Project1_Data612

Samriti Malhotra, Vishal Arora

June 7, 2019

Contents

Building a basic Recommender system.

This is a basic recommender system where we created a sample ratings grid where movie(s) have been rated by users.Few vales are NA and hence this recommender system using BaseLine Predictor(avg mean + rowBias + columnB ias) will try to predict the ratings for the ones which have not been rated and at the end we will check how much improvment it brings to train and test data set .

The libraries used during this project are as following
reshape2
kableExtra

Load Data

Load the sample data from githubSample Movie Data. Using reshape2 library cast and then using base package's apply function we convert the raw data into user-movie matrix.

```
ratings <- read.csv("https://raw.githubusercontent.com/Vishal0229/DATA612_RecommenderSystem/master/Proj
kable(head(ratings,n=9))%>%
  kable_styling(bootstrap_options = c("striped","condensed","responsive"),full_width   = F,position = "
  row_spec(0, background ="gray")
```

user	movie	rating
Nick	Heat	3.0
Nick	Balto	2.0
Nick	Friday	NA
Nick	Clueless	3.5
Nick	Mortal Kombat	1.5
Nick	Atone	NA
Nick	Tangerine	3.0
Mike	Heat	4.0
Mike	Balto	NA

```
#converting the ratings data frame into user-movie matrix
```

```
ratingDT <- acast(ratings, user~movie, value.var="rating")
ratingDT <- apply(ratingDT, 2,as.numeric)
kable(ratingDT) %>%
  kable_styling(bootstrap_options = c("striped","condensed","responsive"),full_width   = F,position = "
  row_spec(0, background ="gray")
```

Atone	Balto	Clueless	Friday	Heat	Heat	Mortal Kombat	Tangerine
0.0	3.5	3.5	3.0	NA	4.0	1.5	NA
3.5	NA	3.5	3.5	NA	4.5	2.5	3.5
NA	4.0	3.5	NA	NA	5.0	3.0	3.5
NA	4.0	3.5	NA	NA	5.0	3.0	3.5
1.0	NA	3.5	4.0	NA	4.0	3.0	4.5
NA	2.0	3.5	NA	NA	3.0	1.5	3.0
2.0	NA	3.5	3.0	5	NA	1.5	NA

Training & Test data set.

Next step we divided iur data set into training(75%) & test dataset(25%) . Then we deduced the average mean on training dataset and also caculated the RMSE on this sparse data set.

```
## 75% of the sample size
smp_size <- floor(0.75 * nrow(ratingDT))

## set the seed to make your partition reproducible
set.seed(123)
train_ind <- sample(seq_len(nrow(ratingDT)), size = smp_size)

train <- ratingDT[train_ind, ]
test <- ratingDT[-train_ind, ]

kable(train) %>%
  kable_styling(bootstrap_options = c("striped","condensed","responsive"),full_width   = F,position = "
  row_spec(0, background ="gray")
```

Atone	Balto	Clueless	Friday	Heat	Heat	Mortal Kombat	Tangerine
NA	4	3.5	NA	NA	5	3.0	3.5
1	NA	3.5	4	NA	4	3.0	4.5
2	NA	3.5	3	5	NA	1.5	NA
NA	4	3.5	NA	NA	5	3.0	3.5
NA	2	3.5	NA	NA	3	1.5	3.0

```
kable(test) %>%
  kable_styling(bootstrap_options = c("striped","condensed","responsive"),full_width   = F,position = "
  row_spec(0, background ="gray")
```

Atone	Balto	Clueless	Friday	Heat	Heat	Mortal Kombat	Tangerine
0.0	3.5	3.5	3.0	NA	4.0	1.5	NA
3.5	NA	3.5	3.5	NA	4.5	2.5	3.5

Raw Average & RMSE on Training data set

Next step we calcualted the mean average of train data set and then we calculated the RMSE on train data set for the raw average.

```r
#deducing the raw mean average on train data set.
train.avg <- mean(train, na.rm =TRUE)
train.avg
```

```
## [1] 3.307692
```

```r
#function to calculate RMSE
RMSE = function(data, data.avg){
  sqrt(mean((data - data.avg)^2, na.rm =TRUE))
}

trainRMSE_beforeBias <-  RMSE(train,train.avg)
trainRMSE_beforeBias
```

```
## [1] 1.038462
```

Calculating Bias & Baseline predictors

Next step we created function calcBias which takes matrix and the raw avergae of that matrix , and caculates the row wise bias and column wise bias . Then creates a matrix adding each row value with each column value and the raw average(mean) of that matrix and assinging the new values to this newly reated matrix.

In short Baseline Predictor = Raw Average(mean) + User Bias + Movie Bias.

Then we calculate the RMSE for the baseline predictor train set matrix.

```r
#Calculating the baseline predictor (raw average + userBias + movieBias)
calcBias <- function(dataMatrix, dataAvg){

  userBias <- rowMeans(dataMatrix, na.rm=T) - dataAvg
  movieBias <- colMeans(dataMatrix, na.rm=T) - dataAvg

  outMatrix <- dataMatrix
  rowcount <-1
  for(item in 1:nrow(dataMatrix))
  {
    colcount <-1
    for(colItem in 1: ncol(dataMatrix))
    {

     outMatrix[rowcount,colcount] <- dataAvg + userBias[[rowcount]] +  movieBias[[colcount]]
      colcount <- colcount +1
    }
    rowcount <- rowcount +1
  }
  return (outMatrix)
}

baselineTrain <- calcBias(train,train.avg)
kable(baselineTrain)%>%
  kable_styling(bootstrap_options = c("striped","condensed","responsive"),full_width   = F,position = "
  row_spec(0, background ="gray")
```

Atone	Balto	Clueless	Friday	Heat	Heat	Mortal Kombat	Tangerine
1.9923077	3.825641	3.992308	3.992308	5.492308	4.742308	2.892308	4.117308
1.5256410	3.358974	3.525641	3.525641	5.025641	4.275641	2.425641	3.650641
1.1923077	3.025641	3.192308	3.192308	4.692308	3.942308	2.092308	3.317308
1.9923077	3.825641	3.992308	3.992308	5.492308	4.742308	2.892308	4.117308
0.7923077	2.625641	2.792308	2.792308	4.292308	3.542308	1.692308	2.917308

```
# clipping the values between 1 and 5, as our movie ratings cannot be below 1 and above 5.
baselineTrain[baselineTrain<1] <- 1
baselineTrain[baselineTrain>5] <- 5

kable(baselineTrain)%>%
  kable_styling(bootstrap_options = c("striped","condensed","responsive"),full_width   = F,position = "
  row_spec(0, background ="gray")
```

Atone	Balto	Clueless	Friday	Heat	Heat	Mortal Kombat	Tangerine
1.992308	3.825641	3.992308	3.992308	5.000000	4.742308	2.892308	4.117308
1.525641	3.358974	3.525641	3.525641	5.000000	4.275641	2.425641	3.650641
1.192308	3.025641	3.192308	3.192308	4.692308	3.942308	2.092308	3.317308
1.992308	3.825641	3.992308	3.992308	5.000000	4.742308	2.892308	4.117308
1.000000	2.625641	2.792308	2.792308	4.292308	3.542308	1.692308	2.917308

```
# Training dataset RMSE
trainRMSE_AfterBias <-RMSE(baselineTrain,train.avg)
trainRMSE_AfterBias
```

```
## [1] 1.046759
```

Calculations for Test data set.

```
# Raw mean of the test dataset
test.avg <- mean(test, na.rm =TRUE)
test.avg
```

```
## [1] 3.041667
```

```
testRMSE_beforeBias <-  RMSE(test,test.avg)
testRMSE_beforeBias
```

```
## [1] 1.16294
```

```
# Test dataset RMSE
baselineTest <- calcBias(test, test.avg)
#clipping the values to keep ratings between 1 & 5.
baselineTest[baselineTest<1] <- 1
baselineTest[baselineTest>5] <- 5

# Training dataset RMSE
testRMSE_afterBias <- RMSE(baselineTest,test.avg)
testRMSE_afterBias
```

```
## [1] 0.9530132
```

Comparing the Results

%Improvement for Train data set

In train data set , improvment dipped(-ve) slightly which can be ignored as it is very minimal hence we can say that for train data set there is no improvment in RMSE through RAw average and through Basline predictor method. This might be due to n number of reason one of which can be very small set of dataset.

```
t1 <- trainRMSE_beforeBias
tb1 <- trainRMSE_AfterBias

trainImprove_perc <- (1-(tb1/t1))*100
trainImprove_perc
```

```
## [1] -0.7989768
```

%Improvement for Test data set

We can clearly see that on our test data set , there has been 18% improvment after using Baseline predictor RMSE.

```
t1 <- testRMSE_beforeBias
tb1 <- testRMSE_afterBias

testImprove_perc <- (1-(tb1/t1))*100
testImprove_perc
```

```
## [1] 18.05142
```