SD\_Assignment 2

Steve Dunn

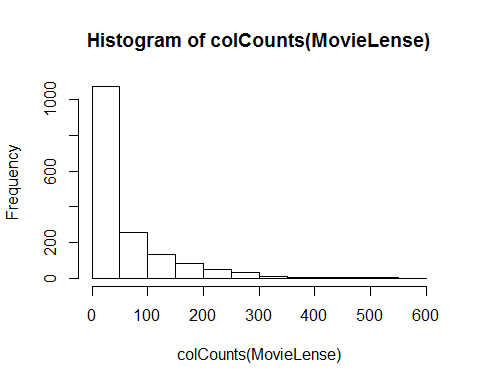
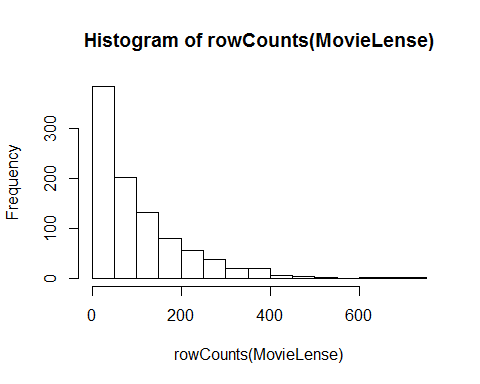
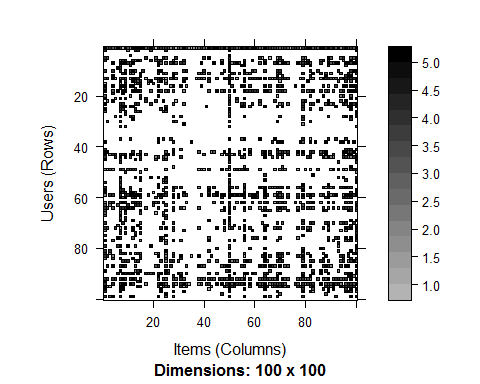
June 26, 2016

Prof. Andy Catlin IS 643 - Special Topics: Recommender Systems Department of Data Analytics, City University of New York

# ======================================================================================

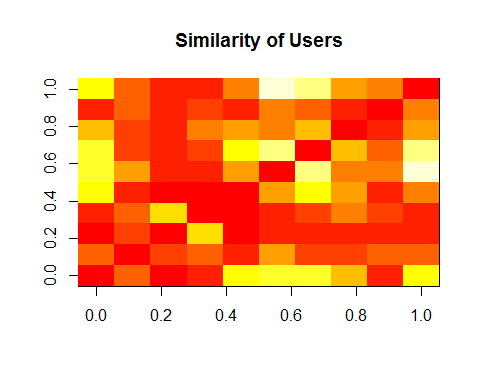
### Using R package to build the similarity and Collaborative Filtering recommender System code and data set Mostly taken from example in Building a Recommendation System with R

# ======================================================================================

While the models use for recommendations do as is expected i.e. make recommendation the accuracy and precision of those recommendation have to be evaluated to see if they make sense either at a user -user or item-item filtering. The accuracy is measured with RMSE, MSE and MAE. Since the measure as calculated differently and apply different weightings I would choose between the RMSE or MAE.I also found that using k-fold cross validation is faster than splitting the data set manually. 

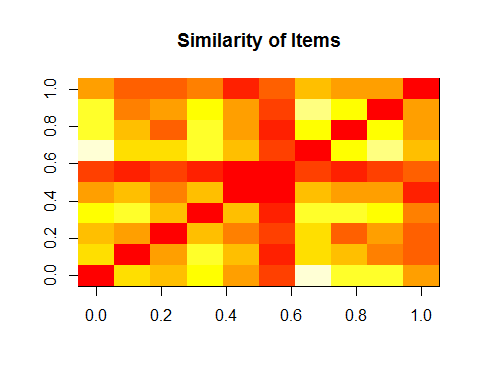
## [1] 3.587565

## 1 2 3 4 5 6  
## 1 0.00000000 0.16893670 0.03827203 0.06634975 0.37967009 0.42968246  
## 2 0.16893670 0.00000000 0.09706862 0.15310468 0.07362338 0.24210639  
## 3 0.03827203 0.09706862 0.00000000 0.33343036 0.02200430 0.06017539  
## 4 0.06634975 0.15310468 0.33343036 0.00000000 0.03268542 0.05730374  
## 5 0.37967009 0.07362338 0.02200430 0.03268542 0.00000000 0.23863583  
## 6 0.42968246 0.24210639 0.06017539 0.05730374 0.23863583 0.00000000  
## 7 0.44048658 0.10867688 0.06368331 0.09410511 0.37498291 0.49185063  
## 8 0.32007948 0.10425672 0.08603051 0.19327073 0.24892997 0.20251387  
## 9 0.07838506 0.16246959 0.06322298 0.10408973 0.05684700 0.18499701  
## 10 0.37773263 0.16127340 0.06748118 0.06254540 0.20142701 0.55485085  
## 7 8 9 10  
## 1 0.44048658 0.32007948 0.07838506 0.37773263  
## 2 0.10867688 0.10425672 0.16246959 0.16127340  
## 3 0.06368331 0.08603051 0.06322298 0.06748118  
## 4 0.09410511 0.19327073 0.10408973 0.06254540  
## 5 0.37498291 0.24892997 0.05684700 0.20142701  
## 6 0.49185063 0.20251387 0.18499701 0.55485085  
## 7 0.00000000 0.28600538 0.14618947 0.48882709  
## 8 0.28600538 0.00000000 0.08594195 0.23328945  
## 9 0.14618947 0.08594195 0.00000000 0.19822253  
## 10 0.48882709 0.23328945 0.19822253 0.00000000



## 1 2 3 4 5 6  
## 1 0.00000000 0.16893670 0.03827203 0.06634975 0.37967009 0.42968246  
## 2 0.16893670 0.00000000 0.09706862 0.15310468 0.07362338 0.24210639  
## 3 0.03827203 0.09706862 0.00000000 0.33343036 0.02200430 0.06017539  
## 4 0.06634975 0.15310468 0.33343036 0.00000000 0.03268542 0.05730374  
## 5 0.37967009 0.07362338 0.02200430 0.03268542 0.00000000 0.23863583  
## 6 0.42968246 0.24210639 0.06017539 0.05730374 0.23863583 0.00000000  
## 7 8 9 10  
## 1 0.44048658 0.32007948 0.07838506 0.37773263  
## 2 0.10867688 0.10425672 0.16246959 0.16127340  
## 3 0.06368331 0.08603051 0.06322298 0.06748118  
## 4 0.09410511 0.19327073 0.10408973 0.06254540  
## 5 0.37498291 0.24892997 0.05684700 0.20142701  
## 6 0.49185063 0.20251387 0.18499701 0.55485085

## Toy Story (1995)  
## Toy Story (1995) 0.0000000  
## GoldenEye (1995) 0.4023822  
## Four Rooms (1995) 0.3302448  
## Get Shorty (1995) 0.4549379  
## Copycat (1995) 0.2867135  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.1163440  
## Twelve Monkeys (1995) 0.6209786  
## Babe (1995) 0.4811139  
## Dead Man Walking (1995) 0.4962884  
## Richard III (1995) 0.2739351  
## GoldenEye (1995)  
## Toy Story (1995) 0.40238218  
## GoldenEye (1995) 0.00000000  
## Four Rooms (1995) 0.27306918  
## Get Shorty (1995) 0.50257077  
## Copycat (1995) 0.31883618  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.08356281  
## Twelve Monkeys (1995) 0.38340339  
## Babe (1995) 0.33700186  
## Dead Man Walking (1995) 0.25525203  
## Richard III (1995) 0.17108221  
## Four Rooms (1995)  
## Toy Story (1995) 0.3302448  
## GoldenEye (1995) 0.2730692  
## Four Rooms (1995) 0.0000000  
## Get Shorty (1995) 0.3248664  
## Copycat (1995) 0.2129566  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.1067223  
## Twelve Monkeys (1995) 0.3729207  
## Babe (1995) 0.2007939  
## Dead Man Walking (1995) 0.2736693  
## Richard III (1995) 0.1581043  
## Get Shorty (1995)  
## Toy Story (1995) 0.45493792  
## GoldenEye (1995) 0.50257077  
## Four Rooms (1995) 0.32486639  
## Get Shorty (1995) 0.00000000  
## Copycat (1995) 0.33423948  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.09030829  
## Twelve Monkeys (1995) 0.48928280  
## Babe (1995) 0.49023553  
## Dead Man Walking (1995) 0.41904357  
## Richard III (1995) 0.25256072  
## Copycat (1995)  
## Toy Story (1995) 0.28671351  
## GoldenEye (1995) 0.31883618  
## Four Rooms (1995) 0.21295656  
## Get Shorty (1995) 0.33423948  
## Copycat (1995) 0.00000000  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.03729866  
## Twelve Monkeys (1995) 0.33476858  
## Babe (1995) 0.25916097  
## Dead Man Walking (1995) 0.27244840  
## Richard III (1995) 0.05545322  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995)  
## Toy Story (1995) 0.11634398  
## GoldenEye (1995) 0.08356281  
## Four Rooms (1995) 0.10672227  
## Get Shorty (1995) 0.09030829  
## Copycat (1995) 0.03729866  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.00000000  
## Twelve Monkeys (1995) 0.13961658  
## Babe (1995) 0.08387647  
## Dead Man Walking (1995) 0.15106449  
## Richard III (1995) 0.20309700  
## Twelve Monkeys (1995)  
## Toy Story (1995) 0.6209786  
## GoldenEye (1995) 0.3834034  
## Four Rooms (1995) 0.3729207  
## Get Shorty (1995) 0.4892828  
## Copycat (1995) 0.3347686  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.1396166  
## Twelve Monkeys (1995) 0.0000000  
## Babe (1995) 0.4235145  
## Dead Man Walking (1995) 0.5274623  
## Richard III (1995) 0.3186228  
## Babe (1995)  
## Toy Story (1995) 0.48111389  
## GoldenEye (1995) 0.33700186  
## Four Rooms (1995) 0.20079389  
## Get Shorty (1995) 0.49023553  
## Copycat (1995) 0.25916097  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.08387647  
## Twelve Monkeys (1995) 0.42351452  
## Babe (1995) 0.00000000  
## Dead Man Walking (1995) 0.42442894  
## Richard III (1995) 0.26776402  
## Dead Man Walking (1995)  
## Toy Story (1995) 0.4962884  
## GoldenEye (1995) 0.2552520  
## Four Rooms (1995) 0.2736693  
## Get Shorty (1995) 0.4190436  
## Copycat (1995) 0.2724484  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.1510645  
## Twelve Monkeys (1995) 0.5274623  
## Babe (1995) 0.4244289  
## Dead Man Walking (1995) 0.0000000  
## Richard III (1995) 0.2885144  
## Richard III (1995)  
## Toy Story (1995) 0.27393511  
## GoldenEye (1995) 0.17108221  
## Four Rooms (1995) 0.15810426  
## Get Shorty (1995) 0.25256072  
## Copycat (1995) 0.05545322  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.20309700  
## Twelve Monkeys (1995) 0.31862281  
## Babe (1995) 0.26776402  
## Dead Man Walking (1995) 0.28851441  
## Richard III (1995) 0.00000000



## Toy Story (1995)  
## Toy Story (1995) 0.0000000  
## GoldenEye (1995) 0.4023822  
## Four Rooms (1995) 0.3302448  
## Get Shorty (1995) 0.4549379  
## Copycat (1995) 0.2867135  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.1163440  
## GoldenEye (1995)  
## Toy Story (1995) 0.40238218  
## GoldenEye (1995) 0.00000000  
## Four Rooms (1995) 0.27306918  
## Get Shorty (1995) 0.50257077  
## Copycat (1995) 0.31883618  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.08356281  
## Four Rooms (1995)  
## Toy Story (1995) 0.3302448  
## GoldenEye (1995) 0.2730692  
## Four Rooms (1995) 0.0000000  
## Get Shorty (1995) 0.3248664  
## Copycat (1995) 0.2129566  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.1067223  
## Get Shorty (1995)  
## Toy Story (1995) 0.45493792  
## GoldenEye (1995) 0.50257077  
## Four Rooms (1995) 0.32486639  
## Get Shorty (1995) 0.00000000  
## Copycat (1995) 0.33423948  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.09030829  
## Copycat (1995)  
## Toy Story (1995) 0.28671351  
## GoldenEye (1995) 0.31883618  
## Four Rooms (1995) 0.21295656  
## Get Shorty (1995) 0.33423948  
## Copycat (1995) 0.00000000  
## Shanghai Triad (Yao a yao yao dao waipo qiao) (1995) 0.03729866

# Splitting the Data set into train and test

#Subsetting the dataset  
ratings\_movies <- MovieLense[ rowCounts( MovieLense) > 50, colCounts( MovieLense) > 100]  
  
which\_train<-sample(x=c(TRUE,FALSE),size = nrow(ratings\_movies),replace = TRUE,prob = c(.8,.2))  
rec\_data\_train<-ratings\_movies[which\_train,]  
rec\_data\_test<-ratings\_movies[!which\_train,]  
  
#build recommender Model  
rec\_model<-Recommender(data=rec\_data\_train,method='IBCF')  
  
#predict 10 movies for the users  
n\_rec<-10  
rec\_predict<-predict(rec\_model,newdata=rec\_data\_test,n=n\_rec)  
  
rec\_predict

## Recommendations as 'topNList' with n = 10 for 119 users.

rec\_matrix<-sapply(rec\_predict@items,  
 function(x){  
 colnames(rec\_data\_test)[x]  
 })  
rec\_matrix[,1:10]

## 5 12   
## [1,] "Kiss the Girls (1997)" "Rumble in the Bronx (1995)"   
## [2,] "Lone Star (1996)" "Quiz Show (1994)"   
## [3,] "Killing Fields, The (1984)" "What's Eating Gilbert Grape (1993)"   
## [4,] "Trainspotting (1996)" "While You Were Sleeping (1995)"   
## [5,] "Raising Arizona (1987)" "Four Weddings and a Funeral (1994)"   
## [6,] "Apocalypse Now (1979)" "Hudsucker Proxy, The (1994)"   
## [7,] "My Left Foot (1989)" "Much Ado About Nothing (1993)"   
## [8,] "As Good As It Gets (1997)" "Truth About Cats & Dogs, The (1996)"  
## [9,] "L.A. Confidential (1997)" "On Golden Pond (1981)"   
## [10,] "Twelve Monkeys (1995)" "Abyss, The (1989)"   
## 15 23   
## [1,] "Babe (1995)" "Big Night (1996)"   
## [2,] "Wizard of Oz, The (1939)" "Secrets & Lies (1996)"   
## [3,] "Mr. Smith Goes to Washington (1939)" "L.A. Confidential (1997)"   
## [4,] "Die Hard (1988)" "Strictly Ballroom (1992)"   
## [5,] "Fish Called Wanda, A (1988)" "Trainspotting (1996)"   
## [6,] "Princess Bride, The (1987)" "Wrong Trousers, The (1993)"  
## [7,] "Aliens (1986)" "Maltese Falcon, The (1941)"  
## [8,] "12 Angry Men (1957)" "Kolya (1996)"   
## [9,] "Sting, The (1973)" "Boot, Das (1981)"   
## [10,] "Sling Blade (1996)" "North by Northwest (1959)"   
## 28 54   
## [1,] "Leaving Las Vegas (1995)" "Babe (1995)"   
## [2,] "First Wives Club, The (1996)" "Dances with Wolves (1990)"   
## [3,] "Singin' in the Rain (1952)" "Wizard of Oz, The (1939)"   
## [4,] "Hoop Dreams (1994)" "Mr. Smith Goes to Washington (1939)"  
## [5,] "True Romance (1993)" "Big Night (1996)"   
## [6,] "Deer Hunter, The (1978)" "Fish Called Wanda, A (1988)"   
## [7,] "Brazil (1985)" "Princess Bride, The (1987)"   
## [8,] "Chasing Amy (1997)" "12 Angry Men (1957)"   
## [9,] "Dead Man Walking (1995)" "Blues Brothers, The (1980)"   
## [10,] "Jackie Brown (1997)" "Henry V (1989)"   
## 57 64   
## [1,] "Postino, Il (1994)" "People vs. Larry Flynt, The (1996)"  
## [2,] "Dances with Wolves (1990)" "Jackie Brown (1997)"   
## [3,] "Lone Star (1996)" "L.A. Confidential (1997)"   
## [4,] "Citizen Kane (1941)" "Chasing Amy (1997)"   
## [5,] "2001: A Space Odyssey (1968)" "Patton (1970)"   
## [6,] "Cinema Paradiso (1988)" "Nikita (La Femme Nikita) (1990)"   
## [7,] "Psycho (1960)" "Close Shave, A (1995)"   
## [8,] "Graduate, The (1967)" "Wrong Trousers, The (1993)"   
## [9,] "Room with a View, A (1986)" "Magnificent Seven, The (1954)"   
## [10,] "As Good As It Gets (1997)" "Ben-Hur (1959)"   
## 75   
## [1,] "2001: A Space Odyssey (1968)"   
## [2,] "Die Hard (1988)"   
## [3,] "Good, The Bad and The Ugly, The (1966)"  
## [4,] "Apocalypse Now (1979)"   
## [5,] "Return of the Jedi (1983)"   
## [6,] "GoodFellas (1990)"   
## [7,] "Alien (1979)"   
## [8,] "Raging Bull (1980)"   
## [9,] "Bridge on the River Kwai, The (1957)"   
## [10,] "Shining, The (1980)"   
## 77   
## [1,] "Kiss the Girls (1997)"   
## [2,] "Peacemaker, The (1997)"   
## [3,] "Great Escape, The (1963)"   
## [4,] "Full Metal Jacket (1987)"   
## [5,] "Dead Poets Society (1989)"   
## [6,] "Heathers (1989)"   
## [7,] "Fantasia (1940)"   
## [8,] "Grosse Pointe Blank (1997)"  
## [9,] "Seven (Se7en) (1995)"   
## [10,] "Professional, The (1994)"

# evaluating Recommendationusing k fold Cross Validation

n\_fold <- 10   
items\_to\_keep <- 15  
rating\_threshold <- 3  
model\_to\_evaluate <- "IBCF"   
model\_parameters <- NULL

eval\_sets <- evaluationScheme( data = ratings\_movies,  
 method = "cross-validation"  
 , k = n\_fold, given = items\_to\_keep  
 , goodRating = rating\_threshold)  
  
eval\_prediction <- predict(object = rec\_model,  
 newdata = getData( eval\_sets, "known"),  
 n = 10, type = "ratings")  
  
eval\_accuracy <- calcPredictionAccuracy(  
 x = eval\_prediction,  
 data = getData(eval\_sets, "unknown"),  
 byUser = FALSE,  
 given = items\_to\_recommend)  
head( eval\_accuracy)

## RMSE MSE MAE   
## 1.059959 1.123512 0.780709

## RMSE MSE MAE

## 1.0114568 1.0230448 0.7419217

eval\_recommender <- Recommender(data = getData(eval\_sets, "train"),  
 method = model\_to\_evaluate,  
 parameter = model\_parameters)  
  
results <- evaluate(x = eval\_sets,  
 method = model\_to\_evaluate,  
 n = seq(10, 100, 10))

## IBCF run fold/sample [model time/prediction time]  
## 1 [0.77sec/0.03sec]   
## 2 [0.78sec/0.06sec]   
## 3 [0.8sec/0.03sec]   
## 4 [0.97sec/0.06sec]   
## 5 [0.79sec/0.03sec]   
## 6 [0.78sec/0.05sec]   
## 7 [0.82sec/0.05sec]   
## 8 [0.8sec/0.04sec]   
## 9 [0.81sec/0.04sec]   
## 10 [0.78sec/0.03sec]

## IBCF run fold/sample [model time/prediction time]

## 1 [0.71sec/0.05sec]

## 2 [0.73sec/0.05sec]

## 3 [0.77sec/0.04sec]

## 4 [0.77sec/0.03sec]

## 5 [0.75sec/0.05sec]

## 6 [0.79sec/0.05sec]

## 7 [0.77sec/0.04sec]

## 8 [0.76sec/0.05sec]

## 9 [0.73sec/0.05sec]

## 10 [0.75sec/0.05sec]

head( getConfusionMatrix( results)[[ 1]])

## TP FP FN TN precision recall TPR  
## 10 2.821429 7.178571 66.41071 240.5893 0.2821429 0.04161942 0.04161942  
## 20 5.839286 14.160714 63.39286 233.6071 0.2919643 0.08402031 0.08402031  
## 30 8.571429 21.428571 60.66071 226.3393 0.2857143 0.12181455 0.12181455  
## 40 11.160714 28.839286 58.07143 218.9286 0.2790179 0.16087556 0.16087556  
## 50 14.071429 35.928571 55.16071 211.8393 0.2814286 0.19859395 0.19859395  
## 60 17.250000 42.750000 51.98214 205.0179 0.2875000 0.24630716 0.24630716  
## FPR  
## 10 0.02857201  
## 20 0.05636302  
## 30 0.08525172  
## 40 0.11496581  
## 50 0.14292819  
## 60 0.17002880

## TP FP FN TN precision recall TPR

## 10 3.232143 6.767857 73.08929 233.9107 0.3232143 0.03688216 0.03688216

## 20 6.607143 13.392857 69.71429 227.2857 0.3303571 0.08214411 0.08214411

## 30 10.089286 19.910714 66.23214 220.7679 0.3363095 0.12863596 0.12863596

## 40 13.375000 26.625000 62.94643 214.0536 0.3343750 0.17386720 0.17386720

## 50 16.750000 33.250000 59.57143 207.4286 0.3350000 0.22388116 0.22388116

## 60 20.178571 39.821429 56.14286 200.8571 0.3363095 0.27266253 0.27266253

## FPR

## 10 0.02767918

## 20 0.05434228

## 30 0.08065479

## 40 0.10792881

## 50 0.13525171

## 60 0.16214875

table\_wide <- reshape( data =table\_Users,#convert table to wide  
 direction = "wide",  
 idvar ="UserID",  
 timevar = "MovieID",  
 v.names = "Rating" )  
  
###Item Based Collaborative Filtering  
head(table\_wide[, 1: 10])

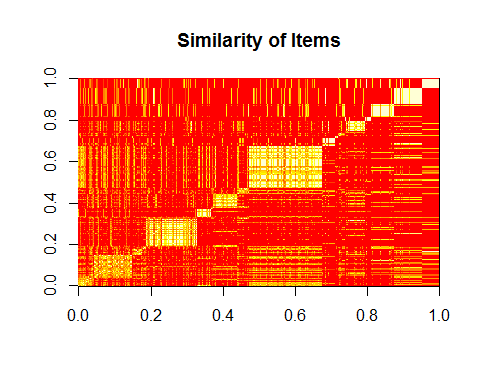
## UserID Rating.914 Rating.3408 Rating.2355 Rating.1197 Rating.1287  
## 1 1 3 4 5 3 5  
## 52 2 NA NA NA NA NA  
## 181 3 NA NA 5 5 NA  
## 232 4 NA NA NA NA NA  
## 253 5 NA 3 5 NA NA  
## 451 6 5 5 NA NA NA  
## Rating.2804 Rating.594 Rating.919 Rating.595  
## 1 5 4 4 5  
## 52 NA NA NA NA  
## 181 NA NA NA NA  
## 232 NA NA NA NA  
## 253 NA NA 4 NA  
## 451 NA NA NA 4

W\_movies<-table\_wide  
W\_movies <- (table\_wide[,!(names(table\_wide) %in% c("UserID"))])  
W\_movies[is.na(W\_movies)] = 0  
head(W\_movies[, 1:10])

## Rating.914 Rating.3408 Rating.2355 Rating.1197 Rating.1287 Rating.2804  
## 1 3 4 5 3 5 5  
## 52 0 0 0 0 0 0  
## 181 0 0 5 5 0 0  
## 232 0 0 0 0 0 0  
## 253 0 3 5 0 0 0  
## 451 5 5 0 0 0 0  
## Rating.594 Rating.919 Rating.595 Rating.938  
## 1 4 4 5 4  
## 52 0 0 0 0  
## 181 0 0 0 0  
## 232 0 0 0 0  
## 253 0 4 0 0  
## 451 0 0 4 0

# Function to calculate cosine Similarity  
Sim\_Cos <- function(a,b)   
{  
 cos\_ine <- sum(a\*b) / (sqrt(sum(a\*a)) \* sqrt(sum(b\*b)))  
 return(cos\_ine)  
}  
  
#Fill table with item rating  
W\_movies\_sim <- matrix(NA, nrow=ncol(W\_movies)  
 ,ncol=ncol(W\_movies)  
 ,dimnames=list(colnames(W\_movies)  
 ,colnames(W\_movies)))  
  
for(i in 1:ncol(W\_movies)) {  
 # Loop through the columns   
 for(j in 1:ncol(W\_movies)) {  
 # Fill matrix with cosine similarities  
 W\_movies\_sim[i,j] <- Sim\_Cos(as.matrix(W\_movies[i]),as.matrix(W\_movies[j]))  
 }  
}

image(as.matrix(W\_movies\_sim),main="Similarity of Items")



W\_movies\_sim <- as.data.frame(W\_movies\_sim)  
  
W\_movies\_n <- matrix(NA, nrow=ncol(W\_movies\_sim),ncol=5,dimnames=list(colnames(W\_movies\_sim)))  
  
for(i in 1:ncol(W\_movies))   
{  
 W\_movies\_n[i,] <- (t(head(n=5,  
 rownames(W\_movies\_sim  
 [order(W\_movies\_sim[,i]  
 ,decreasing=TRUE),][i]))))  
}  
  
#convert to rating Matrix  
ratings\_movies <- as(W\_movies\_sim, "realRatingMatrix")  
ratings\_movies

## 154 x 240 rating matrix of class 'realRatingMatrix' with 300 ratings.

#Splitting the Data set into train and test  
which\_train<-sample(x=c(TRUE,FALSE),size = nrow(ratings\_movies),replace = TRUE,prob = c(.8,.2))  
rec\_data\_train<-ratings\_movies[which\_train,]  
rec\_data\_test<-ratings\_movies[!which\_train,]  
  
#using the training dataset to create recommender model  
rec <- Recommender(rec\_data\_train, method = "IBCF")  
  
#predict recommended Movie for users on test dataset  
rec\_pred <- predict(rec, rec\_data\_test, n=10)  
rec\_pred

## Recommendations as 'topNList' with n = 10 for 32 users.

pre<-sapply(rec\_pred@items,  
 function(x){  
 colnames(ratings\_movies)[x]  
 })