**Project 3 – Association Rule Mining**

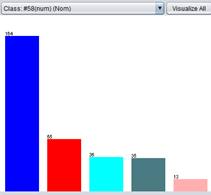
**CS548 Knowledge Discovery and Data Mining - Fall 2016**

**Prof. Carolina Ruiz**

**Students:** Mu Niu, Rohitpal Singh, Suchithra Balakrishnan

|  |  |  |
| --- | --- | --- |
| **Dataset :**   * Dataset Description * Data Exploration * Initial Data Preprocessing (if any) | /05  /10  /05 | |
| **Code Description:** Association Rules and Classification Association Rules | **Weka**  /20 | **Python**  /20 |
| **Experiments:**   * Guiding Questions | /10 | |
| * Sufficient & coherent set of experiments | /15 | /15 |
| * Objectives, Parameters, Additional Pre/Post-processing | /15 | /15 |
| * Presentation of results | /15 | /15 |
| * Analysis of individual experiments’ results | /15 | /15 |
| Quantitative Analysis of Results and Discussion | /20 | |
| Qualitative Analysis of Results, Discussion, and Visualizations | /20 | |
| Advanced Topic | /30 | |
| Total Written Report | /260 = /100 | |

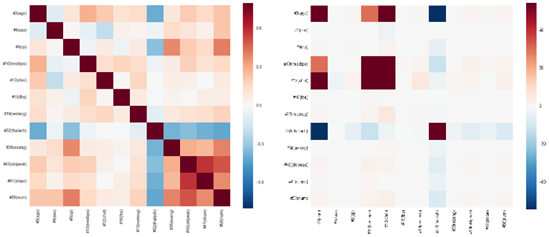
**Dataset Description, Exploration, and Initial Preprocessing: (at most 1 page)**

**[05 points] Dataset Description: (e.g., dataset domain, number of instances, number of attributes, distribution of target attribute, % missing values, …)**

This dataset describes some of the mainly causes for heart disease, such as age, resting blood pressure, serum cholesterol and fasting blood sugar.

The target attribute is prediction of whether a particular person have potential risk to have heart disease or not (0 = false, 1,2,3,4 = true in different level). There are in total 303 instances, 14 attributes in the dataset. There are in total 164 instances are false in target attribute. The dataset has 6 missing values in total, 4 in attribute #44(ca) and 2 in attribute #51(thal).

**[10 points] Data Exploration: (e.g., comments on interesting or salient aspects of the dataset, visualizations, correlation, issues with the data, …)**

We calculated correlation matrix and covariance matrix of the dataset treated all attributes as numerical ones.

We found that the five attributes, #9(cp), #40(oldpeak), #41(slope), #44(ca) and #51(thal), are highly correlated with the target attribute. From the covariance matrix, we can see that the 4th and 5th attributes, resting blood pressure and serum cholesterol, change to the same direction, which suggest they can represent each other.

The interesting thing is that although #16(fbs) doesn’t show significant correlation with target attribute, it is an important attribute to predict whether this person have higher risk to predict heart disease or have other health problem.

**[05 points] Initial data preprocessing, if any, based on data exploration findings: (e.g., removing IDs, strings, necessary dimensionality reduction, …)**

We changed the values in nominal attributes to be texts so that they can be easier to read. For target attribute, although it has values of 0, 1, 2, 3 and 4, it appears we just need to consider whether this particular person have high risk to get heart disease or not. So we replaced value 0 in target attribute to be false and value 1, 2, 3, and 4 to be true for further processes. We removed the 6 instances with missing values in attribute #44(ca) and 2 in attribute #51(thal), so that there are in total 297 instances after preprocessing.

We didn’t do any dimensional reduction in initial data preprocessing since the 14 attributes had already been the result of their dimensional reduction from 76 attributes they got. And we didn’t want to lose much information on the initial preprocessing.

**Weka Code Description: Inputs, output, and process followed by Weka’s code to construct the association rules (at most 2/3 page)**

**[10 points] Association Rules Code Description:**

**Inputs:** Preprocessed dataset and Apriori Algorithm as an associator with default parameters.

**Process:** Apriori Algorithm takes instances as an input. It checks if lower bound is equal to at least one instance If not then lower bound is set to (1/#instances) and Initialize the minSupport value to (1- delta) where default value of delta=0.05 so initial value of minSupport = 0.95 . From computed value of minSupport, necSupport is calculated by using (minSupport \* #instances + 0.5) that gives similar information as minSupport but in terms of number of records. In each cycle, candidate list of K items is generated from which frequent list is extracted using minSupport**.** From the frequent list, candidate and frequent list of K+1 itemsets is generated by merging itemsets until candidate and frequent list is empty. If merging leads to many and large itemsets then pruning is performed. The final frequent list in each cycle is used to form rules based on confidence metrics. This process continues till number of rules specified as parameter in Apriori Algorithm achieved and minSupport is greater than lowerBoundMinSupport. If desired rules are not formed using initial minSupport then minSupport is reduced by 0.5 each time but if minSupport goes below lower bound, then it is set to lowerBound to find remaining rules.

**Outputs:** Minimum Support= 0.25, Minimum Metric <confidence> =0.9, Number of cycles performed=15, Levels=15 and Rules=10.

**[10 points] Classification Association Rules (CARs) Code Description:**

**Inputs:** Preprocessed dataset and Apriori Algorithm as an associator with car parameter as True.

**Process:** CAR is used to perform classification based on association rules. It is a supervised method taking into consideration the target attribute. CAR works same as Apriori while initializing minSupport but CAR must have metrics as confidence. As CAR uses target attribute it checks for the parameter class-index and choose last attribute in the dataset as target attribute. All attributes except class is stored in one variable and class in other variable. For each cycle, it generates Ksets or candidate list of K items and extract frequent items using necSupport= (minSupport \* #instances) that is min support in terms of records. This frequent list is used to form candidate list and frequent list of K+1 itemsets by merging itemsets. This process continues until candidate and frequent list is empty. Pruning is used if merging leads to many and large itemsets.The final frequent list in each cycle is used to form rules and classified to one of the classes based on confidence metrics. This process continues till number of rules specified as parameter in Apriori Algorithm achieved and minSupport is greater than lowerBoundMinSupport. If desired rules are not formed using initial minSupport then minSuppor is reduced by 0.5 each time but if minSupport goes below lower bound, then it is set to lowerBound to find remaining rules.

**Output:** Min Support: 0.15, Minimum Metric <confidence> =0.9, Number of cycles performed=17, Levels=5 and Rules=10.

**[20 points] Python Packages and Functions used (Association Rules and Cars). Describe inputs & outputs (at most 1/3 page)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Package** | **Function** | **Input** | **Output** |
| **Association Rules** | Orange | data.Table | Dataset | Rules with quantity performance |
| associate.AssociationRulesSparseInducer | Parameters (support, classificationRules ) |
| Seaborn | Heatmap | processed.cleveland\_num.csv | Heatmaps on correlation and covariance |
|  | Pandas | Cut | Continuous attribute | Discretized attributes |
|  | read\_csv | processed.cleveland.csv | DataFrame |

**[10 points] Three Guiding Questions: (at most 1/3 page)**

1. **Do men tend to have higher chest pain? Does it mean they have higher risk to have heart disease?**
2. **What is the mainly associated symptoms of lower blood sugar?**
3. **Which medical condition of person is less likely to have heart disease?**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Summary of Association Rule Mining & CAR Experiments in Weka.** *At most 3/4 page.* | | | | | | | | | |
| **Tech** | **Guiding Ques** | **Pre-process** | **Parameters** | **Post-process** | **# of levels** | **# of rules** | **Interesting** **rules** | **Conf of the rule** | **Salient observations about experiment** |
| CAR | 1 | Subset selection | #4(sex), #9(cp), #58(num) | minsup= 0.1  minconf= 0.3 | 2 | 6 | #4(sex)=male 206 ==> #58(num)=<50%  #4(sex)=female 97 ==> #58(num)=<50% | 0.45  0.74 | From these rules we can say that female have higher confidence of not getting the heart disease than men |
| AR | 2 | Discretize | All | Minsup=0.2  Minconf=0.9 | 7 | 6 | #40(oldpeak)='(-inf-0.62]' #44(ca)=noVesselColored #51(thal)=Normal #58(num)=<50%  ==> #16(fbs)=FALSE | conf:0.93  lift:1.09  lev:0.02  conv:1.68 | If a person has no fasting blood sugar, then they tend to not have heart disease and has no major blood vessels colored |
| CAR | 3 | Discretize | All | Minsup=0.15  Minconf=0.9 | 5 | 10 | #19(restecg)=normal #38(exang)=no #40(oldpeak)='(-inf-0.62]' #51(thal)=Normal 49 ==> #58(num)=<50% | 0.94 | a person with no heart disease has health conditions like the rest ecg and thalassemia as normal and no exercise angina pain |

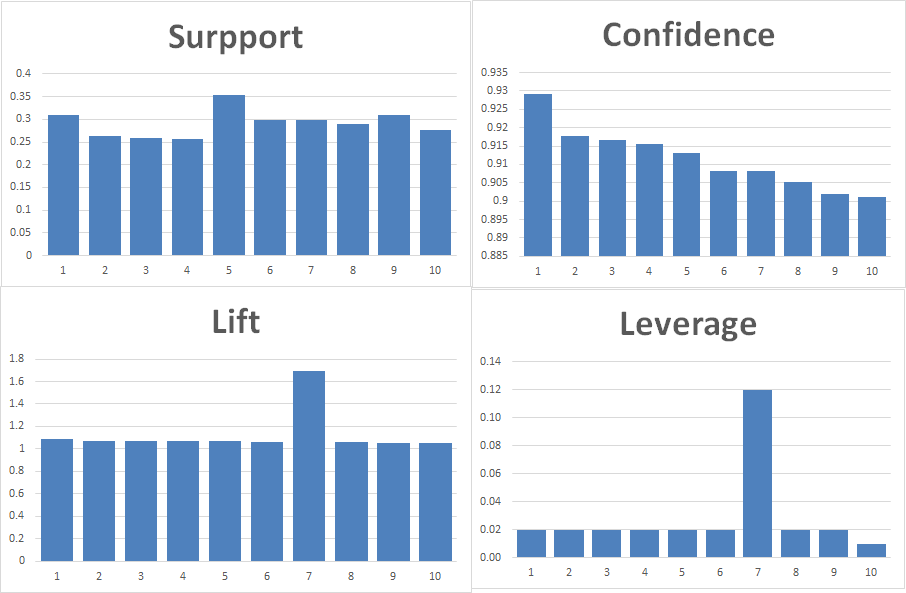
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Summary of Association Rule Mining & CAR Experiments in Python.** *At most 3/4 page.* | | | | | | | | | |
| **Tech** | **Guiding**  **question** | **Pre-process** | **Parameters** | **Post-process** | **# of levels** | **# of rules** | **Interesting rules** | **confidence** | **Salient observations about experiment** |
| AR | 1 | Subset selection | #4(sex), #9(cp), #58(num) | Minsupport: 0.01  Minconfidence: 0.8 | 3 | 9 | 58(num)=TRUE → 4(sex)=male | supp:0.38,  conf: 0.82,  lift: 1.21,  levr:0.06 | Men tends to suffer from higher risk to get heart disease. |
| Minsupport: 0.01  Minconfidence: 0.35 | 3 | 53 | 4(sex)=female → 9(cp)=Non-anginaPain | supp: 0.11  conf:0.35  lift:1.27  levr:0.02 | Female tend to have less chest pain compared to male |
| AR | 2 | Discretize | All | Min support: 0.25  Min confidence: 0.9 | 5 | 10 | 44(ca)=NoVesselColored, 51(thal)=Normal →16(fbs)=FALSE | supp: 0.35  conf:0.91  lift:1.07  levr:0.02 | If a person have lower fasting blood sugar (<= 120 mg/dl), then we can say with more than 90% confidence that this person has no major vessels colored by fluoroscopy and does not have thalassemia. |
| CAR | 1 | Subset selection | #4(sex), #9(cp), #58(num) | Min support: 0.01  Min confidence: 0.35 | 3 | 17 | 4(sex)=female, 9(cp)=AtypicalAngina → 58(num)=FALSE | supp:0.06  conf:0.45  lift:0.84  levr:-0.01 | Female tend to have less chest pain compared to male. Even female with angina does not have higher risk in having heart disease. |
| CAR | 3 | Discretize | All with target attr | Min support: 0.15  Min confidence: 0.9 | 5 | 30 | 9(cp)=asymptomatic, 51(thal)=ReversableDefect → 58(num)=TRUE | supp: 0.24  conf:0.91  lift:1.97  levr:0.12 | If a person does not have heart disease, there is a high confidence to say this person has no major vessels colored by fluoroscopy and does not have thalassemia. But if the person have heart disease, asymptomatic and reversible defected thalassemia could be considered as red flag |
| 38(exang)=no, 44(ca)=noVesselColored, 51(thal)=Normal → 58(num)=FALSE | supp: 0.30  conf:0.91  lift:1.69  levr:0.12 |

**[20 points] Quantitative Analysis of Weka and Python Results and Discussion (at most 1/2 page)**

In quantitative analysis in association rules, the evaluation parameters are support, confidence, lift, leverage and conviction.

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Definition** | **Equation** |
| Support | How frequently a subset appear in the whole dataset |  |
| Confidence | How often correct prediction is made |  |
| Lift | The ratio of the support of a specific subset to the support of A and B separately |  |
| Leverage | The difference of the support of a specific subset and the product between support of A and support of B. |  |
| Conviction | Alternative of confidence |  |

In our AR experiment with the whole dataset, we picked a typical rule, “44(ca)=NoVesselColored, 51(thal)=Normal →16(fbs)=FALSE”, as an example to show how these parameters works. The frequency of subset {44(ca)=NoVesselColored, 51(thal)=Normal} is 115, the frequency of {44(ca)=NoVesselColored, 51(thal)=Normal, 16(fbs)=FALSE } is 105. The frequency of {16(fbs)=FALSE} is 254, there are 297 instances in total. So, **supp(44(ca)=NoVesselColored, 51(thal)=Normal)** is (115/297=) **0.387**, **supp{16(fbs)=FALSE}** is (254/297=) **0.855**, and **supp{44(ca)=NoVesselColored, 51(thal)=Normal, 16(fbs)=FALSE}** is (105/297=) **0.354.** **Confidence** of this rule is (0.354/0.387=) **0.913**. **Lift** of this rule would be (0.354/(0.387\*0.855)=) **1.07**. The **leverage** and **conviction** of this rule is (0.354-0.387\*0.855=) **0.022** and ((1-0.855)/(1-0.913)=) **1.66**. We found these result to be exactly the same as the results we got in Weka and Python.

**[20 points] Qualitative Analysis of Weka and Python Results and Visualizations (at most 1/2 page)**

(Remember also to analyze the results from the point of view of the dataset domain by searching the medical literature, and discuss the answers that the experiments provided to your guiding questions.)

Heart disease in women are developed generally 7 to 10 years later than men. So from the results we can see that the confidence of female who have heart disease is greater than the confidence of male who have heart disease. (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3018605/>)

Studies have shown that people who have thalassemia can develop diabetics. So our results have shown the association between low blood sugar and thalassemia being normal. (*cooleysanemia.org/updates/Diabetes.pdf*)

The visualization to the right gives a bar chart of the confidence, lift, leverage and conviction of the 10 best rule that were extracted when we performed Association Rule mining on all attributes with 0.25 min support.

**Advanced Topic: FP - Algorithm**

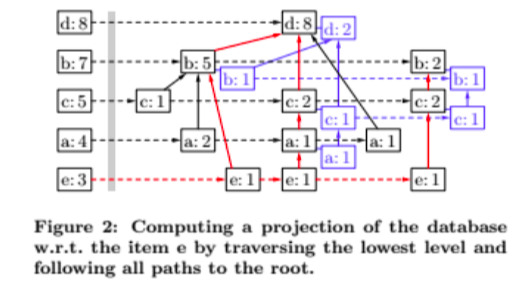
**[7 points] List of sources/books/papers used for this topic (include URLs if available):**

* http://singularities.com/blog/2015/08/apriori-vs-fpgrowth-for-frequent-item-set-mining
* Christian Borgelt, “ An Implementation of the FP-Algorithm”
* <https://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R/Frequent_Pattern_Mining/The_FP-Growth_Algorithm>
* Daniel Hunyadi, “ Performance comparison of Apriori and FP-Growth algorithms in generating association rule”

**[20 points] In your own words, provide an in-depth, yet concise, description of your chosen topic. Make sure to cover all relevant data mining aspects of your topic.**

FP-Growth algorithm identifies frequent itemsets without candidate itemset generation. It is one of the most popular and fastest approach to extract frequent itemsets. It is basically composed of two steps, namely, building a compact data structure called the FP-Tree and extracting the frequent itemsets from the FP-Tree. In the FP-Tree each node represents an item and its frequency over all transactions, and each branch represents a different transaction. In the FP-Tree, overlapping itemsets share the same nodes.

The first step, which is the construction of FP- tree involves preprocessing and is done with just 2 passes over the dataset. In pass 1, the data is scanned and the support of each item is calculated. The items with support less than user specified support are discarded. Then the frequent items based on support value are sorted in descending order over all the transactions (Sorting based on descending is heuristic approach). This order is then used to construct the FP-Tree so that common prefix items can be shared to make more compressed tree. Pointers are maintained between nodes representing same item .In pass 2, FP-Tree is constructed by considering one transaction at a time. The construction is continued until all transactions are mapped to a path in the FP-tree.

In the second step, which is the extraction of frequent itemsets, is done using a bottom-up algorithm and divide and conquer approach. At first, all the prefix path subtrees ending in an itemset are extracted and this is called projection or conditional FP tree (FP tree of transactions containing specific item without including that item). Each prefix path subtree is processed recursively and the solutions are then merged. A conditional FP-Tree is used to find the frequent itemsets with certain item(s) as suffix.

**Pruning:** α-pruning is used to remove less frequent itemsets from the conditional FP-Tree by bypassing that node from the branch. This pruning works in top to bottom manner. If nodes having same parent then they are merged with each other.

**[3 points] How does this topic relate to association rule mining?**

First step in association rule mining is to extract frequent itemsets based on which association rule is constructed.

FP Algorithm is also used to extract frequent itemsets that is having support greater than user specified support value and these selected itemsets are used to construct rules.

**Authorship:** Although each student on the team is expected to be involved in every aspect of the project, describe in detail here the main contributions that each of the team members made to this project. This authorship description must accurately reflect the work done by each team member, and must be approved by all of the members of the team (at most 1/3 page)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Initial preprocessing** | **AR code** | **CAR code** | **Python package** | **Association rules and CAR** | | | **advanced topic** |
| **Weka** | **Python** | **summary** |
| **Rohitpal Singh** | **√** | **√** | **√** | **√** | **√** |  | **√** | **√** |
| **Suchithra Balakrishnan** | **√** | **√** | **√** | **√** | **√** | **√** |  | **√** |
| **Mu Niu** | **√** | **√** | **√** | **√** |  | **√** | **√** | **√** |