GTDM Second Assignment

Group 2

November 8, 2021

Bertoletti, Luca
Brueckner, Xaver Matthias Heinrich
Ly Nguyen, Thi Phuong
Funicello, Alfredo
Degl'innocenti, Giorgio
Luraschi, Matias Santiago

1 Instructions

A survey is submitted to 15 customers of an animal shop asking them if 5 different aspects of the shop should be improved, namely:

- M1=variety of dogs food
- M2=variety and quality of cat litters
- M3=variety of dog beds
- M4=variety of aquariums
- M5=variety of turtle tanks

The collected data are stored in the file data2.csv. Higher grades correspond to an advice of bigger improvement, while low grades mean that the customer is satisfied of the present situation. Can you interpret the results in terms of "concepts" behind the evaluation? And can you group and classify the customers with respect to their attention to such concepts?

	M1	**M2**	**M3**	**M4**	**M5**
1	0	0	1	10	12
2	0	0	0	7	9
3	0	0	0	5	10
4	0	0	0	5	13
5	1	0	0	5	11
6	1	1	0	12	13
7	13	12	10	1	0
8	12	5	9	0	1
9	0	0	1	12	8
10	0	0	0	5	8
11	0	2	1	10	5
12	13	6	9	1	1
13	1	0	0	5	15
14	18	10	5	1	1
15	5	9	10	2	0

2 Initial Estimates

We can assign two logical categories to the five survey fields. The first three fields M1, M2 and M3 are clearly related to **terrestrial animals** while M4 and M5 could be grouped as related to **acquatic animals**.

Higher ratings on the survey correspond to advice of bigger improvement - we can then understand it as a level of dissatisfaction - for example people who gave high ratings in category 1 most likely own either dogs or cats (or both) and would rather see an improvement in the corresponding aspects of the shop and do not care as much about the assortment of items related to aquatic animals surveyed by questions M4 and M5 and will therefore give a low score to them. Satisfaction can in this case be interpreted as indifference. The same holds true the other way around as well.

	M1	**M2**	**M3**	**M4**	**M5**
1	0	0	1	10	12
2	0	0	0	7	9
3	0	0	0	5	10
4	0	0	0	5	13
5	1	0	0	5	11
6	1	1	0	12	13
7	13	12	10	1	0
8	12	5	9	0	1
9	0	0	1	12	8
10	0	0	0	5	8
11	0	2	1	10	5
12	13	6	9	1	1
13	1	0	0	5	15
14	18	10	5	1	1
15	5	9	10	2	0

Taking a look at the starting matrix we can estimate that out of the 15 questioned customers, 5 look like they are more interested in improvements on the **terrestrial animal items** section of the store while the other 10 seem more interested about improvements of the **aquatic animal items** section.

We can verify our estimates by computing the SVD of the matrix.

3 Procedure for the SVD

The *Singular Value Decomposition* of a matrix A is the factorization of A into the product of three matrices.

$$A = UDV^T$$

Where the columns of U and V are orthogonal and D is a diagonal matrix with positive real entries.

The *SVD* is used to find a lower rank matrix which is a good approximation of A and thus reduce the number of variables needed to describe the variability of our data and therefore represent the data in a lower dimensional space.

```r
library(Matrix)

dat <- read.csv('data2.csv')
M <- as.matrix(dat)
rankMatrix(M)[1]
svd_result <- svd(M,nu=5,nv=5)
```

Listing 1: R Code for the computation of the SVD

Through the code shown in the Listing 1 we load up the data and check that the starting rank of the matrix is 5. Using the *svd* function in R we are able to compute the U matrix, the D vector of singular values and the V matrix (Figure 1).

$$D = \begin{bmatrix} 42.5894589547063 & 37.7780174983655 & 10.9861293089865 & 7.87335186814474 & 5.02739226688978 \end{bmatrix}$$

$$U = \begin{bmatrix} -0.33 & -0.18 & -0.10 & -0.01 & -0.13 \\ -0.24 & -0.14 & -0.03 & 0.03 & -0.01 \\ -0.23 & -0.14 & 0.14 & -0.11 & 0.05 \\ -0.28 & -0.17 & 0.28 & -0.22 & 0.09 \\ -0.25 & -0.13 & 0.22 & -0.07 & 0.02 \\ -0.38 & -0.19 & -0.15 & 0.19 & 0.07 \\ -0.24 & 0.46 & -0.12 & -0.18 & 0.37 \\ -0.19 & 0.35 & 0.12 & -0.08 & -0.59 \\ -0.29 & -0.15 & -0.42 & 0.24 & -0.23 \\ -0.19 & -0.12 & 0.05 & -0.03 & 0.02 \\ -0.22 & -0.09 & -0.47 & 0.20 & 0.09 \\ -0.21 & 0.37 & 0.07 & 0.01 & -0.49 \\ -0.32 & -0.17 & 0.41 & -0.22 & 0.08 \\ -0.25 & 0.46 & 0.26 & 0.59 & 0.36 \\ -0.17 & 0.29 & -0.39 & -0.61 & 0.19 \end{bmatrix}$$

$$V = \begin{bmatrix} -0.34 & 0.64 & 0.37 & 0.54 & -0.21 \\ -0.23 & 0.43 & -0.22 & -0.19 & 0.82 \\ -0.23 & 0.42 & -0.28 & -0.65 & -0.51 \\ -0.53 & -0.25 & -0.69 & 0.40 & -0.11 \\ -0.71 & -0.40 & 0.51 & -0.29 & 0.08 \end{bmatrix}$$

Figure 1: Outputs of the svd computation

To reduce the dimensionality of the starting matrix and find its best rank-k approximation we look at the values of the D vector.

$$42,589460 \; 37,778020 \; 10,986130 \; 7,873352 \; 5,027392$$

Following the Feobenius norm approach we calculate the total energy of the matrix:

$$42,5894602^2 + 37,7780202^2 + 10,9861302^2 + 7,8733522^2 + 5,0273922^2 = 3.449,00$$

We have to eliminate iteratively the smallest singular value, keeping in mind that we should retain at least 90% of the energy.

$Val1^2$	$Val2^2$	$Val2^2$	$Val3^2$	$Val4^2$	Sum of squares	% of total energy
1.813, 86	1.427, 18	120, 70	61, 99	25, 27	3.449, 00	100%
1.813, 86	1.427, 18	120, 70	61, 99		3.423, 73	99%
1.813, 86	1.427, 18				3.241, 04	94%
1.813, 86					1.813, 86	53%

Table 1: Different levels of energy while retaining a different number of singular values

We can observe in the Table 1 that the two biggest singular values have to be kept in order to retain about 94% of the total energy and therefore staying above the commonly used 90% threshold. Accordingly, using the code in Listing 2, we slice the matrixes and the vector keeping just 2 columns for the U matrix, 2 rows for the Vt matrix and 2 singular values. We then compute the D diagonal matrix, the result is shown in Figure 2.

```r
U<-svd_result$u[,1:2]
Vt<-t(svd_result$v[,1:2])
D<-diag(svd_result$d[1:2],nrow=2,ncol=2)
```

Listing 2: R code for slicing up U and Vt and creating D

$$U = \begin{bmatrix} -0.33 & -0.18 \\ -0.24 & -0.14 \\ -0.23 & -0.14 \\ -0.28 & -0.17 \\ -0.25 & -0.13 \\ -0.38 & -0.19 \\ -0.24 & 0.46 \\ -0.19 & 0.35 \\ -0.29 & -0.15 \\ -0.19 & -0.12 \\ -0.22 & -0.09 \\ -0.21 & 0.37 \\ -0.32 & -0.17 \\ -0.25 & 0.46 \\ -0.17 & 0.29 \end{bmatrix} \qquad D = \begin{bmatrix} 42.58946 & 0.00000 \\ 0.00000 & 37.77802 \end{bmatrix} \quad Vt = \begin{bmatrix} -0.3395495 & -0.2330685 & -0.2291698 & -0.529335 & -0.7054579 \\ 0.6425794 & 0.4328585 & 0.4202842 & -0.2542393 & -0.3980562 \end{bmatrix}$$

Figure 2: U, D and Vt matrixes after slicing up

We then compute the dot-product of the three matrices through the code shown in Listing 3, the result will be the best rank 2 approximation of the original matrix M, shown in Figure 3.

```
Mbest <- U%*%D%*%Vt
rankMatrix(Mbest)[1]
```

Listing 3: Mbest matrix is created by the product of the U, D and Vt matrices, we then check that it has rank 2.

$$Mbest = \begin{bmatrix} 0.32 & 0.27 & 0.31 & 9.16 & 12.61 \\ -0.03 & 0.02 & 0.05 & 6.69 & 9.23 \\ -0.08 & -0.01 & 0.02 & 6.47 & 8.93 \\ -0.13 & -0.04 & -0.00 & 7.89 & 10.90 \\ 0.43 & 0.34 & 0.36 & 6.96 & 9.57 \\ 0.87 & 0.66 & 0.68 & 10.34 & 14.20 \\ 14.65 & 9.91 & 9.65 & 0.86 & 0.11 \\ 11.24 & 7.61 & 7.41 & 0.87 & 0.37 \\ 0.41 & 0.33 & 0.36 & 7.95 & 10.94 \\ -0.05 & 0.00 & 0.03 & 5.52 & 7.62 \\ 1.14 & 0.81 & 0.82 & 5.86 & 8.01 \\ 12.14 & 8.22 & 8.01 & 1.24 & 0.82 \\ 0.37 & 0.31 & 0.34 & 8.86 & 12.20 \\ 14.82 & 10.03 & 9.77 & 1.32 & 0.73 \\ 9.37 & 6.34 & 6.18 & 1.04 & 0.74 \end{bmatrix}$$

Figure 3: Mbest matrix obtained by the dot product

4 Interpretation of the obtained matrices

The matrix D is related to the strength of the concepts. Since we had to keep only two values of the initial D vector we now know we are working with 2 concepts.

$$D = \begin{bmatrix} 42.58946 & 0.00000 \\ 0.00000 & 37.77802 \end{bmatrix}$$

We can observe that one of the values is bigger than the other one, which means that people are more vocal about one of the two concepts.

The problem is that on a first sight we cannot tell which concept is related to which value of the D matrix.

We solve it by taking a look at the Vt matrix which relates our categories to concepts.

There is a relation between the first element of the main diagonal of the matrix D and the first row of the matrix Vt; and the second element of the main diagonal of the matrix D with the second row of the matrix Vt.

$$Vt = \begin{bmatrix} -0.3395495 & -0.2330685 & -0.2291698 & -0.529335 & -0.7054579 \\ 0.6425794 & 0.4328585 & 0.4202842 & -0.2542393 & -0.3980562 \end{bmatrix}$$

By looking at the rows of the Vt matrix we can see that the SVD has associated the M4 and M5 survey labels to one concept while the M1, M2 and M3 to another one.

Since we now know that the highest value on the diagonal of the D matrix refers to the M4 and M5 concept, which we initially denominated as the **acquatic animals**, we can deduce that more people are outspoken about the need of improvements in that section of the store.

We could further show that by mapping the answers of the users of the survey to the concept space V, through the code shown in the Listing 4, resulting in the matrix *Scores* shown in Figure 4.

```
Scores<-M%*%V
abs(Scores)
```

Listing 4: Mapping users to V matrix

$$Scores = \begin{bmatrix} 13.99 & 6.90 \\ 10.05 & 5.36 \\ 9.70 & 5.25 \\ 11.82 & 6.45 \\ 10.75 & 5.01 \\ 16.10 & 7.15 \\ 10.03 & 17.50 \\ 8.01 & 13.26 \\ 12.22 & 5.82 \\ 8.29 & 4.46 \\ 9.52 & 3.25 \\ 9.11 & 14.08 \\ 13.57 & 6.60 \\ 10.82 & 17.34 \\ 7.15 & 10.80 \end{bmatrix}$$

Figure 4: Matrix obtained by the dot product of the matrix M and the matrix V

These results represent the interest of each user towards the need of improvement in each of the two concepts.

Through the use of the *ggplot2* library we create a visualization of the *Scores* matrix, shown in Figure 5, to have a better understanding of the data. The code is shown in Listing 5.

```r
library(ggplot2)

x <- LETTERS[1:15]
y <- paste0("concept", seq(1,2))
data <- expand.grid(X=x, Y=y)
ve<-as.vector(Scores_users)
data$Z <- abs(ve)

ggplot(data, aes(Y, X, fill= Z)) + geom_tile() + scale_fill_gradient(low="green", high="red"
    )
```

Listing 5: R code for the Gradient Visualization of the *Scores* matrix

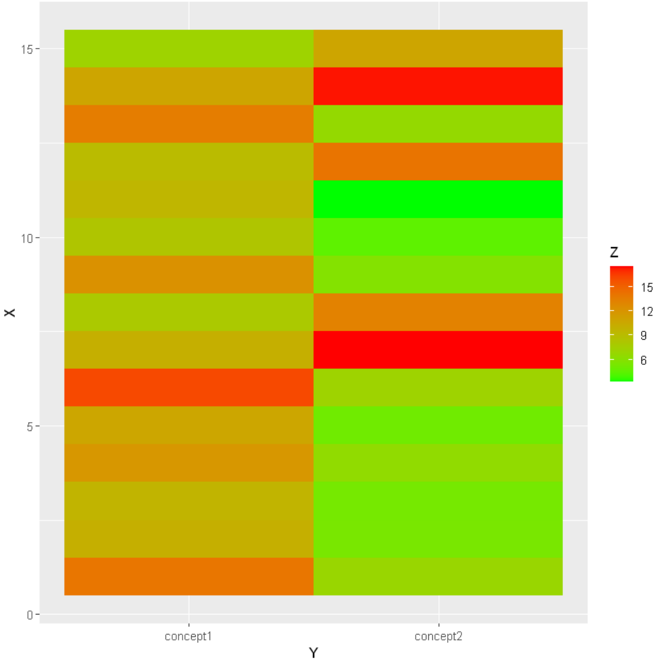

Figure 5: Gradient Visualization of the *Scores* matrix

Through the visualization in Figure 5, where green-like boxes represent lower values and red-like higher ones, we can clearly see that the concept1, being **acquatic animals items** has on average more entries with higher scores, while the concept2, **terrestrial animals items**, has mostly lower scores. Hence we can deduce that more customers are dissatisfied with the store performance regarding **acquatic animals items**.

However it is important to keep in mind that the responses given by the people are not ”pure” as in one of the first theoretical examples given in class. Meaning that even though each group of users seems more dissatisfied with a particular concept, they still believe that there is room for improvement in the other category.

We can compute the similarity matrix between all of the 15 customers, given by the values of the cosine of θ, which is the angle between vectors in the space generated by the columns of the concept matrix V. We use the code shown in Listing 7.

```r
similarity<-c()

for (j in c(1:15)){
  q<-M[j , ]
  scoreq<-q%*%svd_result$v[,1:2]
  for (i in c(1:15)){
    similarity[i]<-dot(scoreq,Scores_users[i,])/(sqrt(sum(scoreq^2)*sum((Scores_users[i,])
    ^2)))
    if (i==15){
      F<-rbind(F,similarity)}
  }}
F<-F[-1,]
rownames(F) <- NULL
```

Listing 6: Computing the similarity matrix between users

The similarity of a customer with themselves is 1 of course, since the angle between identical vectors is 0 and $cos(0) = 1$.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1.00	1.00	1.00	1.00	1.00	1.00	0.06	0.09	1.00	1.00	0.99	0.12	1.00	0.10	0.13
2	1.00	1.00	1.00	1.00	1.00	1.00	0.03	0.05	1.00	1.00	0.99	0.08	1.00	0.07	0.09
3	1.00	1.00	1.00	1.00	1.00	1.00	0.02	0.05	1.00	1.00	0.99	0.08	1.00	0.06	0.09
4	1.00	1.00	1.00	1.00	1.00	1.00	0.02	0.04	1.00	1.00	0.99	0.07	1.00	0.06	0.08
5	1.00	1.00	1.00	1.00	1.00	1.00	0.08	0.11	1.00	1.00	0.99	0.14	1.00	0.12	0.15
6	1.00	1.00	1.00	1.00	1.00	1.00	0.10	0.12	1.00	1.00	1.00	0.16	1.00	0.14	0.17
7	0.06	0.03	0.02	0.02	0.08	0.10	1.00	1.00	0.08	0.03	0.19	1.00	0.07	1.00	1.00
8	0.09	0.05	0.05	0.04	0.11	0.12	1.00	1.00	0.10	0.05	0.21	1.00	0.09	1.00	1.00
9	1.00	1.00	1.00	1.00	1.00	1.00	0.08	0.10	1.00	1.00	0.99	0.13	1.00	0.11	0.14
10	1.00	1.00	1.00	1.00	1.00	1.00	0.03	0.05	1.00	1.00	0.99	0.08	1.00	0.06	0.09
11	0.99	0.99	0.99	0.99	0.99	1.00	0.19	0.21	0.99	0.99	1.00	0.24	0.99	0.23	0.25
12	0.12	0.08	0.08	0.07	0.14	0.16	1.00	1.00	0.13	0.08	0.24	1.00	0.12	1.00	1.00
13	1.00	1.00	1.00	1.00	1.00	1.00	0.07	0.09	1.00	1.00	0.99	0.12	1.00	0.11	0.13
14	0.10	0.07	0.06	0.06	0.12	0.14	1.00	1.00	0.11	0.06	0.23	1.00	0.11	1.00	1.00
15	0.13	0.09	0.09	0.08	0.15	0.17	1.00	1.00	0.14	0.09	0.25	1.00	0.13	1.00	1.00

Values close to 1 indicate strong similarity in behaviour between customers, whereas values close to 0 indicate little to no similarity. To make the table more readable and interpretable we can use the heatmap() function, which also rearranges and clusters the customers (Figure 6). In this case sky-blue corresponds to a similarity value closer to 1 and brown to a value closer to 0. This matrix now gives us a clear separation between the customers 12, 15, 7, 8, 14, which have given higher scores to the **terrestrial animals items**, represented by concept2 in Figure 5 and the rest of the customers, which were less satisfied with the store's assortment of **aquatic animal items**.

```
new_colors <- colorRampPalette(c("brown", "skyblue"))
heatmap(F,col = new_colors(100),main = "Similarity_Matrix",xlab = "Customers", ylab="
    Customers")
```

Listing 7: Use of the heatmap function

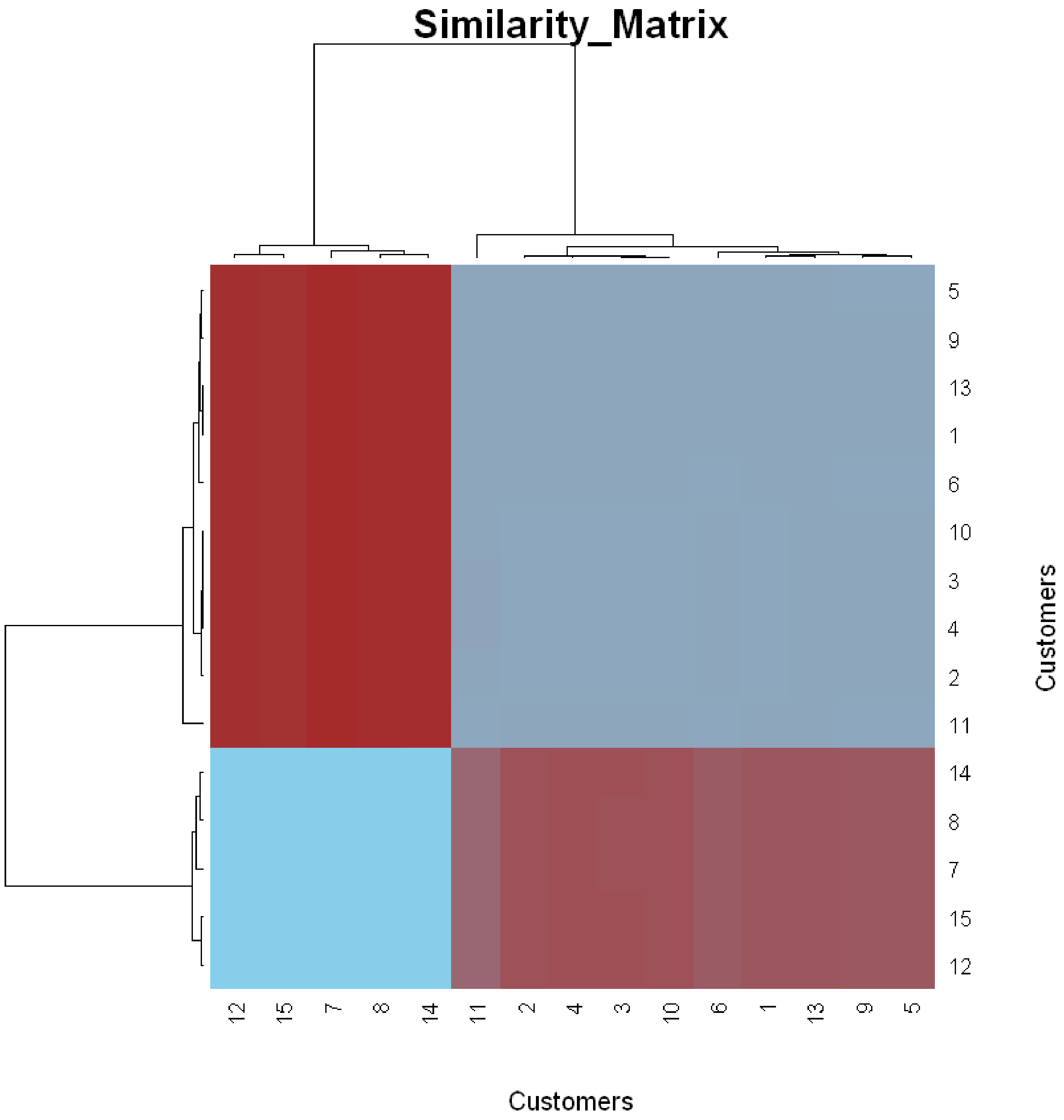

Figure 6: Heatmap visualization of the similarity matrix between customers

5 Integral Codebase

```
library(Matrix)

dat<-read.csv(’data2.csv’)

M<-as.matrix(dat)

rankMatrix(M)[1]

#compute SVD
svd_result<-svd(M,nu=5,nv=5)

U<-svd_result$u[,1:2]
Vt<-t(svd_result$v[,1:2])
V<-svd_result$v[,1:2]
D<-diag(svd_result$d[1:2],nrow=2,ncol=2)

Mbest <- U%*%D%*%Vt
rankMatrix(Mbest)[1]

Scores_users<-M%*%V

library(ggplot2)

x <- seq(1,15)
y <- paste0("concept", seq(1,2))
data <- expand.grid(X=x, Y=y)
ve<-as.vector(Scores_users)
data$Z <- abs(ve)

data

ggplot(data, aes(Y, X, fill= Z)) + geom_tile() + scale_fill_gradient(low="green", high="red")

similarity<-c()

library(pracma)
for (j in c(1:15)){
  q<-M[j , ]
  scoreq<-q%*%svd_result$v[,1:2]
  for (i in c(1:15)){
    similarity[i]<-dot(scoreq,Scores_users[i,])/(sqrt(sum(scoreq^2)*sum((Scores_users[i,])^2)))
    if (i==15){
      F<-rbind(F,similarity)}
  }}
F<-F[-1,]
rownames(F) <- NULL

new_colors <- colorRampPalette(c("brown", "blue"))
heatmap(F,col = new_colors(100),main = "Similarity_Matrix",xlab = "Customers", ylab="Customers")
```