Data Science Techniques and Applications 2020-21

Coursework II:
Multi-dimensional analysis and dimensionality reduction on a Kaggle dataset

This is the second coursework of this module. This document is implicitly extended with the general college rules and prohibitions that were set forth in the text of the first coursework document. Please refer to the text of Coursework I for, e.g., late submission and plagiarism. Likewise for formatting guidelines. In what follow students will be assumed to be acquainted with the text of Coursework I.

The goal of this coursework is to build on your analysis of a Kaggle[1] dataset to develop a Data analytics, applying the concepts seen in class. A light coding and a technical annex describing the method and the results on your dataset of choice will suffice.
The technical annex can assume all the concepts analysis described in Coursework I (CW I), which is available to the markers and will be read back-to-back with Coursework II.
If inclusion from CW I is needed, it won't count for plagiarism. *However, such inclusions should be made as quotations (see example of quotation in the text of CW I), also to avoid Turnitin detection.*
Your work, and the writing of the technical annex, shall be organized in phases as follows.

Phase 1.
Reconsider the Kaggle dataset analysed in CW I, in particular, the analysis of the most important dimensions.
Select three dimensions that could be used as *predictors* of a fourth, *predicted*, dimension.

The predicted dimension could be a numeric value or a class/category.
For instance, in the Iris dataset we could consider Sepal length, Petal length and Sepal width as predictor dimensions and the actual flower classification as the predicted dimension.
Or we could take Petal Width as the (numerical) predicted dimension.

Visualise the values of the predictor dimensions with appropriate Matplotlib[2] scatterplots. (For the Iris case, 2D scatterplots have been seen in class).
The predicted dimension will be an extra dimension in the scatterplot that can be represented, e.g., by colour of the points in the scatterplot.

1 https://en.wikipedia.org/wiki/Kaggle
2 More advanced Python modules such as Bokeh and Dash are allowed but not required.

Discuss the question of whether the three selected dimensions, taken together, could become a good predictor for the predicted dimension.

Phase 2.

Write a simple Python program that loads the reduced (3+1 dimensions) dataset and performs Principal Component Analys using appropriate Scikit-Learn[3] functions.
As an illustration of how to invoke PCA, consider this example by Gaël Varoquaux.
Should the projected dataset contain a high number of datapoints, to the extent that PCA computation is unfeasible on your computer, you can reduce size by randomly sampling a fraction of the datapoints. Please explain your choices in the essay if you do so.

Phase 3.

Describe the results of the analysis and comment them, possibly with a graphical display of the results (see again Varoquaux's example).
Address the following questions:
What motivated your choice of the three predictor dimension? Are there alternatives that could be explored?
Self-assessment: do your selection and the PCA provide a "good" dimensionality reduction?

Important dates: Please refer to dates and times on the Moodle platform.

Submission:

1. **an essay, with the key code fragments shown inset and commented, and**
2. **a source-code file (with extension .py) ready to be inspected and run.**

Please notice that submission of notebooks *is explicitly disallowed* unless it is intended as the first element of the submission (the essay).

Please use your judgement on the right amount of data and length of presentation for a simple technical description. Please edit your essay using the formatting guidelines given in Coursework I. Citation style is also the same as in CW I, e.g., [Narayanan et al., 2011] which is recommended background reading for those working on data about---or produced by---humans.

Plagiarism: The same rules and procedures of Coursework I apply.

3 http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html

References:

[Narayanan et al., 2011]
A. Narayanan, E. Shi, B. I. P. Rubinstein, 2011.
Link Prediction by De-anonymization: How We Won the Kaggle Social Network Challenge
Proc. of the 2011 Int'l Joint Conference on Neural Networks.
https://arxiv.org/abs/1102.4374