Regional Analysis with Topological Data Analysis Ball Mapper

Session 3 Part A: Summarising Data

Session 3 gives you the chance to use Ball Mapper (BM) on an example dataset. In this first part we will see how conventional statistics helps us to understand our data. This document supplements the code file that is available on the GitHub repository as Session 3 Part A.txt and you are strongly encouraged not to try to copy and paste from this document. All code in this file appears in the courier new font

**Preliminaries**

If you have not previously used R then please make sure that you have a version of the software installed on your computer. There is a second alternative to use Google Colab for this session that does not require an installation of R. A separate document gives guidance on using Google Colab. The remainder of the instructions are the same however you are running the code.

Packages

R uses packages of user developed code to run many of its’ useful functions therefore you will need to install the packages for the session. This only needs to be done once, so if you already have the packages installed you may skip this step.

install.packages("dplyr")

install.packages("BallMapper")

Working Directory

R requires you to set a working directory in which the data is located. This will also be the folder where all output is saved. In the example code the working directory is set to setwd("D://oraf/") but you should set this with the right path for your computer. If you wish you can set the working directory through the menus. On Windows you would select File and then Change dir… , whilst on a Mac you would select Misc and then Change Working Directory.

Before starting you must make sure that the **region1.csv** file is in your working directory.

*For those using Google Colab the instructions are slightly different because Google creates a temporary working directory for you. There you will need to upload the data.*

**Setting Up the R Environment**

For this session we will need two packages and so will use the following code to bring them into the R environment

library(dplyr)

library(BallMapper)

Next we should read in the data using the read.table function. The options are the filename, the fact that the file uses a comma to separate columns, and the fact that the first row is the variable names.

dtx<-read.table("region1.csv",sep=",",header=TRUE)

If you wish to view the data then you can use the following code. Seeing the data like this is good to verify that what you have read in matches with your expectations.

head(dtx)

A screenshot of a computer

Description automatically generated with medium confidence

Figure 1: Screenshot of first 6 rows of the dtx data frame

To verify that you have the correct data the output from the head(dtx) command is shown in the screenshot as Figure 1. The variables included are from the 2011 Census for England and Wales. The first two columns allow linking to other geographic databases and/or shapefiles for the construction of maps. Excellent resources to help with mapping in R are available online.

The remaining variables are as follows:

|  |  |
| --- | --- |
| Variable | Interpretation (All are percentages) |
| geog | Name of the Local Authority District |
| Armed | Respondents employed in the armed forces |
| Deprivation0 | Households with no deprivation as assessed against Income, health, Overcrowding and Education |
| Deprivation1 | Households defined as deprived on one of the four measures |
| Deprivation2Plus | Households defined as deprived on two or more of the four measures |
| HealthVeryGood | Respondents who self-identify as having very good health |
| HealthGood | Respondents who self-identify as having good health |
| HealthLow | Respondents who self-identify as having fair, bad or very bad health |
| Agriculture | Respondents working in the agriculture sector |
| Manufacturing | Respondents working in the manufacturing sector |
| Accommodation | Respondents working in the accommodation and travel sector |
| Married | Households where the owners are married |
| Cohabit | Households where the owners cohabit |
| Single | Households with one adult resident who is single |
| Other | Households with one adult resident in a relationship, widowed or divorced |
| QualNone | Highest level of qualification in household is below secondary school |
| QualLevel1 | 1-4 GCSEs at grade A-C |
| QualLevel2 | 5+ GCSEs at grade A-C |
| QualApprentice | Apprenticeships |
| QualLevel3 | Two or more A-Levels |
| QualLevel4 | University degree or higher – includes professional qualifications |
| QualOther | Includes vocational qualificiations |
| OwnedOutright | Household is owned outright |
| OwnedMortgage | Household is owned with support from a mortgage |
| SocialRental | Household is rented from a social housing agency (e.g council) |
| PrivateRental | Household is rented from a private individual or company |

Table 1: Variables used in this session

Table 1 is a useful reference for all of the variables used in this session. However, there are obviously large correlations between the variables. We may see the correlations using the code:

cor(dtx[,3:ncol(dtx)])

Note that this will be a very large correlation matrix, but you can see within it that there are a lot of correlations above 0.8. BallMapper can work with highly correlated variables, but to avoid any problems of correlation in the statistical analysis of the data we will restrict focus to a subset of variables. Selection is made using the following code (note each block is a single line):

dty<-cbind(dtx[,1:2],dtx$QualLevel4,dtx$Deprivation0,dtx$Accommodation,dtx$Married,dtx$HealthVeryGood,dtx$OwnedMortgage)

names(dty)<-c("geog","geogcode","QualLevel4","Deprivation0","Accommodation","Married","HealthVeryGood","OwnedMortgage")

These variables may be edited if you wish to try different combinations of the data.

Finally we create a dummy variable which takes the value 1 if the percentage of households where the highest qualified resident has a university degree or higher. The intuition of this code is that the value 1 is assigned wherever the condition is true.

dty$QL4<-as.numeric(dty$QualLevel4>33)

Our data is now ready for analysis. As a final verification the head of dty should appear as in Figure 2

A screenshot of a computer

Description automatically generated with medium confidence

Figure 2: Head of the reduced dataset dty

**Summary Statistics**

As introduced in Session 1, the first step of an empirical analysis is to look at the summary statistics. The precise table used may include quantiles as well as the minimum and maximum. In some applications skewness and kurtosis for the variables may be tested. Here we present a simple summary statistics table which is created using the user defined function sumstats as on the top of the next page.

To run the function paste it from the accompanying text file. Please note the next lines of code will not run without it.

s001<-sstatsmat(dty[,3:8]) # Creates an object with the summary statistics

s001<-as.data.frame(s001) # Convert to data frame

names(s001)<-c("Variable","Mean","s.d.","Min","Max")

write.table(s001,"summarystats.csv",sep=",",row.names=FALSE)

s001

sstatsmat<-function(characteristics,decp){

if(missing(decp)) decp <- 2

a001<-ncol(characteristics)

sstats<-matrix(0,nrow=a001,ncol=5)

for(i in 1:a001){

j<-i

sstats[i,1]<-names(characteristics)[j]

sstats[i,2]<-round(mean(characteristics[,j]),decp)

sstats[i,3]<-round(sd(characteristics[,j]),decp)

sstats[i,4]<-round(min(characteristics[,j]),decp)

sstats[i,5]<-round(max(characteristics[,j]),decp)

}

return(sstats)

}

We now have the basic information about our dataset

Q1: What inference may be derived from the summary statistics?

Q2: What does the correlation matrix tell us about the dty dataset? (Hint: the correlation code for dtx can be adapted from above. A code to allow output to a csv file is included in the code file.)

Chart, scatter chart

Description automatically generated

**Scatter Plots for Visualisation**

The first argument of today’s session is that to understand data we must view its joint distribution. One way to see the distribution is to plot two variables as a scatter plot. Scatter plots are created by default from the plot() function of R. The plotting code below will give us a basic scatter plot. The scatter plot generated is seen in Figure 3.

The options for plotting are the variables to be plotted, the character used for the points (here a solid circle), the ranges for the X and Y axes and then labels for the axes. Much more may be added to enrich the plots but that is left for you to try.

Figure 3: Example scatter plot

The code for producing Figure 3 is:

plot(dty$Deprivation0,dty$Accommodation,pch=16,xlim=c(0,60),ylim=c(0,60),xlab="Deprivation 0",ylab="Accommodation")

In Figure 3 the axis limits are set based on the maximum value of the characteristics observed when we constructed the summary statistics. We also have no understanding of the relationship between these points and the level of qualification within the areas.

Our next task is to use the dummy variable QL4 to colour the plot. We use the dummy because colouring by the level variable will create too many subtleties of colour. Again those interested can look at guides to plotting in R to see ways of colouring the points on colour scales etc.

As a first step to plotting we will create two subsets from the overall data frame. One where the QL4 dummy is 0 and one where the QL4 dummy is 1.

dty0<-subset(dty,dty$QL4==0)

dty1<-subset(dty,dty$QL4==1)

Here we could undertake some two-sample t-tests to test the differences in the five characteristic variables. However, because the choice of 33% was arbitrary there is less value in such tests. In other applications of BM it may be sensible to conduct such tests.

We now produce a scatter plot on the [0,60] scale where the points where the qualifications are low are coloured black and the points with higher proportions of level 4 qualifications are blue. We add a legend in the bottom left to help with the presentation of the plot.

plot(dty0$Deprivation0,dty0$Accommodation,pch=16,xlim=c(0,60),ylim=c(0,60),xlab="Deprivation 0",ylab="Accommodation")

points(dty1$Deprivation0,dty1$Accommodation,pch=16,col="blue")

leg.text=c("Below 33%","Above 33%")

legend("bottomleft",leg.text,pch=16,col=c("black","blue"))

A gap is left here because the first line wraps around and should actually be a single line of entry in R. Remember it is always easier to take the code from the accompanying .txt file.

Q3. What does the plot tell us about the relationships between the proportion of households who are not classed as deprived on any of the four measures, the proportion of residents working in the accommodation and tourism sector, and the proportion of households where the highest qualification is at level 4?

R allows us to output graphs as .png files using the simple png() function. Below each block of code you will also see a second identical block which is prefaced by a png() command and ended with a dev.off() line. This creates a new graphic device in R, plots as instructed and then closes the device. As a result you will not see the graph, just a message that says the graph is complete.

png("Deprivation0Accommodation60.png")

plot(dty0$Deprivation0,dty0$Accommodation,pch=16,xlim=c(0,60),ylim=c(0,60),xlab="Deprivation 0",ylab="Accommodation")

points(dty1$Deprivation0,dty1$Accommodation,pch=16,col="blue")

leg.text=c("Below 33%","Above 33%")

legend("bottomleft",leg.text,pch=16,col=c("black","blue"))

dev.off()

The 60 on this file name is to represent the fact that 60 was used as the axis limit. After running the code you will see that a new .png has appeared in your working directory. The .png file can then be pasted into a paper, report etc.

Chart, scatter chart

Description automatically generatedChart, scatter chart

Description automatically generated

Figure 4: Scatter plots of the proportion of households recording 0 measures of deprivation and the proportion of households where a resident is employed in the accommodation and tourism sector. Left panel with both axes on scale 0 to 60. Right panel with ranges determined by values in the data set.

Figure 4 plots both the fixed range scatter plot and a scatter plot in which the range of values is determined by the values of the two variables independently. Code for producing the second plot requires first drawing all points in the dataset and then overlaying with the colour of the two datasets.

plot(dty$Deprivation0,dty$Accommodation,pch=16,xlab="Deprivation 0",ylab="Accommodation")

points(dty1$Deprivation0,dty1$Accommodation,pch=16,col="blue")

points(dty0$Deprivation0,dty0$Accommodation,pch=16,col="red")

leg.text=c("Below 33%","Above 33%")

legend("bottomleft",leg.text,pch=16,col=c("red","blue"))

Next we create the png in the same way as before

png("Deprivation0Accommodation.png")

plot(dty$Deprivation0,dty$Accommodation,pch=16,xlab="Deprivation 0",ylab="Accommodation") # Note the limits are set based on summary statistics

points(dty1$Deprivation0,dty1$Accommodation,pch=16,col="blue")

points(dty0$Deprivation0,dty0$Accommodation,pch=16,col="red")

leg.text=c("Below 33%","Above 33%")

legend("bottomleft",leg.text,pch=16,col=c("red","blue"))

dev.off()

Q4. What inference may be derived from Figure 4?

Q5. Repeat the analysis with Married and HealthVeryGood replacing Deprivation0 and Accommodation respectively. What inference can be taken from the resulting scatter plots?

Q6. Repeat the analysis with Deprivation0 and OwnedMortgage as the variables. What is the inference in this third pair of plots?

**Summary**

This first half of the practical session has sought to provide an overview of the dataset using the methods that we would commonly use in statistics. We have seen how to read the data into R, perform simple dataset manipulation and prepare a subset of data for analysis. We then created a summary statistics matrix and a correlation matrix. We outputted the summary statistics and correlation matrix to a csv file so that they may be easily viewed in Excel and then cut and pasted into Word. Finally, we visualised the data using scatter plots. Colouring according to whether the local authority district had a proportion of residents with a university degree or higher was greater than 33% allowed discussion of the relationship between the two axis variables and qualification levels. Part B will introduce Topological Data Analysis Ball Mapper in R and create plots which allow us to see the distribution of qualifications across the joint distribution of local authority district characteristics.