COSC 757 Data Mining Assignment 4

Mary Snyder

Department of Computer & Information Sciences  
College of Science & Mathematics  
Towson University

msnyde8@students.towson.edu

**ABSTRACT**

In this paper, I will be exploring a dataset to become more familiar with frequent itemset and association rule mining through the COSC 757 Data Mining Assignment 4.

**Categories and Subject Descriptors**

H.2.8 **[Database Management]** Database Applications – *Data mining*

**Keywords**

Frequent Itemset; Association Rule Mining; Multivariate; Categorical; arules; Apriori Algorithm; Eclat Algorithm; Support; Confidence;

# INTRODUCTION

## Dataset

I chose a dataset from the UCI Machine Learning Repository whose attributes are categorical in nature. This Congressional Voting Records dataset comprises United States House of Representatives Congressmen voting records for the 16 key votes for 98th Congress second session in 1984 as identified by the Congressional Quarterly Almanac (CQA). The dataset contains information regarding the voter’s party (Democrat or Republican) as well as a simplified vote of yea/nay/abstain for each of the key votes. There were originally 9 different kinds of votes which were simplified into the previously mentioned yea/nay/abstain as follows: voted for, paired for, and announced for are marked as a yea,; voted against, paired against, and announced against are marked as a nay; voted present, voted present to avoid conflict of interest, and did not vote or otherwise make a position known are marked as abstain. There are 435 instances with no missing values. Key votes of abstain are indicated by a ‘?’ value. There are 17 attributes: Class Name (Democrat, Republican) and a Boolean value for each of the 16 key votes.

## Objective of Analysis

The objective of frequent pattern analysis is to find inherent regularizes in the data. A frequent patter reveals an intrinsic and important property of the dataset and mining of these patterns is the foundation for many essential data mining tasks including association, correlation, and causality analysis.

Frequent pattern analysis is achieved through use of itemsets. An itemset is a set of one or more items. The relative support of an itemset is the fraction of transactions that contain the itemset out of the total number of itemsets. An itemset is considered frequent if the support for the itemset is no less than a specified threshold.

# METHODOLOGY

## Preprocessing

The dataset contains voting records for 16 key votes; however, each vote was recorded in the dataset as a simple ‘y’ for yea, ‘n’ for nay, or ‘?’ for abstain (see Table 1). Since each representative (defined by a row) could have voted yea for multiple key votes, this caused issues when trying to treat the data as transactions in order to basket the data for analysis. To overcome this issue I updated the values for each of the key votes to indicate not only the vote (yea/nay/abstain), but also the vote number (see Table 2). For example, a ‘y’ or yea vote for the first key vote (column V2) would translate into ‘1y’ for the processed data, a ‘n’ or nay vote for the fifth key vote (column V6) would translate into ‘5n’, and a ‘?’ or abstain vote for the sixteenth key vote (column V17) would translate into ‘16?’ for the processed data.

Table 1. Sample of Original Data

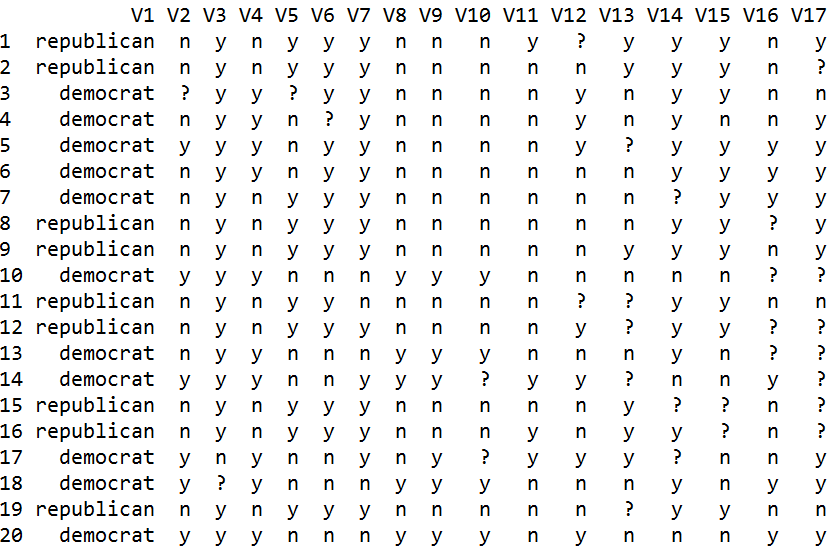
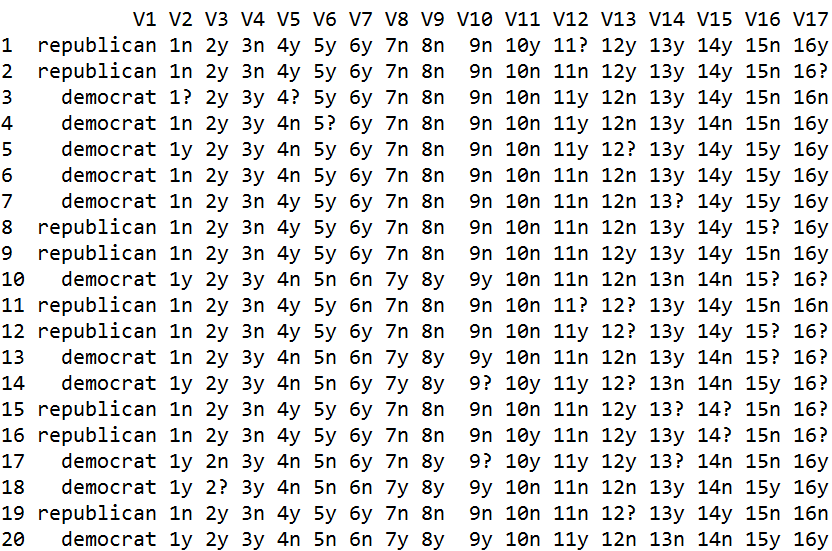


Table . Sample of Processed Data



Determining the frequency of values for the key votes was first done at a low rate to determine a better range. With support of 0.01, the relative item frequency graph (see Figure 1) showed there were many votes, many abstain votes, that were near 0 (zero). There were a few other key vote values with lower frequency, but enough to seem significant. Another relative item frequency graph was completed with a support value of 0.1 (see Figure 2). This second graph shows the relevant key vote value data is displayed much better than a support of 0.01, so a support value of 0.1 was used in the rest of the analysis.

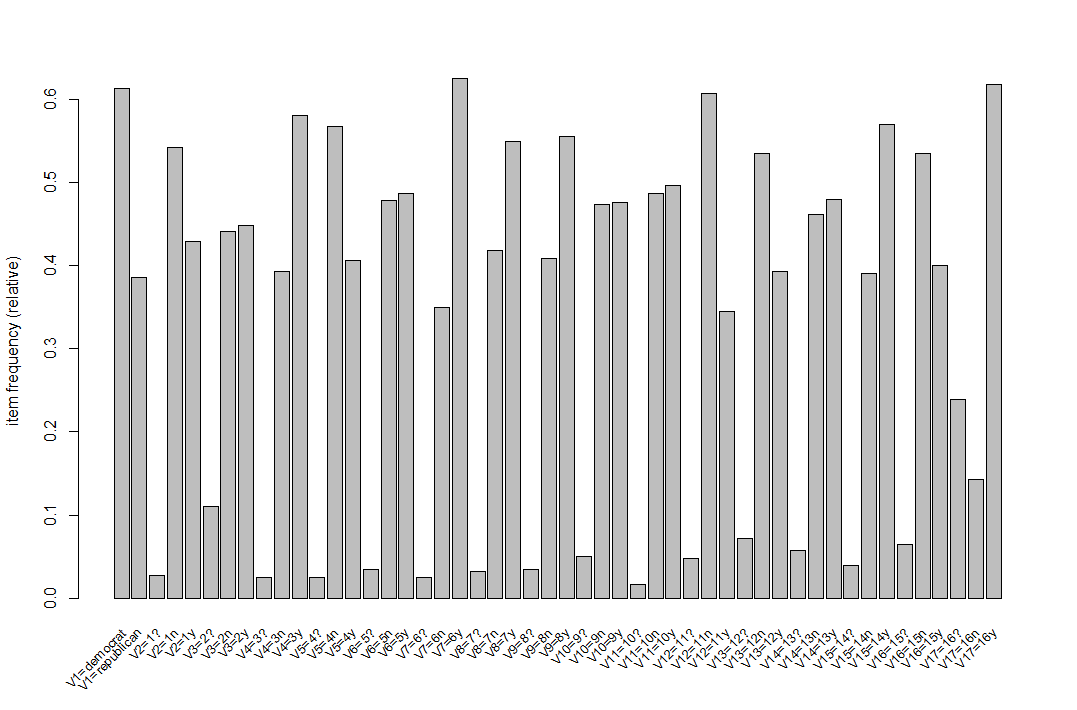


Figure 1. Item Frequency Support of 0.01

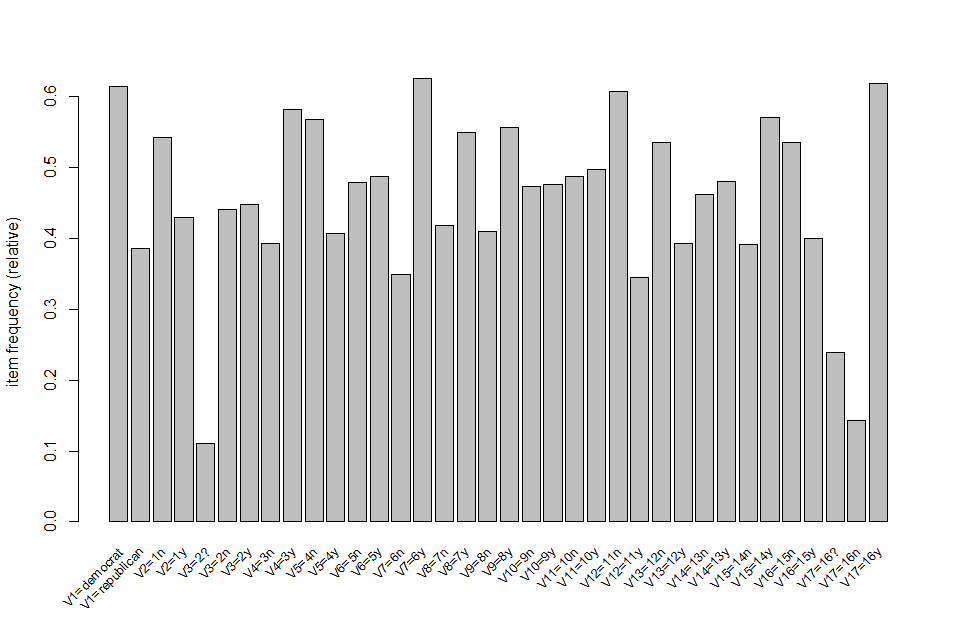


Figure . Item Frequency Support of 0.1

## Tools/Approaches

The statistical computing environment and language R contains a package ‘arules’ to assist in frequent itemset or association rule mining. There are two specific tools in the arules package that will be used for this analysis: Apriori and Eclat.

### Apriori

Apriori in the R arules package is used to mine frequent datasets, association rules, or association hyperedges using the Apriori algorithm. The Apriori algorithm conducts level-wise searches for frequent itemsets as follows:

Ck: Candidate itemset of size k

Lk : frequent itemset of size k

L1 = {frequent items};

for (k = 1; Lk != Ø; k++) do begin

Ck+1 = candidates generated from Lk;

for each transaction *t* in database do

increment the count of all candidates in Ck+1 that are contained in *t*

Lk+1 = candidates in Ck+1 with min\_support

end

return Uk Lk;

There are a few disadvantages of the Apriori algorithm. It requires multiple scans of the database, specifically *n* + 1, where *n* is the length of the longest pattern. In addition, the candidate generation can result in very large candidate sets. For example, if the dataset contains 104 frequent 1-itemsets, the algorithm will need to generate 107 candidate 2-itemsets.

### Eclat

Eclat in the R arules package is used to mine frequent datasets using the Eclat algorithm. The Eclat algorithm uses bottom-up lattice traversal and simple intersection operations for equivalence class clustering. The algorithm is defined recursively with the initial call using all the single items with their transaction ids (tids) and each recursive call examines the intersections of pairs of tids to generate new candidates. Determine support of any k-itemset by intersecting tid-lists of two of its (k-1) subsets.

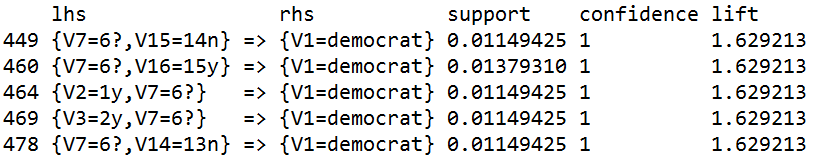
While the Eclat algorithm is very fast in support counting, it is not without its weaknesses. The intermediate tid-lists may become too large for memory. When the tid-list is large, at the time of computation it requires a large amount of space to store the candidate sets. In addition, the algorithm takes more time for computing intersections when the tid-list is large.

# RESULTS

## Apriori

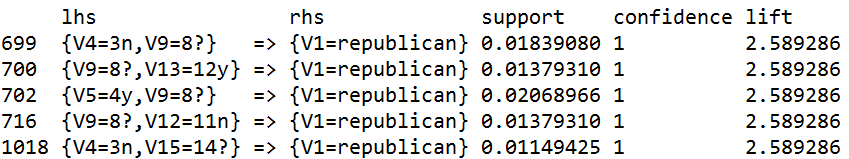
Filtering the Apriori results for only Democrats and sorting by confidence gives a good indication of key vote sets that would be the most important to their party. In this case, it is interesting to note that in the five itemsets with the highest confidence, an abstain vote for key vote 6, Religious Groups in Schools, was a part of each itemset (see Table 3). Another interesting note is no other value in the itemsets showed any overlap at all.

Table . Apriori Democrat Confidence Sort



Filtering the Apriori results for only Republicans and sorting by confidence gives a good indication of key vote sets that would be the most important to their party. In this case, it is interesting to note that in the five itemsets with the highest confidence, an abstain vote for key vote 8, Aid to Nicaraguan Contras, was a part of four out of the five itemsets (see Table 4). Not as high percentage as an abstain vote for key vote 6 was for the Democrats at 100%, but still a very high percentage at 80%. Another interesting note in the five Republican itemsets, a nay vote for key vote 3, Adoption of the Budget Resolution, was seen two out of five times, or 40% of the time. It then makes sense that the highest confidence and support was for the itemset with both the abstain for key vote 8 and the nay for key vote 3.

Table . Apriori Republican Confidence Sort

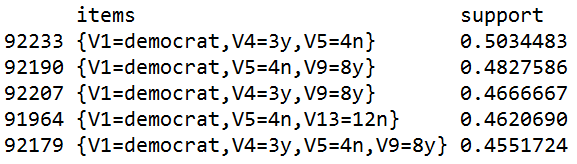


## Eclat

Filtering the Eclat results for only Democrats or only Republicans was a little more challenging with Eclat. There is no confidence value as in Apriori, so support was used instead. In addition, to try to obtain similar results to Apriori and to rule out any empty sets, the minlen was raised to 3.

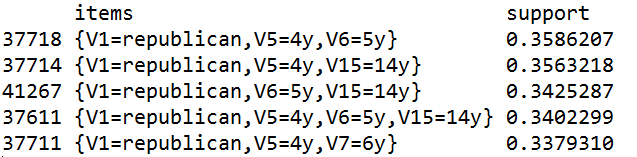
The Eclat results for only Democrats and sorting by support gives a good indication of key vote sets that are important to their party. In this case, it is interesting to note that in the five itemsets with the highest support, a yea vote for key vote 8, Aid to Nicaraguan Contras, was seen in three out of the five itemsets, or 60% of the time (see Table 5). It is also interesting to note that this was frequently in the top itemsets for the Republicans using Apriori only the vote in that case was an abstain, not a yea. In addition, there are four out of the five itemsets where a nay vote for key vote 4, Physician Fee Freeze, for a high percentage of 80% of the total itemsets. It would follow that these two events, yea vote for key vote 8 and nay vote for key vote 4 would have the highest support value; however, this was not the case coming in with the second highest support of 0.48.

Table . Eclat Democrat Support Sort



The Eclat results for only Republicans and sorting by support gives a good indication of key vote sets that are important to their party. In this case, it is interesting to note that in the five itemsets with the highest support, a nay vote for key vote 4, Adoption of the Budget Resolutions, was seen in four out of the five itemsets, or 80% of the time (see Table 6). It is also interesting to note that this itemset was also seen multiple times in the top itemsets for the Republicans using Apriori only there it was only in 40% of the top itemsets. In addition, there are three out of the five itemsets where a yea vote for key vote 14, Crime, as well as three out of the five itemsets where a yea vote for key vote 5, El Salvador Aid, for a percentage of 60% of the total itemsets for both those vote values. Eclat results filtered for Republicans produced more overlapping itemsets with even the itemset containing all three high percentage key vote values.

Table . Eclat Republican Support Sort



# CONCLUSIONS

## Evaluation Metrics

### Support

The support value of an itemset is defined as the proportion of transactions in the database that contain the itemset. It can be written as supp(*X*), where *X* is an itemset.

### Confidence

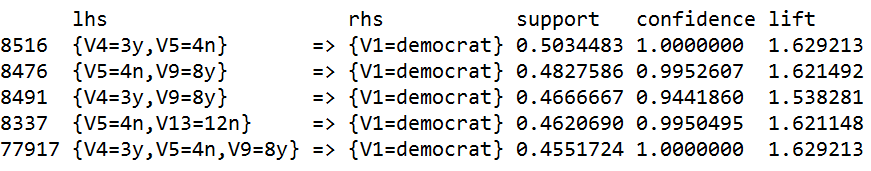
The confidence value of a rule is the proportion of the transactions that contain an itemset, that also contain another non-intersecting itemset of the same dataset. It can be written as conf(*X* => *Y*) = supp(*X* U *Y*) / supp(*X*), where *X* and Y are itemsets such that *X* ∩ *Y* = Ø and *X* => *Y* is a rule.

## Apriori

For a better comparison between Apriori and Eclat, I ran results for Apriori sorting by support as I did for Eclat.

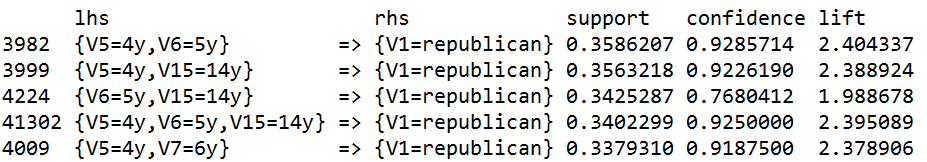
Filtering Apriori results for Democrats only produced very similar results to Eclat results with the top five itemsets matching (see Table 7); however, even with some of the new results having a confidence of 1, none of the results matched the top five itemsets from Apriori sorted by confidence.

Table . Apriori Democrat Support Sort Filter



Filtering Apriori results for Republicans again produced very similar results to Eclat results with the top five itemsets matching (see Table 8); however, even with some of the new results having a confidence near 1, none of the results matched the top five itemsets from Apriori sorted by confidence.

Table . Apriori Republican Support Sort Filter



## Eclat

Looking at the Eclat results sorted by support, I noticed the top five itemsets all contained Democrats even without filtering for them specifically (see Table 9). I also looked at the top ten itemsets to see if they showed a different story (see Table 10). While there are a few itemsets not attributed to either Democrats or Republicans, the rest of the top ten all can be attributed to Democrats. This seems to indicate that Democrats vote more similar to each other as a party on key issues than Republicans do at least for key issues like the 16 key votes in this dataset and at least on a more frequent basis than Republicans do.

Table . Eclat Frequent Itemsets – Top 5

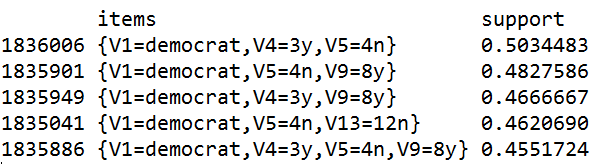
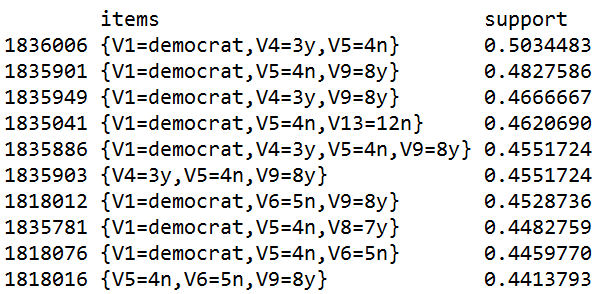


Table . Eclat Frequent Itemsets – Top 10



# REFERENCES

1. Larose, D. and Larose, C.D. *Discovering Knowledge in Data: An Introduction to Data Mining*. Wiley-Interscience, 2014.
2. Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.
3. Congressional Quarterly Almanac, 98th Congress, 2nd session 1984, Volume XL: Congressional Quarterly Inc. Washington, D.C., 1985. (Schlimmer, 1987)
4. Schlimmer, J.C. (1987). Concept acquisition through representational adjustment. Doctoral dissertation, Department of Information and Computer Science, University of California, Irvine, CA.