**20 Thieves Solution by Syed**

**Initial Scenario**

I received email from Tim Brewer at 11:42 and after reading the email and problem readme file. I was up to solve the problem by 12:00 PM. Now 48 hours mean, I must submit it before 12:PM Friday. That’s great. I liked the problem and I liked the approach taken by freelancer.com.

**My Approach Log**

* Initially I just skimmed through the problem to get an idea of what the problem is all about. And after starting writing this solution file, I got back to readme file, but this time to read it with precision. It is important to understand the problem first rather than thinking about the solution. My intial analysis of the problem can be found under Initial Problem Analysis Heading below.
* After completing the initial problem analysis, I moved to the data analysis part. This doesn’t mean programmatically analyzing data, rather it is about reading and understanding the data. The details of initial data analysis can be found under Initial Data Analysis heading below.
* After completing the initial analysis of the data and writing email to Tim Brewer, I moved forward with exploratory data analysis of data to get insight of the data and to identify the need to data preprocessing. Details are available below under initial exploratory data analysis and pre-processing heading.
* After getting my important data ready for analysis. I started analyzing the data. The details of each analysis and my intensions are briefed under initial data analysis heading.

**Initial Problem Analysis**

Few important observations.

* There is at-least one thief (Yes, it is obvious, but I preferred to mark it down)
* Scanner only scan the name and DOB for anyone who enters the club. (Not who left, be careful)
* Ranked list of up-to 20 suspects is needed. From most suspicious to least suspicious. There could be fewer. (Ranking problem)
* Post mid-night time will be considered as previous day
* We have visitor and theft logs
* Freelancer.com need details of how I approached and solved the problem and the problems I faced during the journey. (This is why I have created this document, I hope this will help)

**Initial Data Analysis**

Initial observations about the data:

* theft\_log.csv contains only the dates about the theft. (No time information is provided. This is lack of information, but we data scientist should work it out)
* visitor\_log.csv contains only the visit\_date, name, and dob. (Again we only have the day information. We only know that individual entered the place on the day, but we don’t know either he/she left on the same day or not)
* The joining attribute between visit and theft is date.
* Before moving forward, inquire from Tim Brewer that should be we asking question related to problem? Just to make sure we don’t commit any sins of 09 sins of data mining.

**Initial Exploratory Data Analysis and Pre-processing**

* Before, I start performing the exploratory data analysis, I preferred to get the data into a database. Well, it is not necessary. Many data scientist can process the same data using MS Excel or directly using any data mining tools. But I prefer to work using SQL. And it really help us while working with large datasets. Other tools often start crying when data size grows. I used SQL Server 2016, again only for my personal preference. I created a database nct, short for nightclubthieves. Created two tables each for theft\_log and visitor\_log. Then I bulk inserted data into tables. The queries for this step are provided as initialbulkinsert.sql with this solution.
* *Before moving forward, I must backup my work on GIT. I have version controlled all my work at:* [*https://bitbucket.org/saif137/mybi*](https://bitbucket.org/saif137/mybi) *in freelancertask folder.*
* Now I plan to perform all possible exploratory data analysis, transformations and preprocessing in database using SQL. Once done, with our exploratory data analysis then we will move forward with the problem.
* I started with theft\_log.csv data. It is available in theft\_log\_i table. I first preprocessed the date field to separate out the year, month and day information and stored it into separated table.
* I firsts tried to identify, if there is any patter for theft in terms of month? Either frequency of theft is more in few months or there is gradual increase or decrease in theft over time? For this purpose I wrote following query and found out the yes, July, November, and December has higher theft frequency.

--First check if theft occurence is same or varies across month

select

tyear [Year of theft]

, tmonth [Month of theft]

, count (\*) [Total theft]

from theft\_log

group by

tyear, tmonth

order by

tyear, tmonth

* I also plot the same data using MS Excel PivotChart is show in Figure 1. Visualization is very important for exploratory data analysis. It is also very important task for data scientists in addition to data preprocessing and modeling.

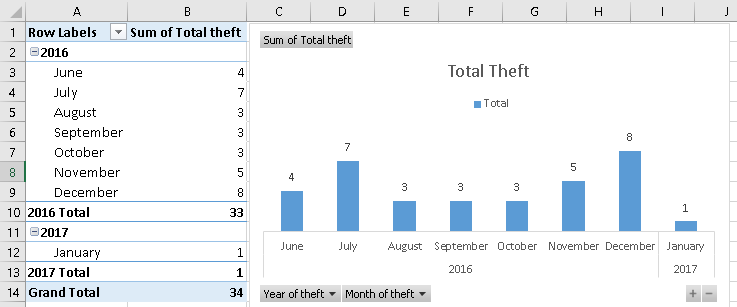


Figure : Month-wise Theft

* Another important exploration will be either, there are any specific days when more thefts have occurred? I identified day-wise theft occurrences using the query given below:

select

tdayname [Day]

, count(\*)

from

theft\_log

group by

tdayname

* Above query result only provided a hint that there has been less theft on Tuesday and Wednesday, but there is no day when there was no theft. Nor we can convincingly say that there are some days with more theft occurrences. It further tried to separate is out on the basis of month using query below:

--Theft occurences on basis of month

select

tmonthname [Month]

, tdayname [Day]

, count(\*)

from

theft\_log

group by

tmonthname, tdayname

order by

tmonthname, tdayname

* Above query result only provided a hint theft occurrence has been more on Sunday and Monday of December. But still I didn’t found any significant patter. Visualization of theft patter for days on montly basis is shown in Figure 2.

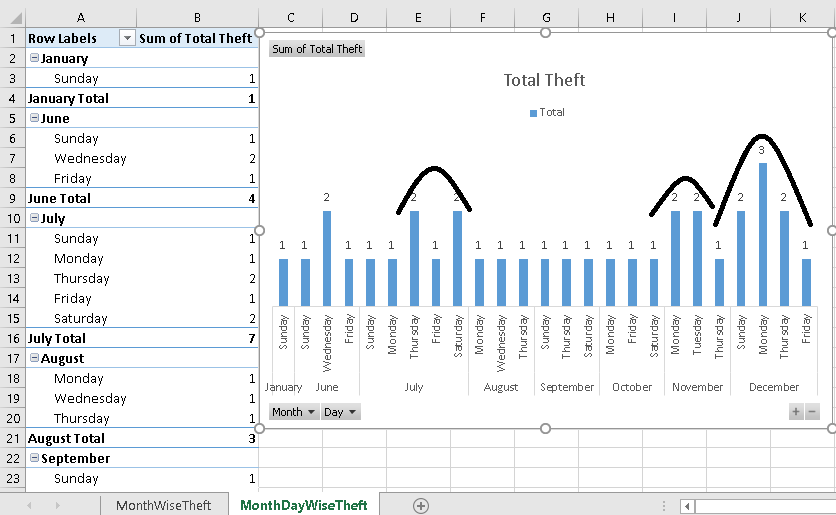


Figure : Theft patter for Days on Monthly Basis

* I immediately realized, that week-wise analysis should also be done to identify if there are some weekly patterns for theft. I added week number column to our table after recreating it with updated query. The updated queries are provided with solution as theft\_log\_pp.sql.

select

tweek [Week]

, count(\*) [Total Theft]

from

theft\_log

group by

tweek

* The weekly theft count also didn’t gave any specific pattern except a minor observation that last three weeks of the year has more theft occurrences together and similarly week 27 to 32 has continuous pattern of theft.
* *Before moving forward, I must backup my work on GIT. I have version controlled all my work at:* <https://bitbucket.org/saif137/mybi> *in freelancertask folder.*
* I will stop working on theft\_log data and now I will move to visitor\_log data that we have already loaded in visitor\_log\_i. I just tried to identify, how each individual can be identified. After checking with distinct and group by queries, it was clear that we should identify each individual using both name and dob.
* Before, I try to find out individuals, who may be guilty. I preferred to eliminate individuals are innocent. I filtered the records/visits for days when no theft was reported. This will reduce our search space. I created a separate table named visitor\_log\_t to store visit information for days when theft has occurred. While moving the data, we also separated the date parts of the visit using query shown below:

insert into visitor\_log

select

visit\_date

, [name]

, [dob]

, convert(integer, SUBSTRING(visit\_date, 1, 4)) as tyear --can also use datepart

, convert(integer, SUBSTRING(visit\_date, 6, 2)) as tmonth

, convert(integer, SUBSTRING(visit\_date, 9, 2)) as tday

, datename(dw, visit\_date) tdayname

, datename(mm, visit\_date) tmonthname

, datepart(wk, visit\_date) tweek

from visitor\_log\_i where visit\_date in (

select theft\_day from theft\_log\_i

)

* Above query reduced our search space from 42727 to 7288. Now we only need to consider individuals in this 7288 records to identify the probable thieves.
* *Before moving forward, I must backup my work on GIT. I have version controlled all my work at:* <https://bitbucket.org/saif137/mybi> *in freelancertask folder.*

**Initial Exploratory Data Analysis and Pre-processing**

* I started my analysis to answer few questions that I had. How many individuals visited on the day, when theft has occurred. I found out that between 195 to 245 visitors were present each day when theft was occurred.
* Next, I was curious to know, either there are visitors who were present on every theft day? Or I can ask the same question as, list the visitors according to their presence of number of theft days. There are 34 distinct theft days. The maximum theft days a visitor was found to be present is Karen Keeny and its 30 days, but beware, Karen could be the most loyal customer of the night club. We are still far away from making any prediction. The only thing that we can take away is that there are more than one thief.
* Next, I was curious to know, either there are visitors who were present together on every theft day? This can also be answered using Association Rule mining. Here I got stuck for two hours, because I tried to get it done using T-SQL and then limit issue arise when I attempted to generate a table with separate column for each visitor. I worked so hard but in the end stuck while creating a query for table with each visitor as a column. But I could do the same thing using MS Excel Pivot table in a minute. The snapshot is given below in Figure 3, where each id on column represent a unique visitor. However, this effort allowed me to replace name and dob with a unique identifier that I will be using for any further processing. This will anyhow reduce the data processed during analytics.

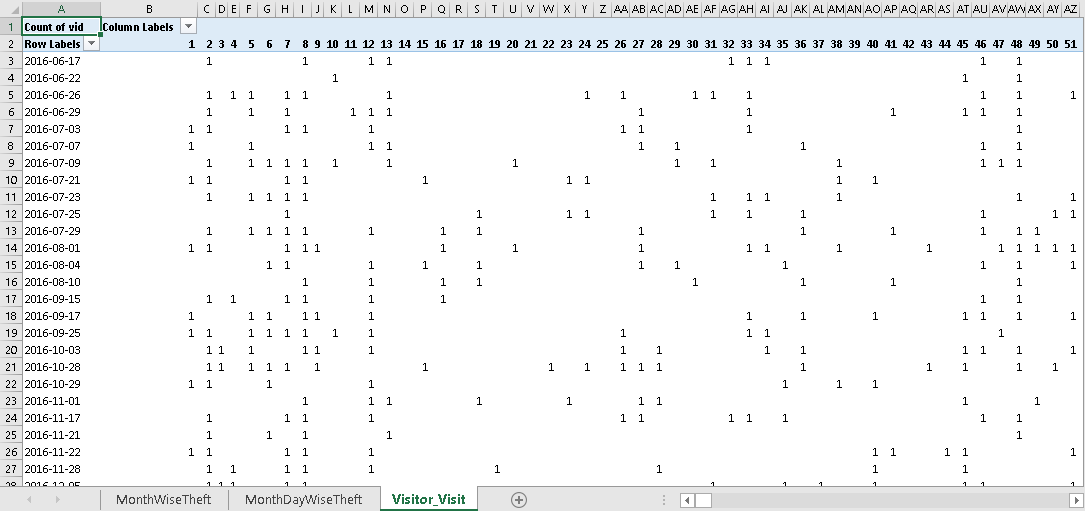


Figure : Visitors Visiting Together on Each Theft Day

* Using this tabular view of the data as shown in Figure 3, I was able to perform association rule mining using FP-Growth algorithm to identify, which visitors visited together during theft days. I was able to complete the initial model execution with default values using rapid miner, but then it became time to take break as library is closing. I will start again on this after2 hours from home.
* After updating the parameters, i.e., minimum confidence to 0.99 (to fetch only important and strong association, following visitors are of interest for us. But this doesn’t mean yet, that they are thieves, they could be our loyal customers as well.

Association Rules

[726] --> [464] (confidence: 0.909)

[538] --> [464] (confidence: 0.909)

[289] --> [464] (confidence: 0.909)

[207] --> [464] (confidence: 0.909)

[773] --> [563] (confidence: 0.909) (Lift = 1.145)

[650] --> [302] (confidence: 0.909) (Lift = 1.145)

[207] --> [302] (confidence: 0.909) (Lift = 1.145)

[100] --> [632] (confidence: 0.909) (Lift = 1.236)

[679, 455] --> [464] (confidence: 0.909)

[563, 744] --> [464] (confidence: 0.909)

[464, 152] --> [563] (confidence: 0.909) (Lift = 1.145)

[563, 145] --> [464] (confidence: 0.909) (Lift = 1.145)

[464, 54] --> [455] (confidence: 0.909)

[744, 145] --> [464] (confidence: 0.909)

[563, 744] --> [152] (confidence: 0.909) (Lift = 1.189)

[744, 152] --> [563] (confidence: 0.909) (Lift = 1.145)

[784] --> [464] (confidence: 0.913)

[748] --> [464] (confidence: 0.913)

[142] --> [464] (confidence: 0.913)

[563, 455] --> [464] (confidence: 0.913)

[464, 302] --> [455] (confidence: 0.913) (Lift = 1.15)

[455, 302] --> [464] (confidence: 0.913)

[709] --> [464] (confidence: 0.917)

[145] --> [464] (confidence: 0.923)

[455] --> [464] (confidence: 0.926)

[701] --> [302] (confidence: 0.952) (Lift = 1.199)

[455, 54] --> [464] (confidence: 0.952)

[848] --> [464] (confidence: 0.955)

[455, 145] --> [464] (confidence: 0.955)

* I also attempted to get another perspective of the generated rules using the graph view. It gave me an interesting insight. As it can be observed in Figure 4, visitors 100 and 632 are present on theft days without any association with other visitors. It may be interesting, but it will need further investigation.

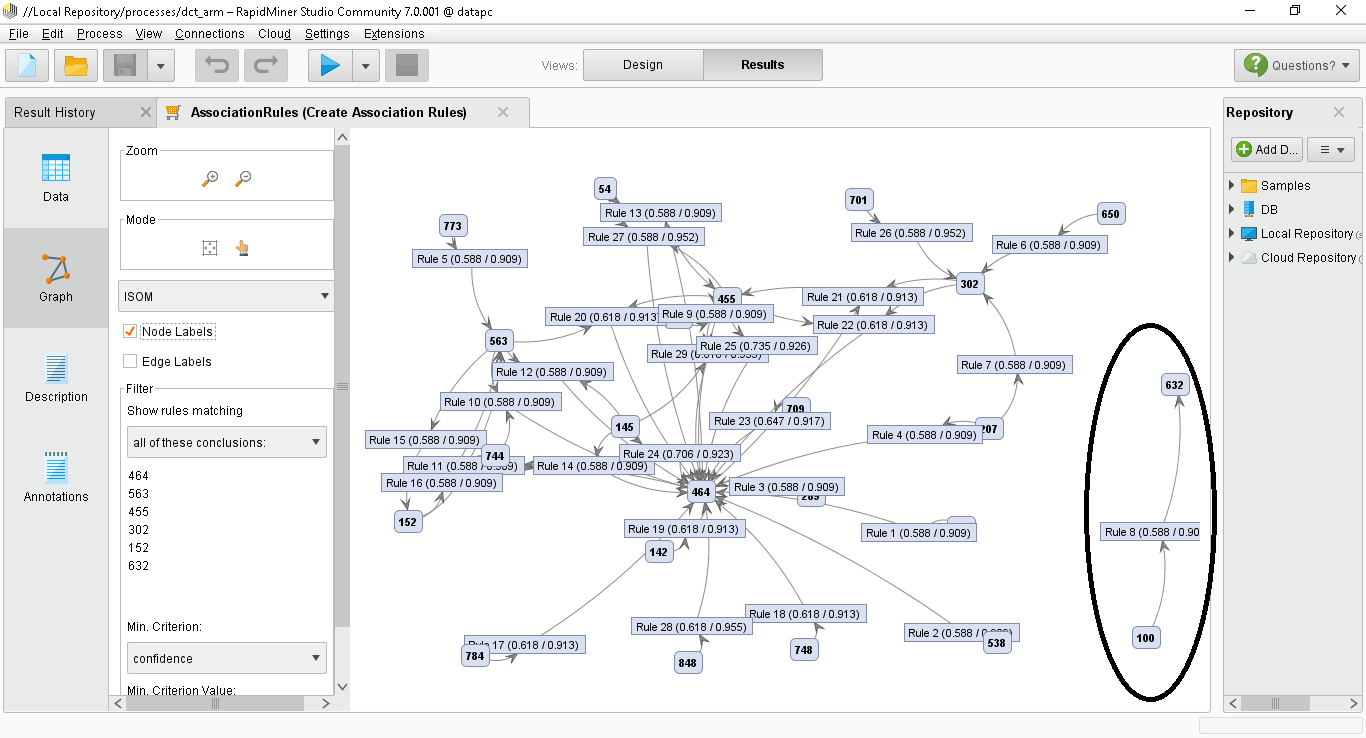


Figure : Graph View of Generated Association Rules

* I will keep focus on following visitors during further analysis: 100, 632, 464, 145, 563, 455, 302, 709, 744, 207, 209, 152, 302 as these visitors has been present in existence of most other visitors.