CO7093 - Big Data & Predictive Analytics

CW Assignment

Classification & Clustering

Assessment Information

Contribution to overall mark	**70%**
Submission Deadline	**07/04/2025, 12:00PM**
Release Date	**19/02/2025**

Any clarifications or updates will be communicated via the Blackboard page.

Overview

The primary goal of this assignment is to familiarize you with key stages of a data science project, including formulating and answering questions about data, visualizing insights, and building predictive models capable of making new predictions. Y you will apply the knowledge gained throughout the module. The dataset is provided in the CSV format.

Learning Objectives

After this homework, students will be able to:

- Work with basic **Python data structures** such as dict, tuple, list etc.
- Use **Pandas** as the primary tool to process structured data in Python with CSV files. Handle extreme cases appropriately. Use appropriate methods to address missing data.
- Use **PyPlot** to make simple plots to investigate a specific phenomenon. Read plotting library documentation and use example plotting code to figure out how to create more complex Seaborn plots.
- Train a machine learning model and use it to make a prediction about the future using the **scikit-learn** library.
- Use **PySpark** to explore efficient approaches to handling big data.

How to submit

For this assignment, you need to submit the followings:

1. **A short report** (about 8-10 pages in pdf including any images) on your findings in exploring the given dataset, a description of your model and its evaluation, a description of your clusters and its justification, local classifiers based on your clusters as well as their evaluation.
2. **The Python source code** written to complete the tasks set in the paper. You should submit the Python code file, group1_solution.py or group1_solution.ipynb. Note that even if you decide to work on your own, you must enrol yourself into a group.

3. **A signed coursework cover** – this should include the names of all the students involved in the work submitted.

Please put your source code, report and signed coursework cover into a zip file CW2_GroupID.zip (e.g., CW2_Group1.zip) and then submit your assignment through the module's Blackboard page by the deadline. Note that to submit, you need to click on the Coursework link on Blackboard and then upload your zipped file. Remember it is **1 submission per group**. We encourage you to use **GitLab** versioning control system to store your code and report. Details about using GitLab will follow shortly.

AI Policy

Generative AI must NOT be used when attempting any part of this assignment.

This is a group assignment, but collaboration between groups is not permitted. Plagiarism will be treated strictly according to standard university and departmental procedures. Your submissions will be sent to a plagiarism detection service. In line with university policy, marking will be done anonymously. Only the Blackboard-supplied userid / student number will be visible in marking. For the above two reasons, do not include your name or any other personally identifiable information in your programs.

Instructions for group work

This assignment may be attempted in groups of size up to 3. Group size of 1 (individual groups) are also allowed. The link to join a group will be available on the module's Blackboard page under Assessment and Feedback Section. Please read the following instructions carefully before you join a group.

1. Please join one of the existing groups. Do not create new groups.
2. The maximum number of members in a single group are 3.
3. You should join a group, even if you intend to work individually.
4. The deadline to join a group is 26/02/2024 at 12:00. After the deadline you will be automatically assigned to your own group of size 1.
5. Before joining a group, please discuss it with other group members and make sure that you are happy working with each other.
6. Once the groups are finalised, you will not be able to leave or join another group.

Normally, a group will be given the same mark unless some members made little or no contributions. Any group can be called for an interview during marking. All group members **must attend**, explain their contributions, and defend the work submitted.

Problem Statement

In this coursework, you will create a classification model that, given a Covid-19 patient's current symptom, status, and medical history, will predict whether the patient is in high risk or not. This dataset contains an enormous number of anonymized patient-related information including pre-conditions. The raw dataset consists of 21 unique features and 1,048,576 unique patients. We have applied some simple data cleansing techniques to reduce data to 200031 unique patients and 21 unique features.

Objective

Using the given dataset, the goal is to determine if the patient is at high risk and will be admitted to ICU or not. You will use appropriate performance metrics to evaluate the performance of your model. The data is not clean, and you will have to apply appropriate methods to clean the data. Additionally, using unsupervised clustering, you will have to implement cluster-based classification model that may improve the performance of the model. The (partially) processed dataset is available to download from the blackboard.

Tasks

Your first task is to prepare the data and carry out data cleansing, bearing in mind the question you would like to answer. For example, which factor is the most important factor in predicting the readmission of a patient.

Part 1: Building up a basic predictive model

Load the dataset `patients.csv` into pandas dataframe and carry out the following tasks.

Data Cleaning and Transformation

If you have a closer look at the dataset, you will see that there are lots of inconsistencies in the dataset. While there are no explicit 'null' values, some binary attributes contain entries such as '?', which represent missing values. These need to be handled appropriately. For the first task, adopt an aggressive approach to address these issues. Below is a list of steps you should consider. This list is not exhaustive, so feel free to explore additional techniques that demonstrate your understanding of exploratory data analysis (EDA).

- Check dataset shape.
- Remove irrelevant columns. Clearly justify any deletions in your report.
- Identify and handle missing values. Some missing values are represented as strings like '?'. Some columns may contain values that fall outside their expected range. Identify all the missing values and convert them to NaN.
- Summarize missing values before and after handling them.
- Verify and adjust data types as needed for consistency.
- Drop rows containing null values.

- Analyse numerical features. Display summary statistics and identify potential outliers. Remove outliers if necessary.
- Normalize features where applicable.
- Check the final dataset shape after preprocessing.

Data Visualisation

Consider the resulting Dataframe. This first aggressive cleaning should give a smaller dataset, which you can start by exploring relationships between the various features of the dataset.

- Plot the distribution of unique classes of the target variable.
- Plot the count of number of ICU cases against age.
- Plot a graph that displays the count of target variable against 'CLASIFFICATION_FINAL'.
- Show the scatter matrix plot and the correlation matrices. Can you identify pairs of highly correlated features.
- Generate additional plots that demonstrate your understanding of the problem and the data. You are free to select the plot and features for visualisation. For better visualisation and understanding of data, consider using seaborn library.

Model Building

Consider the resulting Dataframe:

- Select the predictors that would have impact in predicting `ICU`.
- Build up a first linear model with appropriate predictors and evaluate it. Split the data into a training and test sets. Evaluate your model by using a cross-validation procedure.
- Use different performance metrics to evaluate the performance of your model. You might have noticed that the data is imbalanced. The number of positive examples is less than 8% of the total dataset. Choose appropriate performance metrics to evaluate the performance of your model.
- Balance your data using data balancing technique. Train your model again and evaluate its performance. Did you achieve better prediction accuracies with more balanced data?

Part 2: Improved model

This is an open-ended task, allowing you to apply your problem-solving skills to develop a high-performance model. Your goal is to explore various approaches and build an effective solution. For full credit, you must use PySpark to demonstrate your understanding of handling big data in a distributed environment.

- Consider the entire datasets again. Develop an improved classification model that predicts the patient's risk. You should aim for a model with a higher performance while using a maximum of data points. This implies treating missing values differently for example through imputation rather than dropping them. Validate your model and compare its performance with the performance of the model that you built previously.

- Use the K-Means algorithm to cluster your cleansed dataset and compare the obtained clusters with the distribution found in the data. Justify your clustering and visualise your clusters as appropriate.
- Build up local classifiers based on your clustering and discuss how this clusters-based classification compares to your model obtained in the first part of Improved model.
- As in Part1, balance the data and train and test your model with the balanced data.

Marking Criteria

The following areas are assessed:

1. Cleansing, visualizing, and understanding the data. **[30 marks]**
2. Building up and evaluating the predictive model. **[20 marks]**
3. Improved model, Clustering and evaluating cluster-based model. **[20 marks]**
4. Coding style and use of appropriate data structures. **[10 marks]**
5. Writing the report and interpreting the results. **[20 marks]**

Indicative weights on the assessed learning outcomes are given above and can be found in the marking rubric on Blackboard. The marks are given according to the following guidelines.

- **[90 – 100]:** A predictive model with excellent performance, excellent justification and visualisation of the clusters, great insights from the data, and a report of professional standards, while leveraging PySpark for big data analysis.
- **[80 – 90]:** A comprehensive coverage of data cleansing techniques demonstrating an excellent understanding of the data, a sound comparison of the global predictive model against the clusters-based model, appropriate use of PySpark, and a well-structured, maintainable, and robust code usefully using functions.
- **[70 – 80]:** As in Second Upper plus a well-justified predictive model by the data cleansing with good performance using sound evaluation techniques; a well-justified clusters and a concise report on the results obtained from the dataset. Demonstration of some basic understanding of effective use of PySpark for analysing big data.
- **[60 – 70]:** A good coverage of data cleansing techniques exploring the dataset, a good visualisation of the clusters, a predictive model with an appreciable accuracy with a rationale behind it, a working code and a well- structured report on the results obtained from the dataset.
- **[50 – 60]:** Essential data cleansing techniques are covered, a predictive model partially justified with an appreciable accuracy, a working clustering, a partially commented code with very few functions, and a narrative of the findings about the dataset with few deficiencies.
- **[20 – 50]:** Data is not cleaned appropriately. Some important features are ignored. Dataset is too small, and the model is overfitting the data.
- **[0 – 20]:** Code is not complete and is not compiling. Report is incomplete.