Assignment 4

Made by

Khadija Hesham

E-mail

khesh072@uottawa.com

Supervised by

Dr, Bisi Runsewe

Part A-I)

We have a number of transaction of 8, minsup value of 2 and a minconf value of 0.6.

a) Now we are investigating the occurrence of each item in our transactions.

A	**5**
B	**4**
C	**5**
D	**6**
E	**1**
F	**4**
G	**5**

We can see that all items are satisfying the minsup value except "E", hence it is to be excluded from the next iteration.

AB	**3**
AC	**3**
AD	**4**
AF	**2**
AG	**2**
BC	**2**
BD	**2**
BF	**1**
BG	**2**
CD	**4**
CF	**2**
CG	**3**
DF	**4**
DG	**3**
FG	**2**

We have all items satisfying the minsup value except for "BF", hence to be excluded.

ABC	**1**
ABD	**2**
ABF	**1**

ABG	1
ACD	3
ACF	1
ACG	1
ADF	2
ADG	1
AFG	0
BCD	1
BCF	0
BCG	1
BDF	1
BDG	0
CDF	2
CDG	2
CFG	1
DFG	2

Last iteration:

ABDF	1
ABDG	0
ACDF	1
ACDG	1
ADFG	0
CDFG	0

We should stop in the iteration before the last iteration.

The resulting table after selecting the item satisfying the minsup value:

ABD	2
ACD	3
ADF	2
CDF	2
CDG	2
DFG	2

Rule generation:
for frequent itemset X1={A,B,D}, X2={A,C,D}, X3={A,D,F}, X4={C,D,F}, X5={C,D,G}, and X6={D,F,G}

b) Strong rules are highlighted in yellow.

Rule l.h.s	Rule r.h.s	Confidence
A =>	**{B,D}**	**2/5**
B =>	**{A,D}**	**2/4**
C =>	**{B,D}**	**2/5**
{A,B} =>	**D**	**2/3**
{A,D} =>	**B**	**2/4**
{B,D} =>	**A**	**2/2**
A =>	**{C,D}**	**3/5**
C =>	**{A,D}**	**3/5**
D =>	**{A,C}**	**3/6**
{A,C} =>	**D**	**3/3**
{A,D} =>	**C**	**3/4**
{C,D} =>	**A**	**3/4**
A =>	**{D,F}**	**2/5**
D =>	**{A,F}**	**2/6**
F =>	**{A,D}**	**2/4**
{A,D} =>	**F**	**2/4**
{A,F} =>	**D**	**2/2**
{D,F} =>	**A**	**2/4**
C =>	**{D,F}**	**2/5**
D =>	**{C,F}**	**2/6**
F =>	**{C,D}**	**2/4**
{C,D} =>	**F**	**2/4**
{C,F} =>	**D**	**2/2**
{D,F} =>	**C**	**2/4**
C =>	**{D,G}**	**2/5**
D =>	**{C,G}**	**2/6**
G =>	**{C,D}**	**2/5**
{C,D} =>	**G**	**2/4**

{C,G} =>	**D**	**2/3**
{D,G} =>	**C**	**2/3**
D =>	**{F,G}**	**2/6**
F =>	**{D,G}**	**2/4**
G =>	**{D,F}**	**2/5**
{D,F} =>	**G**	**2/4**
{D,G} =>	**F**	**2/3**
{F,G} =>	**D**	**2/2**

c) Misleading rules are in red specifying the r.h.s having a probability > confidence value (negatively associated items):

Rule l.h.s	Rule r.h.s	Probability of l.h.s
{A,B} =>	**D**	**3/8**
{B,D} =>	**A**	**5/8**
A =>	**{C,D}**	**4/8**
C =>	**{A,D}**	**4/8**
{A,C} =>	**D**	**3/8**
{A,D} =>	**C**	**5/8**
{C,D} =>	**A**	**5/8**
{A,F} =>	**D**	**3/8**
{C,F} =>	**G**	**4/8**
{C,G} =>	**D**	**3/8**
{D,G} =>	**C**	**5/8**
{F,G} =>	**D**	**3/8**

Part A-II):

We are dealing with a transactions data set.
We have read the data set as a transaction data in ‘basket’ format.

```
data<- read.transactions("/home/khadija/Downloads/transactions.csv", format = 'basket', sep=',')
```

Metadata:

```
data

transactions in sparse format with
 7501 transactions (rows) and
 119 items (columns)
```

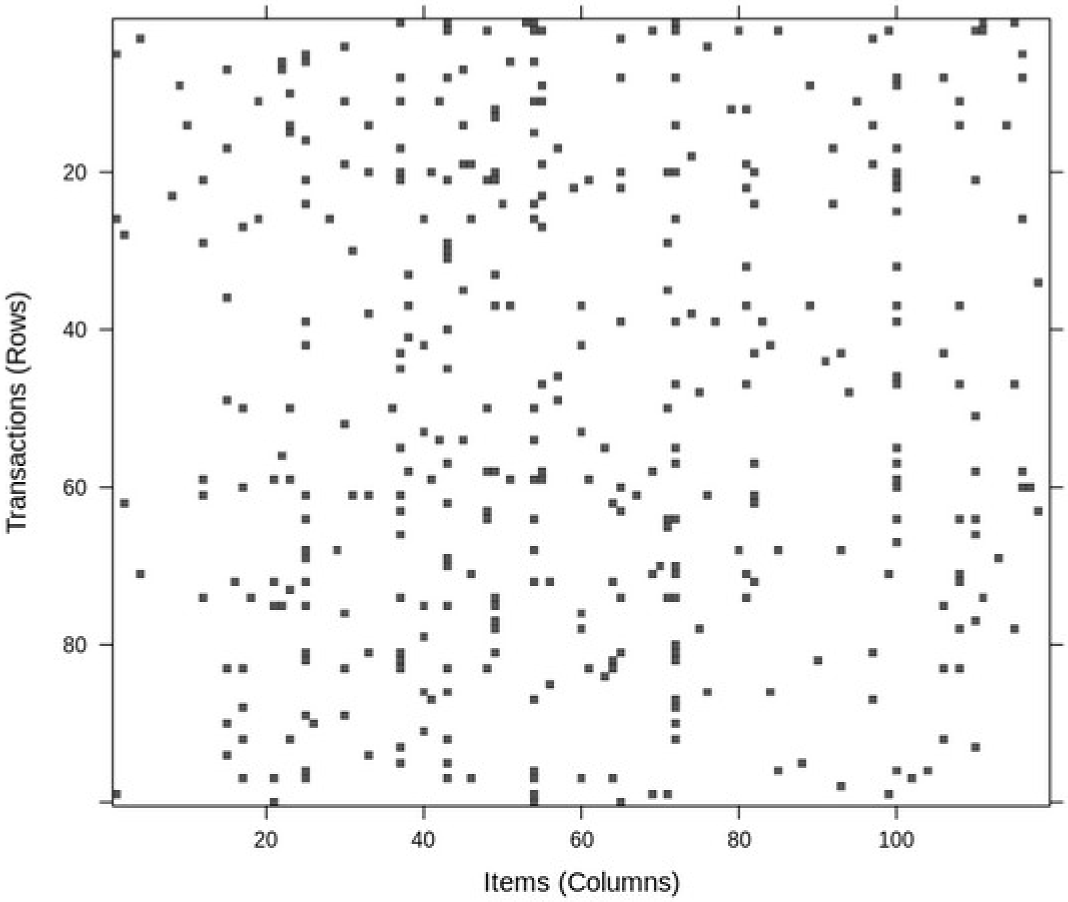
We have 7.5 K transactions including 119 items.
Data summary:

```
summary(data)
```

```
transactions as itemMatrix in sparse format with
 7501 rows (elements/itemsets/transactions) and
 119 columns (items) and a density of 0.03288973

most frequent items:
mineral water          eggs     spaghetti  french fries     chocolate
         1788          1348          1306          1282          1229
      (Other)
        22405

element (itemset/transaction) length distribution:
sizes
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
1754 1358 1044  816  667  493  391  324  259  139  102   67   40   22   17    4
  18   19   20
   1    2    1

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   2.000   3.000   3.914   5.000  20.000

includes extended item information - examples:
             labels
1           almonds
2 antioxydant juice
3         asparagus
```

We can see the most frequent items, such as water ad eggs.

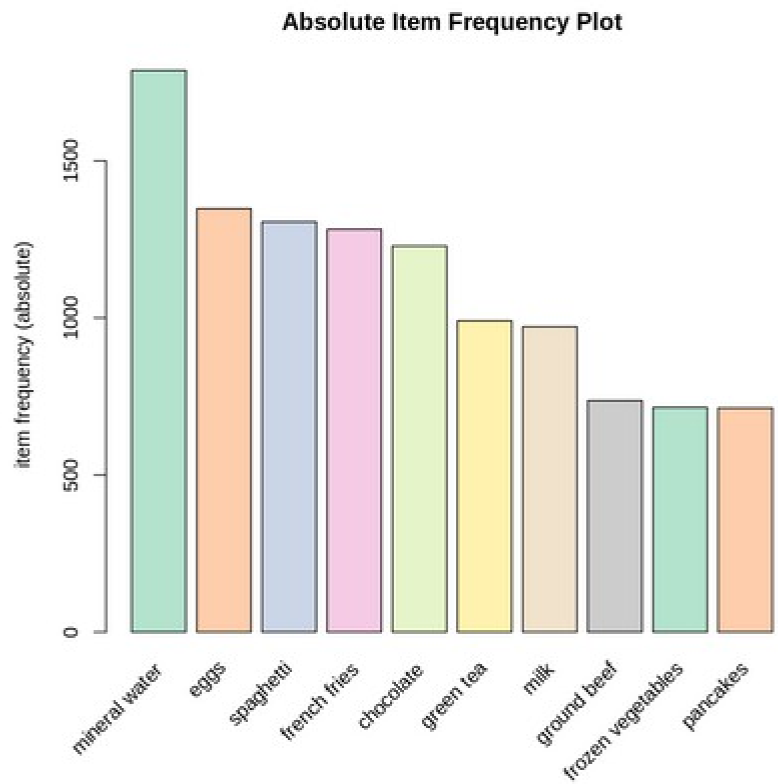
A random sample visual of transactions:



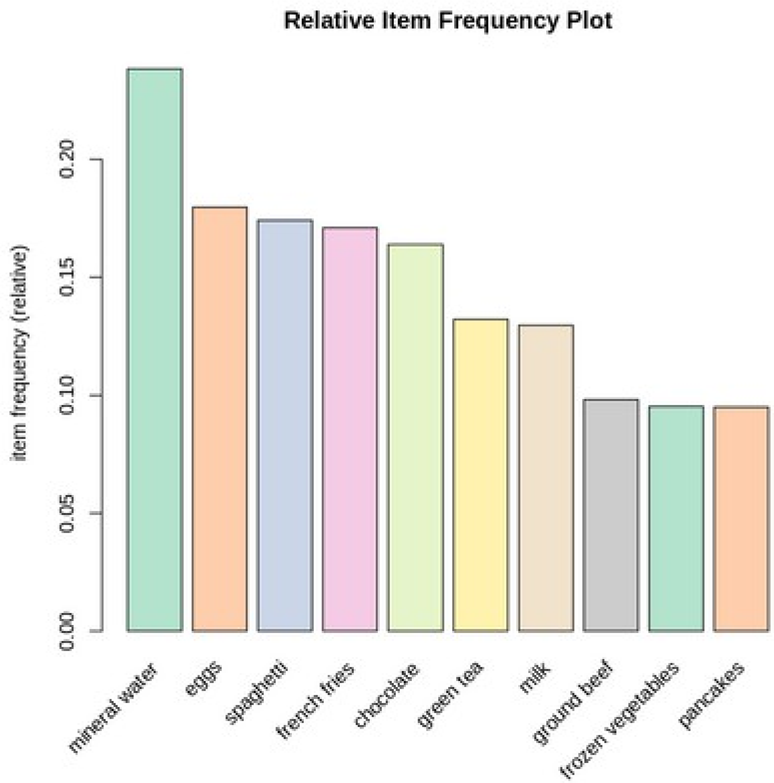
Some transaction that the data contains:

```
In [9]: inspect(data)
                whole weat flour,
                yams}
         [2]    {burgers,
                eggs,
                meatballs}
         [3]    {chutney}
         [4]    {avocado,
                turkey}
         [5]    {energy bar,
                green tea,
                milk,
                mineral water,
                whole wheat rice}
         [6]    {low fat yogurt}
         [7]    {french fries,
                whole wheat pasta}
         [8]    {light cream,
                shallot,
                soup}
         [9]    {frozen vegetables
```

Absolute item frequency plot of first ten transactions:



A relative item frequency plot of first ten transactions:



Generate association rules using minimum support of 0.002, minimum confidence of 0.20, and maximum length of 3:

```
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen
        0.2    0.1    1 none FALSE            TRUE       5   0.002      1
 maxlen target  ext
      3  rules TRUE

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

Absolute minimum support count: 15

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[119 item(s), 7501 transaction(s)] done [0.00s].
sorting and recoding items ... [115 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3

Warning message in apriori(data, parameter = list(supp = 0.002, conf = 0.2, maxlen = 3)):
"Mining stopped (maxlen reached). Only patterns up to a length of 3 returned!"

 done [0.01s].
writing ... [2023 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

Summary of rules generated:

```
set of 2023 rules

rule length distribution (lhs + rhs):sizes
   1    2    3
   1  357 1665

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   3.000   3.000   2.823   3.000   3.000

summary of quality measures:
    support           confidence        coverage             lift
 Min.   :0.002133   Min.   :0.2000   Min.   :0.002666   Min.   : 0.8595
 1st Qu.:0.002533   1st Qu.:0.2405   1st Qu.:0.008266   1st Qu.: 1.5377
 Median :0.003466   Median :0.2941   Median :0.011465   Median : 1.8674
 Mean   :0.005292   Mean   :0.3177   Mean   :0.018647   Mean   : 2.0415
 3rd Qu.:0.005599   3rd Qu.:0.3774   3rd Qu.:0.019064   3rd Qu.: 2.3381
 Max.   :0.238368   Max.   :0.9500   Max.   :1.000000   Max.   :28.0881
     count
 Min.   :  16.0
 1st Qu.:  19.0
 Median :  26.0
 Mean   :  39.7
 3rd Qu.:  42.0
 Max.   :1788.0

mining info:
 data ntransactions support confidence
 data          7501   0.002        0.2
```

A sample of rules generated:

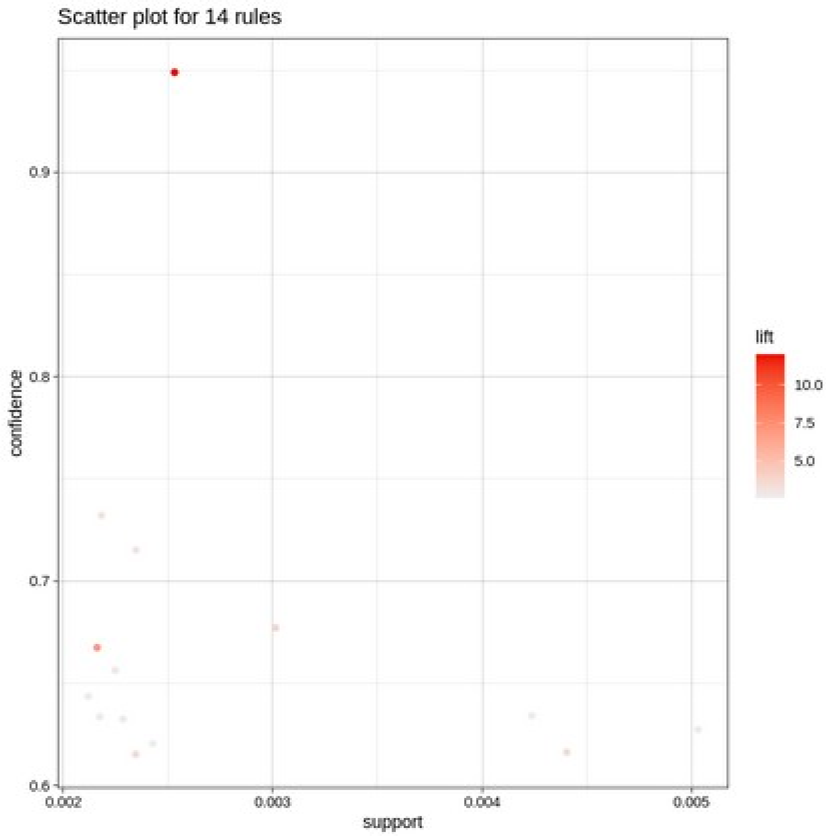
```
     lhs                     rhs             support     confidence coverage
[1]  {}                   => {mineral water} 0.238368218 0.2383682  1.000000000
[2]  {asparagus}          => {mineral water} 0.002133049 0.4444444  0.004799360
[3]  {candy bars}         => {mineral water} 0.002266364 0.2328767  0.009732036
[4]  {shallot}            => {green tea}     0.002266364 0.2931034  0.007732302
[5]  {shallot}            => {french fries}  0.002666311 0.3448276  0.007732302
[6]  {mayonnaise}         => {mineral water} 0.002932942 0.4782609  0.006132516
[7]  {gluten free bar}    => {pancakes}      0.002133049 0.3076923  0.006932409
[8]  {gluten free bar}    => {mineral water} 0.002133049 0.3076923  0.006932409
[9]  {burger sauce}       => {spaghetti}     0.002399680 0.4090909  0.005865885
[10] {burger sauce}       => {mineral water} 0.002399680 0.4090909  0.005865885
     lift      count
[1]  1.0000000 1788
[2]  1.8645290   16
[3]  0.9769621   17
[4]  2.2185358   17
[5]  2.0175910   20
[6]  2.0063953   22
[7]  3.2370266   16
[8]  1.2908277   16
[9]  2.3496102   18
[10] 1.7162142   18
```

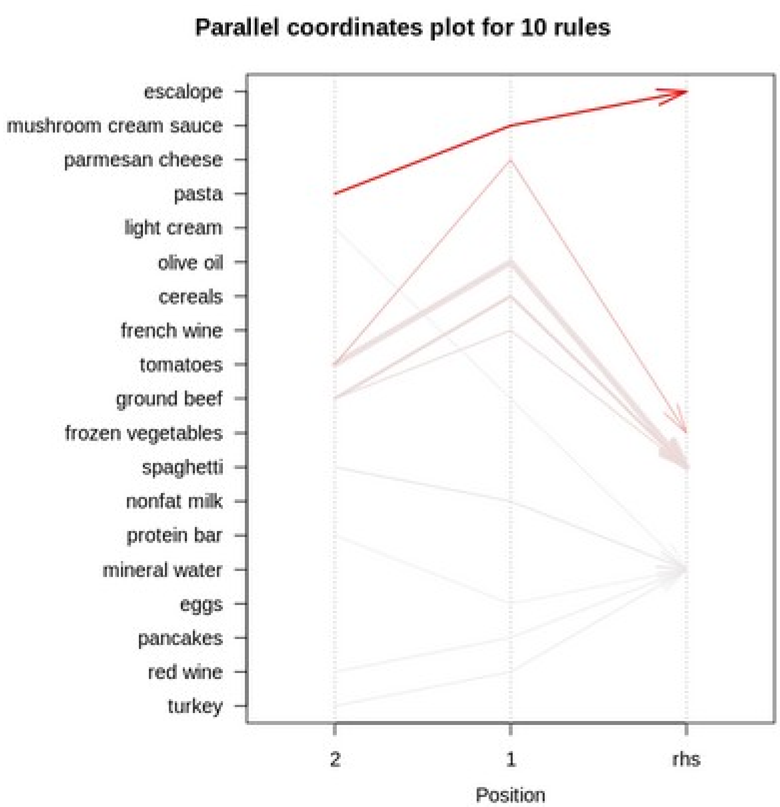
Finding subsets of rules containing any pancakes items:

```
     lhs                            rhs              support
[1]  {gluten free bar}           => {pancakes}       0.002133049
[2]  {whole weat flour}          => {pancakes}       0.002266364
[3]  {bacon}                     => {pancakes}       0.002133049
[4]  {extra dark chocolate}      => {pancakes}       0.002399680
[5]  {light cream}               => {pancakes}       0.003466205
[6]  {light mayo}                => {pancakes}       0.005465938
[7]  {fresh tuna}                => {pancakes}       0.005065991
[8]  {pancakes}                  => {french fries}   0.020130649
[9]  {pancakes}                  => {chocolate}      0.019864018
[10] {pancakes}                  => {eggs}           0.021730436
[11] {pancakes}                  => {spaghetti}      0.025196640
[12] {pancakes}                  => {mineral water}  0.033728836
[13] {fresh tuna,pancakes}       => {spaghetti}      0.002266364
[14] {fresh tuna,spaghetti}      => {pancakes}       0.002266364
[15] {pancakes,pepper}           => {spaghetti}      0.002133049
[16] {pepper,spaghetti}          => {pancakes}       0.002133049
[17] {ham,pancakes}              => {mineral water}  0.002133049
[18] {ham,mineral water}         => {pancakes}       0.002133049
```

Display the rules, sorted by descending lift value:

```
     lhs                                rhs                   support
[1]  {escalope,mushroom cream sauce}     => {pasta}               0.002532996
[2]  {escalope,pasta}                    => {mushroom cream sauce} 0.002532996
[3]  {mushroom cream sauce,pasta}        => {escalope}            0.002532996
[4]  {parmesan cheese,tomatoes}          => {frozen vegetables}   0.002133049
[5]  {mineral water,whole wheat pasta}   => {olive oil}           0.003866151
[6]  {frozen vegetables,parmesan cheese} => {tomatoes}            0.002133049
[7]  {burgers,herb & pepper}             => {ground beef}         0.002266364
[8]  {light cream,mineral water}         => {chicken}             0.002399680
[9]  {ground beef,shrimp}                => {herb & pepper}       0.002932942
[10] {fromage blanc}                     => {honey}               0.003332889
     confidence coverage    lift      count
[1]  0.4418605  0.005732569 28.088096 19
[2]  0.4318182  0.005865885 22.650826 19
[3]  0.9500000  0.002666311 11.976387 19
[4]  0.6666667  0.003199573  6.993939 16
[5]  0.4027778  0.009598720  6.115863 29
[6]  0.3902439  0.005465938  5.706081 16
[7]  0.5483871  0.004132782  5.581345 17
[8]  0.3272727  0.007332356  5.455273 18
[9]  0.2558140  0.011465138  5.172131 22
[10] 0.2450980  0.013598187  5.164271 25
```


Scatter plot for 14 rules
confidence
support
lift
10.0
7.5
5.0
0.9
0.8
0.7
0.6
0.002
0.003
0.004
0.005


Parallel coordinates plot for 10 rules
escalope
mushroom cream sauce
parmesan cheese
pasta
light cream
olive oil
cereals
french wine
tomatoes
ground beef
frozen vegetables
spaghetti
nonfat milk
protein bar
mineral water
eggs
pancakes
red wine
turkey
2
1
rhs
Position

c) Select the rule from Q1 with the greatest lift. Compare this rule with the highest lift rule for maximum length of 2.

Greatest lift rule with maximum length of 3:

```
     lhs                                       rhs                      support
[1]  {escalope,mushroom cream sauce}     => {pasta}                  0.002532996

     confidence coverage    lift      count

[1]  0.4418605  0.005732569 28.088096 19
```

Greatest lift rule with maximum length of 2:

```
     lhs                          rhs           support     confidence coverage
[1]  {fromage blanc}          => {honey}        0.003332889 0.2450980  0.01359819

     lift     count

[1]  5.164271 25
```

i) Which rule has the better lift?

The rule with maximum length of 3 has a lift value of 28 while the The rule with maximum length of 2 has a lift value of 5.1, hence the rule with maximum length of 3 is better.

ii) Which rule has the greater support?

The rule with maximum length of 3 has a support value of 0.002532996 while the The rule with maximum length of 2 has a support value of 0.003332889, hence the rule with maximum length of 2 has greater support.

iii) If you were a marketing manager, and could fund only one of these rules, which would it be, and why?

There is no significance difference between the two rules.
I would go with the rule with maximum length of 3, because it has the greater lift value, hence the escalope,mushroom,cream and pasta are occurring very often.

Part B-I):

The Institute for Statistics Education at Statistics.com asks students to rate a variety of aspects of a course as soon as the student completes it. The Institute is contemplating instituting a recommendation system that would provide students with recommendations for additional courses as soon as they submit their rating for a completed course. Consider the excerpt fromstudent ratings of online statistics courses shown in the Table 14.16, and the problem of what to recommend to student E.N.

1) First consider a user-based collaborative filter. This requires computing correlations between all student pairs. For which students is it possible to compute correlations with E.N.? Compute them.

Measure proximity:

We calculated the average rate for each student, and apply the correlation equation below to calculate the pairwise correlation.

r_LN=(4+3+2+4+2) / 5 = 3
r_MH=(3+4+4)/3= 3.67
r_JH=(2+2)/2=2
r_EN=(4+4+4+3)/4=3.75
r_DU=(4+4)/2=4
r_FL=4
r_GL=4
r_AH=3
r_SA=4
r_RW=(2+4)/2=3
r_BA=4
r_MG=(4+4)/2=4
r_AF=4
r_KG=3
r_DS=(4+2+4)/3=3.33

$$\mathrm{Corr}(U_1, U_2) = \frac{\sum(r_{1,i} - \bar{r}_1)(r_{2,i} - \bar{r}_2)}{\sqrt{\sum(r_{1,i} - \bar{r}_1)^2}\sqrt{\sum(r_{2,i} - \bar{r}_2)^2}},$$

Corr(EN,LN)= (4-3)(4-3.75)+(4-3)(4-3.75)+(2-3)(3-3.75)/sqrt((4-3)^2 +(4-3)^2 +(2-3)^2)* sqrt((4-3.75)^2+(4-3.75)^2 +(3-3.75)^2) =1.25/1.436 =0.87
Corr(EN,MH)=-0.1675/0.1675=-1
Corr(EN,JH)=/0 undefined
Corr(EN,DU)=/0 undefined
Corr(EN,DS)=0.335/0.335=1

2) Based on the single nearest student to E.N., which single course should we recommend to E.N.? Explain why.

We should recommend the student DS courses (SQL or R prog) , as it has a perfect positive correlation between our student.

3) Use R to compute the cosine similarity between users.

We have typed the data to r data frame:

	SQL	Spatial	PA1	DM.in.R	Python	Forcast	R.prog	Hadoop	Regression
LN	4	NA	NA	NA	3	NA	4	NA	2
MH	3	4	NA	NA	4	NA	NA	NA	NA
JH	2	2	NA	NA	NA	NA	NA	NA	NA
EN	4	NA	NA	4	NA	NA	4	NA	3
DU	4	4	NA	NA	NA	NA	NA	NA	NA
FL	NA	4	NA	NA	NA	NA	NA	NA	NA
GL	NA	4	NA	NA	NA	NA	NA	NA	NA
AH	NA	3	NA	NA	NA	NA	NA	NA	NA
SA	NA	NA	4	NA	NA	NA	NA	NA	NA
RW	NA	NA	2	NA	NA	NA	NA	4	NA
BA	NA	NA	4	NA	NA	2	NA	NA	NA
MG	NA	NA	4	NA	NA	4	NA	NA	NA
AF	NA	NA	4	NA	NA	NA	NA	NA	NA
KG	NA	NA	3	NA	NA	NA	NA	NA	NA
DS	4	NA	NA	2	NA	NA	4	NA	NA

We fill nans with zeros and convert the data frame into a matrix and transposed the matrix. Result:

	LN	MH	JH	EN	DU	FL	GL	AH	SA	RW	BA	MG	AF	KG	DS
SQL	4	3	2	4	4	0	0	0	0	0	0	0	0	0	4
Spatial	0	4	2	0	4	4	4	3	0	0	0	0	0	0	0
PA1	0	0	0	0	0	0	0	0	4	2	4	4	4	3	0
DM.in.R	0	0	0	4	0	0	0	0	0	0	0	0	0	0	2
Python	3	4	0	0	0	0	0	0	0	0	0	0	0	0	0
Forcast	0	0	0	0	0	0	0	0	0	0	2	4	0	0	0
R.prog	4	0	0	4	0	0	0	0	0	0	0	0	0	0	4
Hadoop	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0
Regression	2	0	0	3	0	0	0	0	0	0	0	0	0	0	0

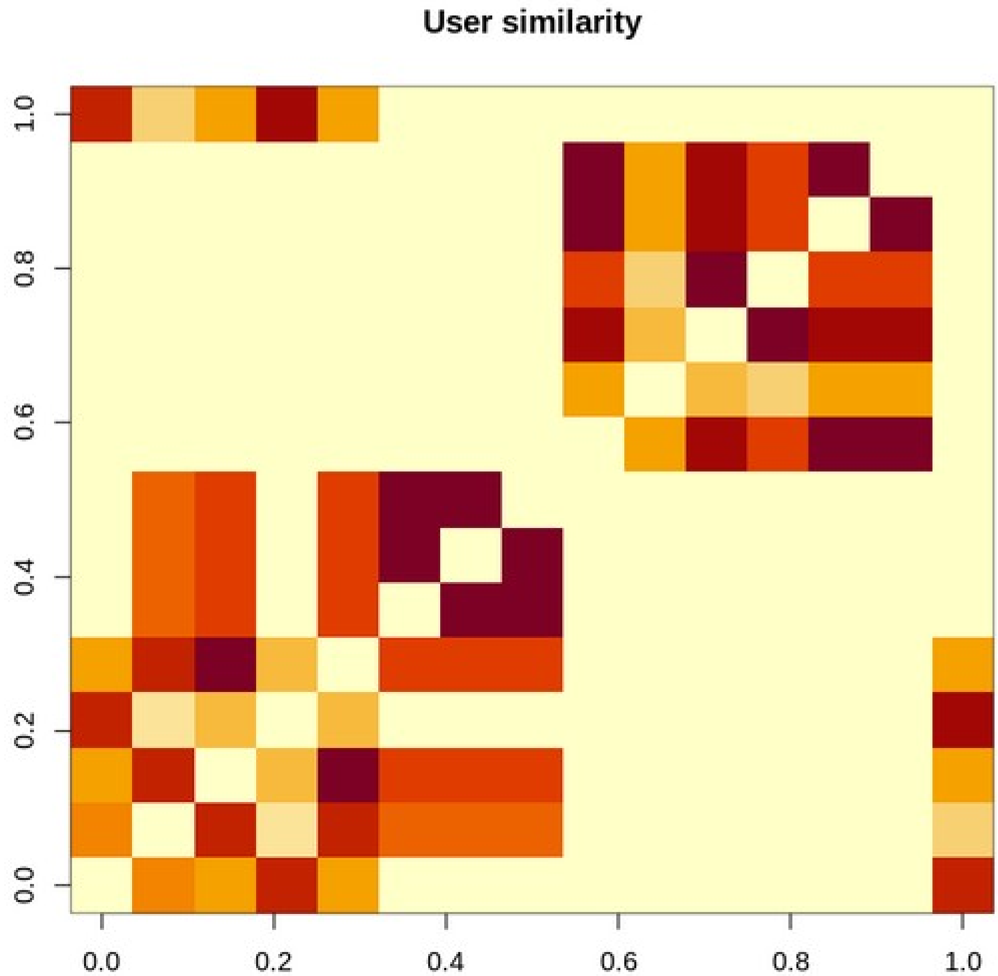
Then to apply the cosine similarity function:

	LN	MH	JH	EN	DU	FL	GL	AH	SA	RW	BA	MG	AF
LN	1.0000000	0.5587442	0.4216370	0.7503086	0.4216370	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
MH	0.5587442	1.0000000	0.7730207	0.2482286	0.7730207	0.6246950	0.6246950	0.6246950	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
JH	0.4216370	0.7730207	1.0000000	0.3746343	1.0000000	0.7071068	0.7071068	0.7071068	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
EN	0.7503086	0.2482286	0.3746343	1.0000000	0.3746343	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
DU	0.4216370	0.7730207	1.0000000	0.3746343	1.0000000	0.7071068	0.7071068	0.7071068	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
FL	0.0000000	0.6246950	0.7071068	0.0000000	0.7071068	1.0000000	1.0000000	1.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
GL	0.0000000	0.6246950	0.7071068	0.0000000	0.7071068	1.0000000	1.0000000	1.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
AH	0.0000000	0.6246950	0.7071068	0.0000000	0.7071068	1.0000000	1.0000000	1.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
SA	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	1.0000000	0.4472136	0.8944272	0.7071068	1.0000000
RW	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.4472136	1.0000000	0.4000000	0.3162278	0.4472136
BA	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.8944272	0.4000000	1.0000000	0.9486833	0.8944272
MG	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.7071068	0.3162278	0.9486833	1.0000000	0.7071068
AF	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	1.0000000	0.4472136	0.8944272	0.7071068	1.0000000
KG	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	1.0000000	0.4472136	0.8944272	0.7071068	1.0000000
DS	0.7950464	0.3123475	0.4714045	0.8830216	0.4714045	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000

	SQL	Spatial	PA1	DM.in.R	Python	Forcast	R.prog	Hadoop	Regression
SQL	1.0000000	0.4155844	0.0000000	0.6115766	0.5470108	0.0000000	0.7895420	0.0000000	0.6321395
Spatial	0.4155844	1.0000000	0.0000000	0.0000000	0.3646738	0.0000000	0.0000000	0.0000000	0.0000000
PA1	0.0000000	0.0000000	1.0000000	0.0000000	0.0000000	0.6115766	0.0000000	0.2279212	0.0000000
DM.in.R	0.6115766	0.0000000	0.0000000	1.0000000	0.0000000	0.0000000	0.7745967	0.0000000	0.7442084
Python	0.5470108	0.3646738	0.0000000	0.0000000	1.0000000	0.0000000	0.3464102	0.0000000	0.3328201
Forcast	0.0000000	0.0000000	0.6115766	0.0000000	0.0000000	1.0000000	0.0000000	0.0000000	0.0000000
R.prog	0.7895420	0.0000000	0.0000000	0.7745967	0.3464102	0.0000000	1.0000000	0.0000000	0.8006408
Hadoop	0.0000000	0.0000000	0.2279212	0.0000000	0.0000000	0.0000000	0.0000000	1.0000000	0.0000000
Regression	0.6321395	0.0000000	0.0000000	0.7442084	0.3328201	0.0000000	0.8006408	0.0000000	1.0000000

4) Based on the cosine similarities of the nearest students to E.N., which course should be recommended to E.N.?

A user similarity graph:



The highest value of cosine similarity is between DS student, for student EN.
But EM has already taken his courses, that's why we looked for another student and the next was DU or JH who are the same in similarity.
The recommendation afterwards was directed to spatial course.

5) Apply item-based collaborative filtering to this dataset (using R) and based on the results, recommend a course to E.N.

Here is the students ordered in similariy between EM students

$LN
$MH
$JH
$EN
$DU
$FL
$GL
$AH
$SA
$RW
$BA
$MG
$AF
$KG
$DS

LN has enrolled in SQL, Python,forecast, and regression.
We know that EM has enrolled in all except forecast, that’s why we are recommending forcast.

References:

[1] Lab code and lecture notes
[2] https://www.kdnuggets.com/2016/04/association-rules-apriori-algorithm-tutorial.html
[3] https://www.datacamp.com/community/tutorials/market-basket-analysis-r
[4] https://www.datamentor.io/r-programming/matrix/
[5] https://datatofish.com/create-dataframe-in-r/
[6] https://stat.ethz.ch/R-manual/R-devel/library/base/html/data.matrix.html
[7] https://stackoverflow.com/questions/8161836/how-do-i-replace-na-values-with-zeros-in-an-r-dataframe
[8] https://rstudio-pubs-static.s3.amazonaws.com/284255_d198d70146db4644860666cb1a5d6c01.html