **Fail**: the output of the code cell does not adequately address the stated requirement.
- **3rd**: the output of the code cell minimally addresses the stated requirement; 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 example, where multiple instances are required, a majority of instances are appropriately provided.
- **2.1**: the output of the code cell fully addresses the stated requirement but has weaknesses in terms of the weakness indicators below.
- **1st**: the output of the code cell fully addresses the stated requirement, as well as meeting the excellence indicators below.

Weakness indicator: results are not presented in a professional and structured manner.
Excellent indicator: results are presented in a professional and structured manner. For example, when multiple instances are provided in the output, clear and concise descriptions are provided to help readers understand the meaning of each instance.

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 using a feedback form. If you have any questions relating to your individual solutions talk to the lecturer.