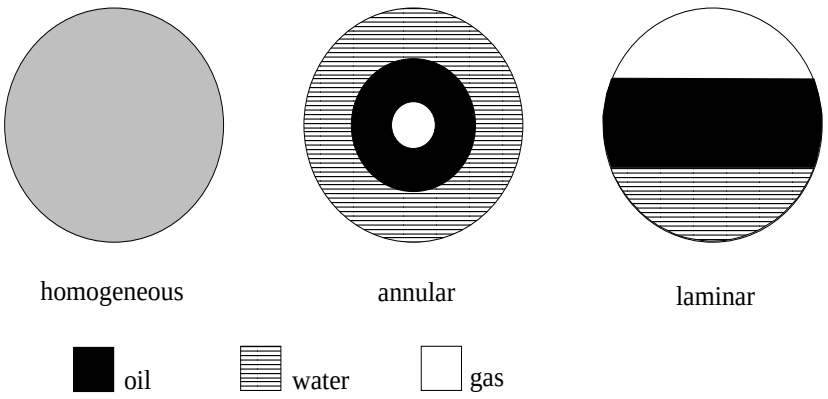
Machine Learning and Neural Computing: Data Classification Coursework

Date to be handed in: by 12 noon 26/03/2021

Introduction

The data that you have been given is the 3 phase oil dataset. Archive of this dataset can be found at `http://inverseprobability.com/3PhaseData.html`. This 12-dimensional dataset is from a physics-based simulation of a non-invasive monitoring system, used to determine the quantity of oil in a multi-phase pipeline containing a mixture of oil, water and gas. The whole dataset includes 3 classes, which are *homogeneous*, *annular* and *laminar (stratified)* namely. The configurations of the flow in the pipe are shown in the following figure.



You have been given two data files, which can be obtained from the module site on Canvas. One includes the training set, named *trndata.csv*, and the other one, *tstdata.csv*, includes the test data. The last column of each file is the class label indicating the configuration of the flow in the pipe: a value of 1 denotes *homogeneous*, a value of 2 denotes *annular*, and a value of 3 denotes *laminar*.

The training set you have been given consists of 1000 instances, 343 labeled as 1, 316 labeled as 2 and 341 labeled as 3. This set can be treated as a *balanced* dataset. The test set is also balanced, and contains 300 instances. You can assume that the data is of satisfactory quality and requires no preprocessing / data cleansing **other than normalisation**.

To classify the data you will be using Support Vector Machines (SVMs). The **type** of SVM you need to use is the **C-SVC (Cost-Support Vector Classifier)** and the **kernel function** you should use is the Gaussian **radial basis function (RBF)**.

Software Required

For this coursework you will need to write your Python code (in version 3 and above) in the Jupyter Notebook. You can use functions from the following packages: Numpy, Pandas, Matplotlib, Seaborn and Sklearn. Your practical session notes should be very useful - these are all available on *Canvas*.

1. Task 1 - Data Exploration (13 marks)
   In this task, you need to use Principal Component Analysis (PCA) to understand the characteristics of the datasets.

   (a) Use Pandas to load both the training set and the test set (1 mark). (Let's denote this original training set as training set (I).)

   (b) Show one scatter plot, that is, two features of the training set against each other. It is your choice to show which two features you want to use. You need to set the label for the x-axis and y-axis, separately, and use different colours to distinguish the three classes (3 marks).

   (c) Normalise the training set and the test set using **StandardScaler()** (Hint: the parameters should come from the training set only) (2 marks).

   (d) Perform a PCA analysis on the scaled training set and plot the scree plot to report variances captured by each principal component (3 marks).

   (e) Plot two subplots in one figure: one for projecting the training set in the projection space constructed using the first principal component (PC1) and the second principal component (PC2); the other one for projecting the training set in the projection space constructed using the second principal component (PC2) and the third principal component (PC3). You need to label the data using different colours in the picture according to its class and set the label for the x-axis and y-axis, separately. (Hint: examples on how to use pyplot.subplot in matplotlib can be found here: `https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.subplot.html`.) (3 marks).

   (f) Obtain projections of the test set by projecting the scaled test data on the same PCA space produced by the training set in Task 1 (d) (1 mark).

2. Task 2 (3 marks)

   (a) Divide the training dataset into a smaller training set (II) and a validation set using the **train_test_split** function and report the number of points in each set. Usually, we

use 20%-30% of the total data points in the whole training set as the validation data. It is your choice on how to set the exact ratio (2 marks).

(b) Normalise both the training set (II) and the validation set (Hint: the parameters should come from the training set (II) only)(1 mark).

3. Task 3 - Non-linear Classification (12 marks)

(a) Basic task (5 marks)

i. Choosing the most suitable parameters (3 marks)
When using the C-SVC SVM with the Gaussian radial basis kernel there are two tunable parameters, *C (cost)* and γ *(gamma)*. You have been given the following combinations: [C=50, γ=10], [C=50, γ=20], [C=100, γ=10], and [C=100, γ=20]. You should train an SVM model for each combination from the given 4 combinations and then test it on the normalised validation set. The accuracy rate for each combination on the validation set should be reported. Finally, you need to select the best combination of parameters and report your result.

ii. Non-linear classification (2 marks)
You should now be in a position to further test your model with the selected parameters by classifying the test data. With the normalised whole training set (I) as the input, you will need to train an SVM model with the suitable parameter values discovered for C and γ in Task 3 (a)i. When the classification model is built you will then need to use it to classify the normalised test set, and report the accuracy rate.

(b) Advanced task - non-linear classification with features reduced using PCA(4 marks)

i. Looking at the scree plot which you have produced in Task 1 (d), how many principal components (PCs) you would like to use to do feature reduction? Explain the reason (1 mark).

ii. Reduce features for both the normalised training set (I) and the normalised test set using the PCA result from Task 1 with the number of principal components you have decided to use (1 mark).

iii. Do the classification using the Gaussian radial basis kernel SVM with parameter values selected in Task 3 (a) (2 marks).

- Normalise the training set and the test set after the feature reduction.
- Train an SVM model on the training set with reduced features.
- Test the model on the corresponding test set, that is the one with reduced features and report the classification result on the test set.

(c) Summarize your findings and write your conclusions in critical thinking. For example, which model gives a better classification result: the one trained on the original features or the one trained on the reduced features? Is this what you have expected? Why? You need to provide evidence to support reasons you give. (3 marks).

What to Submit

The deliverable for this coursework includes

- an experimental report with no more than 10 pages including appendix and less than 1500 words (Please use a single column format. Font size should be set to 11 or 12 point, and the line spacing should be set to 1.5 lines or single) in the PDF format. You need to put your Python code and the corresponding output for each task in the report. All submitted screenshots should be clear and readable.
- a Jupyter Notebook including all code you have written for this coursework.

Please name both submissions using your student ID. For example: 17000000.pdf and 17000000.ipynb.

Overall, there are **2 marks** for presentation and clarity of the submitted report. Note that you must do this coursework individually. You need to submit your coursework via Canvas to the assignment portal: Data Classification.

Please note that the 'Turnitin Submission for Data Classification' portal is for you to obtain a text matching similarity report with a view to improve your academic writing as necessary. You can submit your work to Turnitin as many times as you like. However, please do not submit the work (that is your final submission) that you would like to be marked there.