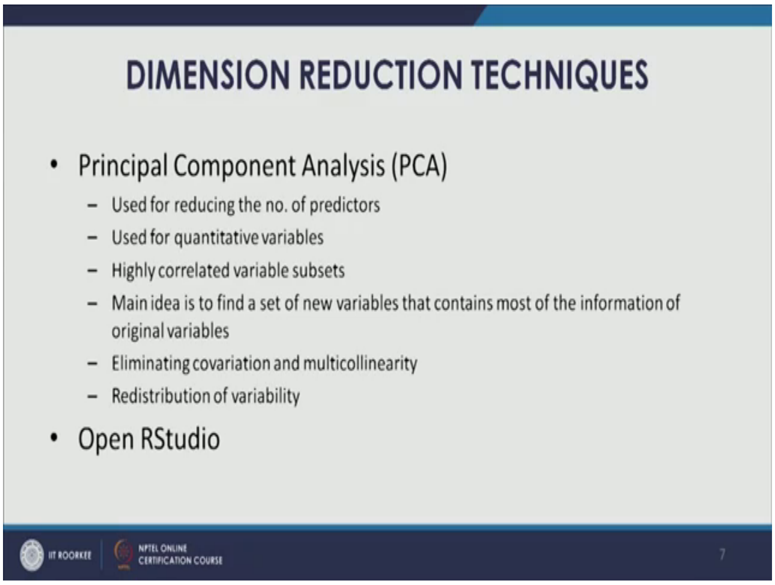
Business Analytics & Data Mining Modeling Using R
Dr. Gaurav Dixit
Department of Management Studies
Indian Institute of Technology, Roorkee

Lecture – 14
Dimension Reduction Techniques- Part II Principal Component Analysis

Welcome to the course business analytics and data mining modelling using R. So, in the previous lecture we were discussing dimension reduction techniques and we covered quite a few. So, today's lecture will start with the automated reduction techniques and the one particular technique that we are going to talk about, that we are going to discuss is principal component analysis also called PCA. So, let us start our discussion on PCA.

(Refer Slide Time: 00:51)



DIMENSION REDUCTION TECHNIQUES

- Principal Component Analysis (PCA)
  - Used for reducing the no. of predictors
  - Used for quantitative variables
  - Highly correlated variable subsets
  - Main idea is to find a set of new variables that contains most of the information of original variables
  - Eliminating covariation and multicollinearity
  - Redistribution of variability
- Open RStudio

So, a principal component analysis is mainly used for reducing the number of predictors, number of predictors as we are discussing the dimension reduction techniques and as we talked, as we discussed in the previous lectures as well, that the idea being reducing the dimension and mainly to achieve the parsimony, to follow the principal of parsimony and many other regions that we have discussed before.

So, the PCA, role of PCA is also similar can we used for reducing the number predictors hence can be used to reduce the dimensions used for quantitative variables. So, only the quantitative variables can be used under this techniques. So, for categorical variables we have to rely on other methods that we have discussed in previous lecture as well.

Now, sometimes when we are dealing with a large number of variables or big pool of predictors, we might encounter highly correlated variable subsets. So, this kind of situation is not desirable in many situation because some of the, some of these variables some of these variables might have information overlap. So, they might be measuring the same kind of information, same information that can that can disturb the model spurious relationship could be there and the model might be useless.

So, therefore, we want to get rid off from these situations. So, principal component analysis PCA can specifically help in this particular kind of scenario, now what is the main idea main idea is to find a set of new variables that contains most of the information of original variables. So, we do not want to lose out on the information because we want to have, we want to retain the explanatory part explanatory power of the model which we could have had using original variables.

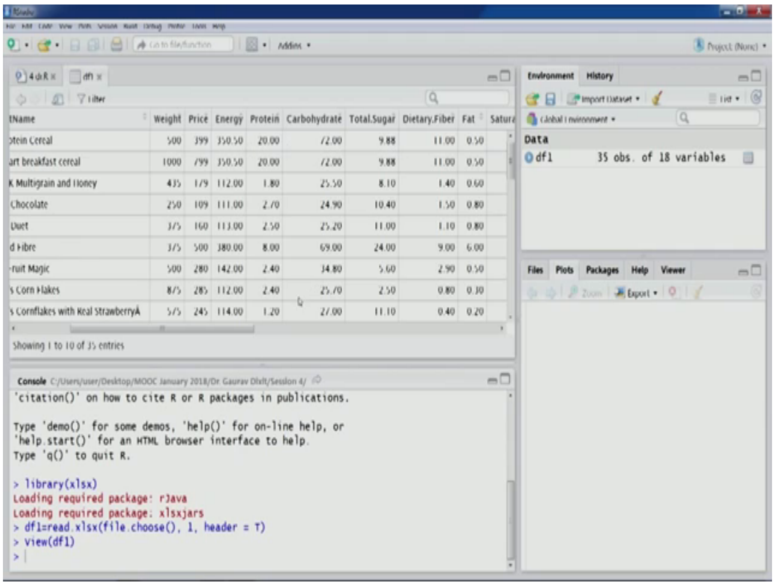
So, with new set of variables also we would like to retain most of the information and hence most of the explanatory power from the model of the model. While idea is to find new set of variables the point to be noted here is we are trying to reduce the dimensions therefore, these set of new variables are going to be less than the number of variables number of original variables, few other objectives could be eliminating co variation and multicollinearity. So, in any information overlap it been 2 variables, it could also be called co variation, now sometimes in regression modelling especially we might encounter this problem multicollinearity.

So, in relation models as we will discuss in coming lectures that this co variation is not desirable and therefore, that leads to multicollinearity that will discuss in the regression related lectures, but through PCA this can also be eliminated. So, eliminating co variation and multicollinearity could be another objective.

Now, essentially while we are looking for a new set of variables which is able to which are few in which are less than number of original variables thereby reducing the dimensions, while we are and also at the same time we are trying to retain the explanatory power of the all the variables put together therefore, the model we need to redistribute the variable this is the, how the when we are looking for a new set of variables we are essentially redistributing the variability that is contained by the original set of variables right.

So, let us go through an exercise using R. So, let us open our studio.

(Refer Slide Time: 05:04)



tName	Weight	Price	Energy	Protein	Carbohydrate	Total.Sugar	Dietary.Fiber	Fat	Satura
otein Cereal	500	399	350.50	20.00	72.00	9.88	11.00	0.50	
art breakfast cereal	1000	799	350.50	20.00	72.00	9.88	11.00	0.50	
K Multigrain and Honey	435	179	112.00	1.80	25.50	8.10	1.40	0.60	
Chocolate	250	109	111.00	2.70	24.90	10.40	1.50	0.80	
Duet	375	160	113.00	2.50	25.20	11.00	1.10	0.80	
d Fibre	375	500	380.00	8.00	69.00	24.00	9.00	6.00	
ruit Magic	500	280	142.00	2.40	34.80	5.60	2.90	0.50	
s Corn Flakes	875	285	112.00	2.40	25.70	2.50	0.80	0.30	
s Cornflakes with Real StrawberryÂ	575	245	114.00	1.20	27.00	11.10	0.40	0.20	

Showing 1 to 10 of 35 entries

Data

df1 35 obs. of 18 variables

Console C:/Users/user/Desktop/MOOC January 2018/Dr. Gaurav Dixit/Session 4/

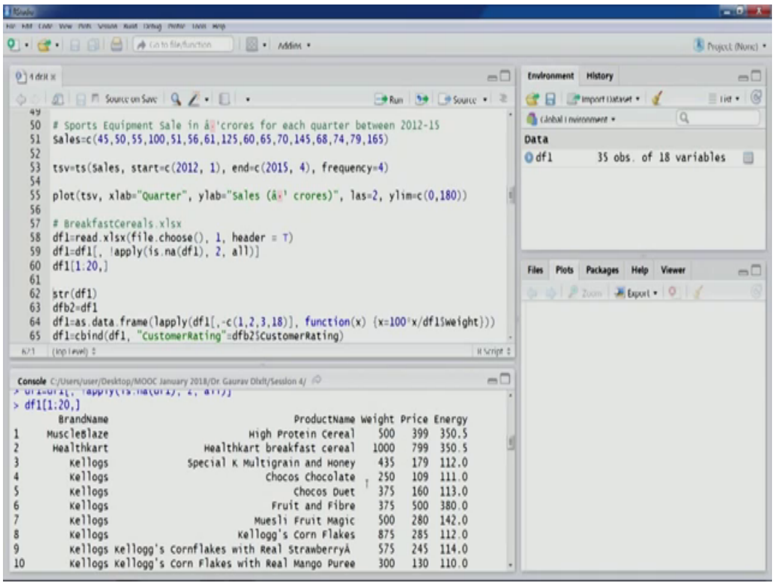
```
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(xlsx)
Loading required package: rJava
Loading required package: xlsxjars
> df1=read.xlsx(file.choose(), 1, header = T)
> View(df1)
> 
```

So, will go back, to go reach to the section where we start our discussion on principal component analysis yes. So, let us import this particular dataset, breakfast cereal data set. Let us import this, will also discuss this data set and how it could be how it this going to be utilized for our exercise. So, you can see 35 observation of 18, 18 variables let us open the this particular data set will try to open here in the our environment itself. You can see the first particular variable that we can see is brand name and you can see few brand names here and then the product name and then specific details about these products the kind of packaging.

So, this is way it is depending on the packaging and the price is the corresponding price and the energy and other contents or ingredients that are there in that particular cereals. So, all those details starting from protein, carbohydrate, sugar, fibre, fat so all those details you can see in this particular data set at the end of it last column after iron related information you would see that customer rating is also there. So, how customers have been rating these particular cereals. So, now, all this data is based on the different cereal packets that are being sold in Indian markets and. So, we have selected few of them and also taken the different details about these cereals.

So, we are going to use this particular data set for our principal component analysis. So, let us eliminate the, if there are any columns, columns having any values and then let us have a look at the rows of data.

(Refer Slide Time: 07:35)



```
# Sports Equipment Sale in â'crores for each quarter between 2012-15
Sales=c(45,50,55,100,51,56,61,125,60,65,70,145,68,74,79,165)

tsv=ts(Sales, start=c(2012, 1), end=c(2015, 4), frequency=4)

plot(tsv, xlab="Quarter", ylab="Sales (â' crores)", las=2, ylim=c(0,180))

# BreakfastCereals.xlsx
df1=read.xlsx(file.choose(), 1, header = T)
df1=df1[, !apply(is.na(df1), 2, all)]
df1[1:20,]

str(df1)
dfb2=df1
df1=as.data.frame(lapply(df1[,-c(1,2,3,18)], function(x) {x=100*x/df1$weight}))
df1=cbind(df1, "CustomerRating"=dfb2$CustomerRating)
```

Console C:/Users/user/Desktop/MOOC January 2018/Dr. Gaurav Dixit/Session 4/

```
> df1=df1[, !apply(is.na(df1), 2, all)]
> df1[1:20,]
     BrandName                              ProductName Weight Price Energy
1  MuscleBlaze                     High Protein Cereal    500   399  350.5
2   Healthkart              Healthkart breakfast cereal   1000   799  350.5
3      Kellogs         Special K Multigrain and Honey    435   179  112.0
4      Kellogs                        Chocos Chocolate    250   109  111.0
5      Kellogs                             Chocos Duet    375   160  113.0
6      Kellogs                         Fruit and Fibre    375   500  380.0
7      Kellogs                      Muesli Fruit Magic    500   280  142.0
8      Kellogs                     Kellogg's Corn Flakes    875   285  112.0
9      Kellogs Kellogg's Cornflakes with Real StrawberryÂ   575   245  114.0
10     Kellogs Kellogg's Corn Flakes with Real Mango Puree   300   130  110.0
```

Environment History

Data

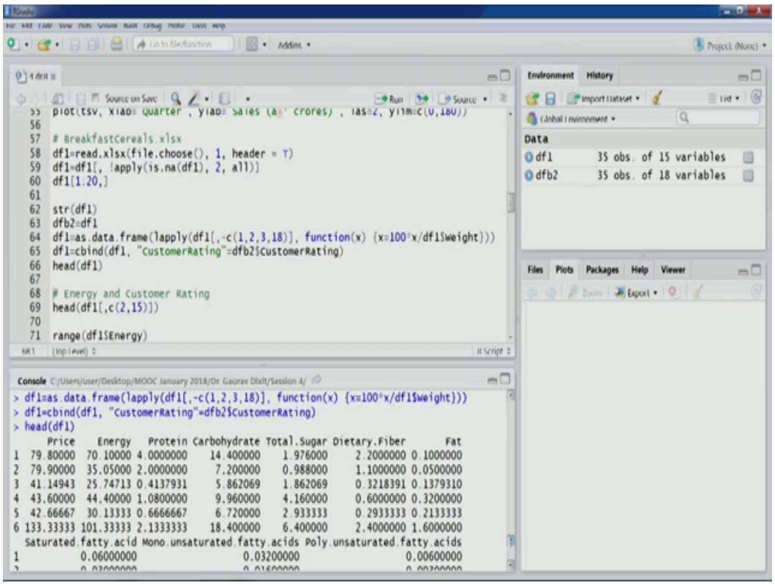
df1 35 obs. of 18 variables

So, the same thing that we did using opening this file in the our environment, same thing we can do through this particular command. You can see in this particular data frame that weight for different the package the details which have been taken from different packages are carrying different weights. So, therefore, it is important for us to have because we are going to compare these serials right later on through principal component analysis essentially. So, these serials are going to be compared. So, therefore, we need to we need to get the details like price energy and other details like protein, carbohydrate, sugar etcetera for the same weight for the same packaging for the same weight of that particular cereals.

So, let us look at the structure of this particular data frame you would see except the brand name and product name all the variables are numerical in nature. Let us take a backup of this full data set, now what we are going to do is we are going to apply this particular function this we have written function this so we are going to divide all the details starting from the price, energy protein with the weight.

So, that we get the details for of all the cereals for similar weight or similar packaging. So, 100, so it is for 100 grams. So, you can see we all these details are going to be now available per 100 grams. So, let us execute this. So, this particular lines will get a new data frame and now once this is done we had earlier eliminate, we have earlier not included customer rating in the earlier line. So, let us combine this one as well customer rating and let us look at the first 6 observations, now you would see that details specific numbers have changed.

(Refer Slide Time: 10:05)



```
plot(tsv, xlab="Quarter", ylab="Sales (₹' crores)", las=2, ylim=c(0,180))

# BreakfastCereals.xlsx
df1=read.xlsx(file.choose(), 1, header = T)
df1=df1[, !apply(is.na(df1), 2, all)]
df1[1:20,]

str(df1)
dfb2=df1
df1=as.data.frame(lapply(df1[,-c(1,2,3,18)], function(x) {x=100*x/df1$Weight}))
df1=cbind(df1, "CustomerRating"=dfb2$CustomerRating)
head(df1)

# Energy and Customer Rating
head(df1[,c(2,15)])

range(df1$Energy)
```

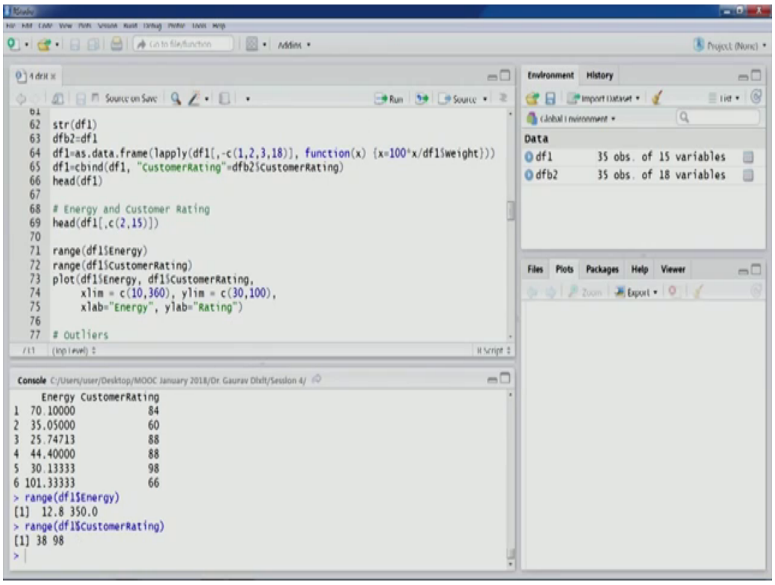
```
Data
df1        35 obs. of 15 variables
dfb2       35 obs. of 18 variables
```

```
> df1=as.data.frame(lapply(df1[,-c(1,2,3,18)], function(x) {x=100*x/df1$Weight}))
> df1=cbind(df1, "CustomerRating"=dfb2$CustomerRating)
> head(df1)
      Price    Energy   Protein Carbohydrate Total.Sugar Dietary.Fiber       Fat
1  79.80000  70.10000 4.0000000    14.400000    1.976000     2.2000000 0.1000000
2  79.90000  35.05000 2.0000000     7.200000    0.988000     1.1000000 0.0500000
3  41.14943  25.74713 0.4137931     5.862069    1.862069     0.3218391 0.1379310
4  43.60000  44.40000 1.0800000     9.960000    4.160000     0.6000000 0.3200000
5  42.66667  30.13333 0.6666667     6.720000    2.933333     0.2933333 0.2133333
6 133.33333 101.33333 2.1333333    18.400000    6.400000     2.4000000 1.6000000
  Saturated.fatty.acid Mono.unsaturated.fatty.acids Poly.unsaturated.fatty.acids
1           0.06000000                   0.03200000                   0.00600000
```

Now, all these numbers are for all these numbers are for each cereal and for 100 grams of each one.

So, now we can move ahead. So, let us select 2 important variables out of this dataset energy and customer rating and let us do our pca run, let us apply our PCA on these two variables and then will proceed further. So, let us focus on energy and customer rating. So, these are first 6 observation for energy and customer rating, now what we are going to do is we will plot a graph between energy and customer rating, let us look at the range you can see 12.82, 35 for energy.

(Refer Slide Time: 10:59)



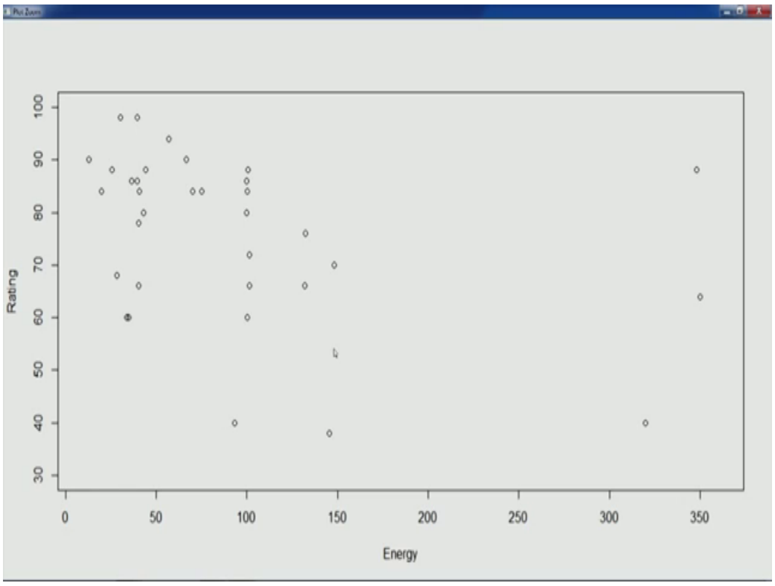
```
str(df1)
dfb2=df1
df1=as.data.frame(lapply(df1[,-c(1,2,3,18)], function(x) {x=100*x/df1$weight}))
df1=cbind(df1, "CustomerRating"=dfb2$CustomerRating)
head(df1)

# Energy and Customer Rating
head(df1[,c(2,15)])

range(df1$Energy)
range(df1$CustomerRating)
plot(df1$Energy, df1$CustomerRating,
     xlim = c(10,360), ylim = c(30,100),
     xlab="Energy", ylab="Rating")

# Outliers
```

```
   Energy CustomerRating
1  70.10000             84
2  35.05000             60
3  25.74713             88
4  44.40000             88
5  30.13333             98
6 101.33333             66
> range(df1$Energy)
[1]  12.8 350.0
> range(df1$CustomerRating)
[1] 38 98
>
```

```
Data
df1     35 obs. of 15 variables
dfb2    35 obs. of 18 variables
```

This is kilocalorie and then the customer rating is there this is percentage customer rating between 0 to 100 so appropriately the limits x and limit and y limits have been specified. So, let us plot this graph.

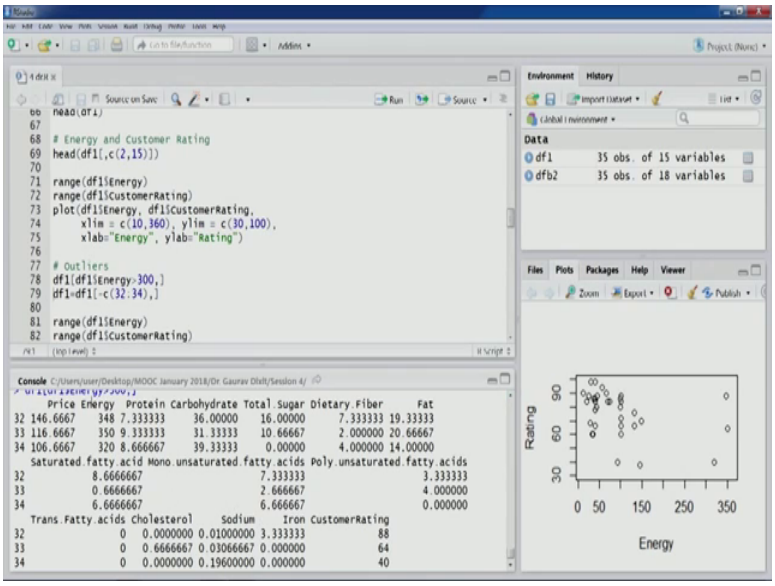
(Refer Slide Time: 11:18)



So, this is the scatter plot that we get, if we look at this scatter plot some of the observation you can see these observations these seem to be way out of the major chunk of the values. So, we will consider them as outliers for our exercise and will try and eliminate them so, that we would end up dealing with only this particular chunk of points.

So, let us find out these outliers. So, most of them looked like having energy value of greater than 300. So, let us identify these points as you can see point object case number, observation number 32, 33 and 34. So, these are the points and you can see the energy values are in excess of 300 kilo calorie.

(Refer Slide Time: 12:00)



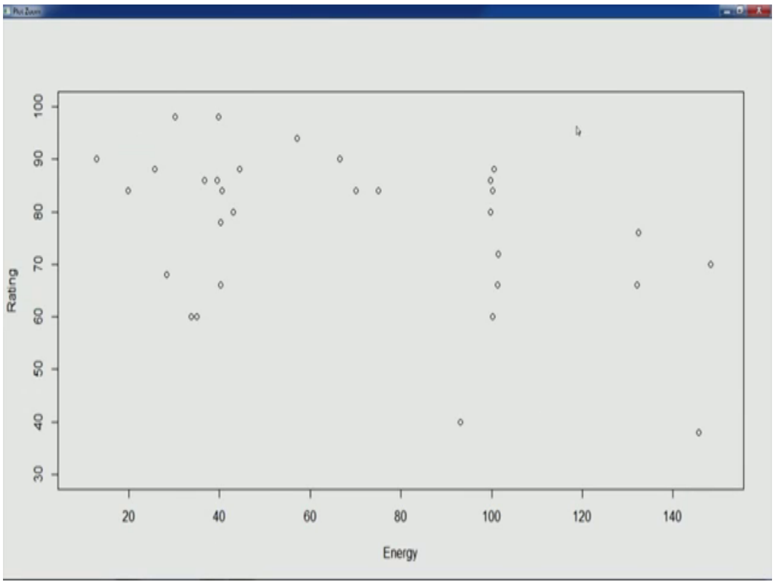
```
head(df1)

# Energy and Customer Rating
head(df1[,c(2,15)])

range(df1$Energy)
range(df1$CustomerRating)
plot(df1$Energy, df1$CustomerRating,
     xlim = c(10,360), ylim = c(30,100),
     xlab="Energy", ylab="Rating")

# Outliers
df1[df1$Energy>300,]
df1=df1[-c(32:34),]

range(df1$Energy)
range(df1$CustomerRating)
```

```
> df1[df1$Energy>300,]
      Price Energy  Protein Carbohydrate Total.Sugar Dietary.Fiber      Fat
32 146.6667    348 7.333333     36.00000    16.00000      7.333333 19.33333
33 116.6667    350 9.333333     31.33333    10.66667      2.000000 20.66667
34 106.6667    320 8.666667     39.33333     0.00000      4.000000 14.00000
   Saturated.fatty.acid Mono.unsaturated.fatty.acids Poly.unsaturated.fatty.acids
32            8.6666667                     7.333333                     3.333333
33            0.6666667                     2.666667                     4.000000
34            6.6666667                     6.666667                     0.000000
   Trans.Fatty.acids Cholesterol     Sodium     Iron CustomerRating
32                 0   0.0000000 0.01000000 3.333333             88
33                 0   0.6666667 0.03066667 0.000000             64
34                 0   0.0000000 0.19600000 0.000000             40
```

Data
df1   35 obs. of 15 variables
dfb2   35 obs. of 18 variables

So, probably these cereals are having high energy content, high energy value so will not include them. So, will only analyze the cereals having a closer by energy value range; let us eliminate these 3 points from our data frame and we are going to plot again. So, now, will we get a much closer plot, rating versus energy?

(Refer Slide Time: 12:30)



So, all these points we can now visualize. So, from this if you look at the points. So, most of the points if we try to draw a line which can go from here to here, its look like you know as the energy, as the energy increases for a particular cereal 4 different cereals as energy increases energy value increases you can see the rating that is slightly that is coming down which is expected.

So, variability in terms of variability also, we can see most of the variability can be captured by the rating and energy itself because they are quite aligned to the x axis and y axis. So, let us look at the mean values of these 2 variables energy and the customer rating. So, these are the mean values. So, mean values representing as we have talked about in previous lectures also the centralized value of a particular variables and gives us a sense of the you know mean value, average values from where other values also other values might be lying around that value. So, it gives such as central value representing value in a way representing value of that particular variable.

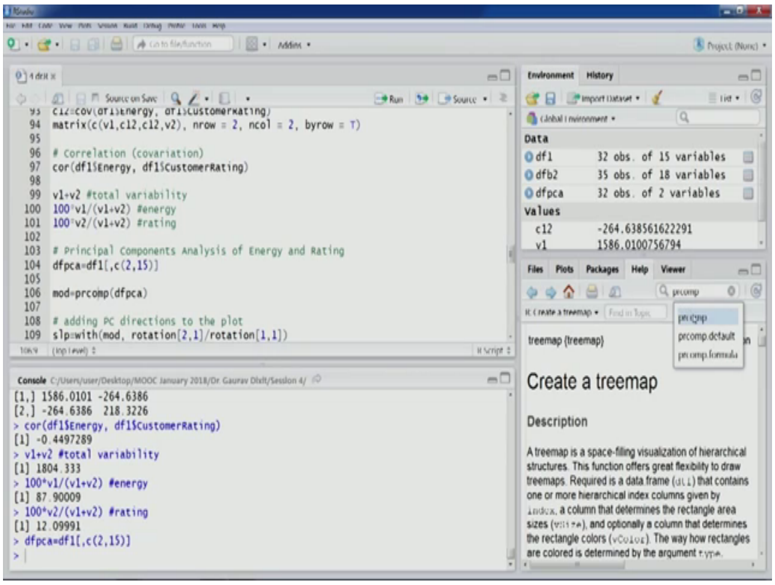
So, let us look at the covariance matrix of this these 2 you know variables. So, let us compute the variance of energy variable then followed by the variance of customer rating, let us also compute their covariance. So, if this is the matrix why we computing all this information is because essentially as we talked about the if these are the original 2 variables energy and customer rating and we are going to apply a principal component, component analysis on these 2 variables essentially we would be redistributing the variability because we want to retain the much of the information right while we are trying to find a set of variables which is fewer than the original number of variables.

So, we are trying to have a look at the variability, if you are interested in finding the correlation between these two. So, you can see these two variables the correlation coefficient seems to be minus 0.45 right. So, now, again let us come back to the variability. So, let us see the total variability within the original variables energy and customer rating that is 1804.333, let us see the contribution of energy the variability that is contributed by energy variable is 87 percent and the variability coming from rating is 12 percent.

So, you would see if we go back to the plot, if you go back to the plot then you would see that it is the along the x axis that is being represented by energy, most of the variability is being captured, You can see from values starting from 22 value starting from 140 and most of the variability can be captured along this dimension. Now, some variability is also in the perpendicular orthogonal direction which is being represented by y axis and rating. So, some variability is also being captured by contributed by rating.

So, now as if the data of energy and rating as it looks like the most of the variability contribution is coming from energy. So, we can actually get rid of rating variable and use the energy because 87 percent of the information is anyway will be able to retain, retain and will get rid of one particular dimension that is rating. So, is there a better way can be increase, can we retain much more information. So, that can be seen through a principal component analysis. So, let us apply principal component analysis of energy and rating. So, first let us select the these 2 variables in a new data frame dfpca, let us apply this function. So, the p R comp is the function that is actually used to apply principal component analysis in R, if you are interested in finding more about this particular. Function you can go into the help section and find out.

(Refer Slide Time: 17:17)



```
c12=cov(df1$Energy, df1$CustomerRating)
matrix(c(v1,c12,c12,v2), nrow = 2, ncol = 2, byrow = T)

# Correlation (covariation)
cor(df1$Energy, df1$CustomerRating)

v1+v2 #total variability
100*v1/(v1+v2) #energy
100*v2/(v1+v2) #rating

# Principal Components Analysis of Energy and Rating
dfpca=df1[,c(2,15)]

mod=prcomp(dfpca)

# adding PC directions to the plot
slp=with(mod, rotation[2,1]/rotation[1,1])
```

Console C:/Users/user/Desktop/MOOC January 2018/Dr. Gaurav Dixit/Session 4/

```
[1,] 1586.0101 -264.6386
[2,] -264.6386  218.3226
> cor(df1$Energy, df1$CustomerRating)
[1] -0.4497289
> v1+v2 #total variability
[1] 1804.333
> 100*v1/(v1+v2) #energy
[1] 87.90009
> 100*v2/(v1+v2) #rating
[1] 12.09991
> dfpca=df1[,c(2,15)]
>
```

```
Data
df1      32 obs. of 15 variables
dfb2     35 obs. of 18 variables
dfpca    32 obs. of 2 variables
Values
c12      -264.638561622291
v1       1586.0100756794
```

treemap {treemap}

Create a treemap

Description

A treemap is a space-filling visualization of hierarchical structures. This function offers great flexibility to draw treemaps. Required is a data frame (dtf) that contains one or more hierarchical index columns given by index, a column that determines the rectangle area sizes (vSize), and optionally a column that determines the rectangle colors (vColor). The way how rectangles are colored is determined by the argument type.

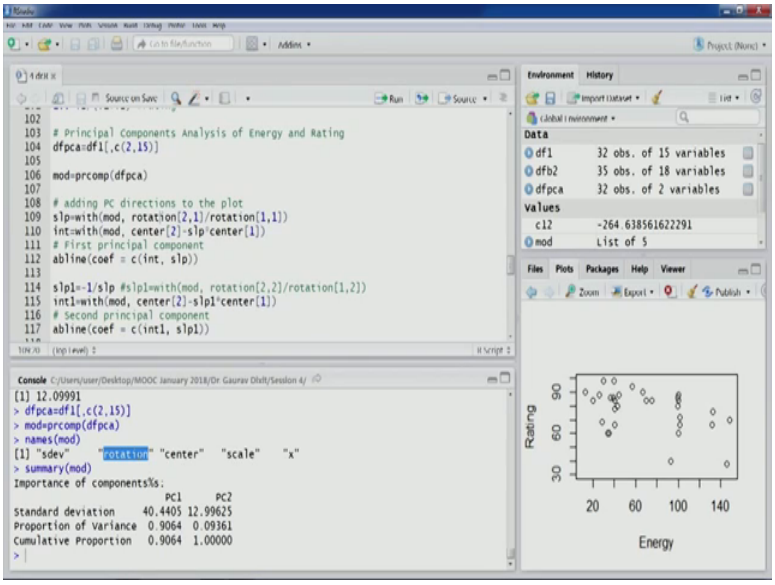
You can see principal component analysis description you can see performs a principal component component analysis on the given data matrix right.

So, we are going to use this particular function and let us run this code. So, this has been run, now many things have been computed as part of as we called this function, now let us look at the new PC direction. So, let us find out, let us look at the details of this mod function you can see different details have been computed p R comp function and if you are interested in the finding out the summary of it, you can see 2 principal components have been computed. A standard deviation for first one is pc 1 for 40 and for pc 2 is 12.99, if you look at the proportion of variance. So, pc 1 is contributing explaining 90 percent of the variance and remaining 9 percent, 9.36 per percent variance variability is contributed by second principal component.

So, you can see from the numbers earlier numbers from 87 and 12 we have redistributed in this particular variability to 90 and 9. So, this is a small change, but in other scenarios if we apply principal component analysis to other data sets or to other variables then the situation could be different, it could be from 60, 40 to 80 20 or 90, 10 kind of redistribution. So, it depends on the particular variables, in this particular case that is redistribution of variability is happening from 87 and 13 to 90 and 10. Now, let us analyze further.

So, because now will have two new dimension and determined by these 2 principal components PC 1, PC 2. So, what we will do? I will add them to our scatter plot which was earlier generated. So, first we need to compute the, this slope and intercept to find out the directions.

(Refer Slide Time: 19:50)



```
# Principal Components Analysis of Energy and Rating
dfpca=df1[,c(2,15)]

mod=prcomp(dfpca)

# adding PC directions to the plot
slp=with(mod, rotation[2,1]/rotation[1,1])
int=with(mod, center[2]-slp*center[1])
# First principal component
abline(coef = c(int, slp))

slp1=-1/slp #slp1=with(mod, rotation[2,2]/rotation[1,2])
int1=with(mod, center[2]-slp1*center[1])
# Second principal component
abline(coef = c(int1, slp1))
```

Console C:/Users/user/Desktop/MOOC January 2018/Dr. Gaurav Dixit/Session 4/

```
[1] 12.09991
> dfpca=df1[,c(2,15)]
> mod=prcomp(dfpca)
> names(mod)
[1] "sdev"     "rotation" "center"   "scale"    "x"
> summary(mod)
Importance of components%s:
                           PC1      PC2
Standard deviation     40.4405 12.99625
Proportion of Variance  0.9064  0.09361
Cumulative Proportion   0.9064  1.00000
> 
```

Environment

```
Data
df1      32 obs. of 15 variables
dfb2     35 obs. of 18 variables
dfpca    32 obs. of 2 variables
Values
c12      -264.638561622291
mod      List of 5
```

So, adding PC directions to the plot you would see that that the rotation is one particular out, one particular returned value that we have in the in the mod, in the mod function in the mod variable right. So, we are going to use this particular, this particular rotation value. So, these are nothing, but weights if you are interested in interested in looking at the rotation value you can check weights for new dimensions z 1 and z 2. So, these are the weights. So, PC 1 and PC 2 you can see the weights. So, these if we want to compute this course for new this direction PC 1 we can use these 2 weights, 1 corresponding to energy minus 0.98 then another one responding to customer rating 0.18 similarly your PC 2 rate is just the reverse.

So, using these weights we can comp compute the newest scores for these new dimensions. So, earlier we had score for energy and customer rating now. So, scores are also pre computed by the P R comm function and written in this particular variable x, let us look at the first 6 values these are the p computed this course. If you want to see how these codes can be computed. So, we can take one example. So, let us compute first score. So, d f PCA 1 let us look at the very first values of the energy and customer rating values where 70 and 84 for the first variable, if we look at the after applying the p R comp or PCA the weights are here. So, we can do we can compute the first code in this fashion.

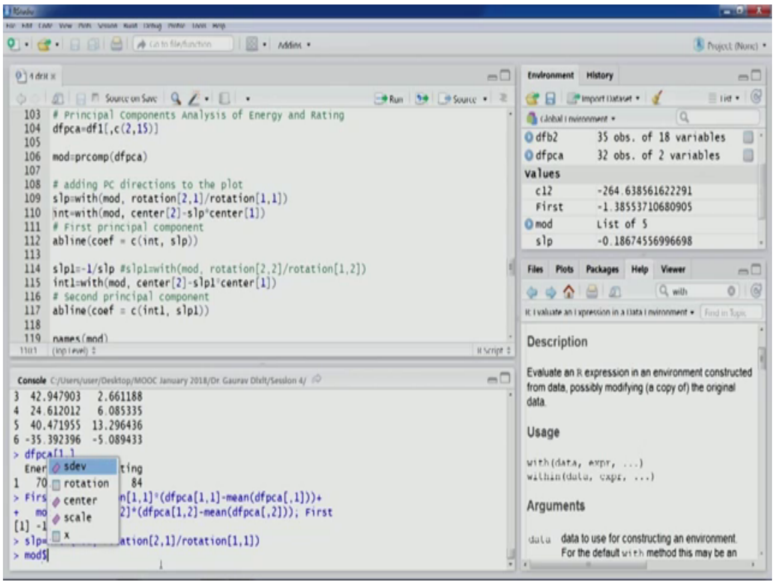
So, these are this is the particular weight that we saw in the earlier table, this is this particular value minus 0.98. So, mod dollar rotation 1 1. So, this is representing that particular weight and then we are subtracting the d f PCA 1 1 by the mean by the mean value. So, the value that we had 1 1 is 70.1 right. So, now, we are also subtracting it by by mean, similarly for second weight that is corresponding to customer rating that is a 0.1835 this is here. So, now, the second value the rating value d f PCA 1 2 is this one 84. Now, this is being subtracted by the mean value of that, mean value of customer rating and then we are weighing it through this weight that we have just computed.

So, let us compute this value you would see that value minus 1.38 has been computed and this is for first is code and for first direction you can see in the results the same value was there. You can see minus 1.38 the same value was this is, this is how we can compute the scores using the weights of new dimensions. So, now, let us plot add the directions. So, let us find out the slope. So, this is nothing, but rotation this is a y value and this is the x value and we are going to use bit one part of function that can be used to do some computations in a particular environment.

So, the environment in this is determined by the first argument that is mod in this case. So, that we do not have to use different notation like mod dollar rotation or mod dollar centre etcetera. We can in the first argument we can specify the environment that is the data and then we can specifically access the variables and do our computation. So, the same thing we are doing this is like y divided by x in this case these weights and this will give a slope.

Similarly, intercept can be fin found out using this if you are interested in looking at the new centre; new centre that can be look looked at.

(Refer Slide Time: 24:02)



```
# Principal Components Analysis of Energy and Rating
dfpca=df1[,c(2,15)]

mod=prcomp(dfpca)

# adding PC directions to the plot
slp=with(mod, rotation[2,1]/rotation[1,1])
int=with(mod, center[2]-slp*center[1])
# First principal component
abline(coef = c(int, slp))

slp1=-1/slp #slp1=with(mod, rotation[2,2]/rotation[1,2])
int1=with(mod, center[2]-slp1*center[1])
# Second principal component
abline(coef = c(int1, slp1))

names(mod)
```

Console C:/Users/user/Desktop/MOOC January 2018/Dr. Gaurav Dixit/Session 4/

```
3  42.947903   2.661188
4  24.612012   6.085335
5  40.471955  13.296436
6 -35.392396  -5.089433
> dfpca[1,]
  Ener        ting
1   70          84
> Firs       n[1,1]*(dfpca[1,1]-mean(dfpca[,1]))+
+    mo      2]*(dfpca[1,2]-mean(dfpca[,2])); First
[1] -1
> slp=        ation[2,1]/rotation[1,1])
> mod$
```

```
dfb2     35 obs. of 18 variables
dfpca    32 obs. of 2 variables
Values
  c12    -264.638561622291
  First  -1.38553710680905
mod      List of 5
  slp    -0.18674556996698
```

Description

Evaluate an R expression in an environment constructed from data, possibly modifying (a copy of) the original data.

Usage

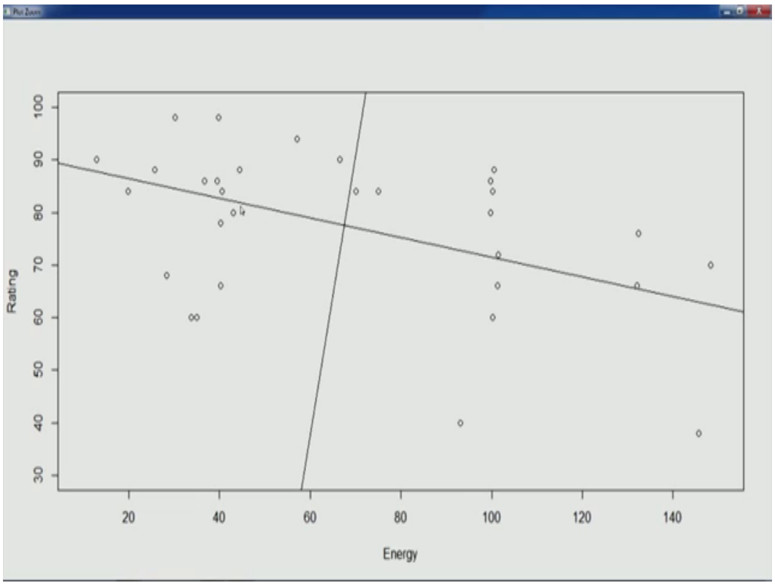
```
with(data, expr, ...)
within(data, expr, ...)
```

Arguments

data data to use for constructing an environment. For the default with method this may be an

So, this is, this is the value 67.47 and 77.5. So, if we go back to the scot scatter plot and let us zoom this particular plot.

(Refer Slide Time: 24:17)



So, the new centre is going to be at 67.47 and 77.5 so somewhere 67.47 and 77.5 somewhere here. So, new centre is going to be here now using this particular new centre, we are trying to compute the intercept. So, slope we have already computed right. So, let us compute intercept once this is done we can add this line. So, this a b line is the function that can draw a line given the intercept and slope. So, let us plot this line you can see a line has been plotted. So, this is PC 1. So, you can see if we compare it to the the original x axis that is represented by energy so there is some angle.

So, now this particular through this line even more variability is being captured that is why we saw a jump from 87 to 90. So, the earlier variability for our captured by energy would have would was 87 which would be represented by a line like this, a line parallel to x axis horizontal line, now this is slightly some slope is there. So, this is capturing even more variability. So, that is why we saw a jump from 87 to 90.

So, let us plot the second direction. So, because these 2 are going to be perpendicular orthogonal, we can compute the slope in this fashion also slope one slope 1, now in this new slope can be minus one divided by the slope for the PC 1 or the other way also the rotation values can also be used to compute the slope. So, let us compute this, we can also find out the intercept for this particular line and we can add this line into the plot. So, if it see the pc 2 has been added. So, these are the 2 lines, now this is now this is the redistribution of variability. So, earlier we had x axis and y axis represented by energy and rating, now we have PC 1 and PC 2. So, re distribution is variability of variability is happening, now earlier it was 87 and you know kind of 13 by these 2 axes, now we have 90 and 10 scenario.

So, same thing we can see here. Now, the new v z 1 value, v z 2 value we can find out the total variability if you look at the total variability is 18, 1804 and if you go back and look at the earlier total variability that we have computed as v 1 plus v 2 both are same. So, you can see variability is same, but the redistribution has happened because of the change in the directions of dimension.

So, now a contribution by energy and rating you can see 90 and 90.6 and 9.36, now principal component analysis can now can be applied to all numerical variables that we have in the data set. So, let us have a look at the data set that we had. So, this is the data set that we saw earlier. So, till now we applied the principal component analysis on just 2 variables energy and rating. So, now, we can apply it on all numerical variables right. So, in this particular data set the variables that we have or all numerical right except 2 variables these trans fatty acids and cholesterol. So, most of the values in these 2 variables are 0. So, therefore, we would not be we would not be including them in this particular analysis. So, we would be eliminate them and other variables would be taken as for the principal component analysis. So, I will stop here and will apply principal component analysis on all the variables that are in the data set in the next lecture.

Thank you.