Data and Text Mining

MCDA5580

Master of Science in Computing and Data Analytics

Assignment-3

Association Mining: Milestones on Website

Submitted by:

Vivekanand Boopathy	A00425792
Parijat Bandyopadhyay	A00430847
Kothai Kannappan Murugappan	A004727876

Submitted to:

Dr. Pawan Lingras

SAINT MARY'S UNIVERSITY SINCE 1802

One University. One World. Yours.

Table of Contents

Executive summary

The objective of this report is to analyze the behavioural patterns of the customer using previous customer data of the website, Simplycast.com and predict their future behaviour by creating a model using association mining and apriori algorithm, which can predict the most commonly associated rules for future user. The analysis is conducted with both user and session level data. The optimal apriori model achieved for user data has 76 rules (Support = 0.3, Confidence =0.5) and for session data has 20 rules (Support = 0.2 , Confidence = 0.5).Considering the top 5 results it is found that the rules, item sets creating rules and maximally occurring items are almost the same for both analyses(from user level and session level data)

Data Summary

The data on user clicks has been provided in MYSQL Database loaded under database name "dataset03" and table name "rawdataDec15" which was exported as CSV for our study. Following are the field in our original dataset:

id: Serially generated id created for each event on the product

user_id: Gives the identity of the user associated with the activity

milestone_name: Name of the milestone selected

Date: Date of the milestone recorded

time: Time of the milestone recorded

The dataset in our consideration contains 3159 unique users and 24713 Session made by them. The total number of records in the original database is 665436.

Data Cleaning and Transformations

The data that has been given to us has a lot of repeated milestones within each of the session recorded. For our association study these duplicates must be removed for passing it to the mining algorithm to avoid biases. The distinct records are extracted from the table using the DISTINCT keyword. Further in order for passing our data to the Apriori algorithm, we require the data regarding each of the session or user in a single row as input. Session field is generating by concatenating user id with the date. This is done using python Pandas data frames and our transformed csv for further analysis has the following fields:

For user analysis:

User<id>, Milestones associated (Categorical in nature)

For session analysis;

<Id+date=Session>, Milestones associated (Categorical in nature)

Data Analysis

For using the A priori algorithm, we use the "arules" package.

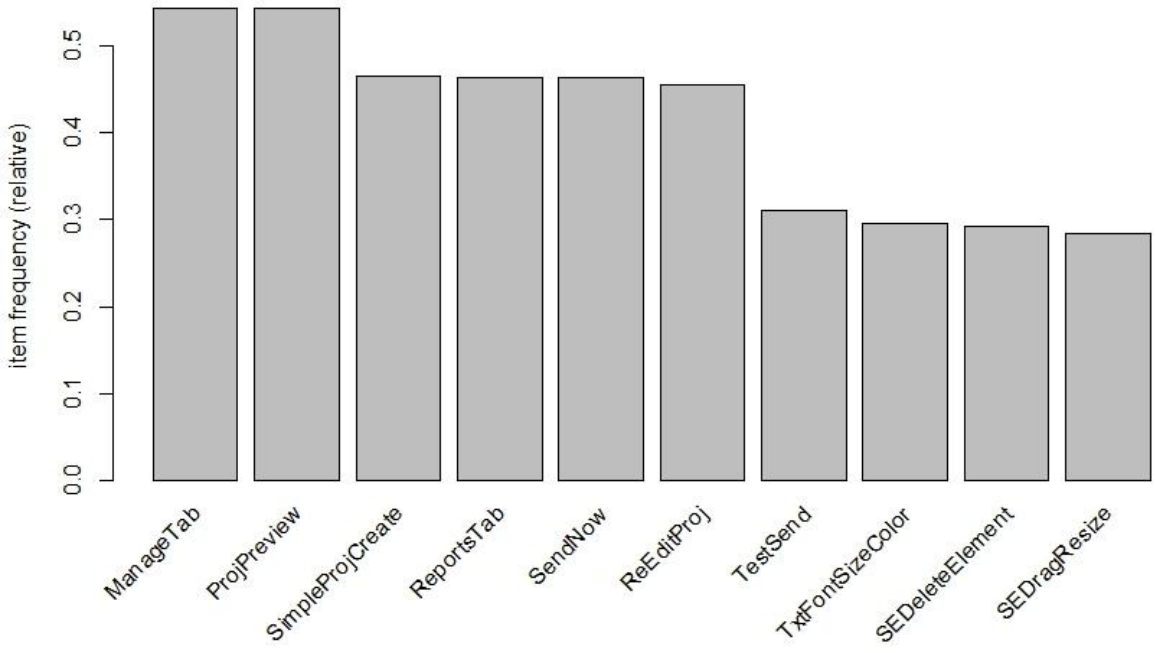
Following are summary results as shows by R on our transformed CSV.

```
> summary(tr)
transactions as itemMatrix in sparse format with
 3159 rows (elements/itemsets/transactions) and
 112 columns (items) and a density of 0.1105006

most frequent items:
       ManageTab      ProjPreview SimpleProjCreate       ReportsTab          SendNow
            1716             1715             1472             1467             1463
         (Other)
           31263

element (itemset/transaction) length distribution:
sizes
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22
807 241 147 108 101 103 101  71  76  72  70  78  54  56  58  65  55  65  42  50  55  46
 25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46
 42  40  24  28  23  27  21  32  33  14  22  24  22  17  21  12  16  14  12  14  11   8
 49  50  51  52  53  54  55  56  57  58  59  60  61  64
  5   9  10   5   8   5   5   6   3   1   1   1   1   1

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    1.00    7.00   12.38   19.00   64.00
```

```
includes extended item information - examples:
           labels
1      ABSplitProj
2     ABSplitTools
3 AccountSettingsA
```

We perform exploratory analysis on the dataset.

Following histogram show the number of occurrences of top 10 recorded milestones.



User Level Analysis

First, we try values of **support = 0.2 and confidence = 0.5** (which is considered an optimum value which provides trustable rules and gives a good number of rules, It remains constant through out our analysis)

661 rules

```
> rules<- apriori(tr, parameter= list(supp=0.2, conf=0.5))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target
        0.5    0.1    1 none FALSE            TRUE       5     0.2      1     10  rules F

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 631

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[112 item(s), 3159 transaction(s)] done [0.00s].
sorting and recoding items ... [21 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 6 done [0.01s].
writing ... [661 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

2nd case:

Support = 0.3 and confidence = 0.5

76 Rules

```
> rules<- apriori(tr, parameter= list(supp=0.3, conf=0.5))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target
        0.5    0.1    1 none FALSE            TRUE       5     0.3      1     10  rules F

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 947

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[112 item(s), 3159 transaction(s)] done [0.00s].
sorting and recoding items ... [7 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 done [0.00s].
writing ... [76 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

3rd case:

Support = 0.4 and confidence = 0.5

8 rules

```
> rules<- apriori(tr, parameter= list(supp=0.4, conf=0.5))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target
        0.5    0.1    1 none FALSE            TRUE       5     0.4      1     10  rules F

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 1263

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[112 item(s), 3159 transaction(s)] done [0.00s].
sorting and recoding items ... [6 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 done [0.00s].
writing ... [8 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

From the observations we see that support = 0.3 and confidence =0.5 gives optimum number of results.

Rules

Sorting by lift values gives following rules.

```
> inspect(sort(rules,by='lift')[1:15])
     lhs                                      rhs           support   confidence lift
[1]  {ManageTab,ReEditProj,ReportsTab}  => {SendNow}    0.3038936 0.9204219  1.987432
[2]  {ManageTab,ProjPreview,ReEditProj} => {SendNow}    0.3206711 0.9200727  1.986678
[3]  {ManageTab,ReEditProj}             => {SendNow}    0.3469452 0.9087894  1.962314
[4]  {ReEditProj,ReportsTab}            => {SendNow}    0.3108579 0.9042357  1.952482
[5]  {ManageTab,ProjPreview,ReportsTab} => {SendNow}    0.3187718 0.8975045  1.937947
[6]  {ManageTab,ReportsTab,SendNow}     => {ReEditProj} 0.3038936 0.8751139  1.918449
[7]  {ManageTab,ProjPreview,SendNow}    => {ReEditProj} 0.3206711 0.8747841  1.917726
[8]  {ManageTab,ProjPreview,ReEditProj} => {ReportsTab} 0.3089585 0.8864668  1.908895
[9]  {ManageTab,ProjPreview,ReportsTab} => {ReEditProj} 0.3089585 0.8698752  1.906964
[10] {ReportsTab,SendNow}               => {ReEditProj} 0.3108579 0.8690265  1.905104
[11] {ManageTab,ProjPreview}            => {SendNow}    0.3665717 0.8812785  1.902911
[12] {ProjPreview,ReportsTab}           => {SendNow}    0.3266857 0.8805461  1.901329
[13] {ProjPreview,ReportsTab}           => {ReEditProj} 0.3194049 0.8609215  1.887336
[14] {ManageTab,ReEditProj,SendNow}     => {ReportsTab} 0.3038936 0.8759124  1.886167
[15] {ManageTab,ProjPreview,SendNow}    => {ReportsTab} 0.3187718 0.8696028  1.872580
```

Itemsets Creating Rules

```
> write(itemsets)
"items" "support"
"1" "{ManageTab}" 0.54320987654321
"2" "{ProjPreview}" 0.542893320671098
"3" "{SendNow,SimpleProjCreate}" 0.322253877809433
"5" "{ReEditProj,SimpleProjCreate}" 0.311490978157645
"7" "{ManageTab,SimpleProjCreate}" 0.330167774612219
"9" "{ProjPreview,SimpleProjCreate}" 0.324786324786325
"11" "{ReportsTab,SendNow}" 0.357708135485913
"13" "{ReEditProj,ReportsTab}" 0.34377967711301
"15" "{ManageTab,ReportsTab}" 0.428300094966762
"17" "{ProjPreview,ReportsTab}" 0.371003482114593
"19" "{ReEditProj,SendNow}" 0.369737258626148
"21" "{ManageTab,SendNow}" 0.41690408357075
"23" "{ProjPreview,SendNow}" 0.397277619499842
"25" "{ManageTab,ReEditProj}" 0.381766381766382
"27" "{ProjPreview,ReEditProj}" 0.397910731244065
"29" "{ManageTab,ProjPreview}" 0.415954415954416
"31" "{ReEditProj,ReportsTab,SendNow}" 0.310857866413422
"34" "{ManageTab,ReportsTab,SendNow}" 0.347261791706236
"37" "{ProjPreview,ReportsTab,SendNow}" 0.326685660018993
"40" "{ManageTab,ReEditProj,ReportsTab}" 0.330167774612219
"43" "{ProjPreview,ReEditProj,ReportsTab}" 0.319404874960431
"46" "{ManageTab,ProjPreview,ReportsTab}" 0.355175688509022
"49" "{ManageTab,ReEditProj,SendNow}" 0.346945235834125
"52" "{ProjPreview,ReEditProj,SendNow}" 0.337448559670782
"55" "{ManageTab,ProjPreview,SendNow}" 0.366571699905033
"58" "{ManageTab,ProjPreview,ReEditProj}" 0.348528015194682
"61" "{ManageTab,ReEditProj,ReportsTab,SendNow}" 0.303893637226971
"65" "{ManageTab,ProjPreview,ReportsTab,SendNow}" 0.318771763216208
"69" "{ManageTab,ProjPreview,ReEditProj,ReportsTab}" 0.308958531180753
"73" "{ManageTab,ProjPreview,ReEditProj,SendNow}" 0.320671098448876
```

Maximal frequent itemsets

Maximally frequent itemsets based on the user data is

"items" "support" "count"
"1" "{TestSend}" 0.311174422285533 983
"2" "{SendNow,SimpleProjCreate}" 0.322253877809433 1018
"3" "{ReEditProj,SimpleProjCreate}" 0.311490978157645 984
"4" "{ManageTab,SimpleProjCreate}" 0.330167774612219 1043
"5" "{ProjPreview,SimpleProjCreate}" 0.324786324786325 1026
"6" "{ManageTab,ReEditProj,ReportsTab,SendNow}" 0.303893637226971 960
"7" "{ManageTab,ProjPreview,ReportsTab,SendNow}" 0.318771763216208 1007
"8" "{ManageTab,ProjPreview,ReEditProj,ReportsTab}" 0.308958531180753 976
"9" "{ManageTab,ProjPreview,ReEditProj,SendNow}" 0.320671098448876 1013

Session Level Analysis

First, we try values of support = 0.1 and confidence = 0.5 (which is considered an optimum value which provides trustable rules and gives a good number of rules, it remains constant through out our analysis)

```
> rules<- apriori(tr, parameter= list(supp=0.1, conf=0.5))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen max
len target   ext
        0.5    0.1    1 none FALSE            TRUE       5     0.1      1
10  rules FALSE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 2471

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[112 item(s), 24713 transaction(s)] done [0.01s].
sorting and recoding items ... [23 item(s)] done [0.00s].
creating transaction tree ... done [0.01s].
checking subsets of size 1 2 3 4 done [0.01s].
writing ... [93 rule(s)] done [0.00s].
```

2nd case:

Support = 0.2 and confidence = 0.5 - > 20 Rules

```
> rules<- apriori(tr, parameter= list(supp=0.2, conf=0.5))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target
        0.5    0.1    1 none FALSE            TRUE       5     0.2      1     10  rules F

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 4942

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[112 item(s), 24713 transaction(s)] done [0.01s].
sorting and recoding items ... [7 item(s)] done [0.00s].
```

```
creating transaction tree ... done [0.01s].
checking subsets of size 1 2 3 done [0.00s].
writing ... [20 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

3rd case:

Support = 0.3 and confidence -> 5 Rules

```
> rules<- apriori(tr, parameter= list(supp=0.3, conf=0.5))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target
        0.5    0.1    1 none FALSE            TRUE       5     0.3      1     10  rules F

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 7413

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[112 item(s), 24713 transaction(s)] done [0.01s].
sorting and recoding items ... [5 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 done [0.00s].
writing ... [5 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

From our experiments, we find that support =0.2 and confidence = 0.5 is giving the optimum number of rules.

Rules

Sorting by lift values gives following rules.

```
> inspect(sort(rules,by='lift'))
     lhs                          rhs           support   confidence lift
[1]  {ManageTab,ReEditProj}    => {SendNow}     0.2413305 0.8135316  1.811244
[2]  {ManageTab,ProjPreview}   => {SendNow}     0.2541982 0.7968037  1.774001
[3]  {ProjPreview,SendNow}     => {ManageTab}   0.2541982 0.9042752  1.638850
[4]  {SendNow}                 => {ManageTab}   0.4047263 0.9010811  1.633061
[5]  {ManageTab}               => {SendNow}     0.4047263 0.7334996  1.633061
[6]  {ReEditProj,SendNow}      => {ManageTab}   0.2413305 0.9000906  1.631266
[7]  {ReEditProj}              => {SendNow}     0.2681180 0.5930368  1.320335
[8]  {SendNow}                 => {ReEditProj}  0.2681180 0.5969369  1.320335
[9]  {ManageTab,SendNow}       => {ReEditProj}  0.2413305 0.5962807  1.318884
[10] {ManageTab,SendNow}       => {ProjPreview} 0.2541982 0.6280744  1.310393
[11] {ProjPreview}             => {SendNow}     0.2811071 0.5864922  1.305764
[12] {SendNow}                 => {ProjPreview} 0.2811071 0.6258559  1.305764
[13] {ProjPreview}             => {ReEditProj}  0.2739854 0.5716336  1.264368
[14] {ReEditProj}              => {ProjPreview} 0.2739854 0.6060145  1.264368
[15] {ReportsTab}              => {ManageTab}   0.2369198 0.6952856  1.260090
[16] {ProjPreview}             => {ManageTab}   0.3190224 0.6655973  1.206285
[17] {ManageTab}               => {ProjPreview} 0.3190224 0.5781754  1.206285
[18] {ReEditProj}              => {ManageTab}   0.2966455 0.6561353  1.189137
[19] {ManageTab}               => {ReEditProj}  0.2966455 0.5376210  1.189137
[20] {}                        => {ManageTab}   0.5517744 0.5517744  1.000000
```

Itemset creating Rules

```
> write(itemsets)
"items" "support"
"1" "{ManageTab}" 0.551774369764901
"2" "{ManageTab,ReportsTab}" 0.23691983976045
"3" "{ProjPreview,SendNow}" 0.281107109618419
"5" "{ProjPreview,ReEditProj}" 0.273985351839113
"7" "{ManageTab,ProjPreview}" 0.319022376886659
"9" "{ReEditProj,SendNow}" 0.268117994577753
"11" "{ManageTab,SendNow}" 0.404726257435358
"13" "{ManageTab,ReEditProj}" 0.296645490227815
"15" "{ManageTab,ProjPreview,SendNow}" 0.254198195281835
"18" "{ManageTab,ReEditProj,SendNow}" 0.24133047383968
```

Maximally frequent itemsets

Maximal sets of timelines based on the user data is

"items" "support" "count"

"1" "{ReportsTab}" 0.340751831020111 8421

"2" "{ReEditProj}" 0.452110225387448 11173

"3" "{ManageTab,SendNow}" 0.404726257435358 10002

"4" "{ManageTab,ProjPreview}" 0.319022376886659 7884

Conclusion

It can be concluded that for user level data the combination of support=0.3 and Confidence=0.5 is giving optimal number of rules where as a support value of 0.2 giving many rules and 0.4 giving a few rules only. Similarly, for session level data a support of 0.2(0.1 giving many and 0.3 giving a few only) giving optimal result with a combination of confidence of 0.5. The top five results for both the analyses show similar results for the rules, item sets creating rules and maximally occurring items.

Note: For R-script kindly refer to the Resource folder.