**PROJECT 2**

**CSE 6339: Advanced topics in Databases**

**Methods of Knowledge Discovery in Healthcare**

**TEAM 101**

**TASK 1**

PRIYA DHARSHINI GUNASEKARAN - 1001054128

REHANA DEVANI - 1001100807

ARAVIND MUNIYAPPA - 1001050288

SUJAY NATARAJAN – 1001086537

**TASK 2 - REGRESSION**

ARAVIND MUNIYAPPA - 1001050288

SUJAY NATARAJAN – 1001086537

**TASK 3 – ASSOCIATION RULE MINING**

PRIYA DHARSHINI GUNASEKARAN - 1001054128

REHANA DEVANI - 1001100807

**Tools Required for tasks 1 and 3:**

1. Eclipse IDE
2. Apache Tomcat Server
3. MySQL Workbench

**TASK 1**

1. Import the folder task 1 in Eclipse. (Task 1 is in the deliverable)
2. Change the credentials by entering your MySQL username and password.
3. Index.php contains the functionality for user to select Age, Sex and Admitting Diagnosis code. main2.css does styling for this page. On clicking submit, control goes to Result.php
4. Result.php contains the MySQL code for all the queries. Results are generated. main\_result.css does styling for this page.
5. Adm\_diag.php calculates top 10 diagnosis codes with highest “death on discharge” ratio. Main\_adm\_diag\_css does styling for this page.
6. Adm\_diag.php calculates top 10 diagnosis codes with highest “death on discharge” ratio. Main\_adm\_diag\_css does styling for this page.
7. exp\_diag.php calculates top 10 most expensive diagnosis codes. Main\_exp\_diag\_css does styling for this page.
8. high\_los.php calculates top 10 diagnosis codes with highest average length of stay. main\_high\_diag\_css does styling for this page.

**TASK 3**

1. Give the credentials to connect to MySQL workbench such as username and password.
2. Import the folder named Task3Web in Eclipse. (Task3Web folder is in the deliverable)
3. It contains one servlet file named apriori, 2 JSP files named AprioriInput.jsp and Result.jsp
4. Servlet file is present in Task3Web -> src -> org -> apriori -> apriori
5. JSP files are present in Task3Web -> WebContent -> AprioriInput.jsp and Result.jsp
6. For each combination different CSV files are generated which contains key and value of all the pairs of selected attributed.
7. AprioriInput.jsp is used to take the input and Result.jsp displays the output.
8. To execute, open the AprioriInput.jsp file, right click and select Run as. Then run on server.
9. The complete functionality the algorithm is in the servlet class and it is explained below.

In this task, we implemented association rule mining functionality. A web interface is created where users selects the features of his interest as well as support and confidence thresholds. The selected attributes are stored in two arrays and all possible unique combinations are found. CSV files are created for each possible combination. Values from array are split using line.split() to store values in CSV. HashMap is used to find the frequency of each unique combination based on selected attributes. This value is divided by total number of values. This gives the support of each rule. Rules are filtered based on the minimum support value provided by user.

Confidence is calculated for all the rules by dividing the number of times each combination of rule occurred by total number of time the antecedent of rule is present. The resultant value of confidence is filtered based on the minimum confidence value given by the user and only the values satisfying the criteria are displayed. Functionality for above is implemented using multiple mappers, each calculating different parts of the rule and are split by line.split() to store temporary results in CSV files thereby giving as output only the rules which satisfy minimum support and confidence values.

ResultMap and combList is used to store the key and value of all the rules. Hence rules are restricted based on support and confidence value given by user. We have not used any predefined method and the code is developed from scratch.

**Technologies Used for Task 2:**

* Java
* JSP
* Weka
* Weka source jar files and servlet jar files.