COMP1832	**Programming Fundamentals for Data Science**	**Faculty Header ID:**	**Contribution 100% of course**
Course Leader **Dr. Jia Wang**	**COMP1832 Portfolio**		**Deadline Date: 08/12/2023**
This coursework should take an average student who is up-to-date with tutorial work approximately 30 hours Feedback and grades are normally made available within 15 working days of the coursework deadline			
Learning Outcomes: 1. Demonstrate knowledge and understanding of commonly used data structures and processing techniques for Data Science. 2. Understand and implement processing pipelines for use in Data Science applications. 3. Build efficient solutions for data manipulation.			

Plagiarism is presenting somebody else's work as your own. It includes: copying information directly from the Web or books without referencing the material; submitting joint coursework as an individual effort; copying another student's coursework; stealing coursework from another student and submitting it as your own work. Suspected plagiarism will be investigated and if found to have occurred will be dealt with according to the procedures set down by the University. Please see your student handbook for further details of what is / isn't plagiarism.

All material copied or amended from any source (e.g. internet, books) must be referenced correctly according to the reference style you are using.

Your work will be submitted for plagiarism checking. Any attempt to bypass our plagiarism detection systems will be treated as a severe Assessment Offence.

Aims

This module provides a hands-on approach to the programming and applied mathematics expertise that is essential to the Data Science programme. Students will develop their understanding and gain practical experience with the data structures and processing algorithms most frequently used in developing Data Science solutions. The module is complementary to the other topics covered in the master's programme and will prepare students in their learning experience as well as in their further pursuit for higher education or work in industry.

Indicative content

This module introduces the basic programming and mathematics knowledge required for successfully building Data Science applications. A range of the following topics will be covered including, but not limited to:

- Basic maths concepts and their computer implementation: Vectors, Matrices, Multi-dimensional Arrays
- Programming data processing tools with Python
- Programming data processing tools with R
- Array-based computation techniques: operation parallelism, memory considerations
- Practical techniques for code reuse and encapsulation.

Learning and Teaching Activities

A combination of weekly 1-hour lectures/tutorials (33%) and 2-hour lab sessions (67%).

This course is to be delivered via several complementary activities: lectures/tutorials, practical work and directed unsupervised learning. The rationale for this mix of activities is to give the students an interesting and varied learning experience combining theory and analysis to underpin the core practical work.

Students will also have extra support through supplemental material in the form of digital media, tutorials and example projects to analyse and disseminate.

Coursework Submission Requirements

- An electronic copy of your work for this coursework must be fully uploaded on **Friday 08/12/2023 latest at 11:30pm** using the links on the module Moodle page for COMP1832.
- For this coursework you must submit **a single PDF** document which include solutions to both Python and R portfolio tasks. This PDF needs to include solutions including answers to questions, justifications of the chosen methods, plots and plot interpretations. **Maximum four pages for the submission**.

- In general, any text in the PDF document must not be an image (i.e. must not be scanned) and would normally be generated from other documents (e.g. MS Office using"Save As .. PDF"). An exception to this is hand written mathematical notation,but when scanning do ensure the file size is not excessive.
- There are limits on the file size (<=500MB).
- Make sure that any files you upload are virus-free and not protected by a password or corrupted otherwise they will be treated as null submissions.
- Your work will not be printed in colour. Please ensure that any pages with colour are acceptable when printed in Black and White.
    - You must NOT submit a paper copy of this coursework.
    - All coursework must be submitted as above. Under no circumstances can they be accepted by academic staff

The University website has details of the current Coursework Regulations, including details of penalties for late submission, procedures for Extenuating Circumstances, and penalties for Assessment Offences.

See http://www2.gre.ac.uk/current-students/regs

Detailed Coursework Specification

Solving data science solutions require considering the nature of the problem addressed, the choices of datasets as well as analytical and visualisation techniques. It is important for our students to be up to date with the current practices of the fundamental programming skills used in data science. Adding these skills from both Python and R to their portfolio will increase the employability of our graduatesand will help them to aim for higher paying jobs in industry, as well as academia.

- **Tasks:**

Portfolio (100 Marks)

The portfolio should include the following components:

1. descriptions of the libraries/functions used and justify your choices
2. summary of datasets (e.g., what is the dataset about, number of rows and columns, and brief introduction of the variables, and etc)
3. code with necessary comments
4. visualisations with detailed plot descriptions (e.g., datasets that are plotted, x, y axis, shapes and colours used in the plot)
5. plot interpretation (e.g., statistic distributions, patterns and trends etc)
6. required answers/solutions by the task

Do NOT include logging information (e.g., errors, warnings, loading libraries and etc) in your submission. They are not part of the data analytics solutions and will consume a lot of space. Keep in mind that the page limit for the coursework submission is 4-page.

- **Deliverables:**

An admissible coursework submission needs to include:

- All required solutions including data, scripts, texts and images with necessary comments and explanations must be arranged and exported into **a single PDF file** and uploaded by **08/12/2023 latest at 11:30pm** using the link on the coursework Moodle page. **Maximum four pages for the submission**.

- **Grading Criteria**

80%-100% Exceptional	You will need to have: an excellent implementation and reflection on understanding your tasks. All requirements are implemented to a higher standard.
70-79% Excellent	A very good implementation showing your solutions with all requirements implemented. Code and plots are clear and readable with justified descriptions and explanations.
60-69% Very good	A good implementation showing your solutions to all the tasks: all required plots are implemented, and both python and R code are working. Explanations of your solutions are provided which reflect good understanding of given data analytics tasks.
50-59% Good	An implementation showing your basic understanding and programming skills of data transformation, basic statistical analyses, and visualisation. Providing solutions with minimum requirements implemented with some justifications.
0-49% Fail	A portfolio with very limited tasks solved. No solution or very few solutions provided for the assignment. A portfolio that fails in reflecting the understanding of the basics of processing and visualising data in Python and R.

Portfolio Tasks

<<<<<<<<<<<<<<<<<<<<<<<Python part>>>>>>>>>>>>>>>>>>>>>>>>>

Select a city with extensive public transport network (your own choice). By utilising the Python programming language, represent part of the public transport network (e.g., bus, tube…) of the selected city by using the graph data structure. **There are in total six tasks that must be completed, with a cumulative mark value of 50.**

Platform for implementation: Jupyter Notebook, PyCharm or any coding platforms supporting Python that you are feeling comfortable with.

Q1. The implementation of the transport network is expected to consist of minimum 5 lines as edges, and minimum 5 stations on each line as nodes **(20 mark)**.

Q2. The network should be completely interconnected, ensuring that there are edges connecting every pair of nodes. **(5 mark)**.

Q3. When visualising the network, use different colours for the different lines and their corresponding stations **(5 mark)**.

Q4. Set up and display attributes for the edges, which represent the distances between the stations. If precise station-to-station distances are unavailable, estimate them by leveraging online mapping services such as Google Maps **(10 mark)**.

Q5. Visualise the names of the lines and the stations on the generated map **(5 mark)**.

Q6. Based on the generated map, suggest two improvements without implementation, each improvement with maximum 3 sentences, which would improve the commute of passengers **(5 mark)**.

<<<<<<<<<<<<<<<<<<<<<<<R part>>>>>>>>>>>>>>>>>>>>>>>>

Data transformation, basic statistics, data analytics and visualisations using the ONS (Office for National Statistics) fraud and computer misuse datasets**.**

There are in total 4 mandatory tasks that must be completed, with a cumulative mark value of 50.

Data used for this task: **Excel spreadsheet "cw_r.xlsx" from Moodle (coursework specification section).**

Platform for implementation: RStudio (recommended) or any coding platforms supporting R that you are feeling comfortable with.

Q1. Get to know your datasets (6 mark).

1. Provide the code to load the Excel spreadsheet file cw_r.xlsx into R console, and use *excel_sheets()* to list all the sheets included in cw_r.xlsx (**1 mark**).
2. Use your own words to briefly describe this Excel file, i.e., the number of data sheets (or called data tables or datasets) contained in the file, and what data/information is recorded in each sheet, and perhaps the links between the different data sheets (**5 mark**).

Q2. Define variables and naming your datasets (4 mark).

1. Provide the code to import the following data sheets to R (Table 3d, Table 5, Table 7, Table 8) by assigning them to four variables (**2 mark**). We will soon use the data from "Table 3d" for Q3, and the data from "Table 5" for Q4.
2. Give these four variables descriptive and meaningful names that accurately describe the content of the dataset. Avoid generic names that provide little content. Use lowercase and hyphen if multiple words are used (**2 mark**)**.**

Q3. Data pre-processing and distribution plotting (25 mark)

The original dataset 'Table 3d' (you rename it in Q2) describes types of fraud and computer misuse in the year range of 2012 to 2021. **Generate two new data frames** by sub-setting/slicing this dataset, with each data frame representing one category of fraud, i.e, one data frame represents **banking and credit industry fraud,** and the other represents **consumer and retail fraud**.

Once the two data frames are defined, **visualise both dispersion and central tendency**

of these two datasets.

To get the full mark, you will need to provide:

1. Code for creating the two new data frames using package **dplyr.** Use head() and str() to examine the basic information of the two data frames (**5 mark).**
2. Code for visualise the dispersion and central tendency of the two data frames (**5 mark).** You have the freedom to choose between R base graphics or ggplot2 for creating visualisations. It's up to you to determine the most suitable types of plots for this specific task.
3. Plot interpretations. Describe the statistical distributions of these two datasets **(10 mark)**
4. A concise description of the workflow to solve this task as well as the justification of your choice of plotting **(5 mark)**

Q4. Data pre-processing and plotting using ggplot2 (15 mark)

What trend/patterns can be found regarding both **number of offences** and **rate per 1000 population** in England (original dataset "Table 5" which you re-name it in Q2)?
You need to visualise such patterns at both region (e.g., North East is a region) and county levels (e.g., Cleveland in North East is a county).

To get the full mark, you will need to provide:

1. Code for creating a new data frame using package **dplyr**. This data frame is sliced and re-arranged from the original dataset and contains only the columns and rows that are relevant to this task **(5 mark)**

2. Plot and explain: Which region(s) has the highest total count of offences as well as offence rate per 1000 populations?

   It is up to you to determine the most suitable types of plots for this specific task. Provide the code for visualisation and the generated plot(s). Justify your choice as part of the explanation. **(5 mark)**

3. Plot and explain**:** based on the visualisation**,** what are the top three counties that have the lowest total count of offences?

   It is up to you to determine the most suitable types of plots for this specific task. Provide the code for visualisation, the generated plot(s). Justify your choice as part of the explanation. **(5 mark)**