COMP2043.GRP Interim Group Report

Team02
Machine Learning Dataset Parsing Tool

Group Members

Boyuan Ma - zy22053 (Team Leader)

Xinyan Li - zy22043

Hao Liu - zy22046

Xinjie Pang - zy22055

Marios Igkiempor - scymi1

Supervisor

Chris Roadknight

December 2018

Contents

1 Description of the Problem

Our team's project is centered around a system that classifies datasets by the best type of machine learning approach to take in order to best analyse the data. It is proposed that the system will be able to parse a dataset, analyse its features and propose a type of learning (supervised, semi-supervised and unsupervised) that can best model the dataset based on the analysed features.

The client intends to use the project as both a prefilter for machine learning and as a teaching aid, and so the program should be able to provide useful information as to how the machine learning approach was derived.

Datasets that the program will analyse will be provided by the client. The possibility of analysing new datasets will be a target that the team will strive for, however it is not essential for the core functionality required by the client.

In essense, the project will take a dataset and tell the user the most optimal machine learning strategy with which to analyse the data.

2 Background Information and Relevant Research

2.1 Machine Learning

The team has to figure out how many machine learning methods would be provided in our website, their respective principles, and how and when to use each one. There is many Test suitability of open source machine learning toolkits such as H2o [1], Weka [2] and various Python and R packages. These machine learning toolkits could be used to analyze different datasets. In this early stage of the project, the team has chosen to use Weka to understand each machine learning method.

2.2 Understanding Machine Learning Algorithms

Section 4.1 lists the machine learning algorithms that the system will be able to produce as output. The team felt it would be beneficial to have a basic understanding of each algorithm and where it is most useful. Some background research was conducted into each algorithm:

Self-Organising Map (SOM)

K-Means Clustering

Principal Component Analysis

Linear Regression

Neural Network

Multiclass Neural Network

Random Forest Regression

Logistic Regression

Multiclass Logistic Regression

Naive Bayesian Network

Support Vector Machine

Self-Training

Forced Clustering

3 Requirements Specification

Following meeting with the client and discussing initial user requirements and system requirements specification, as well as subsequent meeting revising both requirements, the team and the client have agreed on a set of requirements.

3.1 User Requirements

1. Parse datasets and suggests the best machine learning approaches for modeling that dataset
2. The user will supply datasets to be analysed
3. The system will need to be appropriate for use as both a prefilter for machine learning and as a teaching aid.
4. The user requires a degree of data visualization in the front end
5. The user requires a comprehensive, extendible database. The database should allow the user to:
   - 5.1. upload datasets
   - 5.2. modify existing datasets
   - 5.3. provide relevant information about the datasets
6. A rule-based approach will initially be sufficient to assertain what the ideal machine learning approach is for each dataset

7. The client has significant knowledge of machine learning methods and some of this knowledge will be captured to facilitate the decision making process

3.2 Functional System Requirements

1. The system will require a database which can store the data in every dataset
   1.1. The user can upload the dataset
   1.2. The datasets are available for users to download
   1.3. Information about the dataset (what type of data, whether or not there are missing values) can be provided by the user and stored as metadata with each dataset
   1.4. The datasets need preprocessing to find details about the data. These details must be stored as metadata with the dataset. Details will be characteristics of the datasets which help decide the best machine learning approach. Such details could include:
      1.4.1. The type of data
      1.4.2. Size of the dataset
      1.4.3. Number of features
      1.4.4. Number of target outputs provided with the data
      1.4.5. Whether or not there are missing values
      1.4.6. Whether the labels are categories or values
      1.4.7. Complexity of the dataset
      1.4.8. Complexity of relations in the dataset
      1.4.9. Wether or not the dataset is structured
   1.5. The customer needs to be able to add, delete or change information stored in the database
2. The system must be able to analyze a dataset and provide the best machine learning approach to model the dataset
   2.1. The reason why that approach is best needs to be provided to the user
   2.2. The system will have different cataglogues of machine learning. Each cataglogue provides the algorithm for the machine learning and the sample datasets.
   2.3. The system must have a search engine which can search for both the datasets' name and machine learning approaches.
3. The system must provide different types of data visualization
   3.1. Bar charts

3.2. Scatter graphs

3.3. Images

3.4. Interactive visualization at the result page

4. The system should have different mapping opinions for finding the ideal machine learning method:

4.1. Rule-based system

4.2. Deterministic mode

4.3. Probabilistic mode

3.3 Modelling Requirements

Following the elicitation of user and system requirements, the team decided it was useful to model both. We began by making a simple use case diagram, shown in figure 1.

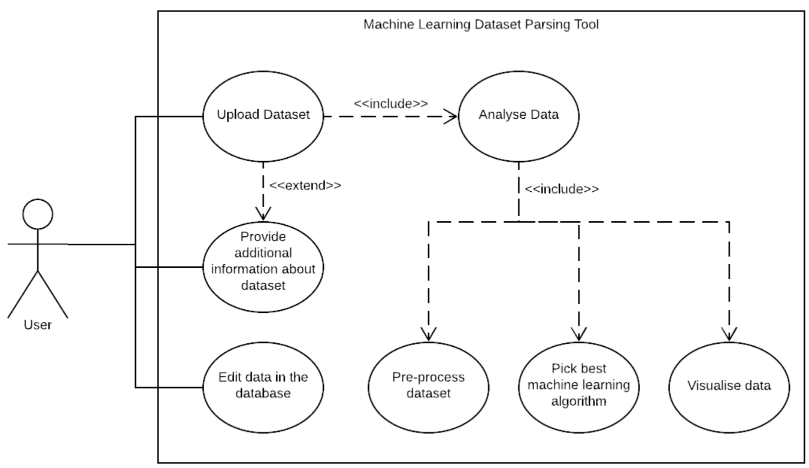

Figure 1: Simple use case diagram of the system

The use case diagram clearly shows how the user will interact with the system. Due to the nature of the system, much of its complexity is hidden from the user and therefore the user does not have many use cases. Regardless, the use case diagram is an important part of the requirements gathering process as it makes it easier for the client to visualise their interactions with the system.

4 Initial Design Ideas

4.1 Picking the Machine Learning Approach

The final goal of the project is to analyze datasets and automatically suggest the best machine learning tools by using our machine learning analyzer. The analyzer will initially be composed of a Decision Tree, which will identify the best machine learning approach based on features of a given dataset.

The short-term goal is to classify dataset into three broad categories: Supervised Learning, Unsupervised Learning and Semi-supervised Learning according to whether there is output data in the dataset. In order to provide a prototype to the client, the system will pick a random machine learning approah. This will let the client have an idea of what a working system would function like.

The team decide to provide 17 potential machine learning tools:

- Multiclass Neural Network
- Linear Regression
- Random Forest Regression
- Sum Regression
- Logistic Regression
- Neural Network
- NB or Sum
- Anti-learning
- Self-Organizing Map
- K-means Clustering
- Principle Component Analysis
- Forced Clustering
- Self-Training
- Deep Learning
- Recurrent Neural Network
- Time Delay Neural Network
- Feature Selection Principal Component Analysis

The best suitable tool will be suggested from these machine learning tools by using decision algorithm in the second iteration of the project, after the prototype.

The website will display the selected most suitable machine learning tool for each dataset uploaded by client. In the long term, the team hopes that the website will be able to analyze the dataset by using the best machine learning tool and visualize the output. However, we are aware that given time constraints, it may not be possible to implement all 17 algorithms, and as such we have agreed with the client that although it would be useful to have, it is not part of the core functionality.

The decision algorithm will be used to determine which is the best analyzing algorithm for datasets. First of all, the algorithm will determine whether the dataset is supervised; according to whether there is output data in the dataset, we divide the dataset into three categories: Unsupervised Learning, Semi-supervised Learning and Supervised Learning, as can be seen in figure 2.

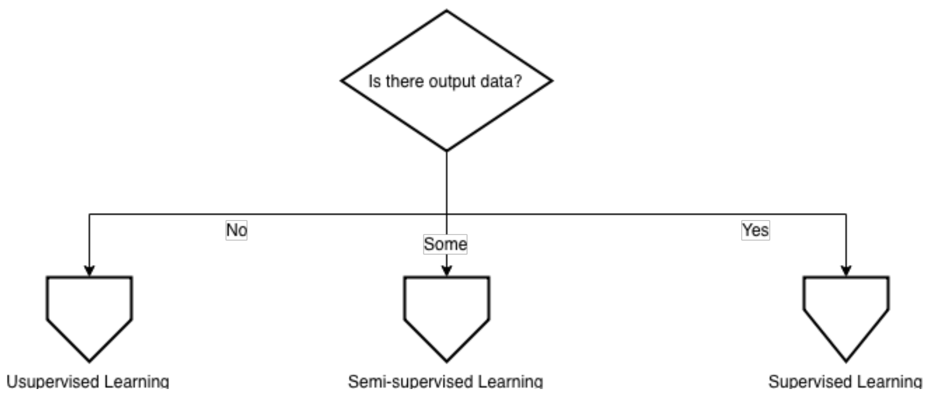

Figure 2: Deciding the class of machine learning to use

If the dataset has no output data, it is classed as an Unsupervised Learning problem. Within Unsupervised Learning, if there are lots of features in the dataset, suggest using a Self-Organizing Map to analyze. Otherwise judge whether the dataset have structure or not. If they have structure, suggest K-means Clustering to analyze, otherwise suggest using Principal Component Analysis. This deicion is modelled in figure 3.

If the dataset is completely labelled with output data, divide it into Supervised Learning. If the outputs are categories divide the dataset to Classification. Else if the outputs are continuous values, divide the dataset to Regression.

If the dataset is best modelled by classification, determine how many categories the dataset contains. If dataset contains 3 or more categories, determine if the dataset is simple or not. For small complex datasets, suggest Multiclass Neural Network or Random Forest Regression. For simple large dataset, suggest using Multiclass Logistic Regression to analyze. If the dataset only contains less than 3 categories, determine whether the relation in the dataset is simple. If

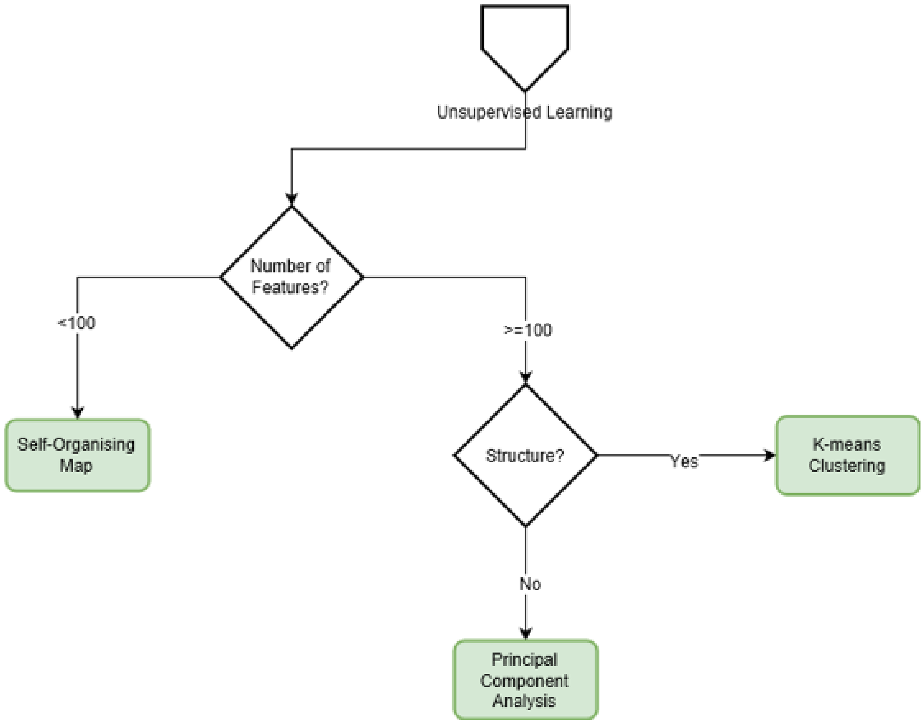

Figure 3: Deciding which unsupervised learning technique to suggest

the dataset has simple relations, suggest Logistic Regression. Otherwise determine how many features in the dataset. If the dataset has less than 100 features, suggest Neural Network, otherwise suggest Nave Bayesian Classifier or Support Vector Machine. If, after modelling the data using a Nave Bayesian Classifier or Support Vector Machine, the output is worse than a guess, suggest Anti-learning.

If the dataset is composed of values (regression), also determine the dataset simple or not, if it is a simple large dataset, suggest Linear Regression. Else if a complex small dataset, suggest Random Forest Regression and Sum Regression. Figure 4 vizualises this set of decisions.

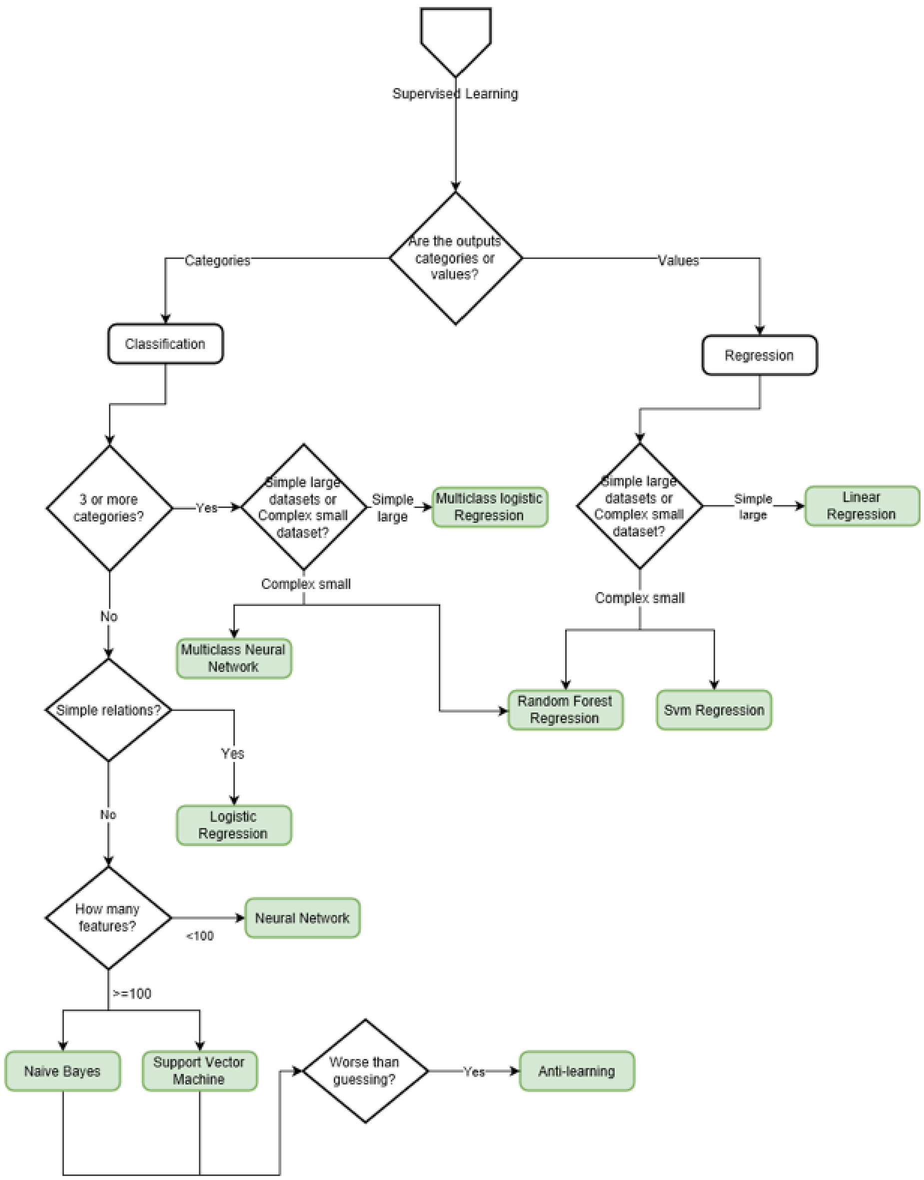

Figure 4: Deciding which supervised learning technique to suggest

5 Key Implementation Decisions

5.1 Database Design

5.1.1 NoSQL vs SQL

Heterogenous big data benefits from NoSQL for a few reasons. Firstly, there is no predefined schema that the data has to conform to, which is useful for our application as we have to save data from a multitude of different data sources in the same data store. We cannot index all the data efficiently beforehand, and therefore cannot define a large schema that all datasets will conform to. Even if we could, a large schema would waste a lot of space, as most datasets would only use a small subset of the schema. This would also make searching the database very inefficient, as we would have to search through potentially hundreds of unused fields.

Removing schemas also makes the database faster to query [3]. Because each dataset is stored in one document rather than spread across multiple tables, the program knows exactly where to look for the data set rather than having to search through multiple tables. This helps when we are having to search through hundred of datasets to find the data set we are interested in.

5.1.2 Lack of Relations Between Datasets

The data sets that our program will use contain almost no relations. As such, using a relational database would waste a lot of the functionality associated with relations, and cause a lot of uneccessary overhead. Instead, a non-relational database will just store data sets independently of each other. Querying the database will simply involve returning a JSON object, with no links to other data sets.

5.2 Accepted File Types in the System

When studying Weka, the team found that there are many file types that machine learning tools could use, for instance Arff data files, CSV data files, XRFF data files, amongst others. After some research into the UCI Machine Learning Repository [4], the team decided to use CSV data files to store datasets into the database and to be used by the machine learning tools. This is because CSV files make it convenient for the team to transform the .data files and .name files found on the UCI repository into a single file. CSV files are also easy to store in the NoSQL database. Knowing the format of the data is useful for data pre-processing and analysing.

6 Problems Encountered

7 Time Planning and Project Management

At the very start of the project, time planning was discussed but was not followed or enforced as much as it should have been. As such, a Gantt Chart was devised and is shown in figure 7. This Gantt chart has helped us stick to deadlines and structure our work.

Alongside the Gantt chart, we have a Kanban board which is hosted on Trello.com [5]. Scrum was discussed as an alternative approach, but Kanban was chosen for some reasons outlined below:

- Teams using Kanban can cope with mutable requirements flexibly. Teamwork will continue with changing project environment.
- Formal and informal meetings can be held as frequently as needed and agile working process can be pushed favorably by team communication.
- Due to the main process of the project having been decided, Kanban is a good system as it allows us to vizualise every stage of work and everyones processing.
- For a team with less development experience, it is difficult to deliver an executable program in a short time. Scrum would be difficult to practice because of its demand on techniques and experience.
- Scrum need to stipulate working time for every iteration, which is difficult for teams with less development experience. Kanban only displays stage missions but not time limits.
- Scrum needs to declare a Product Owner, Scrum manager and Team which might cause confusion in an inexperienced team. Kanban can make team members focus on their work and research.

After agreeing that Kanban would be more suitable to our team's needs, we made a Kanban board on Trello. Trello was chosen as it makes collaboration a lot easier. Being a web based tool, it made it easy for us to manage our workflow from all of our devices, as well as accomodating remote work and management. It also allows team collaboration as it allows people to join the Trello board, which was an essential feature we needed in order to be able to manage our work collaboratively.

7.1 Division of Work

Rather than everyone playing a role in every aspect of the project, we decided it would be easier to divide the work into three distinct subgroups. We decided that the project can clearly be divided into three main sections: *Front-End Development*, *Machine Learning* and *Data Pre-processing and Database Development*. It was decided that Hao and Xinyang would be responsible for Front-End Development, Boyan and Xinjie for Machine Learning and Marios for the Data Processing and Database. The division was made based on both past experience with technologies needed for that area, as well as personal preference. As such, there was no conflict in deciding who would be responsible for what area.

Dividing the project in this fashion allowed the team to be able to divide workload more efficiently, as we could categorise any task into one of the three areas of the project and assign the task to the correct person(s).

Following this division of labour, we decided our Kanban board would be better organised if there was a list for each section. Therefore, we decided to have three lists ("To Do", "Doing" and "Done") for each are of the project, as well as one set of lists for "Other" tasks that had to be done (for example, adding a section to the report, drawing diagrams). This project management workflow kept everything atomic, where everyone was only responsible for a small subset of tasks. It also kept people focused, as everyone could find the correct list for their section and pick up "To Do" tasks that correspond to their section of the project.

References

[1] "Home." [Online]. Available: https://www.h2o.ai/

[2] "Weka 3: Data mining software in java." [Online]. Available: https://www.cs.waikato.ac.nz/ml/weka/

[3] "Mongodb and mysql compared." [Online]. Available: https://www.mongodb.com/compare/mongodb-mysql

[4] "Uci machine learning repository: Flags data set." [Online]. Available: http://archive.ics.uci.edu/ml/datasets.html

[5] "Feature deep dive — getting started with trello." [Online]. Available: https://trello.com/