Data Mining
Home Work 03

aqeel labash

24 February 2016

First Question

To plot the density I used the following code :

```
##### Question 1 #####
rm(list=ls())
setwd("/home/aqeel/Study/DM/HW03")
klient1 <- read.csv('klien  t1.txt',header = FALSE)
klient3 <- read.csv('klient3.txt',header=FALSE)
png("densitywithoutwidth.png",width=500,height = 500)
plot(density(klient1$V1),col="RED",type = "l",main="K1 & K3 Density")
lines(density(klient3$V1),col="green")
dev.off()
```

The previous code will plot the density of klient1 & klient3 without specifying the bandwidth. We notice that there is only a small difference between klient1 and klient3.

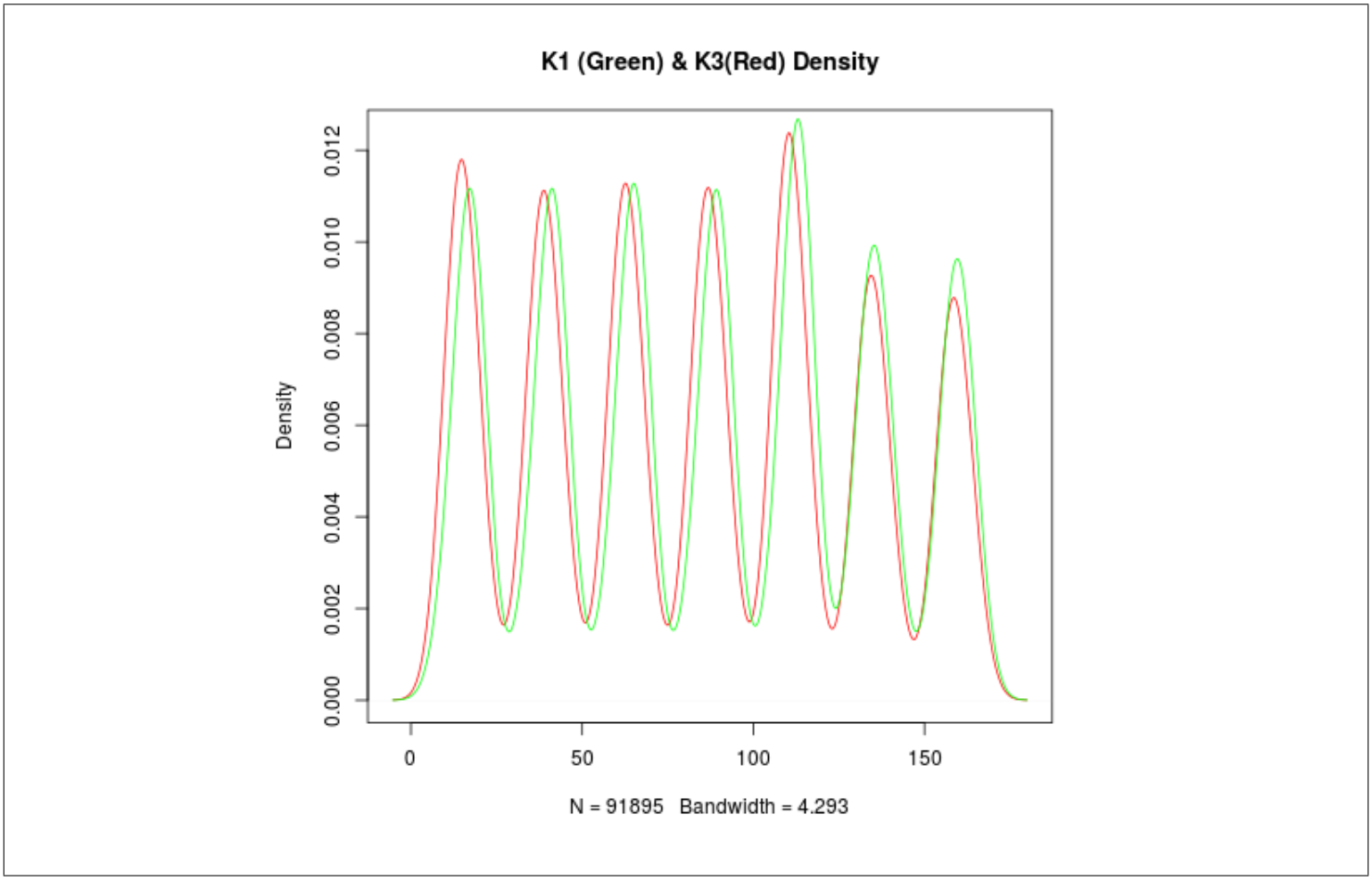

Figure 1: Show the density of K1 & K3 without specifying width

Now to choose the width actually I tried the values to reach a level where I can see the gaps without this periodic behavior in the

density. I selected bandwidth to be 11 where we still can see the changes but it won't affect our judgment of the data (high points,small points etc..).Here is the figure with bandwidth:

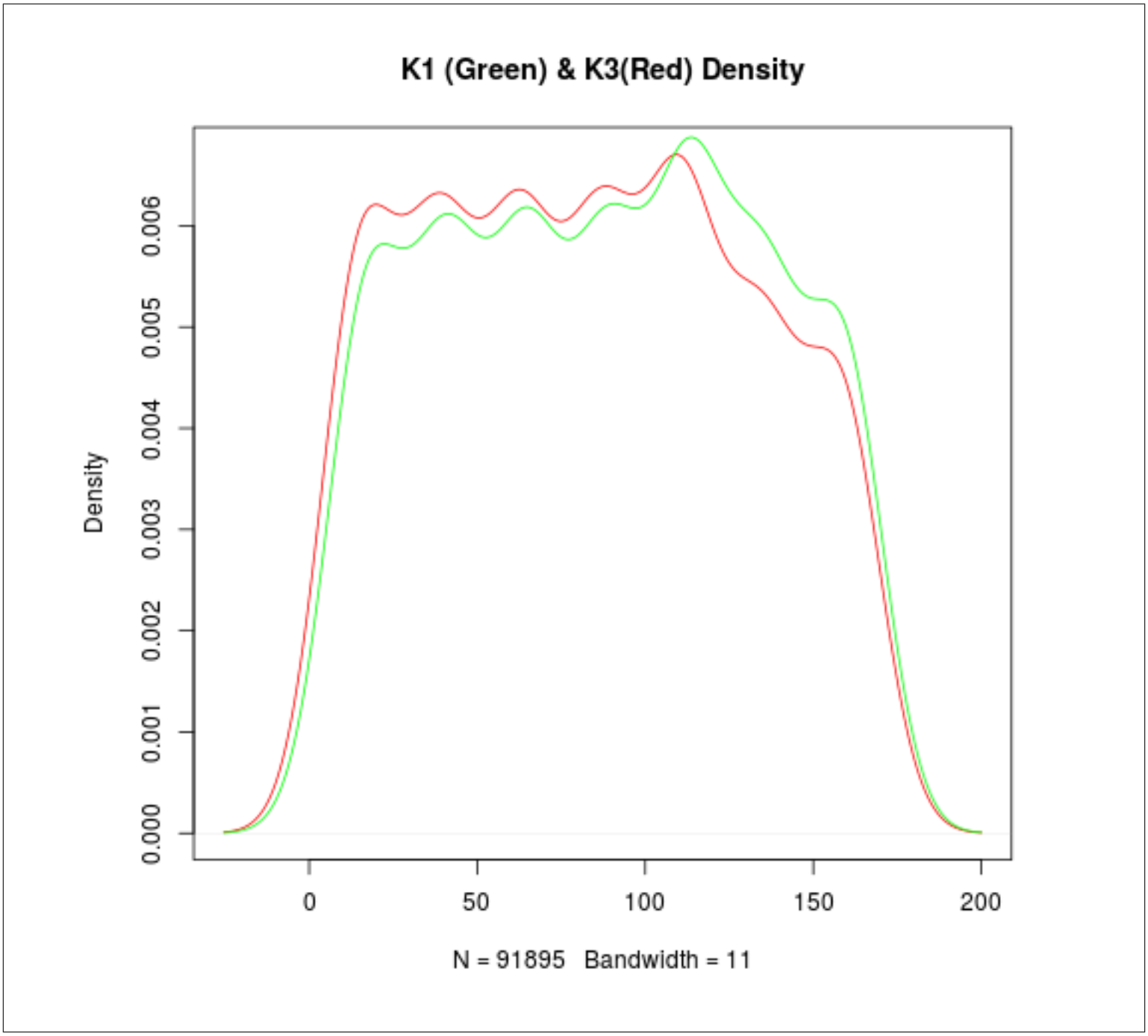

Figure 2: Shows K1&K3 density with specified bandwidth

Figure 2 was generated by the following code :

```
png("densitywithwidth.png",width = 500,height = 500)
bandwidth =11
plot(density(klient1$V1,bw=bandwidth,kernel = "gaussian"),col="RED",type = "l",main="K1 (Green) & K3(Red)
    Density")
lines(density(klient3$V1,bw=bandwidth,kernel = "gaussian"),col="green")
dev.off()
```

To characterize the data first we should understand the data. Week hours : 7*24= 168 , that's how the time is identified. And the holiday is from 120 to 168.So depending on that information here is some characteristics about the data:

1. Both groups shows higher activity at the start of week end.
2. Both groups shows less activity at the week start.

3. First group (in Green) shows less active during the week than second group (K3,in red)
4. First group shows more activity at the start of weekend than the second group.
5. There is a slight shift between both groups.Which made me think that the groups from different time zones

Second Question

The data contain the following information :

1. Date:the date of purchase in syntax of : "YYYYMMDD"
2. Time:the time of purchase in syntax of : "HHMM" or "HMM" depending on time.The time is based on 24 not 12.
3. Product: the name of the product.
4. Shope_id: represent the shop identity

The following Table shows :

1. How many products bought from specific shop.
2. The total amount bought of specific product from all the shops.
3. The total amount bought from specific shop.
4. The total bought from all shops from all products.

Product\Shop	3	4	18	21	32	Total
Banana	6778	8677	4080	1727	3585	24847
Coffee_Cream	4272	7259	4516	1418	3066	20531
Eggs_1	1880	3704	1326	1106	2073	10089
Eggs_2	100	181	0	0	0	281
Grapes	710	1199	495	273	525	3202
Milk_1	3568	5173	2740	836	2557	14874
Milk_2	5629	8309	4968	2440	4369	25715
Sour_Cream_1	2597	3206	1817	848	1236	9704
Sour_Cream_2	2891	5504	3046	1569	2866	15876
Vastlakukkel	939	1784	730	273	383	4109
Whipped_Cream	2285	3815	1168	600	1396	9264
Total	31649	48811	24886	11090	22056	138492

The code used to generate the previous table is here:

```
shopsdata = read.csv("product_time_shop.txt",sep = ';',header = TRUE)
x<-table(shopsdata$product,shopsdata$shop_id)
x<-cbind(x,rowSums(x))
x<-rbind(x,colSums(x))
x

```

Third Question

To create the boxplots I depended on the 2nd question result. in Figure 1 we can notice that Tuesday has the highest selling over all products. I guess that's because people misjudged there needs at the start of the week :) (am not alone :D).Saturday also has a high sell rate over all products. Which could be explained by holiday & people buying for the week.

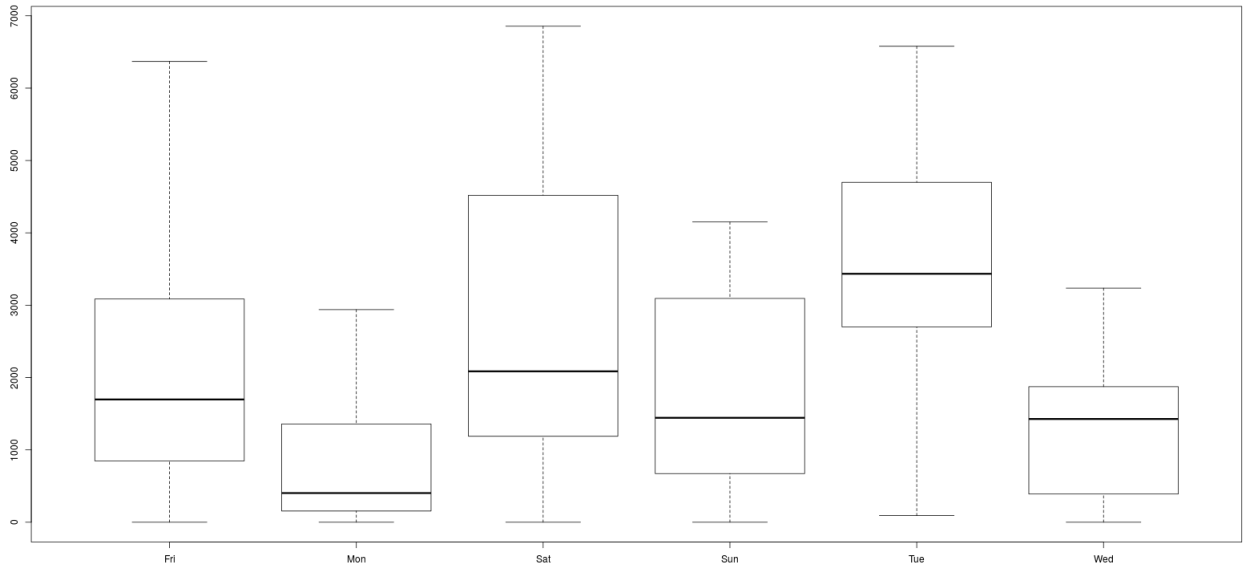

Figure 3: Total sales in the specified dates in all stores (Date Product)

In figure 2 we can see that Milk_2 has a high median which mean it's most sold in most stores. the opposite for Eggs_2 & Grapes which mean they haven't sold much.Banana have a high selling over most of the stores.

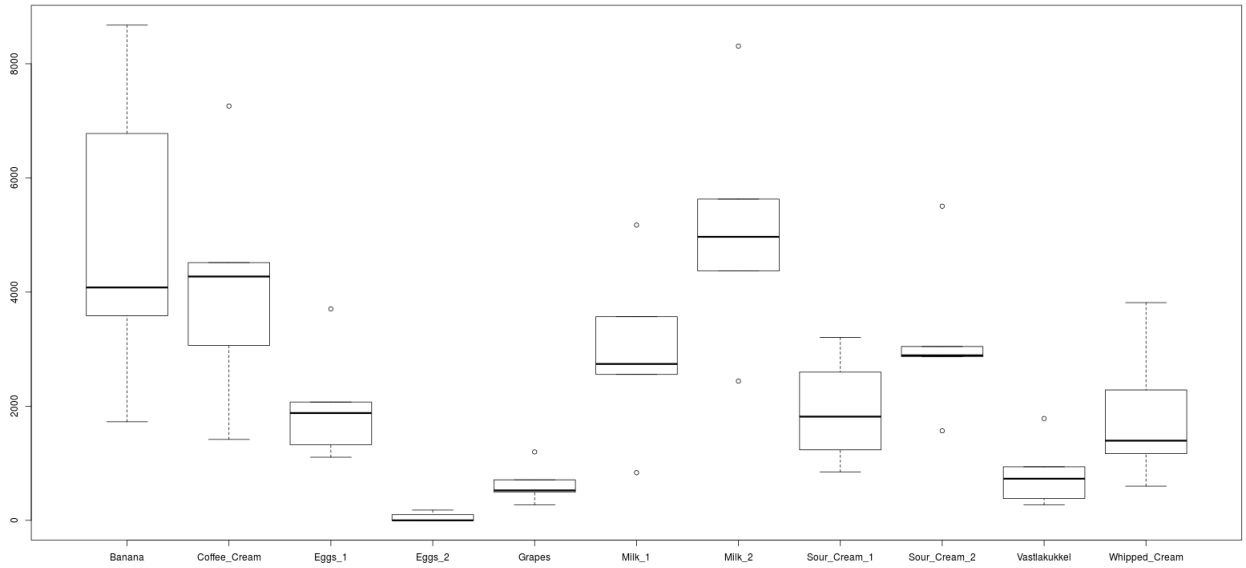

Figure 4: Total sold products in all dates , all stores.(Product Store)

Figure 3 shows us the most product where bought from "Store 4". Also "Store 21" didn't sell much of the products compared to other stores.

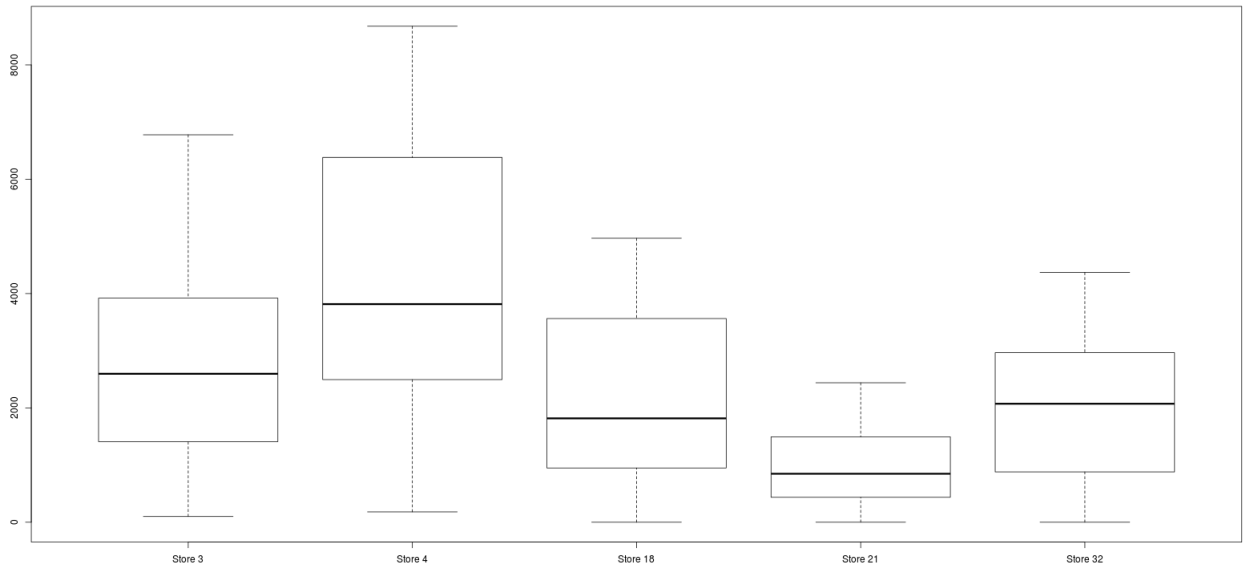

Figure 5: Total sales in specific store in all dates , all products (Store Product)

The previous boxplots where generated by this code :

```
##### Third Question ##########
library(plyr)
days<-c("Sat","Sun","Sat","Tue","Mon","Tue","Sun","Sat","Fri","Fri","Tue","Wed")
shopsdata$date<- mapvalues(shopsdata$date, from = c(unique(shopsdata$date)), to = days)
png("boxplotStores",width=1600,height = 800)
boxplot(x[,(1:5)],names=paste("Store",colnames(x),sep=" "))
dev.off()
png("boxplotproducts",width = 1600,height = 800)
boxplot(t(x)[,(1:11)],names=colnames(t(x)))
dev.off()
png("boxplotdates.png",width=1600,height=800)
boxplot(table(shopsdata$product,shopsdata$date)[,c(1:6)])
dev.off()
```

To represent the data I created a plot compare store vs product.:

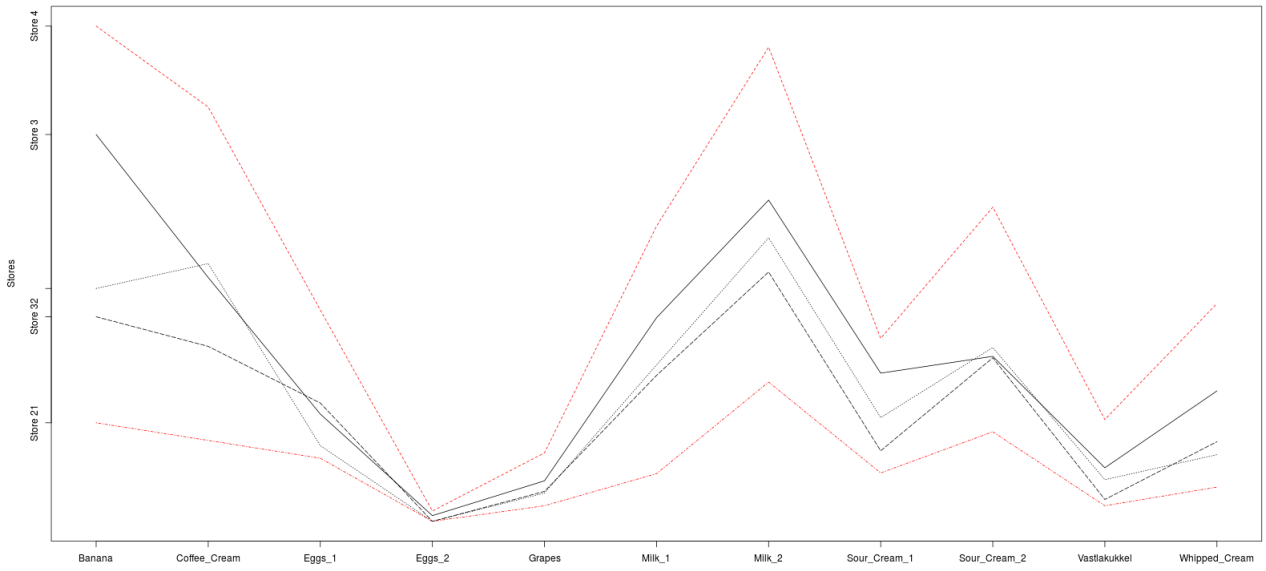

Figure 6: Stores vs products

From the previous figure we can see that store 4 was selling the most.Banana and milk_2 was the most sold thing. I believe this figure satisfy information wanted from stores and products. The previous plot was generated by the following code :

```
png("matplotallvsall.png",width=1600,height=800)
matplot(x, pch = 16,  col = 1:2,xaxt="n",yaxt="n",
ylab = "Stores",type = "l")
axis(1,at=c(1:11),labels = rownames(x))
axis(2,at=x[1,],labels =paste("Store",colnames(x),sep=" "))
dev.off()
```

Fourth Question

For this task I looked at the data. We can know if a specific store run out of some or one product if this products stopped selling while the rest still.So I wrote some code to see this fact and here is the code :

```
######### Fourth Question ###################
shopsdata = read.csv("product_time_shop.txt",sep = ';',header = TRUE)
products <- unique(shopsdata$product)

dd <-subset(shopsdata,date=="20140104" & shop_id==18)[,(2:3)]
dd$product<- factor(dd$product,products,c(1:11))
png("distribution.png",width = 900,height = 1100)
plot(dd,yaxt="n",main="Store 18 at  2014/01/04")
axis(2,at=c(1:11),labels =products)
dev.off()
dd<-subset(dd,product==9)
png("histdensity.png",width = 600,height = 600)
hist(dd$time,prob=TRUE,xlim =c(1000,2400),main = "Density Over Histogram",xlab="Time")
lines(density(dd$time),col="RED")
dd <-subset(shopsdata,date=="20140104" & shop_id==18)[,(2:3)]
dd$product<- factor(dd$product,products,c(1:11))
dd<-subset(dd,product==8)

lines(density(dd$time),col="GREEN")
dev.off()
```

The previous code will plot figure 5 which show the distribution of products over time in Store 18 at 2014/01/04. That allow us to see when a product is not sold anymore while other products still being sold. Same figure show us that product Milk_2 stopped selling around 16 18 (not accurate due lake of space in the image.)

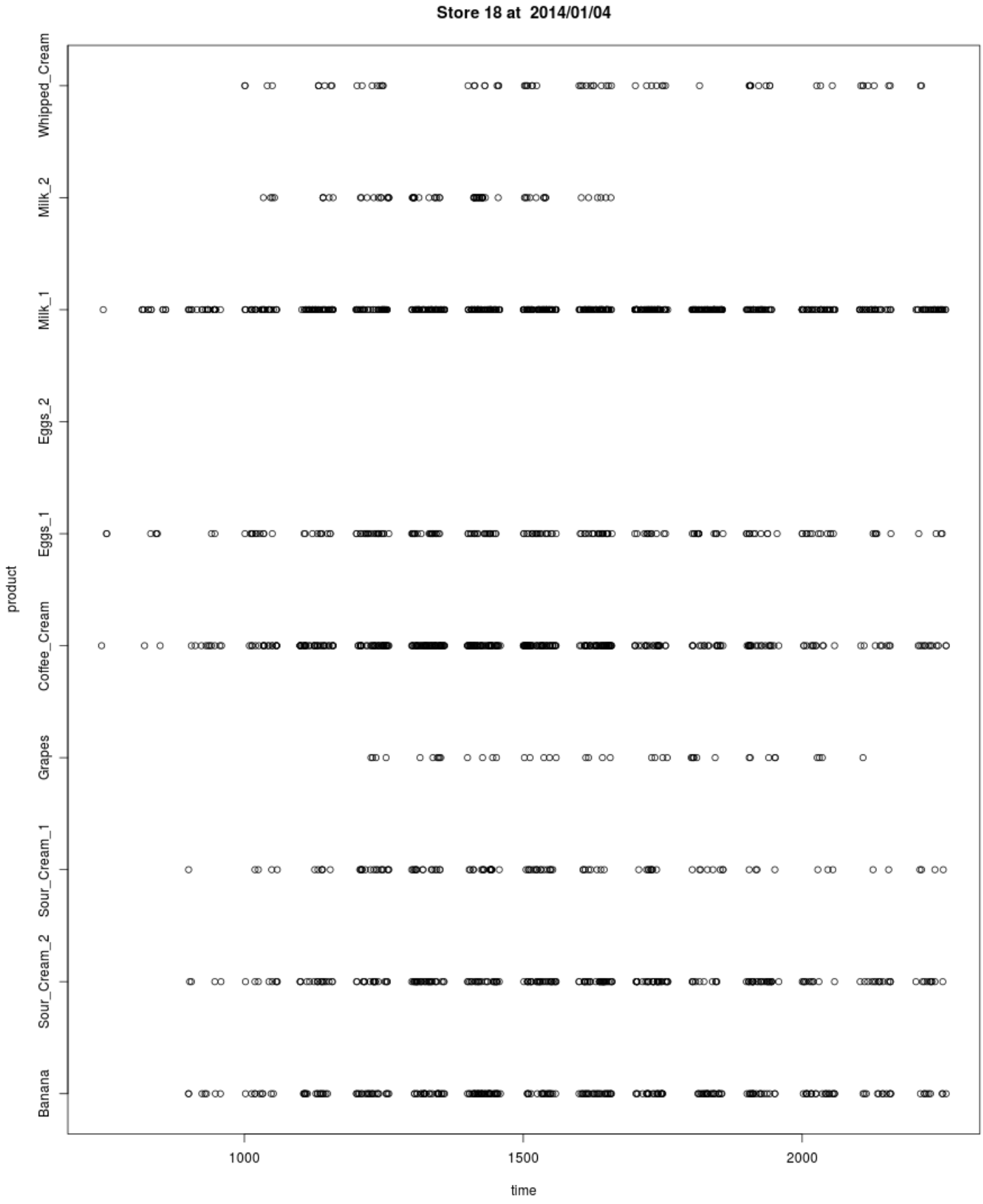

Figure 7: Show the time of selling all products.

After that to convince you more here is a figure 6 which contain the histogram and the density over it. The red line is Milk_2 and the green line is Milk_1 and we can clearly see that after 1700 o'clock there was no more purchases over milk_2.

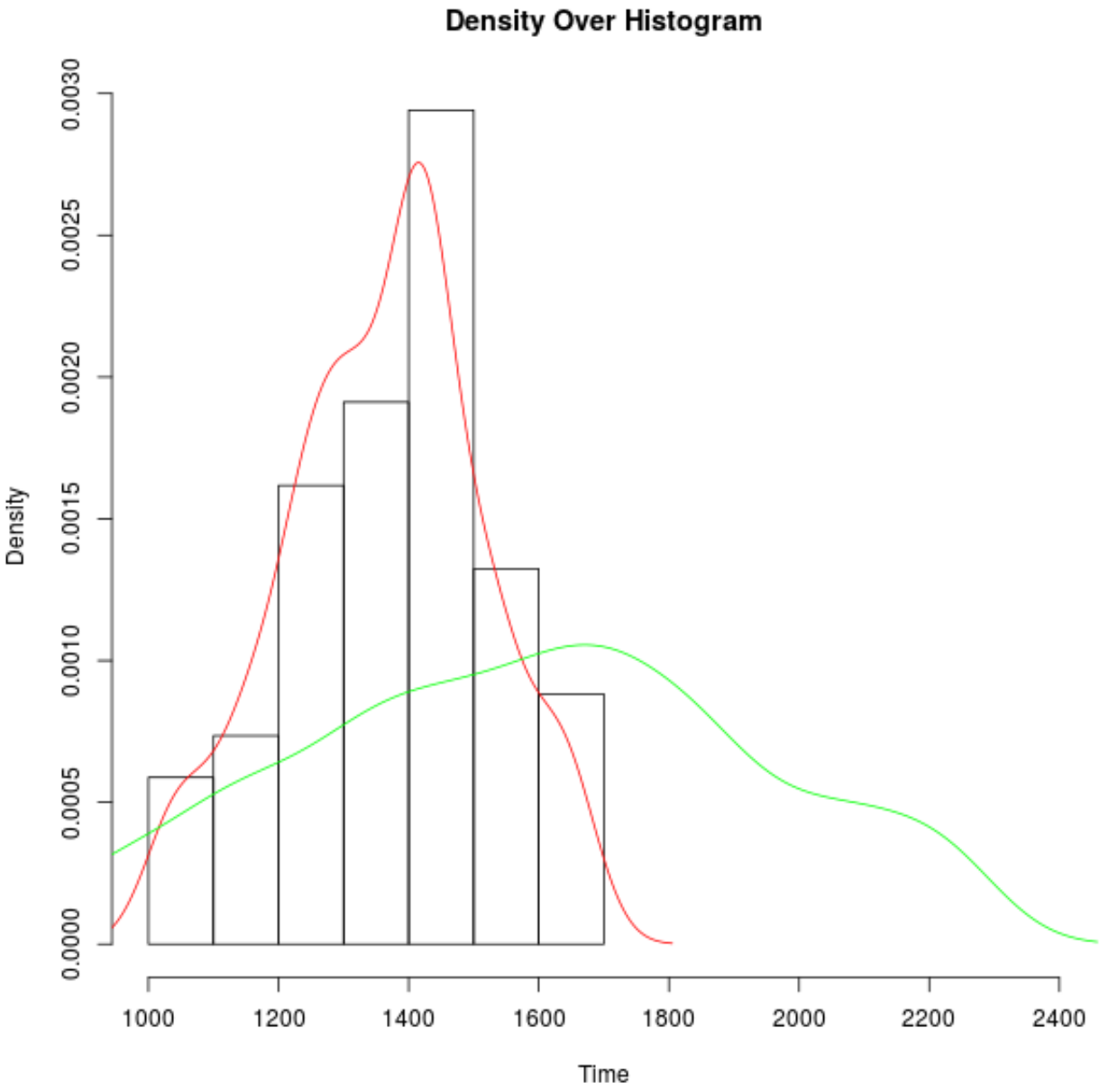

Figure 8: Milk_2 density (in red) vs Milk_1 density (green)

So to formulate it, we can check by fix date,store and then check over all products in one hit to see which one doesn't sell at specific time while other products still selling.

Fifth Question

For this question what I've done is firstly organize the data.So I concatenated the columns (Date,Product,Store) after that I updated the time value then draw the heat map. and here is the code :

```
############ Question 5 ###############
rm(list=ls())
shopsdata = read.csv("product_time_shop.txt",sep = ';',header = TRUE)
shopsdata$info=paste(paste(shopsdata$date, shopsdata$product, sep=";"),shopsdata$shop_id,sep = ";")
shopsdata$date=NULL
shopsdata$product=NULL
shopsdata$shop_id=NULL
for (i in c(7:23))
{
shopsdata[shopsdata$time>=(i*100)&shopsdata$time<((i+1)*100),]$time =i
}
colSums(table(shopsdata$info,shopsdata$time))
```

```
13 png("heatmap2.png",width=800,height = 800)
14 heatmap(table(shopsdata$info,shopsdata$time))
15 dev.off()
```

Here is the heatmap :

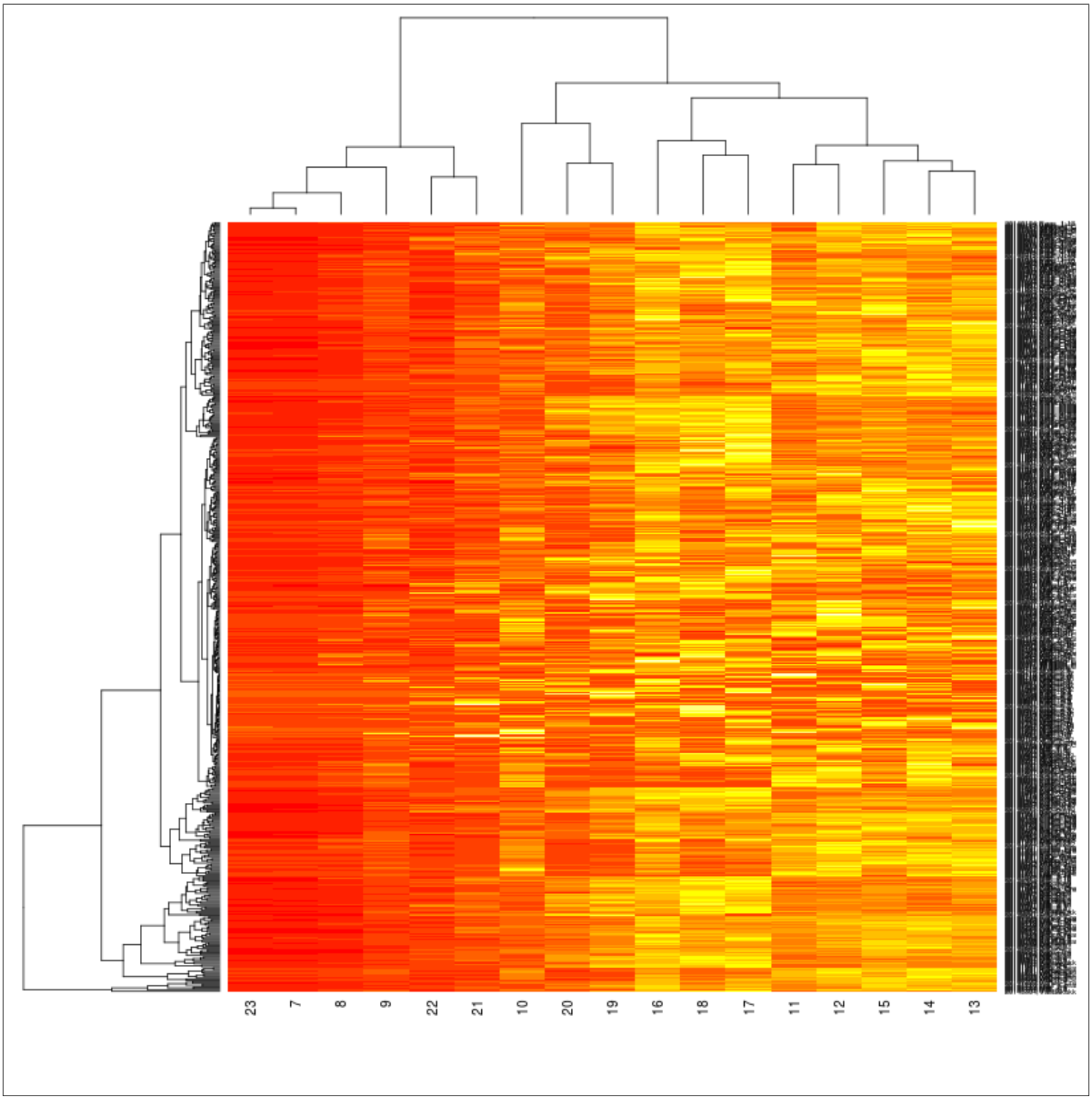

Figure 9: Heatmap (Red to Yellow)

In figure 7 the heat map shows that at 13 (1 PM) it's the most time that people buy in general.
Note: In the heat map the time ordered depending on heat.At 23 the less heat but at 13 the highest heat.

Sixth Question

(1)

To see the behavior, I firstly replaced the dates with week days and then calculated the total sales per day using the following code :

The previous code create the following table :

Fri	Mon	Sat	Sun	Tue	Wed
23948	9649	32260	20460	38174	14001

where we can see that people usually by more on Saturday and that's clear and expected. The shock come from Tuesday which has a high selling rate.I think this high value is due to end of what usually people buy on the week end.Here is the code I used for this task:

```
############### Question 6 ###############
###1st request###
rm(list=ls())
setwd("/home/aqeel/Study/DM/HW03")
shopsdata = read.csv("product_time_shop.txt",sep = ';',header = TRUE)
library(plyr)
days<-c("Sat","Sun","Sat","Tue","Mon","Tue","Sun","Sat","Fri","Fri","Tue","Wed")
shopsdata$date<- mapvalues(shopsdata$date, from = c(unique(shopsdata$date)), to = days)
```

(2)

To do that I used the table of Days-Products.To normalize we can take the maximum value in the matrix and divide the matrix over that number.Here is the code for this request:

```
###2nd request###
products_shops<-table(shopsdata$date,shopsdata$product)
products_shops<- products_shops/norm(products_shops,type ="M")
png("heatmapproductsvsdays.png",width = 500, height = 500)
heatmap(products_shops)
dev.off()
```

After that I created a heat map to have a clear view of products bought which day.

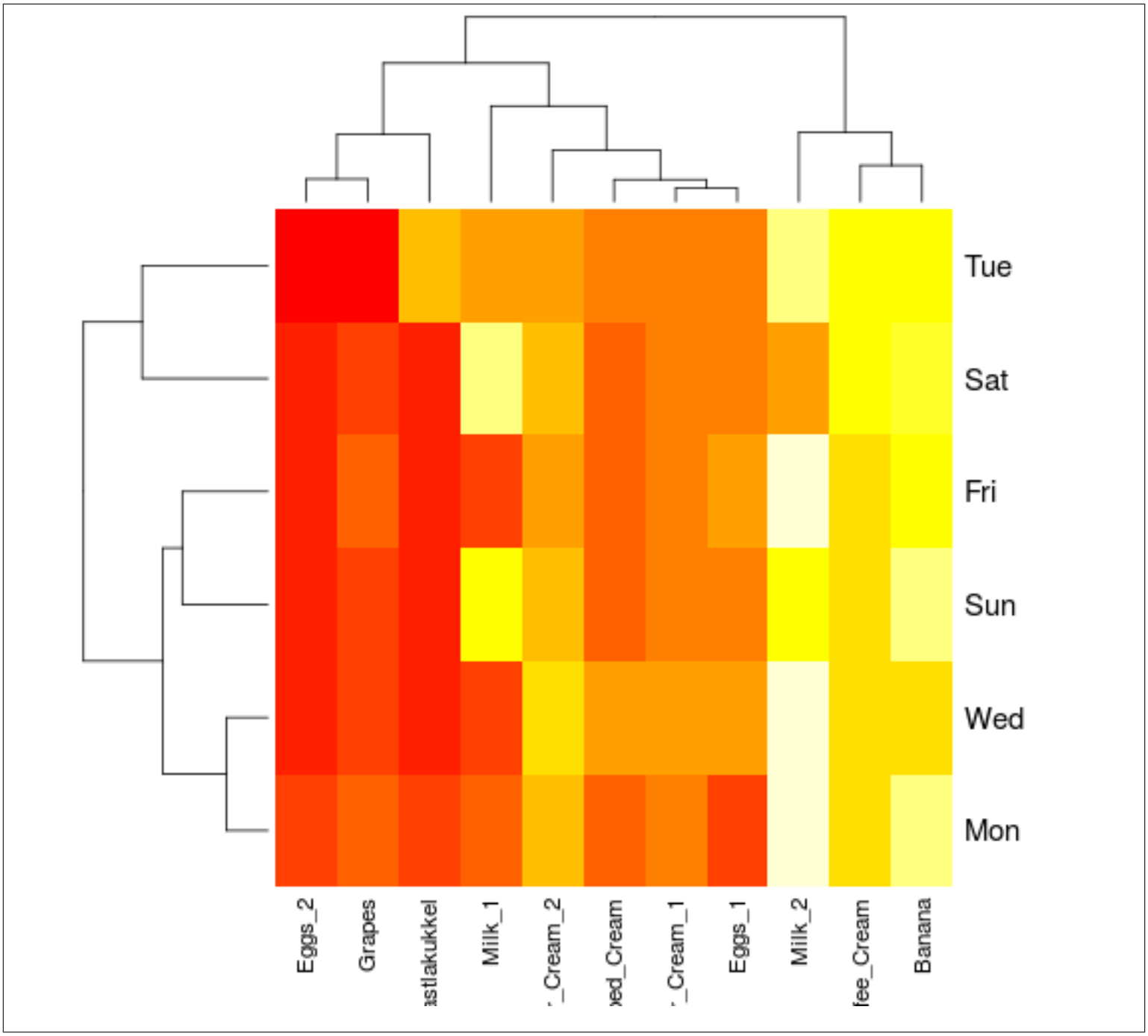

Figure 10: Shows heat map of products over week days

(3)

For this task I changed the data:(dates to week days , time to formual hour, merged date with time) after that created a table (too large 100 row). Then I normalized the matrix and table it to get the following plot:

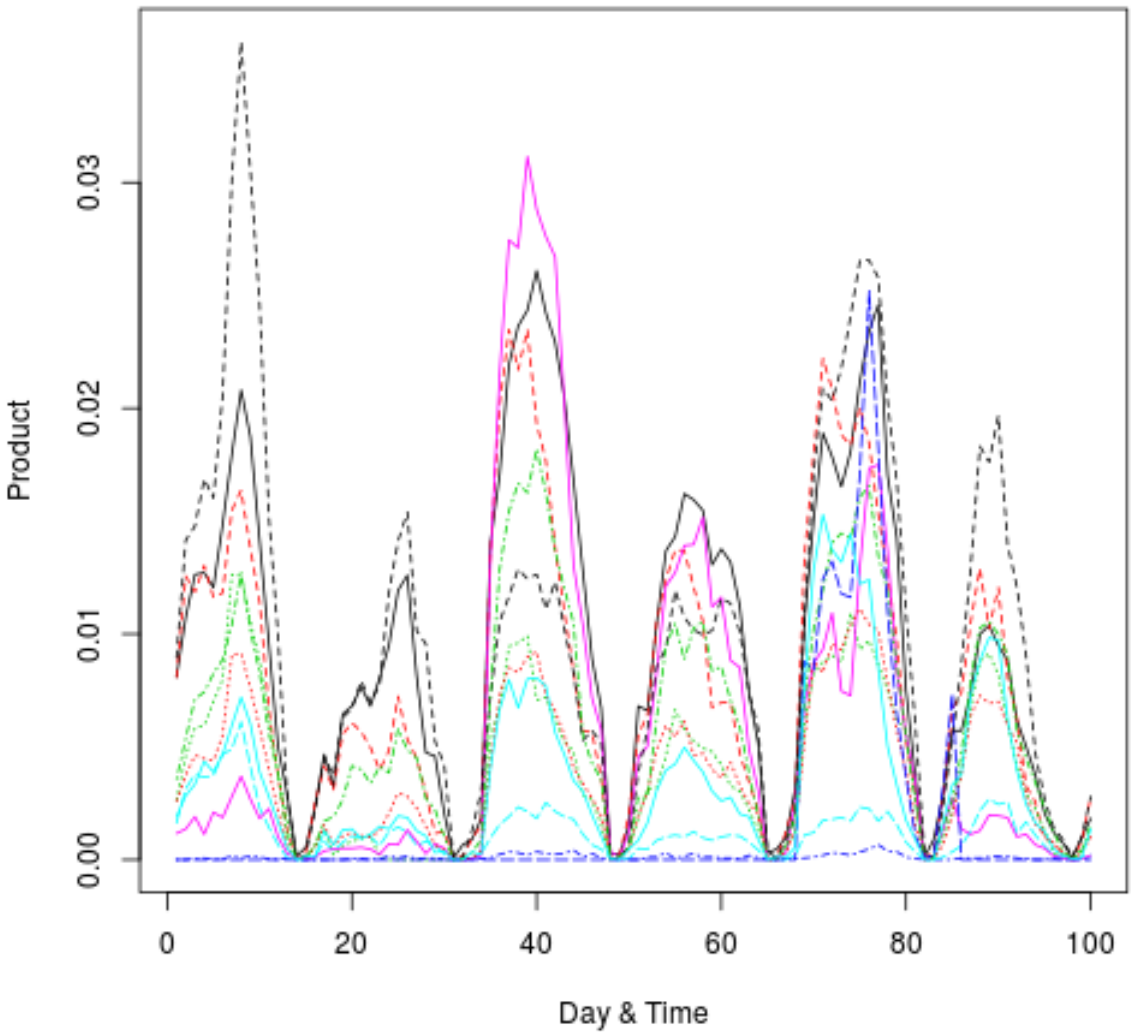

Figure 11: Product vs day and time.

Here is the code for this task :

```
###3rd request###
for (i in c(7:23))
{
shopsdata[shopsdata$time>=(i*100)&shopsdata$time<((i+1)*100),]$time =i
}
days<-c("Sat","Sun","Sat","Tue","Mon","Tue","Sun","Sat","Fri","Fri","Tue","Wed")
shopsdata$date<- mapvalues(shopsdata$date, from = c(unique(shopsdata$date)), to = days)
shopsdata$date<-paste(shopsdata$date, shopsdata$time, sep=" ")
shopsdata$time<-NULL
products_overtime <-table(shopsdata$date,shopsdata$product)
products_overtime<- products_overtime/norm(products_overtime)
png("frequencyoverdays.png",width=500,height=500)
matplot(products_overtime,type = "l",xlab = "Day & Time" , ylab = "Product" )
dev.off()
```

(4)

From the previous figure which shows the products bought at specific day& time.
I believe if we have it large enough to identify the axis we would be able to identify the products that sold out.