Data Management: Finding patterns from records of hospital appointment

Huyen Phan

MSc in Computing and Management

Session 2009/2010

The candidate confirms that the work submitted is their own and the appropriate credit has been given where reference has been made to the work of others.

I understand that failure to attribute material which is obtained from another source may be considered as plagiarism.

(Signature of student) ________________________________

Summary

Pattern recognition comprises a set of approaches which are motivated by its impact in the real world. Treatment arrangement is one of the critical issues for almost hospital around the world. There are many hospitals currently manage appointments manually and paper-based. This kind of management requires an excellent scheduling among doctors, nurses, patients and the availability of regimens without overlapping. The given data set is the records of 866 patients along with their treatments for a period of four months from May to August 2008. Due to paper-based management, the historical records of treatments show that some of treatments do not follow any particular rules, in other word, standard pattern. This project presents a critical discussion of the scientific literature on pattern recognition including knowledge discovery process, data mining and pattern recognition methods. In addition, an automatic program to find standard patterns and repair all regimens with their found standard patterns is beneficial.

Acknowledgments

I am heartily thankful to my supervisor, Natasha Shakhlevich, whose encouragement, guidance and support from the initial to the final level enabled me to develop an understanding of the project and successfully complete my dissertation.

Lastly, I offer my regards and blessings to teachers, professors, advisors, librarians, laboratory assistants, classmates, and friends from the University of Leeds as well as the University of Nottingham, who supported me in any respect during the completion of the project.

Huyen Phan

Contents

List of Figures

List of Tables

Chapter 1: Project Outline

1.1 Hospital's treatment process

The hospital usually serves patients as shown in the following model:

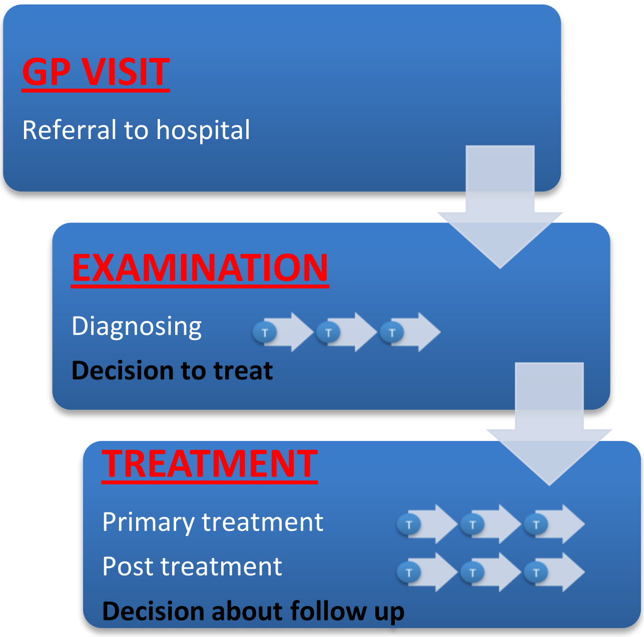


Figure 1: Hospital process of treatment

First of all, a patient would meet a General Practioner (GP) for general diagnose and be referred to a hospital. There are several meetings to examine and diagnose their illness then a treatment schedule will be given. Both the primary treatment and the post treatment are normally conducted over one or more days and spread over a single or several *cycles*. The patient is prescribed one or more than one type of regimen for their treatment. Each regimen has a strict instruction and standard pattern to follow. For example, Table 1 below demonstrates a multiday treatment, which is on the 1st, 8th, 15th and 22nd day of a cycle.

Table 1: Example of multi-day pattern treatment

Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Treatment	T							T							T							T

The patient should strictly follow their multiday pattern with their regimen(s). Traditionally, the hospital will book only one appointment for one patient at a time even if it is known that several visits are needed, which means that the patient will know only one appointment in advance.

The data set in the form of a Microsoft Excel file, is provided by a hospital with information on 866 patients along with the records of their treatments for a period of four months from May to August 2008. Originally, the data set was recorded manually by nurses in hospital's paper diary. Then all of the records over a four-month period were computerised and stored as MS Excel file. There are eight columns corresponding to eight attributes of data:

1. **Patient's name**;
2. **Diagnosis date**: the date when a patient meets a GP to diagnose sickness, and gets a prescription, and a treatment schedule;
3. **Regimen/ drug's name;**
4. **The cycle number**: the number of the current prescription of the multiday pattern;
5. **Day number** of the multi-day pattern related to the cycle;
6. **Appointment date**: The calendar date for treatment accordingly with the day number
7. **The number of days a patient has to wait from decision to treatment until the first visit day**: only few cells in this column have been filled in;
8. **The type of the Multiday/Intraday Pattern**: the name of regimen (drug) used on each treatment day.

However, the data set shows inconsistency in the usages of regimens as shown in the following table containing an extraction of a patient's treatment for three cycles.

Table 2: Example of multi-day pattern treatment for a single patient

Cycle number	Day number of the multi-day pattern related to the cycle	Appointment date
1	1	28/05/2008
1	2	29/05/2008
1	3	30/05/2008
1	6	02/06/2008
1	7	03/06/2008
1	30	26/06/2008
2	1	07/07/2008
2	2	08/07/2008
2	3	09/07/2008
2	4	10/07/2008
2	5	11/07/2008
2	28	03/08/2008

3	1	04/08/2008
3	2	05/08/2008
3	3	06/08/2008
3	4	07/08/2008
3	5	08/08/2008

It is possible to see the inconsistency in regimens used in Table 2; there is no common rule/ pattern across all three cycles. The patient is shown to have had treatment on day number 1, 2, 3, 6, 7, 30 in cycle number 1; day number 1, 2, 3, 4, 5, 28 in cycle number 2; and day number 1, 2, 3, 4, 5 in cycle number 3.

There are various reasons to explain the inconsistencies in the data set; such as the patients not showing up for their appointment, the patients visiting earlier or later than the appointment date, even the patient's state of heath having deteriorated so that nurse has to correct the actual treatment date in hospital's diary.

1.2 Project Aims and Objectives

Due to the inconsistency in historical data on patients' treatments, this project will therefore perform an analysis of the given dataset and describe how to repair it automatically.

The project objectives are:

- Investigate applications of statistics for management to analyse the data set.
- Evaluate and analyse statistical and visualised results, which find the standard patterns for all regiments.
- Summarise applicable methods for pattern recognition and data correction based on research findings.
- Create a computer program to automatically find regimens' pattern
- Produce a pattern database, e.g. an excel file.
- Describe how to repair the dataset

1.3 Minimum Requirements and List of deliverables

The minimum requirement of this project is

1. To produce a report on different techniques applicable to the problem under study.
2. To produce a dataset of standard patterns.
3. A description of the method to automatically repair inaccurate historical data/ records of patients' hospital appointment.

List of deliverables

1. A report on analysis of different data mining algorithms
2. A report on analysis of dataset using statistics tools
3. A repaired dataset with consistence pattern for each regimen.
4. A description of automatic solutions (software) to find standard pattern for each regimen in the given dataset and repair it to be ideal

1.4 Resources required

The Microsoft Excel is the most essential application for this project. I used version 2007 because it supports more statistics tools, well-presented functions, and wider options for graphs and charts. In order to analyze better and provide more convincing evidence, two additional add-ins for Microsoft Excel were installed. They were the Analysis Toolpak add-in and the PhStat2 (Pearson Education Inc., 2010). PhStat2 can be downloaded from the internet or CD attached in a book (Levine et al., 2002). WEKA is a data mining software (The University of Waikato, 2010). It is a professional tool to visualize a huge data set. Its materials can be found online at http://www.cs.waikato.ac.nz/ml/weka/ or module COMP5390M Techniques for Knowledge Management. Python is the main development environment, which is Open Source. The software also can be downloaded online at http://www.python.org/ or module COMP5255M Problem Solving with Computers. The integrated development environment (IDE) used is Eclipse and PyDev is an Eclipse's perspective. The EndNoteX4 software is also used to organise references for this project.

1.5 Proposed research methods

The research methods used for the project were both primary and secondary research methods with personal experimentation for the dataset. I also reviewed related documentation, online materials and literature. The literature reviews included journal articles, research papers, magazine articles, lecture notes and books.

The library of the University of Leeds is an important source of information as it provides not only thousands of books on the computing and business fields but also provides access to online journals and databases. As I am living in Nottingham the City library was also a valuable resource.

The research areas are quite different and therefore have field-specific useful sources. For exploring the concept of data mining and pattern recognition, helpful journals include IEEE (Institute of Electrical and Electric Engineers), Web of Science and online search engine such as Google Scholar, conferences such as the Data Mining, Principles of Data Mining and Knowledge

Discovery which provide many research papers from professionals in the field. Other areas of research such as programming tools, and technical skills for the project were supported by books on development, online programming materials, e.g. Weka and Python and online tutorials on the subjects.

Chapter 2: Project Management

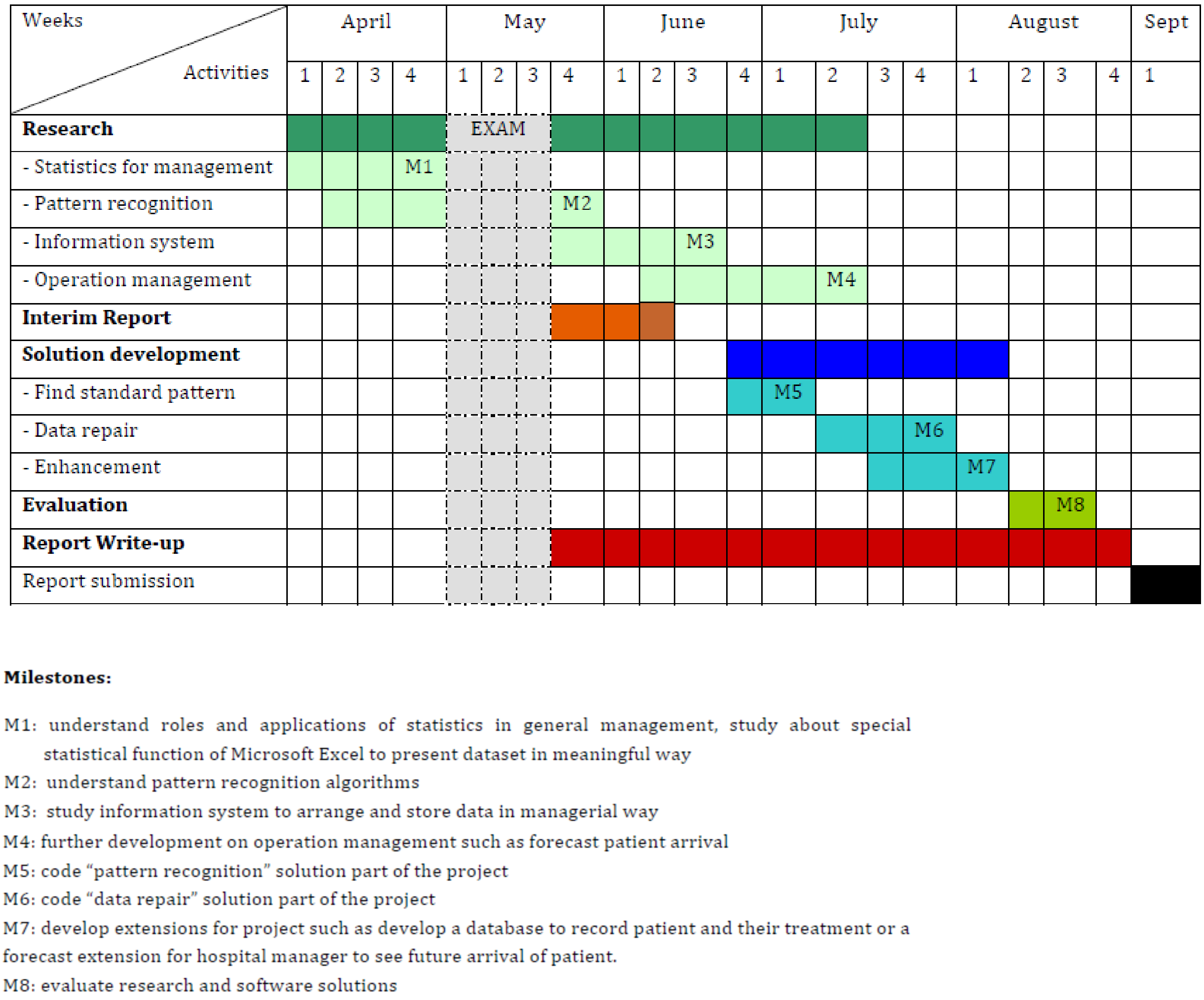
Project Management is an essential element due to the need for responding to the changes in the research environment and project's requirement. In this part the whole process of this project are critically evaluated in order to create profound understanding among the readers on the completion and its effectiveness.

2.1 Project schedule

2.1.1 Initial schedule

To start with, my background learning is commercial management so that it is quite a disadvantage in comparison with other classmate, whose background is computer science or engineering. Therefore, I have struggled for a month to identify which aspect of the given project is the most potential. I have tried to think of as much management idea as possible for this project. In the interim report, the initial schedule was as following:

Table 3: Initial project plan



Weeks / Activities	April				May				June				July				August				Sept
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1
Research	🟩	🟩	🟩	🟩		EXAM		🟩	🟩	🟩	🟩	🟩	🟩	🟩							
- Statistics for management	🟩	🟩	🟩	M1																	
- Pattern recognition		🟩	🟩	🟩				M2													
- Information system								🟩	🟩	🟩	M3										
- Operation management										🟩	🟩	🟩	🟩	M4							
Interim Report								🟧	🟧	🟧											
Solution development												🟦	🟦	🟦	🟦	🟦	🟦				
- Find standard pattern												🟦	M5								
- Data repair														🟦	🟦	M6					
- Enhancement															🟦	🟦	M7				
Evaluation																		🟩	M8		
Report Write-up								🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	
Report submission																					⬛

Milestones:

M1: understand roles and applications of statistics in general management, study about special statistical function of Microsoft Excel to present dataset in meaningful way

M2: understand pattern recognition algorithms

M3: study information system to arrange and store data in managerial way

M4: further development on operation management such as forecast patient arrival

M5: code "pattern recognition" solution part of the project

M6: code "data repair" solution part of the project

M7: develop extensions for project such as develop a database to record patient and their treatment or a forecast extension for hospital manager to see future arrival of patient.

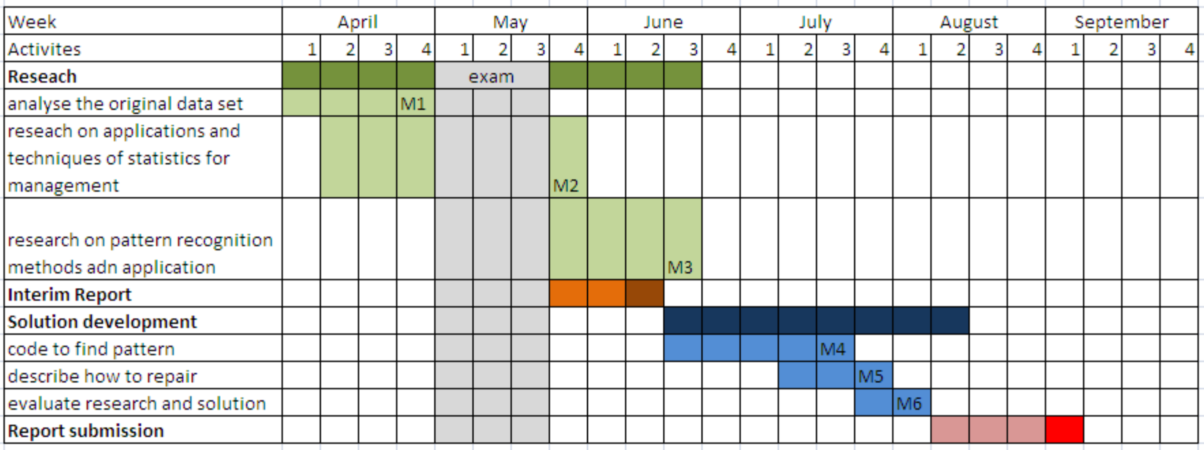
M8: evaluate research and software solutions

The majority of the milestones are under management aspect. I have spent one month (April) to study statistics for management and its techniques. At the same time, I have researched more about pattern recognition. There were so many literatures in this area that confused me. Having discussed with my supervisor, I had clearer idea about the project and its application. At the time of interim report, I have completed milestone M1 and M2 and M3 is half-way reviewed. Thus, I was quite a bit behind the schedule. I am feeling the pressure due to lots of time spending to understand many computing methodologies and algorithms, which are very different to my previous business management background. Writing up report took me more time than expected as I need more time to research and find literatures. However, I have completed in advance the milestone M5 with result as a program to read the excel file and find pattern automatically. In this interim report I have included both my background research chapter and the evaluations of tools/ techniques that I used to visualise the given dataset and find standard pattern. I also included a full analysis for one specific regimen as an illustration for my proposed solution.

2.1.2 Revised schedule

At the end of semester 2, I have been equipped fundamental knowledge especially programming with Python. In the mid-project meeting, my supervisor, my accessor and I have agreed to change the minimum requirements. I absolutely need to push faster in the progress. The revised project plan is as following:

Table 4: Revised project plan



Week	April				May				June				July				August				September			
Activites	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Reseach	🟩	🟩	🟩	🟩	exam			🟩	🟩	🟩	🟩													
analyse the original data set	🟩	🟩	🟩	🟩 M1																				
reseach on applications and techniques of statistics for management		🟩	🟩	🟩				🟩 M2																
research on pattern recognition methods adn application								🟩	🟩	🟩	🟩 M3													
Interim Report								🟧	🟧	🟫														
Solution development											■	■	■	■	■	■	■	■						
code to find pattern											🟦	🟦	🟦	🟦	M4									
describe how to repair														🟦	🟦	M5								
evaluate research and solution																🟦	🟦 M6							
Report submission																		▨	▨	▨	🟥			

Milestones:

M1 analyse the original dataset

M2 report on applications and different techniques of statistics in general management

M3 report on different pattern recognition methods and applications

M4 code to find pattern and export as dataset of regimens' pattern

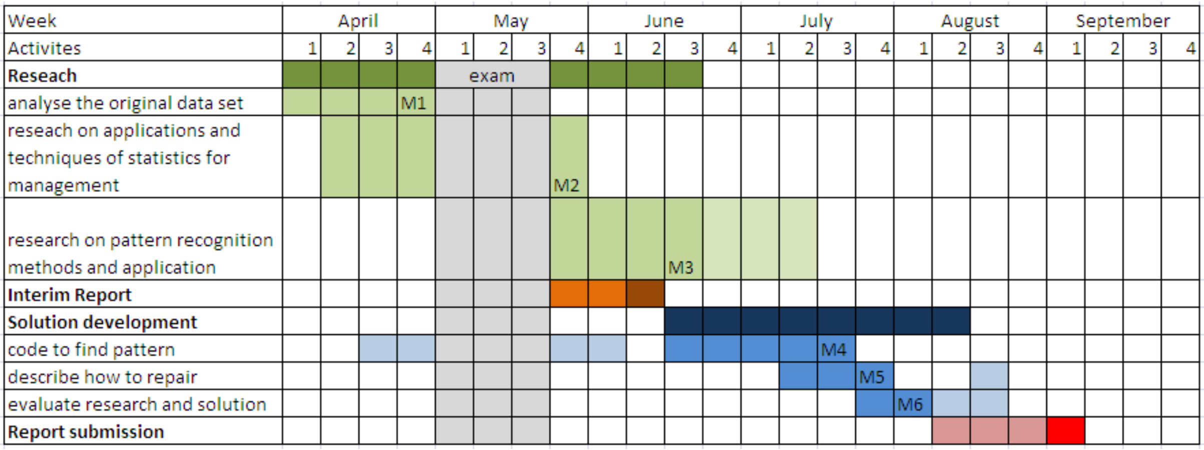
M5 describe how to repair the data set

M6 develop extensions for project. E.g. code “data repair” solution part of the project

M7 evaluate research and software solutions

I have managed to finish all programming tasks and concentrated in writing up the report. Because I have completed the milestone M4 in advance, so that I had the data set of all standard patterns ready for milestone M5. It took me less time (1 week) than expectation (3 weeks) to describe a method to repair the original data set. It also took me only 2 weeks instead of 3 weeks to accomplish milestone M6. The process of writing up is on-going.

Especially for milestone M3, it took me extra 3 weeks to research on pattern recognition methods and application. Based on the reviewed literature, I have clearer understanding of methods and found the most appropriate one to apply in my project. The following table is the actual schedule with highlight the changes.



Week	April				May				June				July				August				September			
Activites	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Reseach	🟩	🟩	🟩	🟩	exam			🟩	🟩	🟩	🟩													
analyse the original data set	🟩	🟩	🟩	M1	▨	▨	▨																	
reseach on applications and techniques of statistics for management		🟩	🟩	🟩	▨	▨	▨	M2																
research on pattern recognition methods and application					▨	▨	▨	🟩	🟩	🟩	M3	🟩	🟩	🟩										
Interim Report					▨	▨	▨	🟧	🟧	🟫														
Solution development					▨	▨	▨				⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛						
code to find pattern			🟦	🟦	▨	▨	▨	🟦	🟦		🟦	🟦	🟦	🟦	M4									
describe how to repair					▨	▨	▨							🟦	🟦	M5			🟦					
evaluate research and solution					▨	▨	▨									🟦	M6	🟦	🟦					
Report submission					▨	▨	▨											🟥	🟥	🟥	🟥			

Overall, I found myself quite behind at the first half. When I understand the project and literatures in depth, I can catch up with the progress much better. It will be on time submission with good quality.

Chapter 3: Project Methodology

3.1 Knowledge Discovery Process and Data Mining

Knowledge Discovery in Databases (KDD) is the method to extract knowledge from databases. There are several research aspects of data mining such as machine learning, pattern recognition, databases, statistics, artificial intelligence, knowledge acquisition for expert systems, and data visualisation. The process of finding and interpreting patterns from a raw dataset is illustrated in Figure 2 (Bulpitt, 2010).

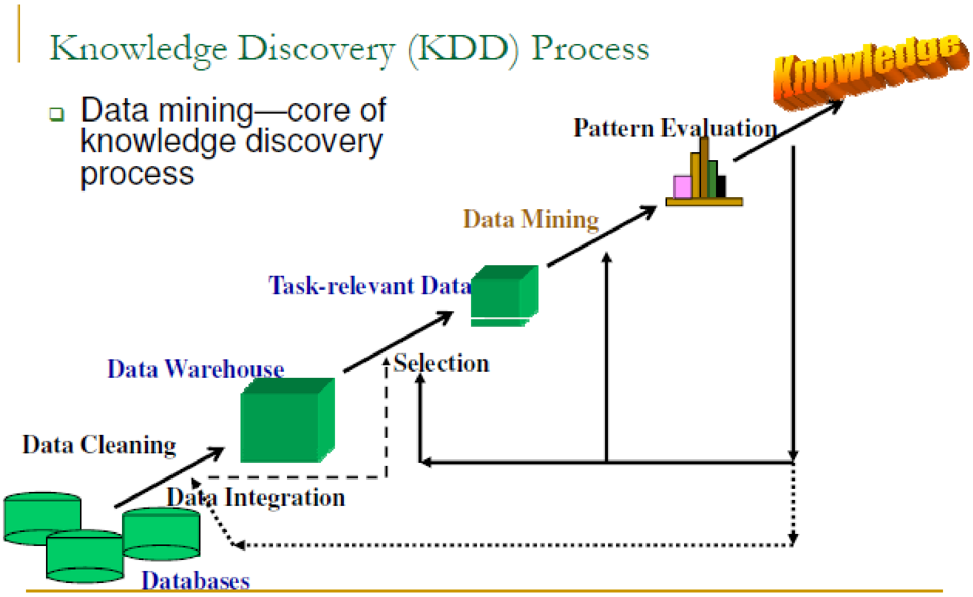


Figure 2: Knowledge Discovery in Databases (KDD) process (Bulpitt, 2010)

The Knowledge discovery process has seven steps (Han and Kamber, 2006):

1. Data Cleaning: to remove noise and inconsistent data. This process often takes 60% of time and effort.
2. Data Integration: where combine multiple data sources
3. Data Selection: where retrieve data relevant to the analysis task from the databases
4. Data Transformation: where transform and consolidate data into forms appropriate for mining
5. Data Mining: where apply intelligent methods in order to extract data patterns
6. Pattern evaluation: to identify the truly interesting pattern representing knowledge based on some measures
7. Knowledge presentation: where present the mined knowledge to the user by visualisation and knowledge representation.

Data is usually “dirty”, which means incomplete, inconsistent, and noisy. The dirty data is caused by human error at data entry, hardware, software problems, different data sources, duplication of records, and data transfer. This explains why it takes the majority of time and effort to clean data by filling in missing values, identifying outliers, smoothing out noisy data, correcting inconsistent data, and removing redundancy (Dasu and Johnson, 2003). There are many ways to handle missing data such as fill in it manually or automatically, or ignore it. It is more complex with noisy data. It can be smoothed by one of four techniques including binning, regression, clustering and combined computer and human inspection.

3.1.1 What is Data Mining?

As shown in Figure 2, data mining is defined as one of the steps in the process of discovering non-trivial and implicit knowledge or patterns from data. It analyses and characterises data, e.g., urban vs. rural areas. It can also be used to discover common patterns, association, and correlations among attributes in databases. The input to a data mining algorithm is in the form of a set of examples, or instances. Each instance has a set of features or attributes, which form *pattern* for that instance (Symeonidis and Mitkas, 2005).

“Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner” (Hand et al., 2001). (Fayyad et al., 2001) also defined “data mining is mechanized process of identifying or discovering useful structure in data”. They use the term *structure* to refer to patterns, models, or relations over the data.

3.1.2 Data Mining tasks

In his book, (Larose, 2005) describes the six most common data mining’ tasks.

- Description
- Estimation
- Prediction
- Classification
- Clustering
- Association

The explanation of each task is following:

Description

This task is to describe patterns and trends within data. The descriptions of patterns and trends in different forms of data interpretation such as graphics, a decision tree, neural networks, and clusters often suggest possible explanations for such patterns and trends. However, for each data set we need to choose the most proper and appropriate method to describe patterns and trends. For instance, decision trees provide a human-friendly explanation of categorical data but are not suitable for nominal data.

Estimation

The target variable for estimation is usually numerical rather than categorical data. The standard models are built using complete records in the past, which provide the predictors and their values. Then, for the new observations, estimates of the value of the target variable are made, based on the values of the predictors. For example, we can estimate the grade-point average (GPA) of a graduate student, based on that student's undergraduate GPA.

The statistical analysis provides several popular estimation methods such as point estimation and confidence interval estimations. Neural networks may also be used for estimation.

Prediction

Prediction is similar to classification and estimation, except that in the case of the prediction, the results lie in the future. For example, predicting the winner of football match, based on a comparison of team statistics. Estimation's methods and techniques may be similar to classification's. Other knowledge discovery methods such as decision tree and k-nearest neighbour are investigated in the later part of this work.

Classification

Classification is appropriate with target categorical data set, which means there are pre-determined categories such as high, medium and low income group. The algorithm would proceed as follows. First, the data set containing both the predictor variables and the (already classified) target variable is examined. The classification software then learns about which combinations of variable are associated with which classes or categories. This data set is called the *training set*. Then the algorithm/ software would look at new records and assign classifications to them based in the classifications in the training set.

For example, we classify a data set of the type of drug a patient would be prescribed, based on certain patient characteristics, such as the age of the patient and the patient's sodium (Na/K ratio). Figure 3 below is a scatter plot of patients' sodium ration against patients' ages for a sample of 200 patients.

The particular drug prescribed is represented by the shade of the points. Light gray points indicate drug Y; medium gray points indicate drug A or X; dark gray points indicate drug B or C. In this scatter plot, Na/K (sodium ratio) is plotted on the Y (vertical) axis and age is plotted on the X (horizontal) axis. The plot gives some recommendations:

- Young patients are on the left in the graph, and high sodium ratios are in the upper half, which indicates that previous young patients with high sodium ratios were prescribed drug Y (light gray points). The recommend prediction classification for such patients is drug Y.
- Patients in the lower right of the graph have been taking different prescriptions, indicated by either dark gray (drugs B or C) or medium gray (drug A or X). Without more specific information, a definitive classification cannot be made here.

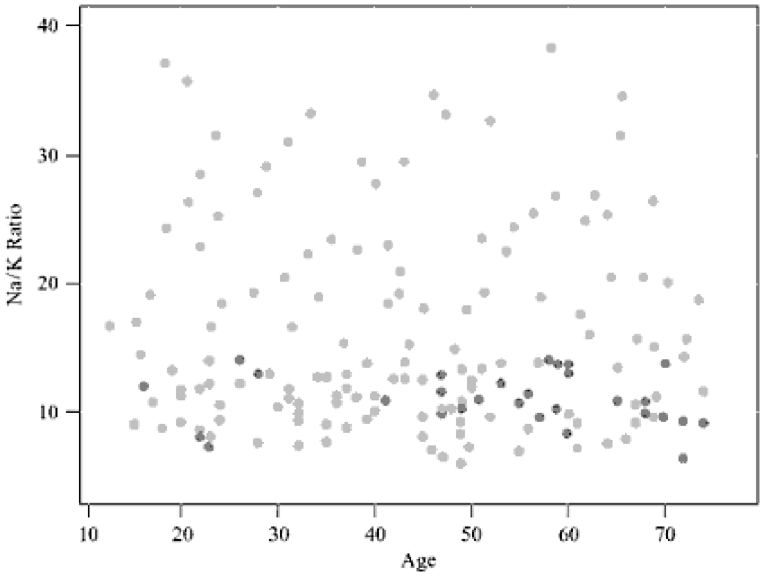


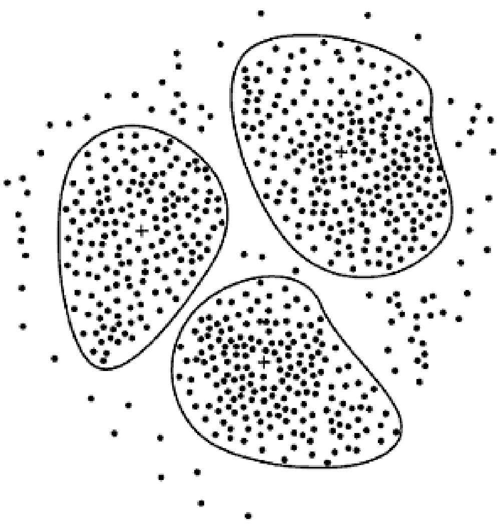
Figure 3: Scatter plot example of Na/K (sodium ratio) vs. the age of patients (Larose, 2005)

Among classification methods, one method can be applied in this project. Instead of trying to create rules, correct directly from the example themselves. This is known as *instance-based* learning. In instance-based learning, each new instance is compared with existing ones using a distance metric, and the closest existing instance is used to assign the class to the new one. This is called the *nearest neighbour* classification method. Sometimes more than one nearest neighbour is used; the largest estimated probability class is assigned to the new instance. This is called the *k-nearest-neighbour* method. This method is straightforward for numeric attributes, when the standard Euclidean distance is used. (Witten and Frank, 2005, Hand et al., 2001).

Clustering

Clustering techniques apply when there is no class to be pre-determined but rather when the instances are to be divided into natural groups. Clustering naturally requires different techniques for the classification and association learning methods. There are different ways in which the result of clustering can be expressed. The groups that are identified may be exclusive so that any instance belongs in only one group. Or they may be overlapping so that an instance may fall into several groups. Or they may be probabilistic, whereby an instance belongs to each group with a certain probability. Or they may be hierarchical, such that there is a crude division of instances into groups at the top level, and each of these groups is refined further.

Clustering is often performed as preliminary step in a data mining process, with the resulting clusters being used as further inputs into a different technique downstream, such as neural networks.



A 2-D plot of customer data with respect to customer locations in a city, showing three data clusters. Each cluster centroid is marked with a "+", representing the average point in space for that cluster. Outliers may be detected as values that fall outside of the sets of clusters.

Figure 4: Clustering method for data smoothing (Han and Kamber, 2006)

Association

The association task for data mining is to find the relationship among attributes. The relationships between two or more attributes are quantified by uncover rules. Association rules are of the form "if something happens, then consequent thing happens" together with a measure of the support and confidence associated with rule.

3.2 Estimation and Prediction Approaches

3.2.1 Statistics for Management

It is essential for managers to get information from large amounts of data for better decision - making. One of the most important issues that managers must deal with is how to effectively use this information when making decisions. Here they face three problems:

- With computerized management information systems and a large number of databases widely available on the web, managers often feel there is too much evidence available. If they are to use evidence when making decisions they must process or transform this data into manageable units which we call information.
- Managers must however do more than passively process available data. They must also actively redesign management information systems to ensure that more relevant data is generated.
- Their most important responsibility is to communicate to others in their organization the need for these activities and the reasons why they have made the decisions based on this evidence.

In order to fully investigate the meaning of data, managers need to learn statistics and have statistical thinking. It helps to present large dataset in informative ways such as graphs, charts; to conclude meaning of large populations based on smaller samples; to improve processes; and to forecast the development trend of the dataset (Levine et al., 2002).

There are two main types of statistics: *descriptive statistics* and *inferential statistics*. Descriptive statistics focuses on collecting, selecting, summarising to present data and describe features of dataset. One of the descriptive statistics' tools is graphical technique, which allows user to present the data in such a useful format such as line chart, bar chart, pie chart, scatter plot, etc. Another tool is numerical technique, which summarise data. For example, it calculates average, mean, mode, standard deviation, etc. Thus, the reader can understand the data better. Inferential statistics is another branch of statistics, which draw characteristics of a huge population based on smaller sample. For instance, instead of interviewing 5,000 employees for their satisfaction toward company's HR policies, they can better interview 500 employees from all departments to have estimated conclusion (Lind et al., 2006)

Statistics has incredible applications in all industries of the economy. First of all, financial management is the function of every business. It provides more precise decisions in providing products and services, e.g. capital budgeting, working capital management, sales forecasts, and revenue estimation. Secondly, marketing management also need statistics for development of products and services by positioning, market segmentation to see profitable segment to invest.

Thirdly, operations management including logistics, process planning, and inventory management also need statistics to determine optimum output and to forecast the amount of required input for production (Keller and Warrack, 2004).

3.2.2 Using Microsoft Office Excel

Microsoft Office Excel is very powerful application for statistics and visualise data. The graphical procedures are used to organise and present data. One of the most popular tools in MS. Excel is to draw Charts, which includes bar chart, line chart, pie chart, column chart, area chart, scatter plot, and so on. It makes very easy to visualise data in informative way.

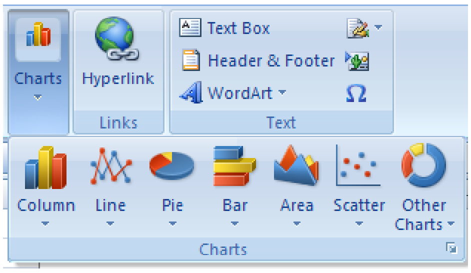


Figure 5: Microsoft Office Excel Charts Insert

The Analysis ToolPak is a MS. Excel add-in program. It includes many analysis tools such as Descriptive Statistics, Histogram, Correlation

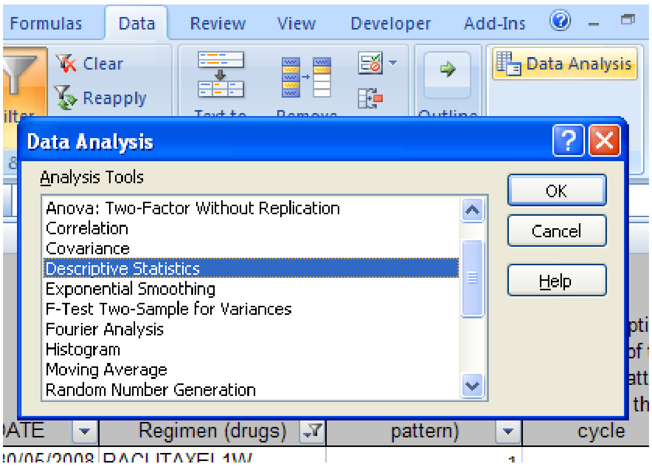


Figure 6: Data Analysis ToolPak Add-in program

3.2.3 Using PhStat2

PhStat2 is a statistical add-in for MS. Excel. It helps produce many procedures not included in standard MS. Excel. Two techniques which will be used in this report are Frequency Distribution and Stem-and-Leaf Display. A frequency distribution counts the number of observations that fall into each of a series of intervals, called **classes** that cover the complete range of observations. A Stem-and-Leaf split observation into two parts: a stem and a leaf. There are several ways to slit up. For example, the number 12.3 can be split into a stem "12" and a leaf "3". This way divides number by the decimal. 12.3 can also be split into a stem "1" and a leaf "2". This definition the stem is the number of tens and the leaf is the number of ones. By using stem-and-leaf display, the given dataset can be clustered and we can see very clearly the popularity of each class or stem.

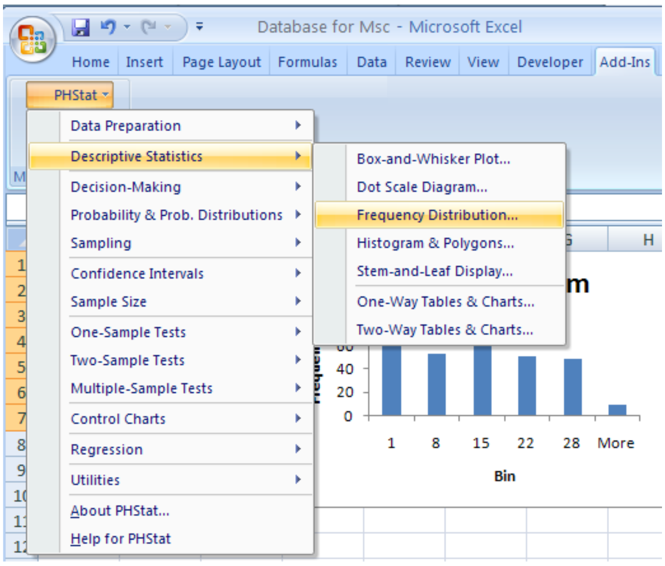


Figure 7: The PhStat menu

3.3 Classification approaches

Data mining methods may be categorized as either supervised or unsupervised. The supervised classification in which the input pattern is a member of a predefined class, unsupervised classification is the opposite (e.g., clustering) in which the pattern is assigned to an unknown class. In unsupervised methods, no target classes are classified in advance. Instead, the data mining algorithm searches for patterns and structure among all the variables. The most common unsupervised data mining method is clustering. This project's data set is one example of unsupervised data set; when the standard patterns for all regimens are unknown. Another data mining method, which may be supervised and unsupervised, is association rule mining (Jain et al., 2000). For example, supermarket may be interested in "which items are purchased together". In fact, there are so many items for sale, that searching for all possible combinations. Most data mining methods are supervised methods, including decision trees, neural networks, and *k*-nearest neighbours (Larose, 2005).

In our case, the data set is unsupervised so that classification is less applicable than clustering.

3.4 Clustering approaches

(Larose, 2005) described the classic clustering technique, called *k*-means. First, the parameter k is specified in advance how many clusters are being sought. Then k points are chosen at random as cluster centres. All instances are assigned to their closest cluster centre. Next the centroid, or mean, of the instances in each cluster is calculated. These centroids are taken to be new centre values for their respective clusters. Finally, the whole process is repeated with the new cluster centres. Iteration continues until the same points are assigned to each cluster in consecutive rounds, at which stage the cluster centres have stabilized and will remain the same forever. The process can be summarized in the table below:

Table 5: The k-means algorithm proceeds (Larose, 2005)

Step 1: Ask the user how many clusters *k* the data set should be partitioned into.
Step 2: Randomly assign *k* records to be the initial cluster centre locations.
Step 3: For each record, fid the nearest cluster centre. Thus, in a sense, each cluster centre "owns" a subset of the records, thereby representing a partition of the data set. We therefore have *k* clusters, $C_1, C_2, \ldots, C_{k.}$
Step 4: For each of the *k* clusters, find the cluster *centroid*, and update the location of each cluster centre to the new value of the centroid.
Step 5: Repeat steps 3 to 5 until convergence or termination.

The "nearest" criterion in step 3 is usually Euclidean distance. The cluster centroid in step 4 is found as follows. Suppose that we have *n* data points $(a_1, b_1, c_1), (a_2, b_2, c_2), \ldots , (a_n, b_n, c_n)$, the *centroid* of these points is the centre of gravity of these points and is located at point $(\sum a_i/n, \sum b_i/n, \sum c_i/n)$ (Larose, 2005). For example, the points (1,2,3), (1,1,1), (1,3,2), and (2,2,1) would have centroid

$$\left(\frac{1+1+1+2}{4}, \frac{2+1+3+2}{4}, \frac{3+1+2+1}{4}\right) = (1.25,2.00,1.75)$$

The algorithms terminates when the centroids no longer change. In other words, the algorithm terminates when for all clusters $C_1, C_2, \ldots , C_k$, all the records "owned" by each cluster center remain in that cluster. Alternatively, the algorithm may terminate when some convergence criterion is met, such as no significant shrinkage in the *sum of squared errors (SSE):*

$$SSE = \sum_{i=1}^{k} \sum_{p \in C_1} d(p, m_i)^2$$

Where $p \in C_i$ represents each data point in cluster *i* and m_i represents the centroid of cluster *i*. An example of *k*-means clustering at work is provided in Larose's book (2005).

This clustering method is simple and effective. It is easy to prove that choosing the cluster centre to be the centroid minimizes the total squared distance from each of the cluster's points to its centre. Once the stabilized, each point is assigned to its nearest cluster centre, so the overall effect is to minimize the total squared distance from all points to their cluster centres. But the minimum is a local one; there is no guarantee that it is the global minimum.

3.4.1 The WEKA machine-learning workbench

WEKA is data mining tool including a collection of machine learning algorithms. A data set can either apply directly the algorithms or called from the Java code. WEKA supports well data, which is in form of ARFF (Attribute-Relation File Format) files or CSV (Comma-separated values) file.

WEKA is open source software written in Java. The software is very useful for research, education, and applications. The WEKA explorer has six tabs for each of its tools (Witten and Frank, 2005).

- Pre-process. Choose and modify the data being acted on.
- Classify. Train and test learning schemes that classify or perform regression.
- Cluster. Learn clusters for the data.
- Associate. Learn association rules for the data.
- Select attributes. Select the most relevant attributes in the data.
- Visualize. View an interactive 2D plot of the data.

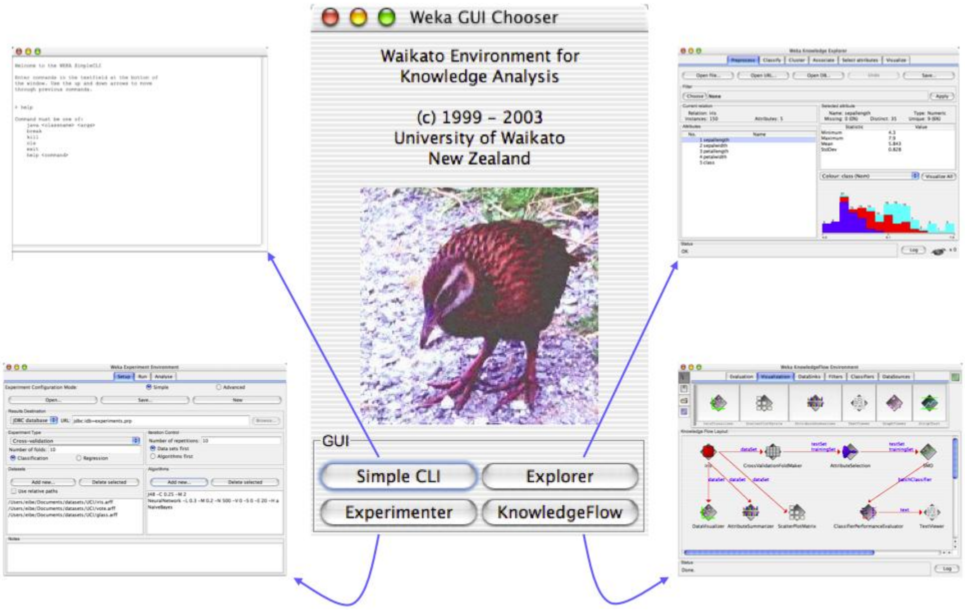
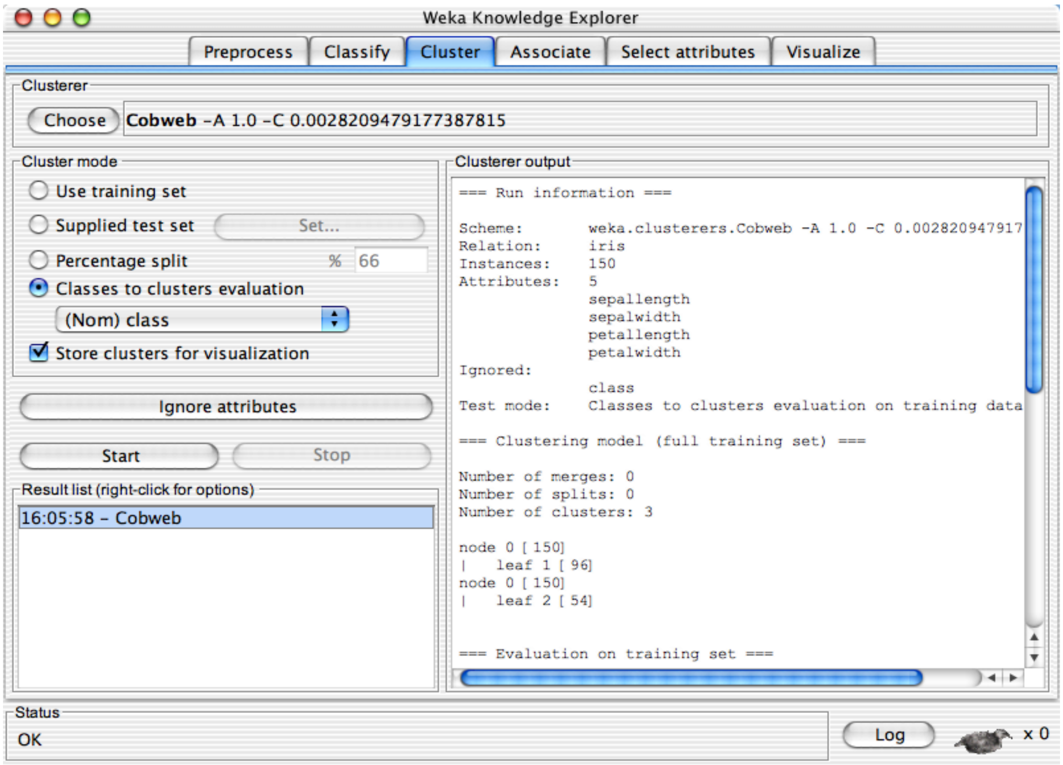


Figure 8: The WEKA Knowledge explorer (The University of Waikato, 2010)

The most useful tool for this project is Cluster. The Cluster mode box is used to choose what to cluster and how to evaluate the results. The first three options are the same as for classification: **Use training set, Supplied test set and Percentage split** – except that now the data is assigned to clusters instead of trying to predict a specific class. The fourth mode, **Classes to clusters evaluation**, compares how well the chosen clusters match up with a pre-assigned class in the data.



Clusterer: **Cobweb** -A 1.0 -C 0.0028209479177387815

Cluster mode
- ○ Use training set
- ○ Supplied test set
- ○ Percentage split % 66
- ● Classes to clusters evaluation
  - (Nom) class
- ☑ Store clusters for visualization

Result list (right-click for options): 16:05:58 - Cobweb

Clusterer output

```
=== Run information ===

Scheme:       weka.clusterers.Cobweb -A 1.0 -C 0.002820947917
Relation:     iris
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
Ignored:
              class
Test mode:    Classes to clusters evaluation on training data

=== Clustering model (full training set) ===

Number of merges: 0
Number of splits: 0
Number of clusters: 3

node 0 [150]
|   leaf 1 [96]
node 0 [150]
|   leaf 2 [54]


=== Evaluation on training set ===
```

Status: OK

Figure 9: The Clustering tool in WEKA (The University of Waikato, 2010)

An additional option in the Cluster mode box, the Store clusters for visualization tick box, determines whether or not it will be possible to visualize the clusters once training is complete. When dealing with datasets that are so large that memory becomes a problem it may be helpful to disable this option.

The ignoring attributes: Often, some attributes in the data should be ignored when clustering. The Ignore attributes button brings up a small window that allows you to select which attributes are ignored.

There is Start/Stop buttons, a result text area and a result list. These all behave just like their classification counterparts. Right-clicking an entry in the result list brings up a similar menu, except that it shows only two visualization options: Visualize cluster assignments and Visualize tree.

In Cluster mode there are five clustering algorithms: SimpleKMeans, EM, Cobweb, MakeDensityBastedClusterer, FarthestFirst. Table 6 lists out five Weka's clustering algorithms.

Table 6: Clustering algorithms (Witten and Frank, 2005)

Name	Function
EM	Cluster using expectation maximization
Cobweb	Implements the Cobweb and Classit clustering algorithms
FarthestFirst	Cluster using the farthest first traversal algorithm
MakeDensityBasedClusterer	Wrap a clusterer to make it return distribution and density
SimpleKMeans	Cluster using the *k*-means method

Cobweb implements both the Cobweb algorithm (Fisher, 1987)for nominal attributes and the Classit algorithm for numeric attributes (Gennaria et al., 1990). *FarthestFirst* implements the farthest-first traversal algorithm of (Hochbaum and Shmoys, 1985). *MakeDensityBasedClusterer* is a meta-clusterer that wraps a clustering algorithm to make it return a probability distribution and density.

SimpleKMeans clusters data using *k*-means clustering algorithm (Section 3.4) in which *k* initial points are chosen to represent initial cluster centers, all data points are assigned to the nearest one; the mean value of the points in each cluster is computed to form its new cluster center, and the iteration continues until there are no changes in the clusters.

How can we choose the number of clusters? Suppose you are using k-means but do not know the number of clusters in advance. One solution is to try out different possible combination and see which is best that is, which one minimizes the total squared distance of all points to their cluster center. A simple strategy is to start from a given minimum, perhaps k=1, and work up to a small fixed maximum, using cross-validation to find the best value. Another possibility is to begin by finding a few clusters and determining whether it is worth splitting them. You could choose k=2, perform k-means clustering until it terminates, and then consider splitting each cluster. Computation time will be reduced considerably if the initial two-way clustering is considered irrevocable and splitting is investigated for each component independently. One way to split a cluster is to make a new seed, one standard deviation way from the cluster's center in the direction of its greatest variation, and to make a second seed the same distance in the opposite direction. Then apply k-means to the points in the cluster with these two new seeds (Witten and Frank, 2005).

Chapter 4: Analysing data

4.1 Identifying erroneous data

In the given initial data set, Saturday and Sunday are supposed to be no working days in the hospital. However, the actual dataset shows records on Saturday and Sunday, 0.48% and 6.47%, respectively.

Table 7: Statistics of all records divided by 7 days of a week

Date	No. of Records	Corresponding Percentage
Monday	769	15.50%
Tuesday	1066	21.49%
Wednesday	1151	23.20%
Thursday	983	19.81%
Friday	647	13.04%
Saturday	24	0.48%
Sunday	321	6.47%
	4961	100%

In addition, it is noticeable that the peak during a week with the highest records is on Wednesday with 23.20% of the dataset. The further analysis may be useful when we have more information about actual hospital activities.

4.2 Identifying the unique records

Studied the given dataset, I have found some special cases:

1. There are some regimens having only one record with one patient having one cycle, and one date of treatment. For such regimens, there are no other records to compare, so that those regimens are considered to have one-day pattern.

Table 8: List of all regimens having only one record

Regimen (drugs)	Treatment day number related to a cycle
AN3ELDOP	1
AN4	1
BLOOD	1
CAV	1
CETUX/IL21(PH1)	1
CIBCPPS(IBAND)	84

CISPLAT+5FU(OP)	1
GATEARMCTEXL1	21
GOSERELINLA10.8	84
PICCOLOIRPAN	21
SORCE	1
TAMOXIFENGYNMT	56
TIPMODIFIED	21
VINOR+EPI	14

The second column in Table 8 is the day number related to a cycle, which mean

2. Some regimens have two recorded rows.

For example, regimen, named CARBO/5FU, has two rows recorded for one patient, who has two different cycles and only one treatment date in each cycle. It is illustrated in the table below:

Patient's name	Diagnose date	Regimen (drug)	Cycle no.	Day no. in cycle	Calendar date
tptcmtprdd	12/05/2008	CARBO/5FU	1	1	14/05/2008
tptcmtprdd	12/05/2008	CARBO/5FU	4	1	17/07/2008

However, there are different cases. The regimens named: CARBO/5FU(OP); E/CARBO/F; GEMTREO; IBANDRONICACID; PACLITAXEL3W; TEMOZOL1STDOSE; TRASTUZ1WLD; TRASTUZUMAB1W; VINOREL(BREAST); and ZALUTETRIAL have the same cycle number but two different treatment dates. We can assume that these regimens having 2-day treatment pattern.

Table 9: List of all regimens having two records

Regimen (drugs)	Cycle number	Treatment day related to a cycle number respectively
CARBO/5FU(OP)	1, 1	1, 2
E/CARBO/F	1, 1	1, 21
GEMTREO	1, 1	1, 1
IBANDRONICACID	1, 1	1, 28
PACLITAXEL3W	4, 4	1, 21
TEMOZOL1STDOSE	1, 1	1, 28
TRASTUZ1WLD	1, 1	1, 7
TRASTUZUMAB1W	1, 1	1, 10
VINOREL(BREAST)	1, 1	1, 8
ZALUTETRIAL	2, 2	1, 1

DARBE5004W	1, 2	28, 1
FEC75	4, 5	21, 1
SU011248	21, 22	42, 1
CARBO/5FU	1, 4	1, 1
VINCRISTINEQ1W	1, 2	1,1

The regimens named DARBE5004W; FEC75; and SU011248 have two different cycles and treatment date, so they are two-day treatment for a cycle. The CARBO/5FU and VINCRISTINEQ1W have two different cycles but the same treatment dates, therefore, they are one-day treatment regimen.

4.3 Pattern representation

Instead of representing the pattern for PACLITAXEL1W as a list of treatment dates in a cycle,

e.g. 1, 8, 15, 22, 28 (1)

We can represent it as a list of time lags between two treatment dates

x days – y days – z days (2)

with x, y, z are the numbers of days between two treatments

For instance,

Table 10: Different way to represent pattern

Date of treatment	1	8	15	22	28
Pattern	**6 days**	**6 days**	**6 days**	**5 days**	

We can also display the pattern as

T, T+a, T+b (3)

with T is the first treatment date, a and b are the time lags between two treatments

For example, the pattern for PACLITAXEL1W is T, T+7, T+14, T+21, T+27

It is discursive to decide which is the most appropriate to represent standard pattern. The simplest way is just treatment date itself as (1). However, for example, if another patient's record shows pattern is 2, 9, 16, 23, 29 for PACLITAXEL1W. There is nothing wrong with this pattern except the dates (each element in this pattern is later than standard list 1 day). The time-lag between two treatments still is the same with 1, 8, 15, 22, 28; and that does not affect quality of patient's treatment. Representation (2) and (3) are quite similar to each other, just based on time between two treatments.

Nevertheless, method (3) is more complex and requires extra effort to calculate the pattern. In conclusion, representation (1) and (2) are more preferable to use but (1) will be the final representation in repaired dataset.

Chapter 5: Generating a data set of standard patterns

5.1 Statistical approaches to estimate and predict

It is necessary to analyse the original dataset to find out its characteristics, features and patterns. Taking CARBO(AUC)21D regimen (drug) as an example, Microsoft Excel's analysis tool (Section 3.2.2) has helped to filter and visualize all data for this regimen (drug).

Each blue point in Figure 10 represents **a treatment date**, when **a patient** visited the hospital for he/she treatment with **a prescribed regimen (drug)**. Figure 10 shows very clearly CARBO(AUC)21D's pattern. All points in this figure have value "1" and "21", which means that CARBO(AUC)21D's pattern is likely day number 1 and day number 21 in a cycle. However, it is noticeable that there are some blanks among points, which are missing data and that need to be filled in the repaired dataset.

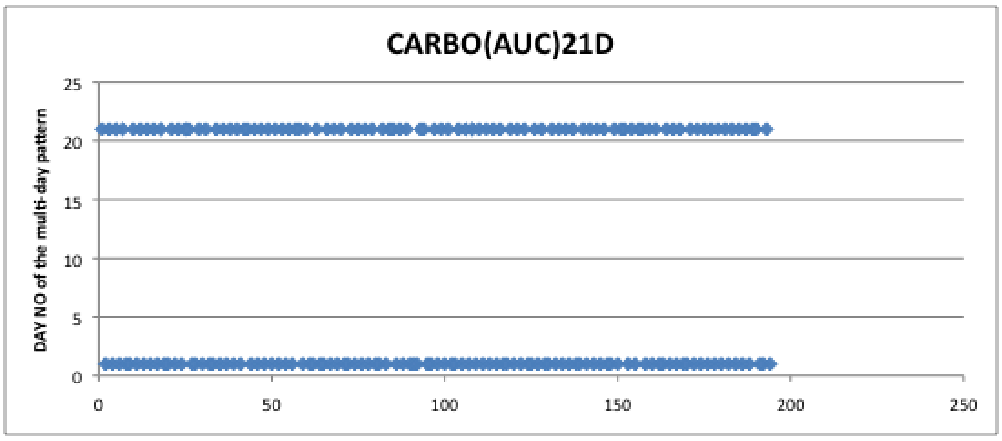


Figure 10: Scatter Plot for CARBO(AUC)21D

Another regimen (PACLITAXEL1W) can be analysed using the same technique with CARBO(AUC)21D. Figure 11 is a scatter plot for the given data of regimen PACLITAXEL1W. It is possible to see that there are four groups of points, which appear mostly at values 1, 8, 15, 22, and 28. However, unlike CARBO(AUC)21D's figure with consistent appearance of values 1 and 21, PACLITAXEL1W's figure does not have equal values. There are some noisy data around defined groups except for the group with value "1". The overall observation shows values 1, 8, 15, 22, 28 as PACLITAXEL1W's standard pattern.

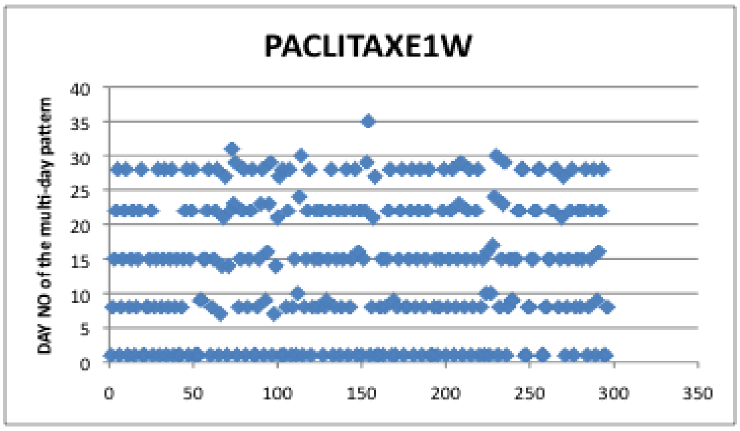
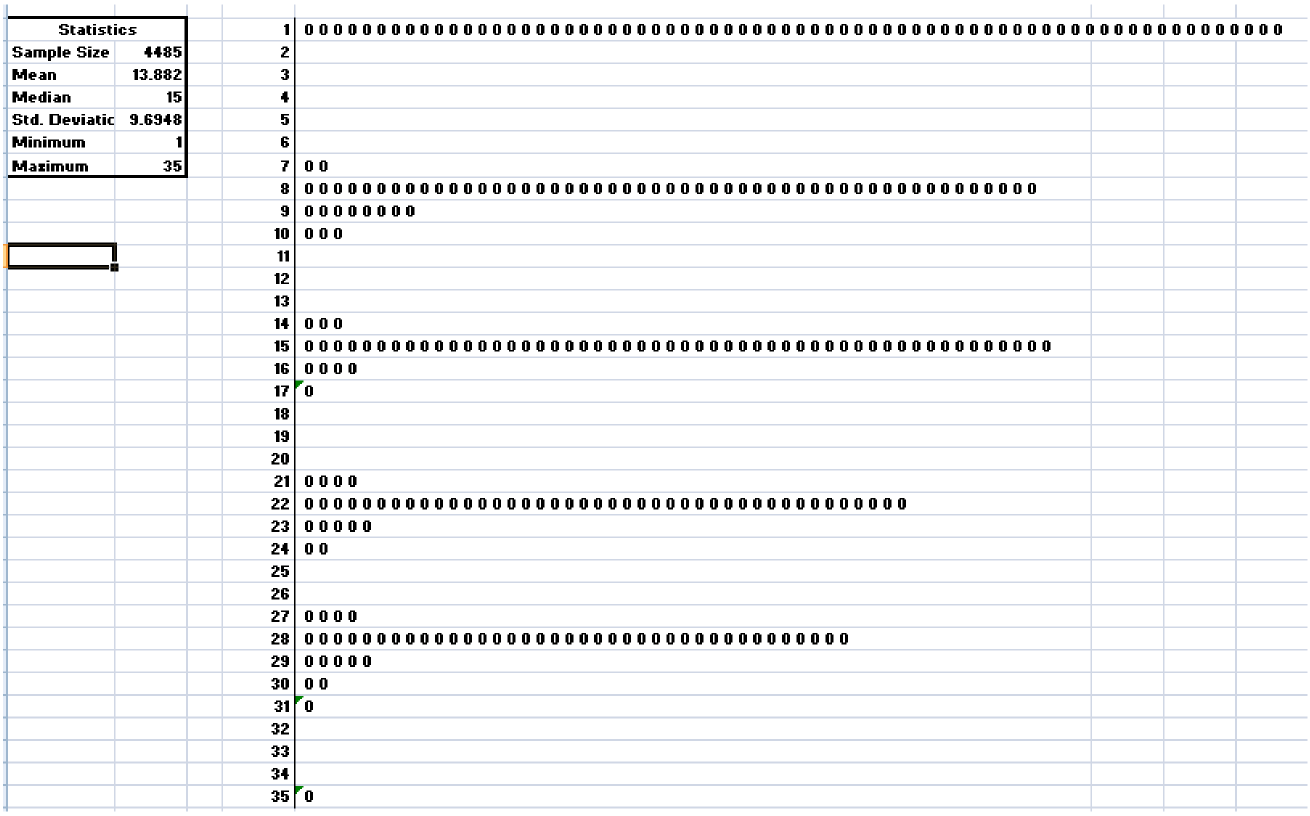


Figure 11: Scatter Plot for PACLITAXE1W

The PhStat2 application (Section 3.2.3) was used to do further analysis. One of the techniques to visualize is the Stem-and-Leaf display function in PhStat2. This technique helps to automatically list all treatment dates in a table, and represent each appointment as "**0**". It also shows that 1, 8, 15, 22, 28 is the pattern of PACLITAXEL1W; when the "**0**" appears almost at value 1, 8, 15, 22, 28. Figure 12 below demonstrates more clearly variations in PACLITAXE1W's data. The percentage of variation (points having values not equal to any of 1, 8, 15, 22, 28) is *45/296 (15.2%).*



Statistics	
Sample Size	4485
Mean	13.882
Median	15
Std. Deviatic	9.6948
Minimum	1
Maximum	35

1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
2	
3	
4	
5	
6	
7	0 0
8	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
9	0 0 0 0 0 0 0 0
10	0 0 0
11	
12	
13	
14	0 0 0
15	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
16	0 0 0 0
17	0
18	
19	
20	
21	0 0 0 0
22	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
23	0 0 0 0 0
24	0 0
25	
26	
27	0 0 0 0
28	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
29	0 0 0 0 0
30	0 0
31	0
32	
33	
34	
35	0

Figure 12: Stem-and-Leaf Display for PACLITAXE1W

Beside the Stem-and-Leaf Display in Figure 12, there is a summary of main statistics number for PACLITAXE1W regimen. In our case, the statistics for one regimen is not quite useful.

There are 296 rows recorded for PACLITAXE1W data in the given dataset. They are now grouped in five "bins", which have been previously determined as possible pattern of this regimen. Frequencies Distribution function in PhStat2 has calculated the frequencies of those five bins including value 1, 8, 15, 22 and 28+ (all points greater than 28).

Table 11: Frequencies Distribution for PACLITAXE1W for bins (1, 8, 15, 22, 28+)

Frequencies for PACLITAXE1W			
Bins	*Frequency*	*Percentage*	*Average value of points in the bin*
1	68	22.97%	1.000
8	53	17.91%	8.188
15	66	22.30%	15.017
22	51	17.23%	22.094
28+	58	19.59%	28.294

The average values for bins rounded are **1, 8, 15, 22, 28**. Those average values are the same as the prediction of PACLITAXEL1W's pattern by previous observation.

We can conclude now that the standard pattern for PACLITAXE1W regimen is **1, 8, 15, 22, 28**.

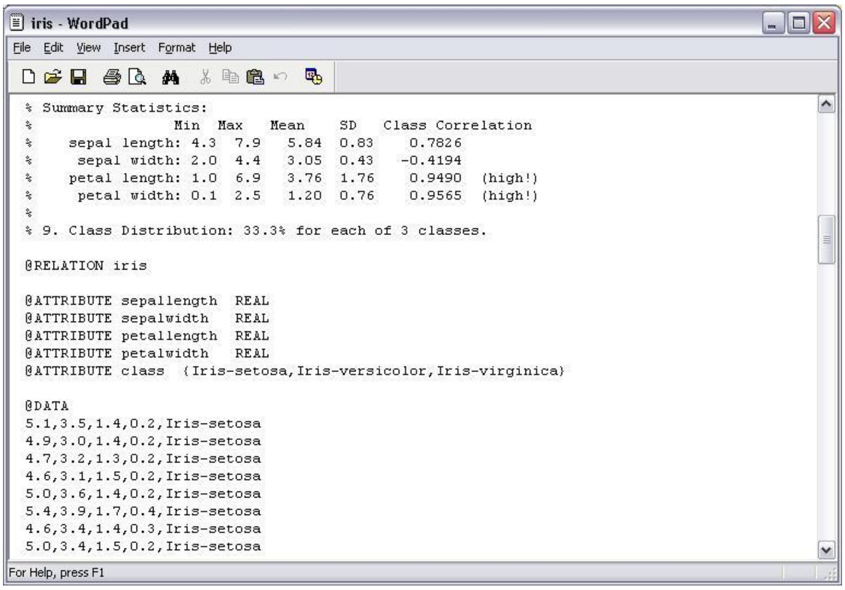
5.2 Using WEKA to cluster the data set

In Section 3.4.1, we have discussed about the WEKA and its functions. As mentioned, classification method is not quite applicable for our data set, which are unsupervised, clustering is implemented (Jain et al., 2000).

5.2.1 Preparing the data

In order to have pattern analysis of one particular regimen, the first step is to filter the original data set for all records to view only that regimen. Then a new spreadsheet is created to contain only records of that regimen. The most important attribute is *the day number of the multi-day pattern related to the cycle,* so that other attributes can be deleted. In some cases, the cycle number is useful; therefore, this attribute can be kept. The PACLITAXEL 1W regimen is chosen for experiment in the WEKA.

The data is given in a spreadsheet. However, WEKA's native data storage method is ARFF format. It can easily convert from a spreadsheet to ARFF. In addition, most spreadsheet programs, e.g. Microsoft Excel allow us to export data into a file in comma-separated value (CSV) format as a list of records with commas between items. Having done this you need only load the file into a text editor, e.g Microsoft Word, WordPad, and convert it manually into the ARFF file. Nevertheless, we do not actually have to go through these steps to create the ARFF file, because the Explorer of the WEKA can read CSV spreadsheet files directly.



```
% Summary Statistics:
%                 Min  Max   Mean    SD   Class Correlation
%    sepal length: 4.3  7.9   5.84  0.83    0.7826
%     sepal width: 2.0  4.4   3.05  0.43   -0.4194
%    petal length: 1.0  6.9   3.76  1.76    0.9490  (high!)
%     petal width: 0.1  2.5   1.20  0.76    0.9565  (high!)
%
% 9. Class Distribution: 33.3% for each of 3 classes.

@RELATION iris

@ATTRIBUTE sepallength  REAL
@ATTRIBUTE sepalwidth   REAL
@ATTRIBUTE petallength  REAL
@ATTRIBUTE petalwidth   REAL
@ATTRIBUTE class  {Iris-setosa,Iris-versicolor,Iris-virginica}

@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
```

Figure 13: Example of an ARFF file (Yang, 2009)

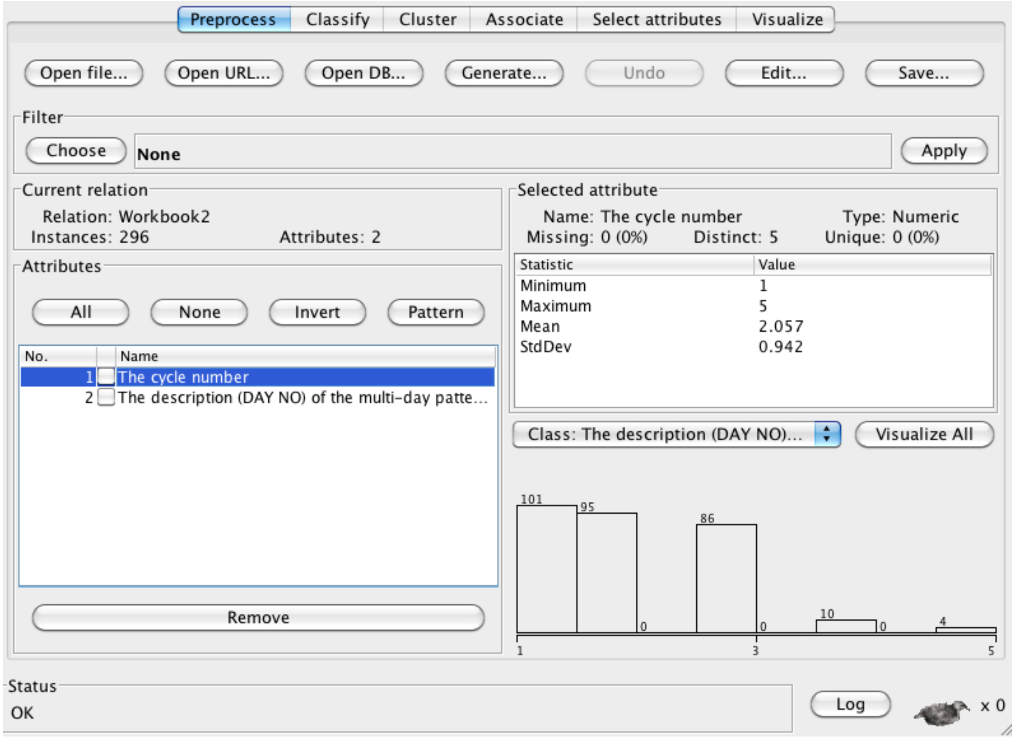
5.2.2 Loading the data into the Explorer

After loading the data (.csv file) into the Explorer, we can start analyzing it. Having loaded the file, the screen will be as shown in Figure 14. This tells readers about the dataset: this regimen has 296 instances and two attributes. The first attribute, *the cycle number*, is selected by default and has no missing values, five distinct values, and no unique values; the statistics are *Minimum, Maximum, Mean, Standard Deviation*, and values are 1, 5, 2.057, 0.942, respectively. A histogram at the lower right shows how many records for each of the values of the cycle number from 1 to 5. For example, 101 records have value 1 or 1.5; 95 records have values 1.5 or 2.

If the data is nominal, the histogram will show how often each of the values occurs for each value of the attribute. In addition, it is possible to delete an attribute by clicking its checkbox and using the *Remove* button. *All* selects all the attributes, *None* selects none, and *Invert* inverts the current selection.

5.2.3 Clustering

Use the *Cluster* panel to invoke clustering algorithm (Section 3.4). The cluster preferences are shown in the Figure 15. The name of output is *weka.clusterers.SimpleKMeans.* Cluster data using the k-means algorithm. There are some options in preferences: *displayStdDevs* is to display standard deviations of numeric attributes and counts of nominal attributes; *distanceFunction* is the distance function to use for instances comparison (default is weka.core.EuclideanDistance). If the Manhattan distance is used, then centroids are computed as the component-wise median rather than mean; *dontReplaceMissingValues* is to replace missing values globally with mean/mode; *maxIterations* is to set maximum number of iterations; *numClusters* is to set number of clusters; *preserveInstancesOrder* is to reserve order of instances; and *seed* is the random number seed to be used.



Filter: None

Current relation
Relation: Workbook2
Instances: 296 Attributes: 2

Selected attribute
Name: The cycle number Type: Numeric
Missing: 0 (0%) Distinct: 5 Unique: 0 (0%)

Statistic	Value
Minimum	1
Maximum	5
Mean	2.057
StdDev	0.942

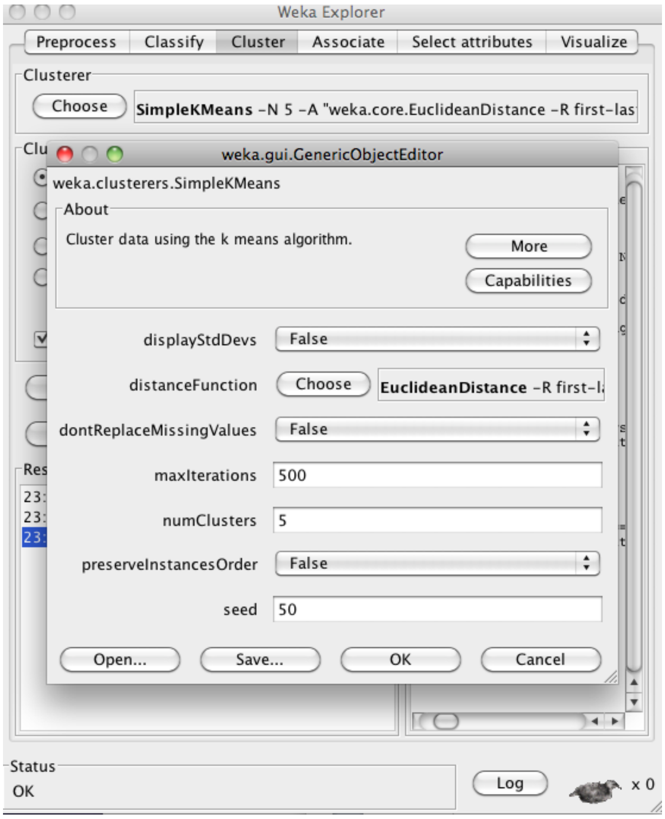
Attributes

No.		Name
1	□	The cycle number
2	□	The description (DAY NO) of the multi-day patte...

Class: The description (DAY NO)...

Status
OK

Figure 14: The WEKA Explorer: reading in the PACLITAXEL 1W regimen's records



SimpleKMeans -N 5 -A "weka.core.EuclideanDistance -R first-las

weka.clusterers.SimpleKMeans

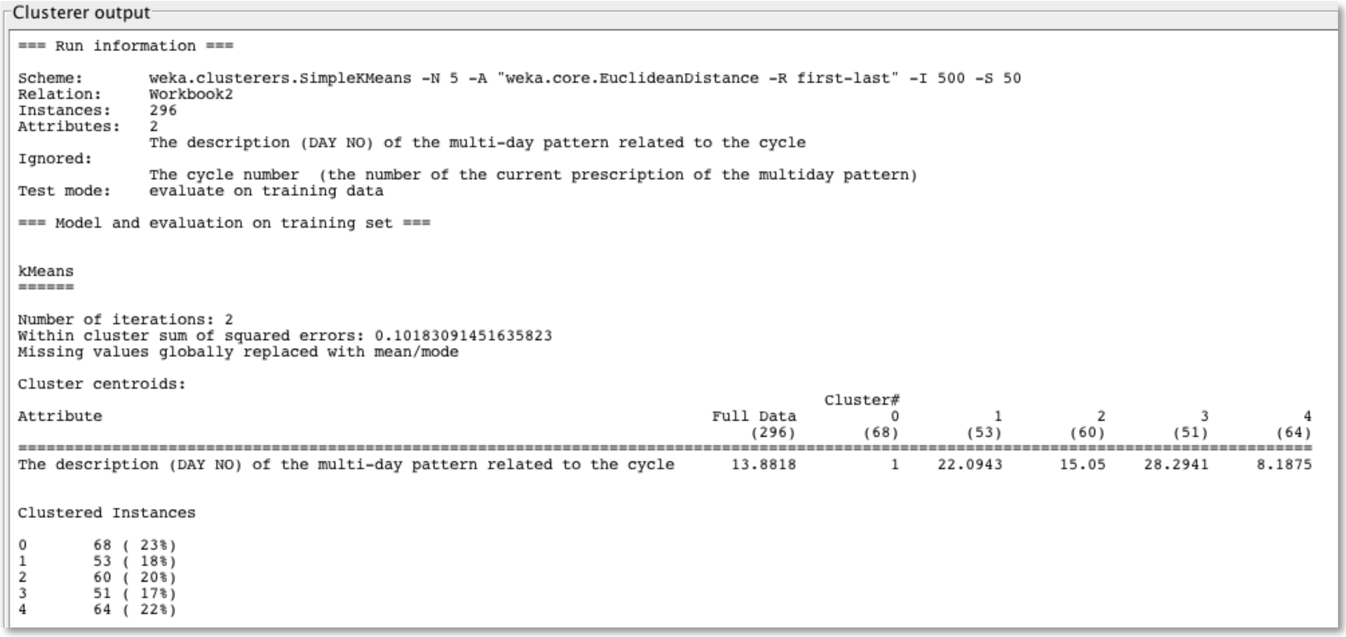
Cluster data using the k means algorithm.

displayStdDevs	False
distanceFunction	**EuclideanDistance** -R first-l
dontReplaceMissingValues	False
maxIterations	500
numClusters	5
preserveInstancesOrder	False
seed	50

Status
OK

Figure 15: Cluster Preferences

I chose five clusters to generate as experienced in Microsoft Excel before. “The cycle number” attribute is ignored. The cluster result shows in Figure 16 below. The result window shows the inputs, e.g. *Scheme, Source*: “workbook2”, *Number of Instances*: 296, *Number of attributes*: 2, *Ignored attribute:* “The cycle number”, and *Test mode*: Use training data set. The outputs are summarized the number of iterations: 2, the centroid of each cluster as well as statistics on the number and percentage of instances assigned to different clusters. Cluster centroids are the mean vectors for each cluster. Thus, centroids can be used to characterize the clusters. In our case, the centroid for cluster from 0 to 4 has value approximately **1; 22; 15; 28; 8**. This result is obviously the same with observed and calculated results in previous section.



```
=== Run information ===

Scheme:       weka.clusterers.SimpleKMeans -N 5 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 50
Relation:     Workbook2
Instances:    296
Attributes:   2
              The description (DAY NO) of the multi-day pattern related to the cycle
Ignored:
              The cycle number  (the number of the current prescription of the multiday pattern)
Test mode:    evaluate on training data

=== Model and evaluation on training set ===


kMeans
======

Number of iterations: 2
Within cluster sum of squared errors: 0.10183091451635823
Missing values globally replaced with mean/mode

Cluster centroids:
                                                                                Cluster#
Attribute                                                          Full Data          0          1          2          3          4
                                                                       (296)       (68)       (53)       (60)       (51)       (64)
=====================================================================================================================================
The description (DAY NO) of the multi-day pattern related to the cycle   13.8818          1    22.0943      15.05    28.2941     8.1875


Clustered Instances

0       68 ( 23%)
1       53 ( 18%)
2       60 ( 20%)
3       51 ( 17%)
4       64 ( 22%)
```

Figure 16: The WEKA Clusterer Output

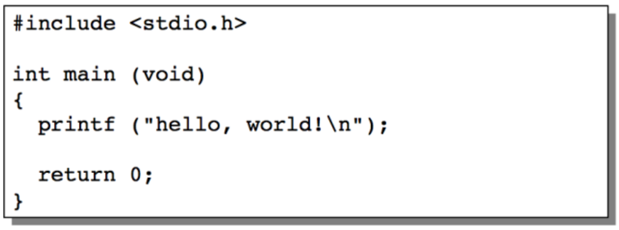
Not only MS. Excel but also WEKA gives standard pattern for PACLITAXE1W is 1, 8, 15, 22, 28. Nevertheless, these two methods are not quite fully automatic. The next section will introduce a new and promising fully automatic solution.

5.3 Using Python to find standard pattern

5.3.1 Introduction to Python

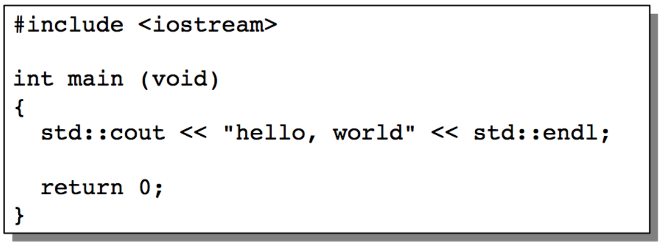
Python is a free and open source programming language. It has a community-based development model. This programming language is much simpler, quicker to learn and turn ideas into working code than other programming languages such as C, C++, and Java (Jenkins, 2010). For example, a small program to print “*Hello World*”

In C,



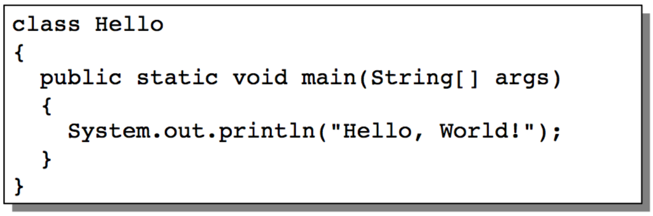
```
#include <stdio.h>

int main (void)
{
  printf ("hello, world!\n");

  return 0;
}
```

In C++,



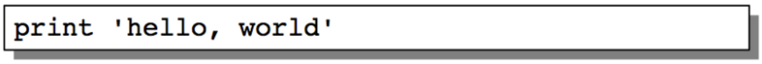
```
#include <iostream>

int main (void)
{
  std::cout << "hello, world" << std::endl;

  return 0;
}
```

In Java,



```
class Hello
{
  public static void main(String[] args)
  {
    System.out.println("Hello, World!");
  }
}
```

But in Python programming seems to be very simple, just like this:



```
print 'hello, world'
```

Python has been taught as a very attractive programming language. It is worthwhile to create a Python program to find standard pattern automatically.

5.3.2 Algorithm description

In order to define a Standard Pattern: we need all patterns having the same length and the same day number in one cycle. Therefore, the program should follow two stages:

- Stage 1: Select records corresponding to the same cycle.
- Stage 2: Produce a standard pattern based on the selected records.

Stage 1: Select records

I suggest two methods to select records from the data set: **the longest list** and **the highest frequency**.

The longest list method: Select patterns consisting at the largest number of visit days. For example, some patients taking PACLITAXEL 1W regimen have multiday pattern treatment as below table. If using the longest list method, the standard pattern would be calculated based on pattern number 1 and 11 because it has the longest list of numbers of elements in one cycle – 6 elements.

The highest frequency: Select patterns appear most frequently. if using the highest frequency method for patients in the below table, all patterns having five elements, which are pattern number 2, 3, 4, 6, 7, and 8 will be selected to calculate the standard pattern.

PACLITAXEL 1W	
	Pattern
1	1,8,15,22,28,33
2	1,8,15,22,28
3	1,8,15,22,28
4	1,8,15,22,28
5	1,9,15
6	1,9,15,22,28
7	1,8,15,22,28
8	1,7,14,21,27
9	1,8,22,28
10	1,8,15
11	1,8,14,20,26,31

Stage 2: Produce a standard pattern

After selecting patterns with the same length, the next step is to calculate standard pattern from those patterns. There are two methods to calculate standard pattern from chosen patterns: one uses **the mean** and another one uses **the mode** of the elements in chosen patterns, which is suggested in (Jagannathan and Petrovic, 2009).

For instance, if I used the highest frequency method previously to select the patterns I would have results:

3	6, 6, 6, 5	1,8,15,22,28
4	6, 6, 6, 5	1,8,15,22,28
6	7, 5, 6, 5	1,9,15,22,28
7	6, 6, 6, 5	1,8,15,22,28
8	5, 6, 6, 5	1,7,14,21,27
Mean		1; 8; 14.8; 21.8; 27.8
Mode		1; 8; 15; 22; 28

The method based on mean or mode is quantitative; but the method based on Stem-and-Leaf Display is estimation. It might be possible to get some "evidence figures" in Excel; this was not explored in this project.

In conclusion, we have four options for finding standard pattern by combining two methods to find standard pattern's length and two methods to get standard elements in one pattern. The combination is summarized in Table 12:

Table 12: Options to find a Standard Pattern

	The longest list	The highest frequency
The mean	The longest patterns + The mean of each element in those cycles	The highest frequency of the number of elements in patterns + The mean of each element in those cycles
The mode	The longest patterns + The mode of each element in those cycles	The highest frequency of the numbers of elements in patterns + The mode of each elements in those cycles

5.3.3 Program Design

In order to automatically find standard patterns for all regimens and (will) repair the dataset with their found standard patterns; the software contains three main modules: Pre-processing, Get Standard and Data Repair. Figure 17 below summarises main idea for

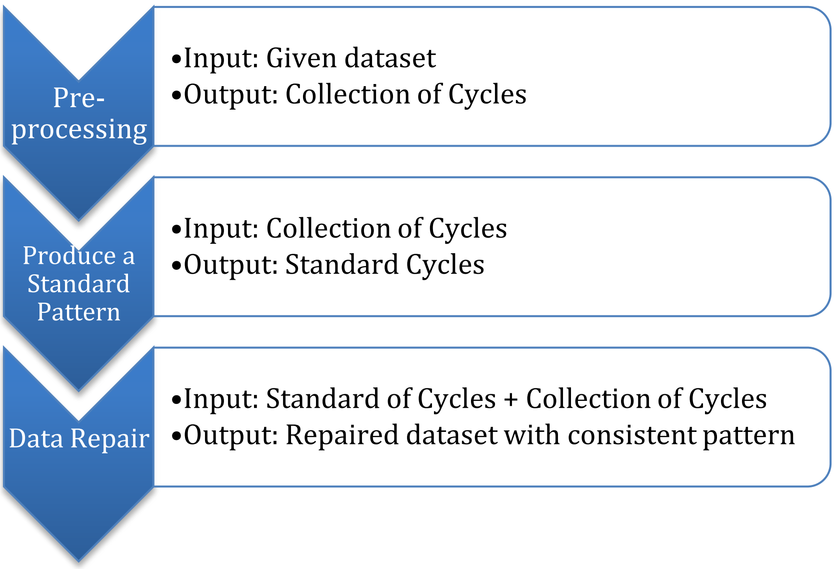

Pre-processing
•Input: Given dataset
•Output: Collection of Cycles
Produce a Standard Pattern
•Input: Collection of Cycles
•Output: Standard Cycles
Data Repair
•Input: Standard of Cycles + Collection of Cycles
•Output: Repaired dataset with consistent pattern

Figure 17: Program Design

Module 1: Pre-processing

This module **reads** the given dataset. The given dataset is in format of MS Excel (.xls) file. However, the Excel file is quite complex with formatted columns and rows so it was simplified by converting from '.xls' to '.csv' file. This process can be done very easily by choosing '.csv' from SaveAs options list when opening MS Excel. Another reason for using .csv file because programming language – Python is built-in supported .csv. That could save much time and effort to create database for software development.

Module Pre-processing also **lists all Regimen (Drug) names** by rows. Then collection of **all cycles and names of all Regimens are extracted** from the list.

Module 2: Produce a standard pattern

Previous section has mentioned four options to find the Standard Pattern for one regimen. In the first place, I have implemented *the longest list method and the mean method* for one regimen (PACLITAXEL1W). Other combination was described in Section 5.3.2.

```
Initial RegimenCycleList

LongestCycleList = getLongest Cycles(RegimenCycleList)

For i from 0 to  LongestCycleLength

        Standard[i] = Average (LongestCycle[i])

End For
```

(Campbell et al., 2009)

The whole dataset is imported to get the data set of all Standard Patterns.

Module 3: Data Repair

When having the list of Standard Pattern from module 2, they are used to repair the original data set. There are two methods to repair data with found Standard Pattern. The first method is **Case Deletion** and the second is **Nearest-neighbour fixed point correction**.

Case Deletion method simply deletes all records being not as same as the found standard patterns.

Nearest-neighbour fixed point correction method based on nearest neighbour classification method and instance-based learning method, which has been described in Section 3.1.2. This module will be explained more in Chapter 6:

5.3.4 Results

Due to time constraint, I have not finished all three modules. I have finished the module 2, to make a data set of standard patterns automatically in Excel format as following.

Regimen name	Standard Pattern
5FU+FA(5DAY)	[1, 2, 3, 4, 5, 28]
5FU+FACON	[1, 2, 3, 4, 6, 28]
5FU+FAWEEKLY	[1, 8, 15, 22, 29, 37, 43]
ABC-02GEM	[1, 8, 15, 28]
ABC-02GEM/CIS	[1, 9, 22]
AN1AND2(OP)	[1, 29]
…	…

The full list of standard pattern is provided in the appendix D.

Chapter 6: Methods to automatically repair the data set

In the Chapter 5:, the combination of the longest list and the means value was applied in the Python program to find out the data set of standard pattern (Appendix D). The ideal repaired data set would be the one having consistent patterns throughout all regimens. Repairing task uses the standard patterns to edit inconsistent patterns in the original data set, therefore, if the data set of standard patterns changes due to different pattern recognition methods then the repaired result will be different.

In this part, the regimen, named PACLITAXEL1W, is an illustration to repair. In appendix D, PACLITAXEL1W’s standard pattern is [1, 8, 15, 22, 28]. In Section 4.3, we have discussed different pattern representations. The following table extracts of some patterns of PACLITAXEL1W regimen that are not consistent.

PACLITAXEL 1W’s patterns		
	Day number representation(1)	*Time-lag representation (2)*
1	1,8,15,22,28	6 days – 6 days – 6 days – 5 days
2	1,8,15,22,28	6 days – 6 days – 6 days – 5 days
3	1,8,15,22,28	6 days – 6 days – 6 days – 5 days
4	1,8,15,22,28	6 days – 6 days – 6 days – 5 days
5	1,9,15	7 days – 5 days
6	1,9,15,22,28	7 days – 5 days – 6 days – 5 days
7	1,8,15,22,28	6 days – 6 days – 6 days – 5 days
8	1,7,14,21,27	5 days – 6 days – 6 days – 5 days
9	1,8,22,28	6 days – 13 days – 5 days
10	1,8,15	6 days – 6 days
11	1,8,14,20,26,31	6 days – 5 days – 4 days – 5 days – 4 days

There are two methods to repair the data set: case deletion and nearest-neighbour fixed point correction.

6.1 Case deleting

If any pattern does not coincide, in other word, *exactly the same* with the standard pattern then delete it. So, those patterns remaining are definitely the same with the standard pattern. Implementing this method for PACLITAXEL1W’s examples above, the patterns number 5, 6, 8, 9, 10, 11 are deleted from the data set and patterns number 1, 2, 3, 4, 7 are remained in the data set. This method is the simplest and easiest to implement and repair the data. On the other hand, this method the repaired data set will lose so many records. This downside makes this method inappropriate to implement. It is possible to delete element(s) in longer patterns than the standard pattern.

6.2 Nearest-neighbour fixed point correction

As case deletion method does not consider different cases and scenarios in the original data set. The *nearest-neighbour fixed point* correction is more complex and does consider various scenarios within the data set such as patterns with number of elements not equal to standard pattern's number of elements; patterns with the same number of elements to standard pattern's number of elements but incorrect pattern. This method is based on the idea of the *instance based learning* and the *nearest neighbour classification method* (Section Classification under 3.1.2). An example pattern "X" has n elements [n_1, n_2, n_3, n_4, n_5, ... ,n_k] and the standard pattern "S" has m elements [m_1, m_2, m_3, m_4, m_5, ... ,m_k]. When comparing two patterns, the elements of X and the elements of S are neighbourhood. For example, n_1 has neighbour $m_{1,}$ m_2, m_3, m_4, m_5, ... ,m_k. When me match X with S; firstly we compare their number of elements, secondly we calculate the distance between a element in X and its neighbour in S. The "distance" in our case is the Euclidean distance between two neighbours, which is explained in Section 3.4. When distance between two neighbour equal 0, they have the same value.

In order to repair the data set with patterns inconsistently, I suggest two heuristics, which are called *left-right heuristic* and *right-left heuristic*.

- The left-right heuristic compares and repairs a pattern "**X**" to the standard pattern "**S**" from the left to the right; try to match the first element of X with the any of its neighbour (element of S). If the first element of X is not equal to any element of S then check the other element of X to see whether any neighbours have the same value. If there is no element of X equals with any element of S then the nearest-distance neighbour is chosen to fix. The first matched element of X is now the starting point to repair. From the starting point, any element of X different than the respective element of S need to be repaired to completely match with S. If the pattern is shorter than the standard pattern, then add the missing elements with the remained elements in the standard pattern.
- The right-left heuristic is basically similar with the left-right heuristic except that the program (software) starts to compare the last element in S. from the first-right element or the last element in X instead of the first one.

In fact, these heuristics prefer the time-lag representation (2) to day-number representation (1). The reason is that some patterns have the same time-lag between treatments with the standard pattern's time-lag but not always exactly the same in day number. Taking one PACLITAXEL1W's pattern [1, 7, 14, 21, 27] or [5 days – 6 days – 6 days – 5days] as an example. As we know, PACLITAXEL1W's standard pattern is [1, 8, 15, 22, 28] or [6 days – 6 days – 6 days – 5 days]. If the program uses the day-number representation to compare and identify differences, four elements [7, 14, 21, 27] in pattern [1, 7, 14, 21, 27] will absolutely be changed. However, if the time-lag representation is used, the program will identify only one difference between the example pattern and the standard pattern, which is the first element. Thus, the program edits only the first element in the example pattern instead of four elements.

Pattern	1	7	14	21	27
Using representation 1 to repair	1	***8***	***15***	***22***	***28***
Using representation 2 to repair	***0***	7	14	21	27

The repair using the first representation makes (4/5) 80% change of data but using second representation to repair only changes (1/5) 20% of the example pattern. So that the second representation is more appropriate.

Another scenario is presented in following table. Data set shows this patient *[tpomdodtut]* used regimen PACLITAXEL1W, of which standard pattern is [1, 8, 15, 22, 28] in two cycles. Pattern for cycle 1 is [1, 15, 22, 29, 35] or [13 days – 6 days – 6 days – 5 days] and pattern for cycle 2 is [1, 8] or [6 days].

Patient	DIAGDATE	Regimen (drugs)	The cycle number	The description (DAY NO) of the multi-day pattern related to the cycle
tpomdodtut	01/08/2008	PACLITAXEL1W	1	1
tpomdodtut	01/08/2008	PACLITAXEL1W	1	15
tpomdodtut	01/08/2008	PACLITAXEL1W	1	22
tpomdodtut	01/08/2008	PACLITAXEL1W	1	29
tpomdodtut	01/08/2008	PACLITAXEL1W	1	35
tpomdodtut	01/08/2008	PACLITAXEL1W	2	8
tpomdodtut	01/08/2008	PACLITAXEL1W	2	15

The representation 2 shows that there is one difference between the example pattern and the standard pattern for the first cycle. The difference is the first element [13 days] instead of [6 days]. Using left-right heuristic, the program will correct the day number on the left of [13], which is [1] to [8]. The new repaired pattern is [8, 15, 22, 29, 35] with 1/5 (20%) of repairing.. However, using the right-left heuristic, the program will correct the day number on the right of [13], which is [15] to [8] and onward. The repaired pattern is [1, 8, 15, 22, 28] with 4/5 (80%) of repairing. Thus, it is suggested to choose left-right heuristic's result.

Representation 1	1	15	22	29	35
Representation 2	13 days	6 days	6 days	5 days	
Right-left heuristic	1	**8**	**15**	**22**	**28**
Left-right heuristic	**8**	15	22	29	35

The second cycle is [8, 15] or [6 days]. The program uses the representation [6 days] and finds it matching with second element in the standard [6days] so it is kept the same. It is possible to add the missing elements with the remained elements in the standard pattern. Thus, the second cycle would become [8, 15, 22, 29, 35].

Chapter 7: Evaluation

In general, all three minimum requirements mentioned in part 1 have been completed. The first requirement is about techniques applicable to the problem under the study. In part 2: *Project Methodology* has discussed about Knowledge Discovery Process and Data Mining. In principle, data mining is ideal for the problem in this report. While we have a raw data set at the first step, and at the end of data mining process, we want the *knowledge* about standard patterns. Part 2 has provided us different techniques applicable to the pattern recognition task. The second requirement is a data set of standard patterns for all regimens. In order to achieve this requirement, three different approaches have been applied; they are statistical approach, using WEKA software and using Python. The final requirement is the description about methods to repair the data set. Two approaches have been used; they are case deletion and nearest-neighbor fixed point correction.

7.1 Evaluation of Approaches

Different approaches certainly leads to different results. To decide which approach is the most appropriate, this section considers pros and cons of each approach for each requirement. This takes into account every attributes: how much they complete task, how complex they are, what requirements are needed, etc.

7.1.1 Evaluate approaches for Pattern recognition

Pattern recognition is the most significant and important task to be done. The second task of repairing the original data set can carry out if and only if the regimens' patterns are found. Three approaches applied in this project are different in level of complexity, implemented scope, and ability to accomplish task. The following table shows advantages and disadvantages of each method to have better overview of all possible solutions.

Table 13: Pros and Cons of pattern recognition methods

	Advantages	Disadvantages
Statistics with Microsoft Excel & PhStat2	✓ Build-in application so no installment is required ✓ Easy to use ✓ Able to visualize clusters, which is the potential pattern of regimens using charts and figures ✓ Keep the same format of the	✗ Need to filter dataset manually ✗ Do not produce all regimens' patterns, just one at a time ✗ The significant evidence to confirm regimen's standard was not explored in this project

	original dataset	
WEKA	✓ Semi-automatically generate the cluster ✓ WEKA automatically produces a pattern with explanation and justification from well-known pattern recognition algorithms.	✗ Records of one regimen is extracted to a new Excel file ✗ Original file is converted into other format (CSV) ✗ Produce one regimens' patterns at a time ✗ Semi-automatic: Number of cluster is manually chosen
Python	✓ Find all regimens' patterns automatically ✓ The result is more precise and reliable ✓ The most powerful solution: further extensions can be developed e.g. automatically repair.	✗ Original file is converted into other format (CSV) ✗ Require to learn about programming ✗ The most complex method among three

The first, statistics approach with Microsoft Excel is quite simple. It helps analyse and visualize possible cluster or potential pattern of one regimen. Microsoft Excel is very popular application for any personal computer, its statistics toolkit is an add-on, just need to activate to use. PhStat2 is also an add-on application to Microsoft Excel but not ready-to-use. It requires some understanding of its functions and statistics to use this tool effectively. The graphs and figures generated are very clear and understandable. They are useful for the first learner about this data set of hospital's appointments. Though this approach does show the pattern in groups of points it is not very persuasive about pattern. The scatter plot e.g. Figure 10 shows clearly distinct groups of points or clusters when there is no outliner or noise in the data set. When the data set contains noise like Figure 11, it is difficult to define clusters exactly. PhStat2 Stem-and-Leaf Display gives value for points so that reader can identify values of outliners and make assumption about the modes and mode's neighbours. For example, in Figure 12 the assumption is that the mode is "8" and all points around 8 such 7, 9 10 are all in one group/ cluster. One element in the pattern therefore is the mode "8".

The second approach is to use WEKA software. WEKA is a powerful data mining tool that provides user-friendly explorer to classify, cluster, generate tree-view, etc. This software is open source therefore a big advantage while people can easily download and make use of it. Though WEKA is easy to use user still need to understand the concept of its algorithms such k-means, Cobweb, and EM to choose the most proper clustering mode. Furthermore, the clusterer tool can provide meaningful centroid numbers, which are clusters' mean numbers if and only if the input is a regimen's data. Thus, one regimen's records should be extracted to a new Excel sheet and then converted to CSV format, the format is supported by WEKA. In addition, user needs to test several different cluster numbers to find out how many clusters are there exactly. This method is parameter sensitive and not robust enough to satisfy pattern recognition requirement.

It may reduce time to try several cluster numbers by combining statistics method and WEKA. For instance, we use scatter plot or Stem-and-Leaf Display to have general idea about how many cluster would be, then use that estimated number of clusters in WEKA to calculate the centroid/ cluster values. Although each method separately and the combination of these two give pretty precise results of regimen's pattern; they only can generate pattern for one regimen at a time and just semi-automatic. The intervention of human still is critical.

The third method is a program written using Python. Python is simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance (Python, 2010). The beginner can easily find a lot of online Python's tutorials and instructions. There are four options to generate standard patterns (Table 12). In the scope of this report, due to short of time, only one version program is developed based on one combination – the longest list and the mean. If I had more time, I would have developed more versions and try to implement other three combinations. When using other combination, the patterns would be different and possibly lead to better solution. I have tested the results of my Python program (Appendix D) by observations of all standards not start at 1. For example, in appendix D, the regimen "BEP3DAY" has pattern [7, 8, 15, 21]. However, its scatter plot shows pattern [8, 15, 21], which is the correct pattern. The reason of difference is that there is only one cycle contained four elements, which are [7, 8, 15, 21] and the program based on the longest list and the mean to define patterns. In this example, if the program based on the highest frequency and the mode, it would have provided a correct pattern. In general, almost patterns are correct. The biggest advantage of this method is fully automatic. In conclusions, the Python program is a complete solution for pattern recognition requirement, to find regimens' patterns automatically.

7.1.2 Evaluate approaches to repair the data set

Another requirement is a description of method to repair the data set. Two approaches have been applied to accomplish task: the case deletion and the nearest-neighbor fixed point correction. While, the case deletion approach is very simple and easy, the other approach is much more sophisticated and robust. The following table shows advantages and disadvantages of each method.

	Advantages	**Disadvantages**
Case deletion	✓ Simple ✓ The result is always correct and consistent	× Data lost while correcting × Not flexible
Nearest-neighbor fixed point correction	✓ Sophisticated ✓ More flexible in different cases	× Not robust × Results is depended on standard pattern's representation

The first approach is case deletion. This method is absolutely simple and basic. If all wrong patterns are deleted then remained ones must be right. Basically, nothing is wrong with this assumption but it is not quite a repair method, just an eliminating method to get a consistent data set. Therefore, the final result may be ideal and consistent but not satisfied requirement.

The second approach, nearest-neighbour fixed point correction, is a more proper method. It covers many possible cases such as longer patterns, shorter patterns, and same length patterns. On the other hand, under some circumstances, this solution does not completely create a good amendment. An instance is pattern [22, 29] or [6 days] for regiment PACLITAXEL1W. If we use time-lag representation and apply the *nearest-neighbour fixed point* method, the program will realize that there is no difference at the first place. The example standard [6 days] matches with the first element of the standard pattern [6 days] when using left-right heuristic. The program would find no error at all and change nothing. When using right-left heuristic, the program will match this example pattern with the second last element of standard and also change nothing. In fact, the day-number representation helps us see the pattern [22, 29] likely match with standard pattern [...22, 28] so this example pattern should be corrected. Let's assume we have a standard pattern **S** [1, 4, 7, 11] or [2 days – 2 days – 3 days] and a pattern **Y** [5, 8, 11] or [2 days – 2 days].

Representation 1	5	8	11
Representation 2	2 days	2 days	
Standard	2 days	2 days	3 days

In this case, both the left-right and the right-left heuristic would match the first two elements of Y with the first two elements of S. The program would change nothing because of any difference. However, the day-number representation is straightforward; to match Y with the last three elements of S and change [5] to [4] and [8] to [7].

Therefore, it is debatable about the efficiency of this method. The representation is very important to consider before applying any method to repair. In my point of view, if the length of pattern is equal to the length of standard pattern then use the time-lag representation otherwise use the day-number representation.

Given various methods, one can test whether

- The output they produce fully satisfies standard patterns
- All treatments happen on working days;
- The methods can be compared using a quantitative characteristic (percentage of changed data).

Chapter 8: Conclusion

To sum up, there are three minimum requirements for this project which are all completed. The relevant literatures about pattern recognition, and knowledge discovery process as well as data mining tasks have been reported in Chapter 2:. There are three possible methods to get standard patterns which are statistics by Microsoft Excel and PhStat2, WEKA and Python program. Among three approaches, statistical approach is the simplest one and the Python program is the most sophisticated. Only Python program satisfies all requirements: automatically find standard patterns for all regimens. Both WEKA and statistical method are semi-automatic and only find standard pattern for one regimen at a time. The program can be edited by using different combinations to define standard patterns. The description of repairing methods includes two possible approaches: case deletion and nearest-neighbour fixed point correction. The case deletion method is very simple, not quite useful for this project because of lost data. The other approach is more proper when it considers different scenarios to repair the data set. It does not delete the data; however, this method has not repaired shorter patterns than standard pattern correctly. Further possible research and methods can be applied to repair the original data set.

Chapter 9: References

BULPITT, A. 2010. Data Mining Lecture Note. *Knowledge Management*.

CAMPBELL, J., GRIES, P., MONTOJO, J. & WILSON, G. 2009. *Practical Programming: An Introduction to Computer Science Using Python*.

DASU, T. & JOHNSON, T. 2003. *Exploratory Data ming and Data cleaning*, John Wiley & Sons, Inc.

DAVIES, C. & LOWE, T. 2010. *Kolb Learning Cycle Tutorial* [Online]. University of Leeds. Available: http://www.ldu.leeds.ac.uk/ldu/sddu_multimedia/kolb/static_version.php [Accessed 15/02/2010].

FAYYAD, U., GRINSTEIN, G. & WIERSE, A. 2001. *Information Visualization in Data Mining and Knowledge Discovery*, Morgan Kaufmann.

FISHER, D. Year. Improving inference through conceptual clustering. *In:* AAAI Conference, 1987 Seattle Washington. 461-465.

GENNARIA, J., LANGLEYA, P. & FISHER, D. 1990. Models of incremental concept formation. *Artificial Intelligence,* 40**,** 11-61.

HAN, I. & KAMBER, M. 2006. *Data Mining: Concepts and Techniques*.

HAND, D., MANNILA, H. & SMYTH, P. 2001. *Principles of Data Mining*, The MIT Press.

HOCHBAUM, D. & SHMOYS, D. 1985. A best possible heuristic for the k-center problem. *Mathematics of Operations Research,* 10**,** 180-184.

JAGANNATHAN, R. & PETROVIC, S. 2009. Dealing with Missing Values in a Clinical Case-Based Reasoning System. Nottingham, United Kingdom: School of Computer Science, University of Nottingham.

JAIN, A., DUIN, R. & MAO, J. 2000. Statistical Pattern Recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 22.

JENKINS, T. 2010. Lecture 0 - What is...? Lecture Note. *Problem Solving with computers*.

KELLER, G. & WARRACK, B. 2004. *Statistics for management and economics*, Thomson/Brooks/Cole.

LAROSE, D. 2005. *Discovering Knowledge in Data - An Introduction to Data Mining*, John Wiley & Sons, Inc.

LEVINE, D., STEPHAN, D., KREHBIEL, T. & BERENSON, M. 2002. *Statistics for managers using Microsoft Excel*, Upper Saddle River, N.J. : Prentice Hall.

LIND, D., MARCHAL, W. & WATHEN, S. 2006. *Basic statistics for business & economics,* Boston [Mass.] : London, McGraw-Hill/Irwin.

LIU, J., SUN, J. & WANG, S. 2006. Pattern Recognition: An overview. *IJCSNS International Journal of Computer Science and Network Security,* 6**,** 57-61.

MCSHANE, S. & TRAVAGLIONE, T. 2006. *Organisational Behaviour on the Pacific Rim*, McGraw-Hill Higher Education.

PEARSON EDUCATION INC. 2010. *Statistics add-in system for Microsoft Excel* [Online]. Available: http://www.prenhall.com/phstat/ [Accessed 2010].

PYTHON. 2010. *What is Python? Executive Summary* [Online]. Available: http://www.python.org/doc/essays/blurb/ [Accessed 02/2010].

SOMMERVILLE, I. 2008. *Software engineering*, Pearson/ Addison-Wesley.

SYMEONIDIS, A. & MITKAS, P. 2005 *Agent intelligence through data mining*, Springer.

THE UNIVERSITY OF WAIKATO. 2010. *Weka Machine Learning Project* [Online]. The University of Waikato. Available: http://www.cs.waikato.ac.nz/ml/weka/ [Accessed 2010].

WITTEN, I. & FRANK, E. 2005. *Data Mining: Practical machine learning tools and techniques. 2nd edition,* San Francisco, Morgan Kaufmann.

YANG, Q. 2009. COMP 337 Tutorial 1 - Introduction to WEka. *http://ihome.ust.hk/~wxiang/TA/COMP337/comp_337_t1.htm*.

Chapter 10: Appendixes

Appendix A: Reflection

William Glasser said that "Education is the process in which we discover that learning adds quality to our lives. Learning must be experienced."

David Kolb developed a theory of experiential learning that can give us a useful model by which to develop our practice. This is called The Kolb Learning Cycle or The Experiential Learning Cycle. The cycle comprises four different stages of learning from experience and can be entered at any point but all stages must be followed in sequence for successful learning to take place. It is necessary to reflect on the experience to make generalisations and formulate concepts which can then be applied to new situations. This learning must then be tested out in new situations. The learner must make the link between the theory and action by planning, acting out, reflecting and relating it back to the theory (Davies and Lowe, 2010). I would like to review my learning progress by put all my actions in this model. The figure below is the model of Kolb Learning Cycle.

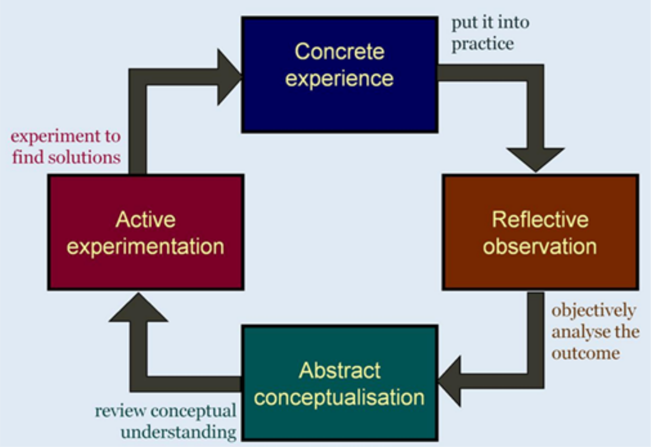
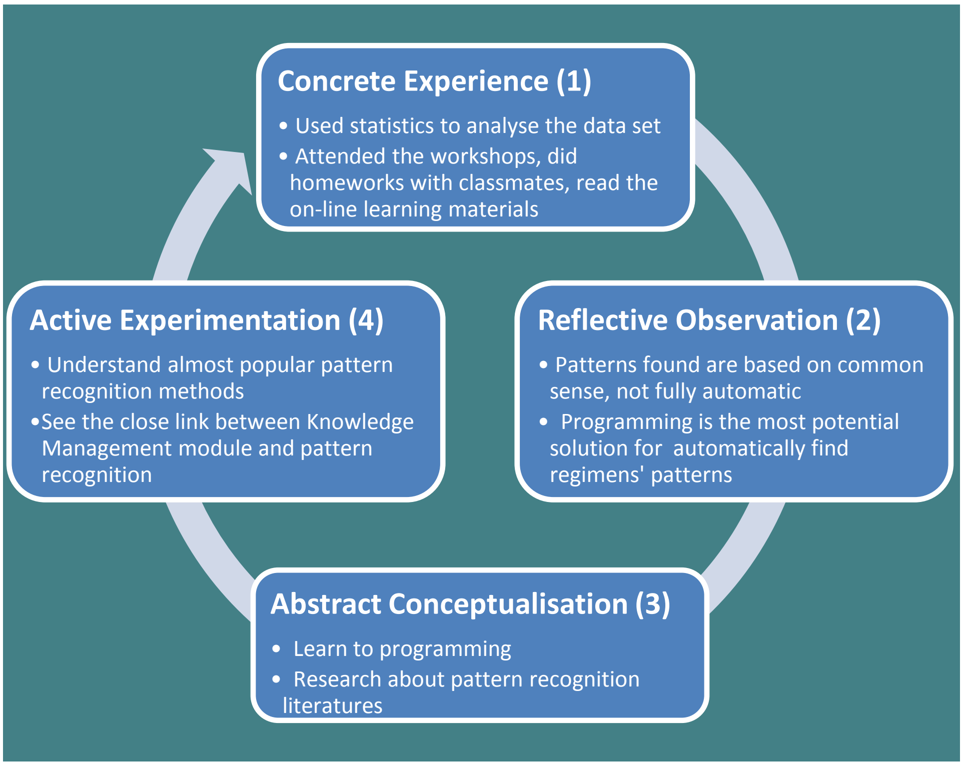

Concrete experience
put it into practice
Reflective observation
objectively analyse the outcome
Abstract conceptualisation
review conceptual understanding
Active experimentation
experiment to find solutions

Figure 18: Kolb's experiential learning model (McShane and Travaglione, 2006)

Learning Cycle 1:

About week 4 in semester 2, I have received the topic for my dissertation. Thought about modules are relevant, maybe business information system and Python. However, programming for me at the beginning of semester 2, is a challenging task. Due to the first degree in business management and statistics was a crucial part of my past experience, I quickly applied statistical tools and techniques into the given data set. At the same time, I have struggled with all computing modules and Programming with Python was the hardest one. Because the majority of classmates have engineering or computer science background, they can easier catch up with progress, theories and teaching method than I can. It took me much time for independent learning. I have practiced Python at home, attended the workshops, read the on-line learning materials, actual experienced of programming plus asked teachers and friends for more instructions. I have tried to do all homework as well as in-class practice tasks.



Concrete Experience (1)

- Used statistics to analyse the data set
- Attended the workshops, did homeworks with classmates, read the on-line learning materials

Reflective Observation (2)

- Patterns found are based on common sense, not fully automatic
- Programming is the most potential solution for automatically find regimens' patterns

Abstract Conceptualisation (3)

- Learn to programming
- Research about pattern recognition literatures

Active Experimentation (4)

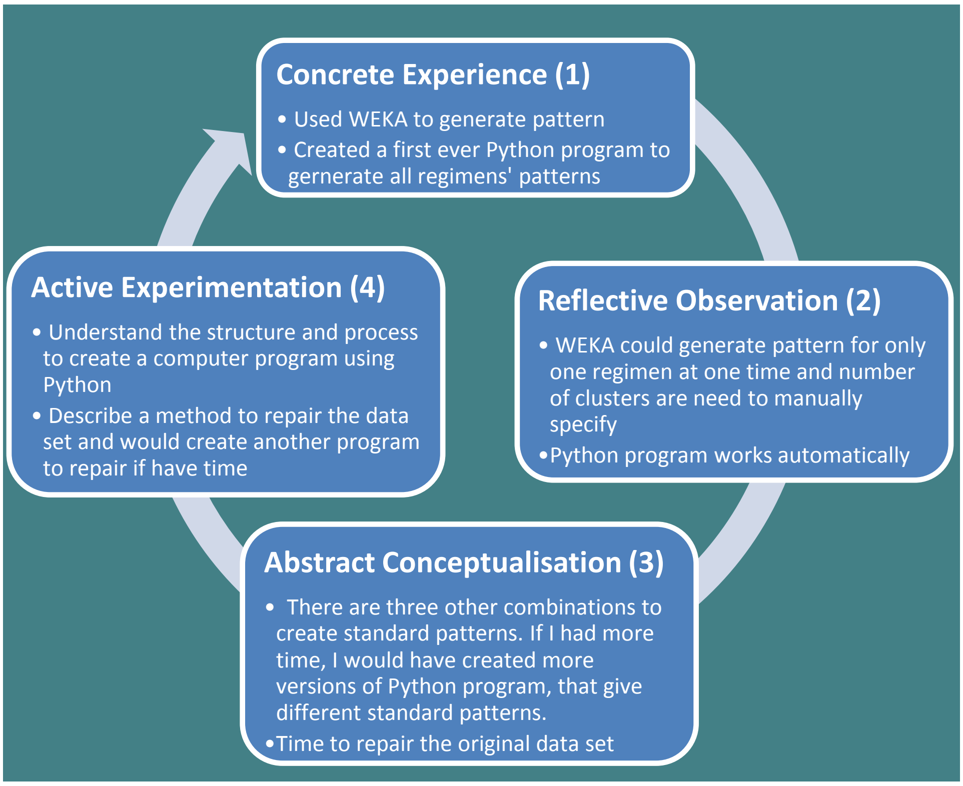
- Understand almost popular pattern recognition methods
- See the close link between Knowledge Management module and pattern recognition

The results from Microsoft Excel and PhStat2 gave me the initial understanding of cycles, and regimes' patterns. However, the observed figures did sketch out patterns quite well for simple and already consistent regimens but not every regimen. In order to find all regimens' patterns automatically, a computer program is a must. I told myself to study harder and practice more on Python. I also need to read more literatures about pattern recognition to find out how people solve the same problem as mine.

During the first two semesters at the University of Leeds with four modules in the Computing school, I have learnt business information system, techniques for knowledge discovery, programming, and e-system. Those modules are beneficial for this final thesis especially the Knowledge Management and Programming module. I have been taught the introductory of programming (Campbell et al., 2009), software engineering (Sommerville, 2008), process of knowledge discovery (Larose, 2005). All theories about software design stages, Python's codes, and data mining tools such as WEKA are totally new for me because my previous background is Commercial Management. I have created some small programs with Python, and understood fundamentally of how to make written code run. In knowledge management module, I have learnt about data mining and its task. It seems to be very close to my project. In addition, I researched about pattern recognition and its popular algorithms and methods such as decision tree, neural network, template matching, clustering.

The literatures I read helped me to explain the fundamental process to classify and cluster data in a huge data set. Some literatures I have read about pattern recognition and its methods are difficult for me to understand because they contain lots of mathematics, and computing terminologies. I discussed with my supervisor, and she told me about other possible aspects of this project closely related to business management, which makes me more confident. Have researched around operation management in hospital, I set one of the milestones in the schedule: operation management. But it is a huge area to research and may not solve the main problem of pattern recognition. Then, I took the lecture notes of Knowledge Management/ data mining to review and did extensive research on the topic. The reading gave me clearer ideas of clustering and how to use WEKA to find standard pattern. The algorithms in pattern recognition guide me to think of possible implementation in my Python program.

Learning Cycle 2:



Concrete Experience (1)

- Used WEKA to generate pattern
- Created a first ever Python program to gernerate all regimens' patterns

Reflective Observation (2)

- WEKA could generate pattern for only one regimen at one time and number of clusters are need to manually specify
- Python program works automatically

Abstract Conceptualisation (3)

- There are three other combinations to create standard patterns. If I had more time, I would have created more versions of Python program, that give different standard patterns.
- Time to repair the original data set

Active Experimentation (4)

- Understand the structure and process to create a computer program using Python
- Describe a method to repair the data set and would create another program to repair if have time

Though it took me much more time than expected to create a program by Python, I have created the first ever program in my life. The program does generate all regimens' patterns automatically. I also used WEKA to generate standard patterns of some regimens. The results of two programs are the same.

During semester 2, I have been taught to use WEKA but I did not fully understand how it can do clustering, classifying, etc. After an extensive research, I now fully understand the algorithms behind WEKA explorer. I gain more knowledge about knowledge management and its techniques by myself than from lectures. I realized that classes just gave me the very introductive idea and it required me to self-study, to deeply research and be able to implement it. Python is not hard as I thought from the beginning. If I keep practicing, I can do programming. Instead of fear feeling with programming, I am now much more confident in it. The Python program proves that I can do programming.

There are three other possible methods that can produce different standard patterns. I do not have enough time to implement all of them. However, in the future, I would develop more versions of pattern recognition program in Python, or in other language if I have chance to learn such as Java. It was about a month before the deadline, I finish two-third of requirements. In meeting with my supervisor and accessor, we agreed to change the final requirement from actually create program to repair the data set to describe a method to repair the dataset. I have already brainstormed some ideas to repair the data set in about week 7 of semester 2. The last month was just for writing and completing report.

In conclusion, it may derive from my own experience of being student at the University of Leeds and especially at the School of Computing. It was a challenge for me to study programming as a previous commercial management background. There are several things that I would love to do if I had more time such as implement other methods to find standard pattern, create a program to repair the original data set. Before studying this course, “programming is impossible for business student” but now I totally agree with the sentence “nothing is impossible”; I’m a business student and I did really code a computer program. And finally, the most important achievement for me is the confidence and knowledge of brand new field: Computer Science.

Appendix B: The Interim Report

See next page.

School of Computing, University of Leeds

MSc Interim Project Report

All MSc students must submit an interim report on their project to the CSO ***by 9 am*** ***Friday 18th June.*** Note that it may require two or three iterations to agree a suitable report with your supervisor, so you should let him/her have an initial draft well in advance of the deadline. The report should be a maximum of 15 pages long and be attached to this header sheet. It should include:

- the overall aim of the project
- the objectives of the project
- the minimum requirements of the project and further enhancements
- a list of deliverables
- resources required
- project schedule and progress report
- proposed research methods
- a draft chapter on the literature review and/or an evaluation of tools/techniques

The report will be commented upon both by the supervisor and the assessor in order to provide you with feedback on your approach and progress so far.

Student:	Huyen Phuong Thi PHAN
Programme of Study:	**MSc in Computing and Management**
Title of project:	Data management: processing records of hospital appointment
Supervisor:	**Natasha Shakhlevich**
External Company (if appropriate):	**N/A**

Signature of student: **Date:**

Supervisor's comments on the Interim Report

Assessor's comments on the Interim Report

Table of Contents

List of Tables

List of Figures

General Information

Problem Description

Hospital usually serves patients as following model:

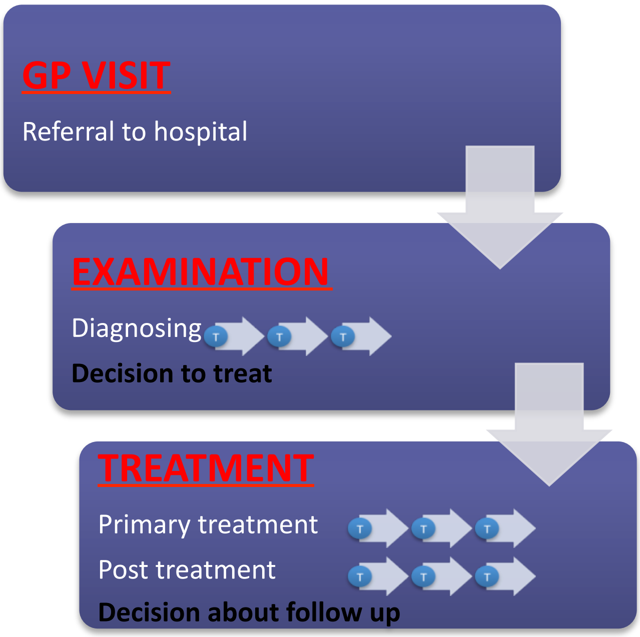


Figure 1: Hospital process of treatment

First of all, a patient should meet General Practioner (GP) for general diagnose and be referred to hospital. There are several meetings to examine and diagnose illness then a treatment schedule will be given. From the time of meeting GP to the first day of treatment should no longer than 18 weeks and 2 weeks for cancer. Primary treatment and post treatment both are prescription of multiday pattern. Each regimen has its own routine such as below regimen has repeated pattern on the 1st, the 8th, the 15th and the 22nd day of one cycle and it may be required more than one cycle to complete a patient's multi-day treatment.

Table 1: Example of multi-day pattern treatment

Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Treatment	■							■							■							■

The patient should strictly follow their multiday pattern with prescription of a regimen. Traditionally, the hospital books only one appointment for one patient at a time even if it is known that several visits are needed, which means patients would be noticed a next appointment in advance.

Pathway management method suggests that patient should be informed their multiday pattern in advance and all of their appointments for multiday treatment should be booked in advance. This method would help both patients and hospital effectively and efficiently manage their time and effort. This may help reduce number of nurses outsourcing.

The dataset (.xlsx file) is provided by hospital with information about patients and the records of their treatments for a period of 4 months in 2008 (from May to August). However, the dataset was recorded manually by nurses in hospital's paper diary. There were various reasons for imperfect dataset, e.g. patients did not show up on appointed date or they came earlier or later than appointed date or patient's state of heath may deteriorate so that nurse corrected actual treatment date in hospital's diary. That made dataset not showing correct pattern for regimen.

Project Aim and Objective

Due to inaccurate historical patients' appointments recorded in hospital, this project will therefore perform analysis of the given dataset and repair it automatically.

The **project objectives** are:

- Investigate applications of statistics for management by comparing and contrasting several statistics tools, including PhStat2, R, Microsoft Excel function.
- Evaluate and analyse statistical and visualised results which help to understand meaning beyond the dataset and find proper way to display pattern.
- Evaluate potential solutions, programming languages and tools, and select the most appropriate method to implement
- Develop several methods to solve problems based on research findings.
- Identify some knowledge management tools such as data mining especially for pattern recognition

Minimum Requirements and Further Enhancements

The minimum requirement of this project is

1. Report with analysis of the whole dataset to find standard pattern for each regimen
2. Produce an automatic repair inaccurate historical data/records of patients' hospital appointment
3. Report with analysis of original and repaired data.
4. Report with analysis of different data mining techniques especially pattern recognition

The successful achievement of this project will help:

- A software developer with perfect dataset which can be used to perform experiments evaluating scheduling software.
- A hospital manager understands the rate of patient arrivals based on historical data and analysis, nurse peak time and regimen usages. So then they can manage logistic better to avoid under-staff, utilise staffs' working hour and fulfil patient's multiday pattern treatment.
- The public to have better schedules for their treatment by successfully apply pathway method and achieve national target (18 days from GP visit to the first treatment).

This project can be deeper analysed to understand operations management aspect of hospital. In addition, a simple booking appointment database with friendly user interface could be developed.

List of deliverables

1. Report on analysis of different data mining algorithms: including nearest neighbour, association rules and decision trees
2. Report on analysis of dataset using statistics tools
3. Produce the repaired dataset with consistence pattern for each regimen.
4. Automatic solutions (software) to find standard pattern for each regimen in the given dataset and repair it to be ideal

Resources required

In addition to the most essential application for this project, Microsoft Excel (version 2007), there are several special add-in applications of MS. Excel needed such as Analysis Toolpak add-in and PhStat2 (already installed on personal computer). Python will be the main development environment, which is Open Source. The software was installed as extension for Eclipse. Additionally, Weka – a data mining tool, was already installed on my PC and will be used to visualise the given dataset.

Project schedule and progress report

Schedule

Table 2: Project plan

Weeks / Activities	April				May				June				July				August				Sept
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1
Research	🟩	🟩	🟩	🟩	EXAM			🟩	🟩	🟩	🟩	🟩	🟩	🟩							
- Statistics for management	🟩	🟩	🟩	M1																	
- Pattern recognition		🟩	🟩	🟩				M2													
- Information system								🟩	🟩	🟩	M3										
- Operation management										🟩	🟩	🟩	🟩	M4							
Interim Report								🟧	🟧	🟧											
Solution development												🟦	🟦	🟦	🟦	🟦	🟦				
- Find standard pattern												🟦	M5								
- Data repair														🟦	🟦	M6					
- Enhancement															🟦	🟦	M7				
Evaluation																		🟩	M8		
Report Write-up								🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	
Report submission																					⬛

Milestones:

M1: understand roles and applications of statistics in general management, study about special statistical function of Microsoft Excel to present dataset in meaningful way

M2: understand pattern recognition algorithms

M3: study information system to arrange and store data in managerial way

M4: further development on operation management such as forecast patient arrival
M5: code "pattern recognition" solution part of the project
M6: code "data repair" solution part of the project
M7: develop extensions for project such as develop a database to record patient and their treatment or a forecast extension for hospital manager to see future arrival of patient.
M8: evaluate research and software solutions

Progress

Up to date, I have completed milestone M1 and M2 and M3 is half-way reviewed. Thus, I seem to be quite a bit behind the schedule. I am feeling the pressure due to lots of time spending to understand many computing methodologies and algorithms, which are very different to my previous business management background. Writing up report took me more time than expected as I need more time to research and find literatures. However, I have completed in advance the milestone M5 with result as a program to read the excel file and find pattern automatically. In this interim report I have included both my background research chapter and the evaluations of tools/ techniques that I used to visualise the given dataset and find standard pattern. I also included a full analysis for one specific regimen as an illustration for my proposed solution.

Proposed research methods

The research methods for the project are both primary and secondary research with personal experiment for dataset. I also reviewed related documentation and online materials, and literature. The literature reviews will include journal articles, research papers, magazine articles, lecture notes and books.

The library of the University of Leeds is the most important source for information as they provide not only thousands of books in computing and business field but also access to online journals and databases. Google Scholar is also useful to find literatures. EndNoteX3 is also used to organise references for this project.

The research areas are quite different and therefore have field-specific useful sources. For exploring the concept of data mining, pattern recognition, helpful journals include IEEE, Web of Science and online search engine, conferences such as the DAGM Symposium, which provide research papers in relevant field. Other areas of research such as programming tools, and technical skills for project are supported by books on development, online programming materials such as Python and online tutorials on the subjects.

Literature review and evaluation of tools/techniques

Statistics for management:

It is essential for manager to get information from large amounts of data to make better decision - making. One of the most important issues that managers must deal with is how to effectively use information when making decisions. Here they face 3 problems:

- With computerized management information systems and a large number of databases widely available on the web, managers often feel there is too much evidence available. If they are to use evidence when making decisions they must process or transform this data into manageable units which we call information.

- Managers must however do more than passively process available data. They must also actively redesign management information systems to ensure that more relevant data is generated.

- Their most important responsibility is to communicate to others in their organization the need for these activities and the reasons why they have made the decisions based on this evidence.

In order to fully investigate the meaning of data, managers need to learn statistics and have statistical thinking. It helps to present large dataset in informative ways such as graphs, charts; to conclude meaning of large populations based on smaller samples; to improve processes; and to forecast the development trend of the dataset (Levine et al., 2002).

There are two main types of statistics: *descriptive statistics* and *inferential statistics*. Descriptive statistics focuses on collecting, selecting, summarising to present data and describe features of dataset. One of the descriptive statistics' tools is graphical technique, which allows user to present the data in such a useful format such as line chart, bar chart, pie chart, scatter plot, etc. Another tool is numerical technique, which summarise data. For example, it calculates average, mean, mode, standard deviation, etc. Thus, the reader can understand the data better. Inferential statistics is another branch of statistics, which draw characteristics of a huge population based on smaller sample. For instance, instead of interviewing 5,000 employees for their satisfaction toward company's HR policies, they can better interview 500 employees from all departments to have estimated conclusion (Lind et al., 2006)

Statistics has incredible applications in all industries of the economy. First of all, financial management is the function of every business. It provides more precisely decisions in providing products and services, e.g. capital budgeting, working capital management, sale forecast, revenue estimation. Secondly, marketing management also need statistics for development of products and services by positioning, market segmentation to see profitable segment to invest. Thirdly, operations management including logistics, process planning, and inventory management also need statistics to determine optimum output and forecast amount of required input for production (Keller and Warrack, 2004).

Knowledge Discovery Process

Knowledge Discovery in Databases (KDD) is the method to extract knowledge from databases. There are several research aspects of data mining such as machine learning, pattern recognition, databases, statistics, artificial intelligence, knowledge acquisition for expert systems, and data visualisation. The process of finding and interpreting patterns from raw dataset is illustrated as below figure (Bulpitt, 2010).

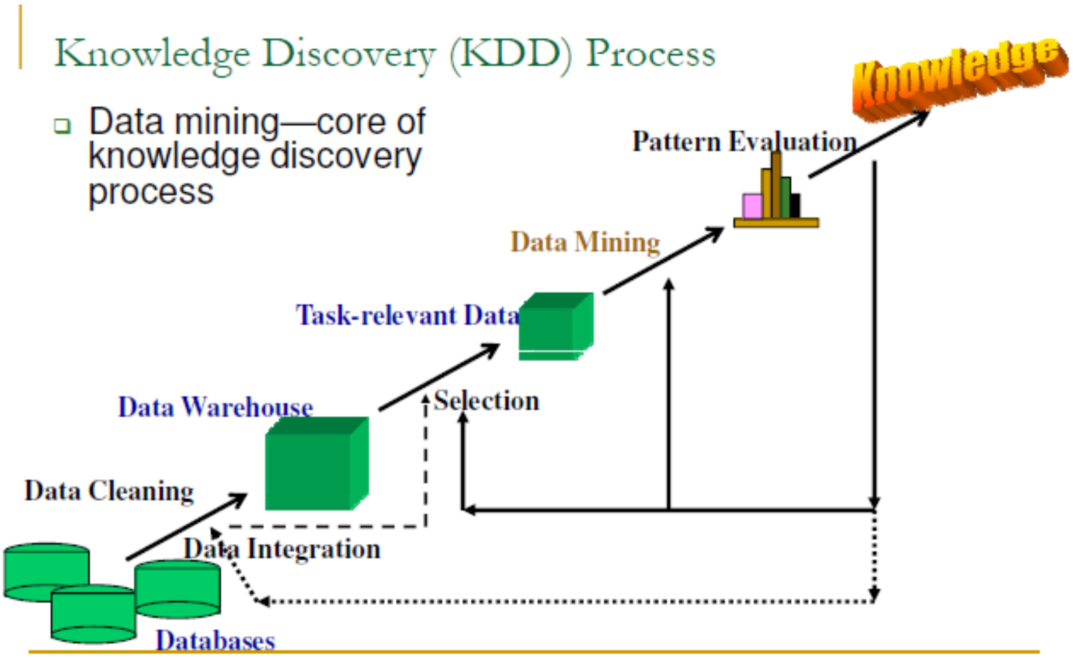
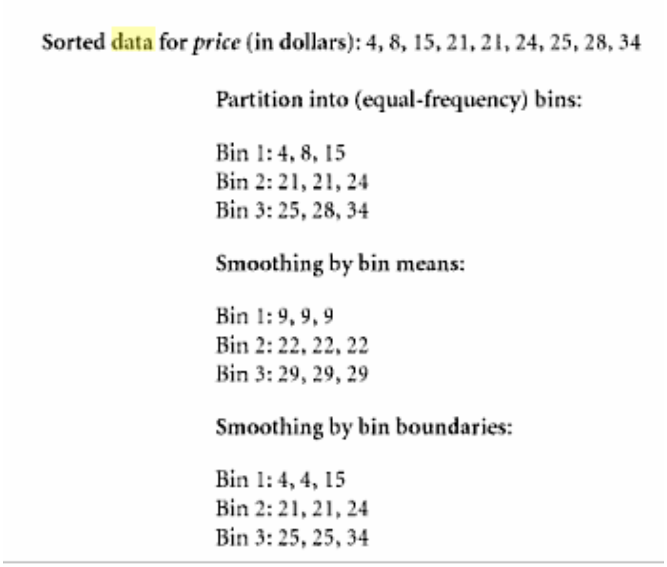


Figure 2: Knowledge Discovery in Databases (KDD) process

The Knowledge discovery process has 7 steps:

1. Data Cleaning: to remove noise and inconsistent data. This process often takes 60% of time and effort.
2. Data Integration: where multiple data sources may be combined
3. Data Selection: where data relevant to the analysis task are retrieved from the databases
4. Data Transformation: where data are transformed or consolidated into forms appropriate for mining by performing summary
5. Data Mining: where intelligent methods are applied in order to extract data patterns
6. Pattern evaluation: to identify the truly interesting pattern representing knowledge based on some measures
7. Knowledge presentation: where visualisation and knowledge representation techniques are used to present the mined knowledge to the user

Data usually are "dirty", which means incomplete, inconsistent, and noisy (e.g., containing errors or outliers). The dirty data may be caused by human error at data entry, hardware, software problems, different data sources, duplication of records, and data transfer. That explains why it takes majority of time and effort to clean data by filling in missing values, identifying outliers, smoothing out noisy data, correcting inconsistent data, removing redundancy. There are many ways to handle missing data such as fill in it manually or automatically, or ignore it. It is more complex with noisy data. It can be smoothed by one of four techniques including binning, regression, clustering and combined computer and human inspection. Binning method needs firstly sorting data and partition into equal frequency or *bins* then data can be smoothed by bin means, median or boundaries. For example,



Sorted data for *price* (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into (equal-frequency) bins:

Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing by bin means:

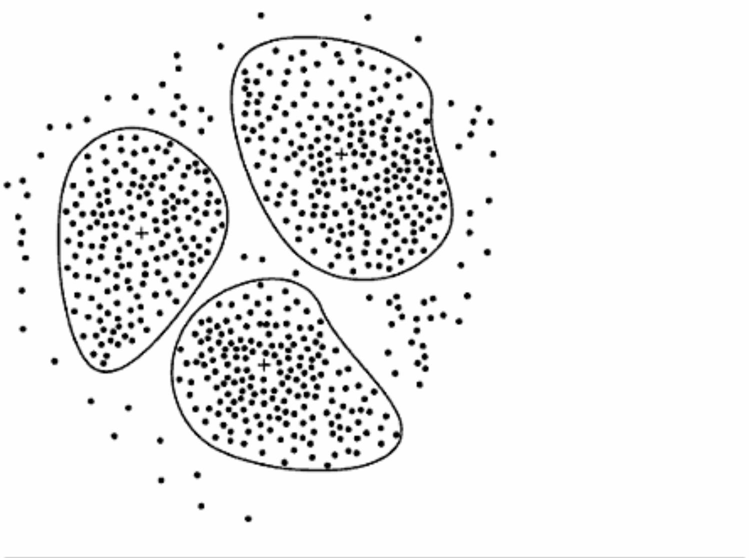
Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Smoothing by bin boundaries:

Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

Figure 3: Binning methods for data smoothing

Regression is method to smooth data by fitting it into regression functions. Clustering method detects and removes outliers of data.



A 2-D plot of customer data with respect to customer locations in a city, showing three data clusters. Each cluster centroid is marked with a "+", representing the average point in space for that cluster. Outliers may be detected as values that fall outside of the sets of clusters.

Figure 4: Clustering method for data smoothing

(Han and Kamber, 2006)

Data mining is defined as a one of steps in the process of discover non-trivial and implicit knowledge or patterns from data. It analyses and characterises data, e.g., urban vs. rural areas. It also discovers the common patterns, association, and correlations among attributes in databases. The input to a data mining algorithm is in the form of a set of examples, or instances. Each instance has a set of features or attributes, which form *pattern* for that instance (Symeonidis and Mitkas, 2005).

Microsoft Office Excel analysis tools

Microsoft Office Excel is very powerful application for statistics and visualise data. The graphical procedures are used to organise and present data. One of the most popular tools in MS. Excel is to draw Charts, which includes bar chart, line chart, pie chart, column chart, area chart, scatter plot, and so on. It makes very easy to visualise data in informative way.

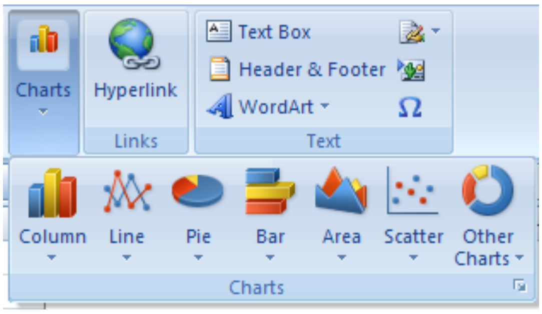


Figure 5: Microsoft Office Excel Charts Insert

The Analysis ToolPak is a MS. Excel add-in program. It includes many analysis tools such as Descriptive Statistics, Histogram, Correlation

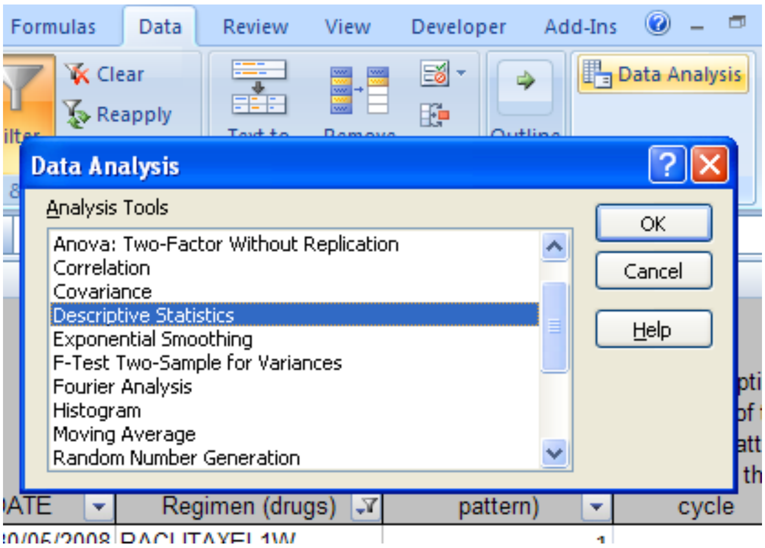


Figure 6: Data Analysis ToolPak Add-in program

PhStat2

PhStat2 is a statistical add-in for MS. Excel. It helps produce many procedures not included in standard MS. Excel. Two techniques will be used in this report as Frequency Distribution and Stem-and-Leaf Display. A frequency distribution counts the number of observations that fall into each of a series of intervals, called **classes** that cover the complete range of observations. A Stem-and-Leaf split observation into two parts: a stem and a leaf. There are several ways to slit up. For example, the number 12.3 can be split into a stem "12" and a leaf "3". This way divides number by the decimal. 12.3 can also be split into a stem "1" and a leaf "2". This definition the stem is the number of tens and the leaf is the number of ones. By using stem-and-leaf display, the given dataset can be clustered and we can see very clearly the popularity of each class or stem.

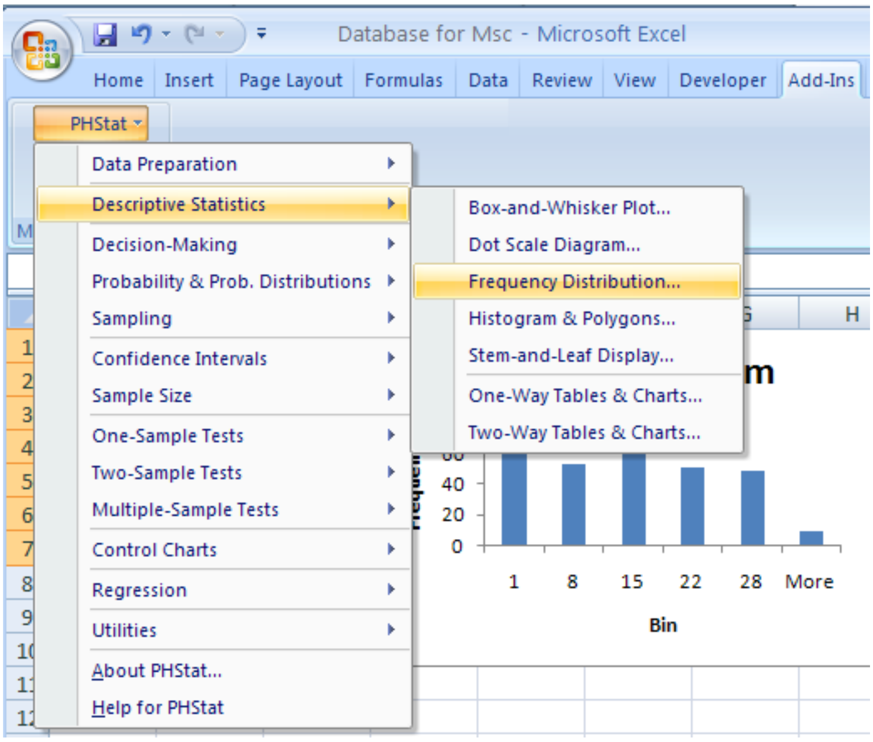


Figure 7: The PhStat menu

Proposed solution

Finding patterns

Each row in the dataset has 8 attributes: Patient's name, Diagnosed date, Regimen (drugs), The cycle number (the number of the current prescription of the multiday pattern), day number of the multi-day pattern related to the cycle, Appointment date, The number of days a patient has to wait from decision to treatment until the first visit day, and The type of the Multiday/Intraday Pattern. The most important column is 'Appointment date'. Each patient is booked several dates on several cycles for their multiday pattern. For example, Patient A has 30-day pattern with appointments as following table:

Table 3: Example of multi-day pattern treatment for 1 patient

Cycle number	DAY Number of the multi-day pattern related to the cycle	Appointment date
1	1	28/05/2008
1	2	29/05/2008
1	3	30/05/2008
1	6	02/06/2008
1	7	03/06/2008
1	30	26/06/2008
2	1	07/07/2008
2	2	08/07/2008
2	3	09/07/2008
2	4	10/07/2008

2	5	11/07/2008
2	28	03/08/2008
3	1	04/08/2008
3	2	05/08/2008
3	3	06/08/2008
3	4	07/08/2008
3	5	08/08/2008

The day number for this particular patient is 1,2,3,6,7,30 for cycle 1; and 1,2,3,4,5,28 for cycle 2; and 1,2,3,4,5 for cycle 3; which does not show the pattern for his/her treatment. One of the objectives of this project is to find pattern for all patients and their treatments; to repair inaccurate dataset into perfect dataset with consistent patterns.

Data analysis

Each point in the chart represented **a time**, when **a patient** visited hospital for their treatment with **a regimen**. Therefore when filtering one particular regimen, a chart will show regimen's pattern. For example, below data chart for *CARBO(AUC)21D* present its pattern is day 1 and day 21 of a cycle. However, there are some blanks between points, which is missing data and that maybe need to be fulfil in repaired dataset.

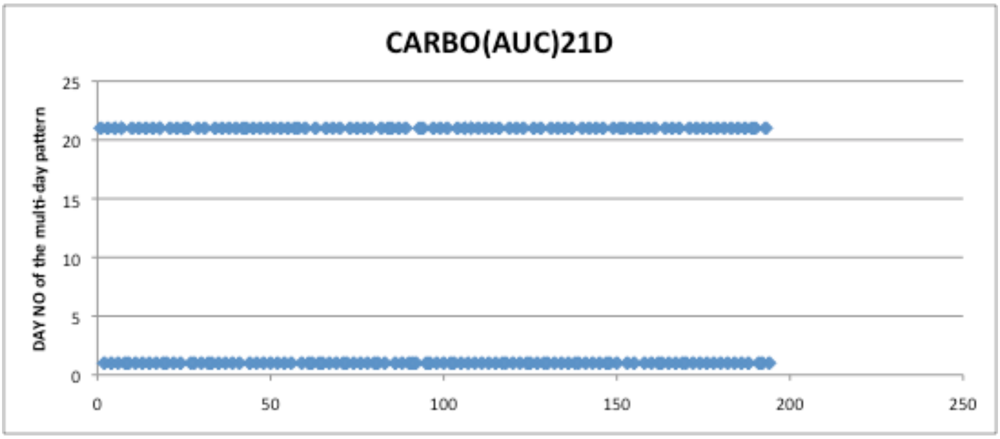


Figure 8: Scatter Plot for CARBO(AUC)21D

Another data can be plotted in several 'clusters', For instance, PACLITAXEL1W has data presented in 4 'clusters': 1, 8, 15, 22, 28. However, data in cluster 1 all has same value "1", other clusters have several points have lower or higher value than majority.

If we assume that majority of points representing regimen's pattern then the points not equal to 1, 8, 15, 22, 28 need to be amended.

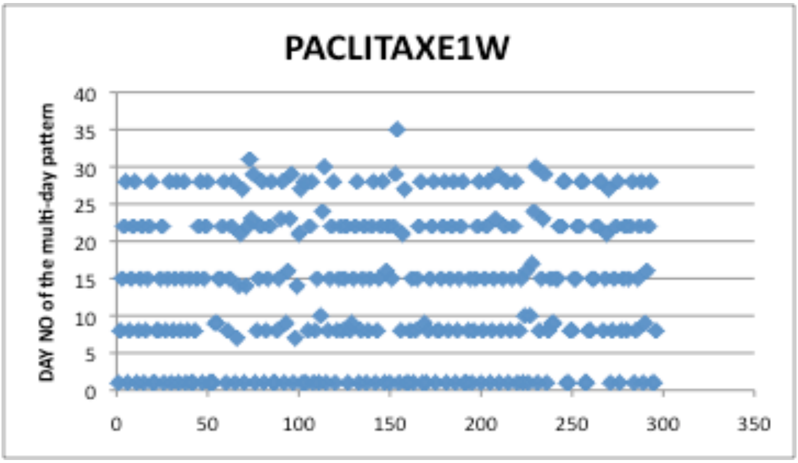


Figure 9: Scatter Plot for PACLITAXE1W

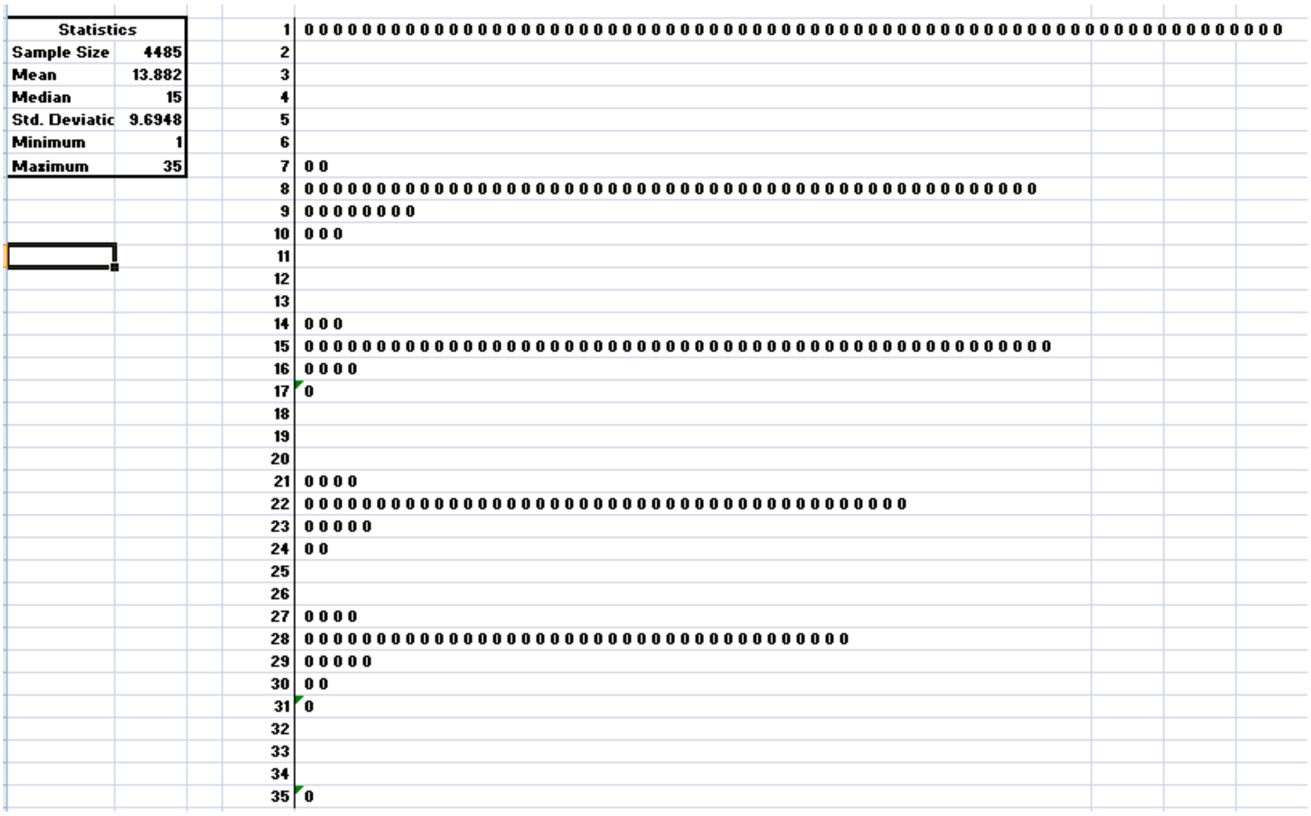
We can use PhStat2 add-in application for MS Excel to do some visualization for dataset. For example, with PACLITAXE1W regimen above, the first observation for its pattern is day number 1, 8, 15, 22, 28. We group 296 rows into 5 "bins" and calculate the frequencies of 5 bins for this regimen.

Table 4: Frequencies Distribution for PACLITAXE1W for bins (1,8,15,22,28+)

Frequencies for PACLITAXE1W			
Bins	*Frequency*	*Percentage*	*Average date*
1	68	22.97%	1.000
8	53	17.91%	8.188
15	66	22.30%	15.017
22	51	17.23%	22.094
28+	58	19.59%	28.294

Calculate the average date for each bin, we have rounded to **1, 8, 15, 22, 28**.

The stem-and-Lead display as following demonstrates more clearly variations of the multiday-pattern for PACLITAXE1W. The percentage of variation (numbers are not equal to 1, or 8, or 15, or 22, or 28) is 45/296 (15.2%).



Statistics	
Sample Size	4485
Mean	13.882
Median	15
Std. Deviatic	9.6948
Minimum	1
Maximum	35

```
 1| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 2|
 3|
 4|
 5|
 6|
 7| 0 0
 8| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 9| 0 0 0 0 0 0 0 0
10| 0 0 0
11|
12|
13|
14| 0 0 0
15| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
16| 0 0 0 0
17| 0
18|
19|
20|
21| 0 0 0 0
22| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
23| 0 0 0 0 0
24| 0 0
25|
26|
27| 0 0 0 0
28| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
29| 0 0 0 0 0
30| 0 0
31| 0
32|
33|
34|
35| 0
```

Figure 10: Stem-and-Leaf Display for PACLITAXE1W

With 45/296 (15.2%) points not equal to 1, 8, 15, 22, 28 we could make a decision of standard pattern for PACLITAXE1W is **1, 8, 15, 22, 28**

Instead of representing pattern for PACLITAXEL1W as a list of treatment date in a cycle, we can represent it as list of time-lag between 2 treatment dates. For example,

Table 5: Different way to represent pattern

Date of treatment	1	8	15	22	28
Pattern	**6 days**	**6 days**	**6 days**	**5 days**	

Applying this method for some patients using this regimen, we have the following table:

PACLITAXEL 1W		
	Pattern	Treatment date
1	6, 6, 6, 5	1,8,15,22,28
2	6, 6, 6, 5	1,8,15,22,28
3	6, 6, 6, 5	1,8,15,22,28
4	7, 5	1,9,15
5	7, 5, 6, 5	1,9,15,22,28
6	6, 6, 6, 5	1,8,15,22,28
7	5, 6, 6, 5	1,7,14,21,27
8	6, 13, 5	1,8,22,28
9	6, 6	1,8,15

Different ways to calculate pattern.

The next step, I will eliminate all cases which do not have enough 5 treatment dates (5 is the longest set of number of treatment dates) such as record number 4, 8 and 9. Those with enough 5 dates recorded, now can used to calculate **the mean** for each date. I have list of all 5 means are 1, 8, 14.8, 21.8, 27.8 respectively and can be rounded up to 1, 8, 15, 21, 28. This is the pattern for regimen PACLITAXEL1W.

Jagannathan and Petrovic (2009) suggested a slightly different way to find pattern. Instead of using *the mean*, their method used **the mode** to represent pattern. In fact, the stem-and-leaf display in Figure 10 shows very clearly the mode for this regimen. Points almost appear in date number 1, 8, 15, 22, 28. This method also finds the same pattern for regimen PACLITAXEL1W.

Automatic program to find pattern (milestone M5)

```
Initial RegimenCycleList
LongestCycleList = getLongest Cycles(RegimenCycleList)
For i from 0 to  LongestCycleLength
        Standard[i] = Average (LongestCycle[i])
End For
```

Explaination:

1. transfer normal .xls file to .csv
2. Read the .csv file
3. For each regimen, get all records has the most number of elements in the set of treatment date
4. Standard pattern for that regimen is the set of average of each element in the set .

Some observations and evaluation

1. There are some regimens recorded in only one row (1 customer, 1 cycle, and 1 date of treatment). There is nothing to compare and contrast. So that those regimens only need one treatment.

Table 6: List of all regimens with one treatment only

Regimen (drugs)	No. of rows
AN3ELDOP	1
AN4	1
BLOOD	1
CAV	1
CETUX/IL21(PH1)	1
CIBCPPS(IBAND)	1
CISPLAT+5FU(OP)	1

GATEARMCTEXL1	1
GOSERELINLA10.8	1
PICCOLOIRPAN	1
SORCE	1
TAMOXIFENGYNMT	1
TIPMODIFIED	1
VINOR+EPI	1

2. Regimen has only 2 recorded rows:
   Each regimen has 2 rows for 1 patient, 2 cycles and the same date of treatment. Therefore, these regimens probably also have one-day pattern.

Table 7: List of all regimens with one-day pattern

Regimen (drugs)	No. of rows	No. patient
CARBO/5FU	2	1
CARBO/5FU(OP)	2	1
DARBE5004W	2	1
E/CARBO/F	2	1
FEC75	2	1
GEMTREO	2	1
IBANDRONICACID	2	1
PACLITAXEL3W	2	1
SU011248	2	1
TEMOZOL1STDOSE	2	1
TRASTUZ1WLD	2	1
TRASTUZUMAB1W	2	1
VINCRISTINEQ1W	2	1
VINOREL(BREAST)	2	1
ZALUTETRIAL	2	1

References

BULPITT, A. 2010. Knowledge Management: Data Mining Lecture Note.
HAN, I. & KAMBER, M. 2006. *Data Mining: Concepts and Techniques*.
KELLER, G. & WARRACK, B. 2004. *Statistics for management and economics*, Thomson/Brooks/Cole.
LEVINE, D., STEPHAN, D., KREHBIEL, T. & BERENSON, M. 2002. *Statistics for managers using Microsoft Excel*, Upper Saddle River, N.J. : Prentice Hall.
LIND, D., MARCHAL, W. & WATHEN, S. 2006. *Basic statistics for business & economics,* Boston [Mass.] : London, McGraw-Hill/Irwin.
SYMEONIDIS, A. & MITKAS, P. 2005 *Agent intelligence through data mining*, Springer.

MARWALA T. 2009. *Computational Intelligence for Missing Data Imputation, Estimation, and Management: Knowledge Optimization Techniques*, Information Science Reference
CHESNEY T., 2009, *Searching for Patterns – How we can know without asking*, Nottingham University Press
JAGANNATHAN R. & PETROVIC S. 2009, *Dealing with Missing Values in a Clinical Case-Based Reasoning System*, School of Computer Science, University of Nottingham

Appendix C: Pattern recognition methods (Liu et al., 2006)

Pattern recognition undergoes an important developing for many years. Pattern recognition include a lot of methods which impelling the development of numerous applications in different filed. The practicability of these methods is intelligent emulation.

1. Statistical pattern recognition
2. Data clustering
3. The application of fuzzy sets
4. Neural networks
5. Structural pattern recognition
6. Syntactic pattern recognition
7. Approximate reasoning approach to pattern recognition
8. A logical combinatorial approach to pattern recognition
9. Applications of Support Vector Machine (SVM) for pattern recognition
10. Using higher-order local autocorrelation coefficients to pattern recognition.
11. A novel method and system of pattern recognition using data encoded as Fourier series and Fourier space

Appendix D: Regimens' patterns found by Python program

Regimen's name	Pattern	Regimen's name	Pattern
5FU+FA(5DAY)	[1, 2, 3, 4, 5, 28]	GOSERELIN3.6	[14]
5FU+FACON	[1, 2, 3, 4, 6, 28]	GOSERELINLA10.8	[84]
5FU+FAWEEKLY	[1, 8, 15, 22, 29, 37, 43]	IBANDRONICACID	[1, 28]
ABC-02GEM	[1, 8, 15, 28]	ICON6CHEMO	[1, 21]
ABC-02GEM/CIS	[1, 9, 22]	ICON7ARMA	[1, 21]
AN1AND2(OP)	[1, 29]	ICON7ARMB	[1, 21]
AN3ELDOP	[1]	IMATINIB400MG	[1, 28]
AN4	[1]	IRCS(3WEEKLY)	[1, 21]
ANASTROZOLE	[19]	IRINO/TEMOZOL	[1, 2, 3, 4, 5, 8, 9, 10, 11, 12]
AVASTM	[1]	IRINO+BEV2W	[1, 14]
BEP(ADJ.)3DAY	[8, 15, 20]	IRINOTECAN(2W)	[1, 14]
BEP3DAY	[7, 8, 15, 21]	IRINOTECAN(3W)	[1, 21]
BEPACCELERATED	[7, 12]	IRMDG	[1, 14]
BEPTE3LONG	[21]	IRONIVINFUSION	[1, 21]
BEPTE3SHORT	[8, 15, 21]	LANREOTIDE60	[1, 28]
BLOOD	[1]	LANREOTIDE90	[1, 28]
BROSTEORTC62061	[1, 21]	LETROZOLE	[1, 28]
CAPECITABINE	[1, 21]	LIPODOXKAPOSI	[1]
CAPRTWEEKS1-5	[1, 35]	LIPOSOMALDOXO	[1, 28]
CARBO(AUC)21D	[1, 21]	MARS(PEM/CARBO)	[1, 21]
CARBO(AUC)28D	[1, 28]	MARS(PEM/CISP)	[1, 21]
CARBO(AUC)GCT	[1, 21]	MDG	[1, 14]
CARBO(AUC6)ETOP	[1, 2, 21]	METHOTREXATE(W)	[1, 8, 15, 22, 28]
CARBO/5FU	[1]	MITOCAP	[1, 22, 42]
CARBO/5FU(OP)	[1, 2]	MITOXANTRONE	[1, 21]
CARBO/DOX	[1, 21]	NP(OP)	[1, 8, 21]
CARBO/ETOP	[1]	OCAP	[1, 21]
CARBO/ETOPGCT	[1, 2, 3, 4, 21]	OXALICAP(2W)	[1, 14]
CARBO/GEM	[1, 8]	OXALICAP(3W)	[1, 21]
CARBO/PACLITAX1W	[1, 8, 15, 21]	OXMDG	[1, 14]
CARBO21ALLERGY1	[1]	PAC/CARBONSCL	[1, 21]
CARBO21ALLERGY2	[1]	PACLITAX/CARBO	[1, 21]
CARBO21ALLERGY3	[1]	PACLITAX90CARBO4	[1, 16, 29]
CARBO28ALLERGY1	[1]	PACLITAXEL1W	[1, 8, 15, 22, 28]
CARBOAUC5	[1, 21]	PACLITAXEL3W	[1, 21]
CARBOMV	[1, 15, 22, 28]	PAMCYCLES1-4	[35]
CAV	[1]	PAMIDRONATE(3W)	[1, 21]
CETUX/IL21(PH1)	[1]	PAMIDRONATE(4W)	[1, 28]
CETUXIMABH+N	[1, 9, 16, 23, 30, 37, 44]	PAMIDRONATE(6W)	[1, 42]
CIBCPPS(IBAND)	[84]	PAMIDRONATE(HCM)	[1, 7]
CIS/DOX(ENDOM).	[21]	PCV	[1, 42]
CIS/TOPOTECAN	[2, 3]	PICCOLOIRCS	[1, 21]
CISPLAT+5FU(OP)	[1]	PICCOLOIRINO3W	[1, 21]
CISPLATIN/5-FU	[21]	PICCOLOIRPAN	[21]

COINOXMDG	[1, 14]
COINXELOX	[1, 21]
CYCLOETOPVINC	[1, 2, 3, 4, 8, 15, 28]
DARBE3003W	[1, 21]
DARBE5003W	[1, 21]
DARBE5004W	[14]
DENOSUMABTRIAL	[1]
DOCETAX/PRED3W	[1, 21]
DOCETAXEL	[1, 21]
DOCETAXEL1W	[1, 8, 15]
DOCETAXELADJ.	[1, 21]
DOXORUBICIN	[1, 21]
DTIC	[1, 21]
E/CARBO/F	[1, 21]
EC90	[1, 21]
EOCAP	[1, 2, 21]
EOF	[1, 21]
EP(SCLC)	[1, 2, 4, 22]
EPIRUBWEEKLY	[1, 8, 15]
ERLOTINIB	[8, 22]
FEC75	[11]
FOCUS2MDG	[1, 14]
FOCUS2OXMDG	[1, 14]
FOCUSOXMDG	[1, 14]
FOXTROTOXMDG	[1, 14]
FOXTROTOXMDGPAN	[1, 14]
GATEARMATEL2	[1, 21]
GATEARMBTEFL2	[1, 14]
GATEARMCTEXL1	[21]
GEM/CARBO	[1, 8, 21]
GEM/CIS(SPARE)	[8, 21]
GEM/CIS21D	[8, 21]
GEM/SPLITCIS21	[9, 22]
GEMCAP(PANCREAS)	[1, 8, 15, 28]
GEMCISCHOLANGIO	[1, 9, 22]
GEMCITABINE	[1, 8, 15, 28]
GEMLOWDOSE	[1, 15, 28]
GEMTREO	[1]
QUASAR2BEV+CAP	[1, 21]
REO011LEVEL+3	[1, 2, 3, 4, 5]
ROFERONINDUCT	[1, 5, 8, 14]
ROFERONMAINT	[1, 28]
SATINEC	[1, 21]
SOCCARNP	[1, 8, 21]
SOFEAFULVMAINT	[1, 28]
SORCE	[1]
STAMPEDEARMC	[1]
STIMUVAXCYCLO	[1, 3]
SU011248	[22]
SUNITINIBINDUCT	[14, 15, 42]
SUNITINIBMAINT.	[1, 42]
TACT2CAPECIT	[1]
TACT2CMF	[1, 8, 28]
TACT2EPI(3W)	[1, 21]
TACT2EPI+G(2W)	[1, 14]
TAMOXIFENGYNMT	[56]
TE23B15EP	[9, 16, 22]
TE23CBOP	[10, 24, 31, 38, 44]
TEMOZOL1STDOSE	[1, 28]
TEMOZOLOMIDE	[12]
TIPMODIFIED	[21]
TOPOTECAN1W	[1, 8, 16]
TRASTADJMD4#	[22, 43, 64, 84]
TRASTUZ1WLD	[1, 7]
TRASTUZADJ.LD	[1, 21]
TRASTUZUMAB1W	[1, 10]
TRASTUZUMAB3W	[1, 1, 21]
VINCRISTINEQ1W	[1]
VINOR+EPI	[14]
VINOREL(BREAST)	[1, 8]
VINORELBINEORAL	[1, 8, 15, 21]
ZALUTETRIAL	[1]
ZOLEDRONATE3W	[1, 21]
ZOLEDRONATE4W	[1, 28]
ZOLEDRONATE6W	[1, 42]