Assignment 4

There are two questions in assignment 4 – you must answer both.

Question 1 - CARET (Marks: 1/2)

In the resources for Assignment 4 is an R notebook that uses caret to perform a classification investigation using Support Vector Machines. The data for this notebook is from kaggle and can be downloaded from the assignment. Please create a similar notebook to undertake a **functionally equivalent** classification using Sk-Learn (Python).

Hint: Since Sk-Learn can optimise preprocessing parameters it should be possible to use less code that in caret to find the best *num_components*.

The goal is not to win the kaggle contest, but to create an equivalent solution in a different framework. There is an example tutorial here to show to implement the Sk-Learn pipeline framework. Note that this example does not do an initial train/test split so you will need to engineer that from scratch.

You need to research Python documation to learn how to:

- read a CSV data file.
- view the structure of a dataset
- convert a categorical variable to binary
- how to do a 70/30 split stratified on the the target
- ignore the **id** variable
- rebalance the training data using Borderline SMOTE
- normalize the predictor data
- perform UMAP dimensional reduction (with a variable *num_components*)
- classify using SVM with a radial basis function (with a variable hyper-parameter *C*)
- turn as many of the above steps as possible into a “pipeline”
- tune the pipeline using 10 fold cross validation of the train data
- predict the test results
- turn the test results into a confusion matrix
- visualise the confusion matrix (an alluvial plot is too hard; just use a *ConfusionMatrixDisplay.plot()*)

Turn these code snippets into a python notebook and then submit this.

Question 2 - Quality Control (Marks: 1/2)

The file ***monitor.csv*** contains comma separated data. The columns are
Timestamp - the time-stamp of a model prediction being run
ProcessMemory - the allocated memory (MB) of the relevant server process
Prediction - the value predicted by the model
PredictionTimeMS - the duration of the prediction task in milliseconds

You will need to add a day-of-the-year column or something similar to marks the days.

Using the supplied CSV data, generate control charts and answer the following questions:

a) Is the memory usage of the server in control?
b) Is the prediction time of the model in control?
c) Is the stream of predictions in control?

The relevant control charts would be “xbar” and “s”. The values belonging to a single day constitute each group of values. Assume the first 40 days of data can be used to establish the control limits for the remainder of the data.

Have a read of https://cran.r-project.org/web/packages/qicharts/vignettes/controlcharts.html

Present your results as an **R notebook** document (RStudio file menu, new file, R Notebook) showing the various charts and summary tables. Please submit the *.nb.html file created by previewing your notebook.

I recommend an overall summary that looks something like this:

Measurement	Xbar breaches	Xbar runs signal	S breaches	S runs signal	Overall
Memory	3	1	1	1	Out of control
Prediction	0	0	1	0	Out of control
Time	0	0	0	0	In Control