Aston University
BIRMINGHAM UK

College of Engineering & Physical Sciences
Assignment Brief

AM41DP Data Science Programming	Final Coursework

Adam Farooq a.farooq6@aston.ac.uk

Assignment Brief/ Coursework Content:

- This coursework consists of a full data exploration activity, including exploratory data analysis, merging relational data, data pre-processing and clean-up, feature engineering and modelling. Data from a real application will be provided, together with a set of specific objectives.
- Most of the coursework activities will feel familiar to students who have engaged with the lectures and tutorials. The activities will require some independent exploration and critical thinking, but the core concepts are covered in the lecture videos and live sessions.
- The coursework is aimed at assessing all four Learning Outcomes of this module, using **Python or R** for a variety of data science-related activities.
- The coursework must be completed **individually**. This does not mean that you cannot discuss with your peers or exchange ideas and insights, but only that your solution and report need to reflect your own thinking and understanding.

Please read these instructions carefully. *There are relevant points throughout the sections, as well as some potentially interesting tips.*

Task Details

Introduction to the Problem

Experimental determination of biological properties of protein sequences, such as their ability to elicit an immune response, is an expensive and time-consuming activity that remains an important bottleneck in the development of new vaccines and serological tests for infectious diseases. As a consequence, the development of computational approaches for predicting these properties from publicly-available proteomic data have become increasingly relevant.

For more information see[1].

Task Details

In this coursework you will explore a number of data science tasks related to a problem.

Two data sets are available in the Blackboard page:

- **Sp_epitopes:** this data set contains 18 columns describing B-cell epitopes (protein fragments that are associated with the immune response of a given host) of the parasite Streptococcus pyogenes[2]. 1 This csv file contains the following variables (the most relevant ones are highlighted in **boldface**):
    - pubmed_id –ID of the paper where the epitope was described
    - year – Year of publication
    - epit_name – Name of the epitope as used in the reference
    - **epitope_id** – Unique epitope ID
    - evid_code – Evidence code
    - epit_struc_def – Observation type
    - sourceOrg_id – taxonomy ID of the source organism
    - **protein_id** – ID of protein where the epitope was identified
    - **epit_seq** – aminoacid sequence of the epitope
    - **start_pos** – initial position in the protein (position count starts from 1)
    - **end_pos** – final position in the protein (position count starts from 1)
    - n_assays – number of assays used in validating the epitope
    - host_id – ID of the host organism
    - bcell_id – ID of the B-cells used in each assay
    - assay_type – Type of the experimental setup used to measure epitope binding
    - **n_Positive** – number of assays that returned a Positive value
    - **n_Negative** – number of assays that returned a Negative value
    - **assay_class** – individual results of each assay
- **proteins.csv:** this data set contains information about approximately 14,000 proteins that are related to several pathogens, including but not limited to the proteins containing the epitopes from **Sp_epitopes**. This csv file contains the following variables (the most relevant ones are highlighted in **boldface**):
- TSeq_seqtype – type of sequence
    - TSeq_accver – Accession number

[1] https://academic.oup.com/bioinformatics/article/37/24/4826/6325084 or https://www.news-medical.net/life-sciences/Amino-Acids-and-Protein-Sequences.aspx

[2] Streptococcus pneumoniae: For Clinicians | CDC

- TSeq_taxid – taxonomic ID of the organism from which the protein was isolated
- TSeq_orgname – name of the organism from which the protein was isolated
- TSeq_defline – metadata related to protein description
- **TSeq_length** – total length of the protein sequence
- **TSeq_sequence** – aminoacid string representing the protein
- **UID** – unique ID value (either the Accession Number or a Uniprot ID)
- DB – Database from which the protein was retrieved.
- TSeq_sid – SID identifier

These data sets were created by parsing existing proteomic sequences, retrieved and consolidated from the online databases IEDB[3], NCBI GenBank[4] and Uniprot[5], using the development version of the R package epitopes[6]. The two data sets are related by the variable pair **protein_id – UID**, which should be used for joining the relevant protein information onto the epitopes data set.

Descriptive Details and Task Requirements

Your task in this coursework is to develop a **fully reproducible** report either as a *Jupyter notebook* or *R Markdown notebook***,** documenting your work on this data together with your code chunks. Your notebook must represent an actual structured document (containing Title, Author, a short Executive Summary, Problem Definition, a well-structured description of your Data Analysis process, and Conclusions), not just a sequence of code blocks.

The specific requirements of this coursework are listed below, organised using the MoSCoW prioritisation method[7].

ID	Description	Priority
M1	A short summary of the dataset.	Must (2.5 marks)
M2	Adequately join the protein sequences onto the epitopes table	Must (5marks)
M3	Remove invalid observations: (i) those without corresponding protein sequences; (ii) those for which the epitope substring is not located in the correct position of the protein string (based on start_pos and end_pos); (iii) those with start_pos < 8 or with end_pos > (TSeq_length – 8)	Must (10 marks)

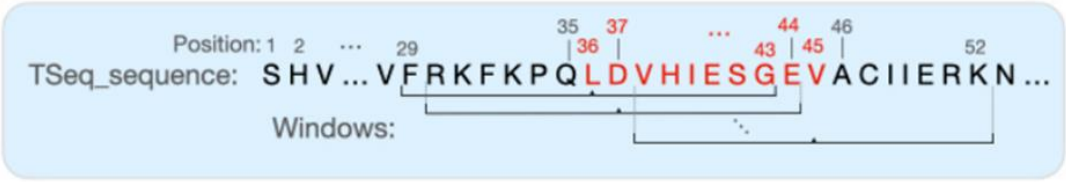
[3] https://www.iedb.org/
[4] https://www.ncbi.nlm.nih.gov/genbank/
[5] https://www.uniprot.org/
[6] https://github.com/fcampelo/epitope
[7] https://www.projectsmart.co.uk/moscow-method.php

	(iv) those containing any non-specific aminoacid letters (namely B,J,X or Z) in the epitope sequence. Report the dimension of the data after each step.	
M4	Calculate the Class attribute for this data set based on the number of positive and negative assays (if n_Positive >= n_Negative then Class = 1, otherwise Class = 0)	Must (5 marks)
M5	Develop high-quality visualisations of the main characteristics of the resulting joined data set. The minimal requirement here is (i) a graphical investigation of Class balance; (ii) a graphical investigation of the distribution of epitope lengths; (iii) A grouped bar chart of the frequencies (%) of each aminoacid letter in Positive vs Negative observations. Discuss what these visualisations are showing	Must (10 marks)
M6	Based on the resulting data set of (M1)-(M3), assemble an **expanded data set**, such that each aminoacid of each epitope sequence is represented in an individual row. This expanded data set must have a new variable called AA_window , containing a substring of length 15 centred on the specific aminoacid. See **Figure 1** below for details. Report the dimension of the data after each step.	Must (10 marks)
M7	Adequate documentation of all analysis steps in the form of a *fully reproducible* report, either as a Jupyter notebook or R Markdown notebook. By *fully reproducible* what is meant is that all analyses, figure generation etc. contained in the report must be able to be re-executed by an independent assessor, assuming that the data files are in the same folder as the report notebook.	Must (7.5 marks)
S1	Report the class imbalance of the **expanded data set.** To deal with this class imbalance you must keep all 'positive class' observations and randomly select 1.5*num_positive number of 'negative class' observations from the **expanded data set;** where num_positive is the number of positive observations. Report the dimension of the data after each step.	Should (2.5 marks)
S2	Calculation of features related to the letters in the *AA_window* sequences calculated in (M5): - 20 features for the frequency of each individual letter; - 400 features related to the frequency of each possible pair of letters.	Should (10 marks)
S3	Splitting of the expanded dataset into training (80%) and testing (20%) set based on the **protein_id** value (i.e., all epitopes with the same **protein_id** should be under the same split) after M5.	Should (2.5 marks)

S4	Fit **a classification** model using the training set and report the model's predictive performance. Note the model should only use 420 features from (S2) and the class labels from (M3).	Should (5 marks)
C1	Calculate additional features after (S2): - Shannon entropy of *AA_window*.[8] - TMM and MHI of *AA_window*.[9] [*Note: TMM is calculated as the weighted sum of the letter counts in AA_window, with weights being given by column Molecular mass (Da) in the table provided in the footnote. MHI is calculated as the weighted arithmetic mean of the letter counts in AA_window, with weights being given by column Hydropathy index* in the table provided on the footnote.] - Total number of Carbon, Hydrogen, Oxygen, Nitrogen and Sulphur atoms in *AA_window*.[10] [*Note: this is calculated as a weighted sum of the count of each letter in AA_window, with the weight of each letter being given by the respective column* in the table provided]	Could (15 marks)
C2	Develop high-quality visualisations of the main characteristics of the results from (C1)	Could (5 marks)
W1	Code is written in a consistent and neat manner	Would (5 marks)
W2	Results are explained by bringing new insight	Would (5 marks)

[8] https://en.wiktionary.org/wiki/Shannon_entropy
[9] http://www.imgt.org/IMGTeducation/Aide-memoire/_UK/aminoacids/abbreviation.html
[10] https://www.dropbox.com/s/mz8tmta5kkxp2gn/Atoms.csv

protein_id	epitope_id	epitope_seq	start_pos	end_pos	...	Class
YZX	XYZ	LDVHIESGEV	36	45	...	
ABC	BCA	...	17	25	...	



An epitope of length *N* generates *N* rows in the expanded data set:

protein_id	epitope_id	AA_position	AA_window	Class
YZX	XYZ	36	RKFKPQLDVHIESG	
YZX	XYZ	37	KFKPQLDVHIESGE	
YZX	XYZ	38	FKPQLDVHIESGEV	
YZX	XYZ	39	KPQLDVHIESGEVA	
YZX	XYZ	40	PQLDVHIESGEVAC	
...	...	...	...	
YZX	XYZ	44	DVHIESGEVACIIER	
YZX	XYZ	45	VHIESGEVACIIERK	
ABC	BCA	17	...	

Figure 1: Example of data set transformation for Requirement (M5). The Class column of the resulting rows should receive the same value as the original epitope from which the windows were extracted. The only columns that are required in the expanded data set at the end of M5 are the ones shown above.

Other solution Guidelines:
This coursework can be approached using **Python** and/or **R** (including combined solutions). Carry out your CW solution process in a systematic way, and take good notes on what you have done.

On Blackboard you can find two solutions that were submitted to a previous Data Science Programming CW. to a loosely similar task: **do not follow this model solution blindly**, since the data and CW requirements were very different! The model solution is only there to give you an initial idea idea of the sorts of thought processes and structure that you could use.

Assessment:
Your coursework will be assessed in two parts: quality of the report (20%, including reproducibility, adherence to format, quality of text, etc.) and use of adequate data science programming tools to solve the problems (80%, including the correct use of R or Python tools, development of good visualisations, correct data filtering and selection, etc.).

Completion (to reasonable level of quality) of *all* requirements described as **Must** are necessary to reach the pass threshold (50). Completion (to reasonable level of quality) of *all* **Must + Should** are required to reach the levels of Merit (60+) or Distinction (70+). **Could** and **Would**-type requirements are additional activities that can be used to add extra marks to the report (again, if completed at a reasonable level of quality).

Recommended reading/ online sources:

- All course materials are available on Blackboard: Lecture notes, recordings and free textbooks stored under *Learning resources.*

Key Dates:

27th Jan 2022	Coursework set
28th April 2022 11:59am (UK time)	Submission date
TBC	Expected feedback return date (individual coursework feedback summary, available on Blackboard.)

Submission Details:

Coursework files must be submitted electronically on Blackboard, using the link available under the "**Assessment**" area. Late submissions will be given a 10% penalty per day.

Report submission guide:

- Please clearly indicate your name on the report.
- The report will be submitted through a specific link on Blackboard, which will be located under **Assessment - Coursework**. The file name of the report should follow the naming convention ***Lastname_Firstname_AM41DP_Report***.
- The report **must** be submitted as either a Jupyter Notebook (**ipynb)** or R Markdown notebook (**Rmd)** file, and must be able to be parsed assuming that the original CSV data files are available under the same folder as the report file (if your notebook works in RStudio Cloud or Google Colab it should be ok).
- Reports submitted as any document format other than the ones above will be subject to a flat penalty of **5 marks**. DO NOT compress your files.
- The total length of the report (excluding template section headers, cover sheet and references) must not exceed 40 pages (do a "print preview" to check). A **10 marks** penalty will be given if the page count exceeds this.
- The report should be well-organised to describe your data analysis steps. A suggested general structure is provided at the beginning of section *Descriptive Details and Task Requirements*.

Marking Rubric:

Pass threshold	
50	All **Must** requirements are fulfilled at a minimal standard of quality.
51 - 59	All **Must** requirements are fulfilled at a minimal standard of quality, with some showing evidence of more development. Some **Should** requirements are explored as well.
Merit	
60 - 69	All **Must** and **Should** requirements are fulfilled at or above a minimal standard of quality. Some of these requirements show evidence of a more critical or insightful implementation or analysis. Some **Could** requirements are explored as well.
Distinction	
70 - 89	All **Must** and **Should** requirements are fulfilled above a minimal standard of quality, showing strong evidence of high-level development. Most **Could** requirements are explored and developed, with insightful discussions into the

	specific requirements and the programming solutions used to tackle them. Some **Would** requirements are included.
90+	All aspects of the coursework are presented at a solid professional level, similar to what could be found in an academic publication. All **Must, Should, Could** and **would** requirements are developed showing evidence of new insights into the exploration of the data.