**Prepare rules for the all the data sets**

**1) Try different values of support and confidence. Observe the change in number of rules for different support,confidence values**

**2) Change the minimum length in apriori algorithm**

**3) Visulize the obtained rules using different plots**

> movies <- read.csv(file.choose())

> View(movies)

> head(movies)

V1 V2 V3 V4 V5 Sixth.Sense Gladiator LOTR1 Harry.Potter1 Patriot LOTR2

1 Sixth Sense LOTR1 Harry Potter1 Green Mile LOTR2 1 0 1 1 0 1

2 Gladiator Patriot Braveheart 0 1 0 0 1 0

3 LOTR1 LOTR2 0 0 1 0 0 1

4 Gladiator Patriot Sixth Sense 1 1 0 0 1 0

5 Gladiator Patriot Sixth Sense 1 1 0 0 1 0

6 Gladiator Patriot Sixth Sense 1 1 0 0 1 0

Harry.Potter2 LOTR Braveheart Green.Mile

1 0 0 0 1

2 0 0 1 0

3 0 0 0 0

4 0 0 0 0

5 0 0 0 0

6 0 0 0 0

>

> summary(movies)

V1 V2 V3 V4 V5 Sixth.Sense Gladiator

Gladiator :6 Harry Potter2:1 :3 :8 :9 Min. :0.0 Min. :0.00

Harry Potter1:1 LOTR :1 Braveheart :1 Green Mile:2 LOTR2:1 1st Qu.:0.0 1st Qu.:0.25

LOTR1 :1 LOTR1 :1 Gladiator :1 Median :1.0 Median :1.00

Sixth Sense :2 LOTR2 :1 Harry Potter1:1 Mean :0.6 Mean :0.70

Patriot :6 Sixth Sense :4 3rd Qu.:1.0 3rd Qu.:1.00

Max. :1.0 Max. :1.00

LOTR1 Harry.Potter1 Patriot LOTR2 Harry.Potter2 LOTR Braveheart

Min. :0.0 Min. :0.0 Min. :0.0 Min. :0.0 Min. :0.0 Min. :0.0 Min. :0.0

1st Qu.:0.0 1st Qu.:0.0 1st Qu.:0.0 1st Qu.:0.0 1st Qu.:0.0 1st Qu.:0.0 1st Qu.:0.0

Median :0.0 Median :0.0 Median :1.0 Median :0.0 Median :0.0 Median :0.0 Median :0.0

Mean :0.2 Mean :0.2 Mean :0.6 Mean :0.2 Mean :0.1 Mean :0.1 Mean :0.1

3rd Qu.:0.0 3rd Qu.:0.0 3rd Qu.:1.0 3rd Qu.:0.0 3rd Qu.:0.0 3rd Qu.:0.0 3rd Qu.:0.0

Max. :1.0 Max. :1.0 Max. :1.0 Max. :1.0 Max. :1.0 Max. :1.0 Max. :1.0

Green.Mile

Min. :0.0

1st Qu.:0.0

Median :0.0

Mean :0.2

3rd Qu.:0.0

Max. :1.0

>

> corrplot(cor(movies[,6:15]),method = "square",type = "upper")

>



> movies\_rules <- apriori(as.matrix(movies[,6:15]),parameter = list(support = 0.005,confidence= 0.05,minlen=3))

Apriori

Parameter specification:

confidence minval smax arem aval originalSupport maxtime support minlen maxlen target ext

0.05 0.1 1 none FALSE TRUE 5 0.005 3 10 rules TRUE

Algorithmic control:

filter tree heap memopt load sort verbose

0.1 TRUE TRUE FALSE TRUE 2 TRUE

Absolute minimum support count: 0

set item appearances ...[0 item(s)] done [0.00s].

set transactions ...[10 item(s), 10 transaction(s)] done [0.00s].

sorting and recoding items ... [10 item(s)] done [0.00s].

creating transaction tree ... done [0.00s].

checking subsets of size 1 2 3 4 5 done [0.01s].

writing ... [77 rule(s)] done [0.00s].

creating S4 object ... done [0.00s].

> movies\_rules

set of 77 rules

> inspect(head(sort(movies\_rules,by="lift")))

lhs rhs support confidence coverage lift count

[1] {Gladiator,Green.Mile} => {LOTR} 0.1 1.0 0.1 10 1

[2] {Sixth.Sense,Gladiator,Green.Mile} => {LOTR} 0.1 1.0 0.1 10 1

[3] {Gladiator,LOTR} => {Green.Mile} 0.1 1.0 0.1 5 1

[4] {Sixth.Sense,LOTR} => {Green.Mile} 0.1 1.0 0.1 5 1

[5] {Sixth.Sense,Green.Mile} => {LOTR} 0.1 0.5 0.2 5 1

[6] {LOTR1,Harry.Potter1} => {LOTR2} 0.1 1.0 0.1 5 1

>

> inspect(head(sort(movies\_rules,by="confidence")))

lhs rhs support confidence coverage lift count

[1] {Patriot,Braveheart} => {Gladiator} 0.1 1 0.1 1.428571 1

[2] {Gladiator,Braveheart} => {Patriot} 0.1 1 0.1 1.666667 1

[3] {LOTR,Green.Mile} => {Gladiator} 0.1 1 0.1 1.428571 1

[4] {Gladiator,LOTR} => {Green.Mile} 0.1 1 0.1 5.000000 1

[5] {Gladiator,Green.Mile} => {LOTR} 0.1 1 0.1 10.000000 1

[6] {LOTR,Green.Mile} => {Sixth.Sense} 0.1 1 0.1 1.666667 1

>

> inspect(head(sort(movies\_rules,by="support")))

lhs rhs support confidence coverage lift count

[1] {Gladiator,Patriot} => {Sixth.Sense} 0.4 0.6666667 0.6 1.111111 4

[2] {Sixth.Sense,Patriot} => {Gladiator} 0.4 1.0000000 0.4 1.428571 4

[3] {Sixth.Sense,Gladiator} => {Patriot} 0.4 0.8000000 0.5 1.333333 4

[4] {Patriot,Braveheart} => {Gladiator} 0.1 1.0000000 0.1 1.428571 1

[5] {Gladiator,Braveheart} => {Patriot} 0.1 1.0000000 0.1 1.666667 1

[6] {Gladiator,Patriot} => {Braveheart} 0.1 0.1666667 0.6 1.666667 1

> inspect(head(sort(movies\_rules,by=c("count","lift"))))# max count = 4

lhs rhs support confidence coverage lift count

[1] {Sixth.Sense,Patriot} => {Gladiator} 0.4 1.0000000 0.4 1.428571 4

[2] {Sixth.Sense,Gladiator} => {Patriot} 0.4 0.8000000 0.5 1.333333 4

[3] {Gladiator,Patriot} => {Sixth.Sense} 0.4 0.6666667 0.6 1.111111 4

[4] {Gladiator,Green.Mile} => {LOTR} 0.1 1.0000000 0.1 10.000000 1

[5] {Sixth.Sense,Gladiator,Green.Mile} => {LOTR} 0.1 1.0000000 0.1 10.000000 1

[6] {Gladiator,LOTR} => {Green.Mile} 0.1 1.0000000 0.1 5.000000 1

>

> head(quality(movies\_rules))

support confidence coverage lift count

1 0.1 1.0000000 0.1 1.428571 1

2 0.1 1.0000000 0.1 1.666667 1

3 0.1 0.1666667 0.6 1.666667 1

4 0.1 1.0000000 0.1 1.428571 1

5 0.1 1.0000000 0.1 5.000000 1

6 0.1 1.0000000 0.1 10.000000 1

>

> plot(movies\_rules,method = "scatterplot",jitter=0)



> plot(movies\_rules,method = "grouped matrix")



> plot(head(sort(movies\_rules,by="lift"),n=10),method = "graph")

>



> plot(movies\_rules,method = "paracoord")



> plot(movies\_rules,method = "matrix")

Itemsets in Antecedent (LHS)

[1] "{Sixth.Sense,Gladiator,Green.Mile}" "{Gladiator,Green.Mile}"

[3] "{Sixth.Sense,LOTR1}" "{Sixth.Sense,Harry.Potter1}"

[5] "{Sixth.Sense,LOTR2}" "{Sixth.Sense,Gladiator,LOTR}"

[7] "{Sixth.Sense,LOTR1,Harry.Potter1}" "{Sixth.Sense,LOTR1,LOTR2}"

[9] "{Sixth.Sense,Harry.Potter1,LOTR2}" "{Sixth.Sense,LOTR1,Green.Mile}"

[11] "{Sixth.Sense,Harry.Potter1,Green.Mile}" "{Sixth.Sense,LOTR2,Green.Mile}"

[13] "{Sixth.Sense,LOTR1,Harry.Potter1,LOTR2}" "{Sixth.Sense,LOTR1,Harry.Potter1,Green.Mile}"

[15] "{Sixth.Sense,LOTR1,LOTR2,Green.Mile}" "{Sixth.Sense,Harry.Potter1,LOTR2,Green.Mile}"

[17] "{LOTR1,Harry.Potter1}" "{Harry.Potter1,LOTR2}"

[19] "{LOTR1,Green.Mile}" "{Harry.Potter1,Green.Mile}"

[21] "{LOTR2,Green.Mile}" "{Gladiator,LOTR}"

[23] "{LOTR1,Harry.Potter1,LOTR2}" "{LOTR1,Harry.Potter1,Green.Mile}"

[25] "{LOTR1,LOTR2,Green.Mile}" "{Harry.Potter1,LOTR2,Green.Mile}"

[27] "{Sixth.Sense,LOTR}" "{Sixth.Sense,Green.Mile}"

[29] "{LOTR1,LOTR2}" "{Gladiator,Braveheart}"

[31] "{Gladiator,LOTR,Green.Mile}" "{LOTR1,Harry.Potter1,LOTR2,Green.Mile}"

[33] "{LOTR,Green.Mile}" "{Sixth.Sense,Gladiator}"

[35] "{Patriot,Braveheart}" "{Sixth.Sense,Patriot}"

[37] "{Sixth.Sense,LOTR,Green.Mile}" "{Gladiator,Patriot}"

Itemsets in Consequent (RHS)

[1] "{Gladiator}" "{Patriot}" "{Sixth.Sense}" "{Braveheart}" "{Harry.Potter1}" "{Green.Mile}"

[7] "{LOTR2}" "{LOTR1}" "{LOTR}"

>



**Applying apriori algorithm**

* **For support=0.002,confidence=0.5,minlen=2**

> movies\_rules2 <- apriori(as.matrix(movies[,6:15]),parameter = list(support =0.005, confidence =0.5,minlen=2))

Apriori

Parameter specification:

confidence minval smax arem aval originalSupport maxtime support minlen maxlen target ext

0.5 0.1 1 none FALSE TRUE 5 0.005 2 10 rules TRUE

Algorithmic control:

filter tree heap memopt load sort verbose

0.1 TRUE TRUE FALSE TRUE 2 TRUE

Absolute minimum support count: 0

set item appearances ...[0 item(s)] done [0.00s].

set transactions ...[10 item(s), 10 transaction(s)] done [0.00s].

sorting and recoding items ... [10 item(s)] done [0.00s].

creating transaction tree ... done [0.00s].

checking subsets of size 1 2 3 4 5 done [0.00s].

writing ... [105 rule(s)] done [0.00s].

creating S4 object ... done [0.00s].

>

> inspect(head(sort(movies\_rules2,by="lift")))

lhs rhs support confidence coverage lift count

[1] {Gladiator,Green.Mile} => {LOTR} 0.1 1.0 0.1 10 1

[2] {Sixth.Sense,Gladiator,Green.Mile} => {LOTR} 0.1 1.0 0.1 10 1

[3] {Harry.Potter2} => {Harry.Potter1} 0.1 1.0 0.1 5 1

[4] {Harry.Potter1} => {Harry.Potter2} 0.1 0.5 0.2 5 1

[5] {LOTR} => {Green.Mile} 0.1 1.0 0.1 5 1

[6] {Green.Mile} => {LOTR} 0.1 0.5 0.2 5 1

>

> inspect(head(sort(movies\_rules2,by="confidence")))

lhs rhs support confidence coverage lift count

[1] {Harry.Potter2} => {Harry.Potter1} 0.1 1 0.1 5.000000 1

[2] {Braveheart} => {Patriot} 0.1 1 0.1 1.666667 1

[3] {Braveheart} => {Gladiator} 0.1 1 0.1 1.428571 1

[4] {LOTR} => {Green.Mile} 0.1 1 0.1 5.000000 1

[5] {LOTR} => {Gladiator} 0.1 1 0.1 1.428571 1

[6] {LOTR} => {Sixth.Sense} 0.1 1 0.1 1.666667 1

> > inspect(head(sort(movies\_rules2,by="support")))

lhs rhs support confidence coverage lift count

[1] {Patriot} => {Gladiator} 0.6 1.0000000 0.6 1.428571 6

[2] {Gladiator} => {Patriot} 0.6 0.8571429 0.7 1.428571 6

[3] {Gladiator} => {Sixth.Sense} 0.5 0.7142857 0.7 1.190476 5

[4] {Sixth.Sense} => {Gladiator} 0.5 0.8333333 0.6 1.190476 5

[5] {Patriot} => {Sixth.Sense} 0.4 0.6666667 0.6 1.111111 4

[6] {Sixth.Sense} => {Patriot} 0.4 0.6666667 0.6 1.111111 4

>

> plot(movies\_rules2,method = "scatterplot",jitter=0)

>



> plot(head(sort(movies\_rules2,by="lift"),n=20),method = "graph")

>



> plot(movies\_rules2,method = "paracoord")

>



> plot(movies\_rules,method = "matrix")

Itemsets in Antecedent (LHS)

[1] "{Sixth.Sense,Gladiator,Green.Mile}" "{Gladiator,Green.Mile}"

[3] "{Sixth.Sense,LOTR1}" "{Sixth.Sense,Harry.Potter1}"

[5] "{Sixth.Sense,LOTR2}" "{Sixth.Sense,Gladiator,LOTR}"

[7] "{Sixth.Sense,LOTR1,Harry.Potter1}" "{Sixth.Sense,LOTR1,LOTR2}"

[9] "{Sixth.Sense,Harry.Potter1,LOTR2}" "{Sixth.Sense,LOTR1,Green.Mile}"

[11] "{Sixth.Sense,Harry.Potter1,Green.Mile}" "{Sixth.Sense,LOTR2,Green.Mile}"

[13] "{Sixth.Sense,LOTR1,Harry.Potter1,LOTR2}" "{Sixth.Sense,LOTR1,Harry.Potter1,Green.Mile}"

[15] "{Sixth.Sense,LOTR1,LOTR2,Green.Mile}" "{Sixth.Sense,Harry.Potter1,LOTR2,Green.Mile}"

[17] "{LOTR1,Harry.Potter1}" "{Harry.Potter1,LOTR2}"

[19] "{LOTR1,Green.Mile}" "{Harry.Potter1,Green.Mile}"

[21] "{LOTR2,Green.Mile}" "{Gladiator,LOTR}"

[23] "{LOTR1,Harry.Potter1,LOTR2}" "{LOTR1,Harry.Potter1,Green.Mile}"

[25] "{LOTR1,LOTR2,Green.Mile}" "{Harry.Potter1,LOTR2,Green.Mile}"

[27] "{Sixth.Sense,LOTR}" "{Sixth.Sense,Green.Mile}"

[29] "{LOTR1,LOTR2}" "{Gladiator,Braveheart}"

[31] "{Gladiator,LOTR,Green.Mile}" "{LOTR1,Harry.Potter1,LOTR2,Green.Mile}"

[33] "{LOTR,Green.Mile}" "{Sixth.Sense,Gladiator}"

[35] "{Patriot,Braveheart}" "{Sixth.Sense,Patriot}"

[37] "{Sixth.Sense,LOTR,Green.Mile}" "{Gladiator,Patriot}"

Itemsets in Consequent (RHS)

[1] "{Gladiator}" "{Patriot}" "{Sixth.Sense}" "{Braveheart}" "{Harry.Potter1}" "{Green.Mile}"

[7] "{LOTR2}" "{LOTR1}" "{LOTR}"

>

> plot(movies\_rules2,method = "matrix",engine = "3d")

Itemsets in Antecedent (LHS)

[1] "{Sixth.Sense,Gladiator,Green.Mile}" "{Gladiator,Green.Mile}"

[3] "{Harry.Potter2}" "{Sixth.Sense,LOTR1}"

[5] "{Sixth.Sense,Harry.Potter1}" "{Sixth.Sense,LOTR2}"

[7] "{Sixth.Sense,Gladiator,LOTR}" "{Sixth.Sense,LOTR1,Harry.Potter1}"

[9] "{Sixth.Sense,LOTR1,LOTR2}" "{Sixth.Sense,Harry.Potter1,LOTR2}"

[11] "{Sixth.Sense,LOTR1,Green.Mile}" "{Sixth.Sense,Harry.Potter1,Green.Mile}"

[13] "{Sixth.Sense,LOTR2,Green.Mile}" "{Sixth.Sense,LOTR1,Harry.Potter1,LOTR2}"

[15] "{Sixth.Sense,LOTR1,Harry.Potter1,Green.Mile}" "{Sixth.Sense,LOTR1,LOTR2,Green.Mile}"

[17] "{Sixth.Sense,Harry.Potter1,LOTR2,Green.Mile}" "{LOTR1,Harry.Potter1}"

[19] "{Harry.Potter1,LOTR2}" "{LOTR1,Green.Mile}"

[21] "{Harry.Potter1,Green.Mile}" "{LOTR2,Green.Mile}"

[23] "{Gladiator,LOTR}" "{LOTR1,Harry.Potter1,LOTR2}"

[25] "{LOTR1,Harry.Potter1,Green.Mile}" "{LOTR1,LOTR2,Green.Mile}"

[27] "{Harry.Potter1,LOTR2,Green.Mile}" "{Sixth.Sense,LOTR}"

[29] "{LOTR1}" "{LOTR2}"

[31] "{LOTR}" "{Harry.Potter1}"

[33] "{Sixth.Sense,Green.Mile}" "{Green.Mile}"

[35] "{LOTR1,LOTR2}" "{Gladiator,Braveheart}"

[37] "{Gladiator,LOTR,Green.Mile}" "{LOTR1,Harry.Potter1,LOTR2,Green.Mile}"

[39] "{Braveheart}" "{LOTR,Green.Mile}"

[41] "{Patriot,Braveheart}" "{Sixth.Sense,Patriot}"

[43] "{Sixth.Sense,LOTR,Green.Mile}" "{Sixth.Sense,Gladiator}"

[45] "{Gladiator}" "{Patriot}"

[47] "{Sixth.Sense}" "{Gladiator,Patriot}"

Itemsets in Consequent (RHS)

[1] "{Gladiator}" "{Patriot}" "{Sixth.Sense}" "{Harry.Potter1}" "{Green.Mile}" "{LOTR1}"

[7] "{LOTR2}" "{Harry.Potter2}" "{LOTR}"

>

>



**So From Above cases we observed that,**

**1. Lower the Confidence level Higher the no. of rules.**

**2. Lower the minlen(minimum length), Higher the no. of Rules are getting generated.**

**3. Higher the Support, lower the no. of rules.**

**4. So, we can conclude that these 3 parameters different combinations can generate different rules.**