Rrrr - Software for Pirates

# Setup your environment

## Installing

Install R and R studio. Think of R studio as a kind of remote control for R which is the main “engine” that actually does all the work

## Working directory

It’s good practice .

As I said in class, you have to worry a bit about your working environment. The following chunck shows you how you could change that. However, I have simplified things a bit (from what you saw in class) to make it easier for to begin with.

For that to work you have to keep all your data files in a directory called that and all the code files (such as this one) in a directory that sits in the same sub directory. e.g. if you have a directory “data stories course” you want to have in there a directory called “data”" with the data files and a directory “code” with the code files.

Note that by default in a markdown document (i.e. an Rmd document…things are a bit different in a R document) the default active working directory is always the directory where the document is stored. So below when I load data, this is what I assume; e.g. I will load “../data/foreigners.dta”. Note that this tells the computer that to find the file you need to go one level up (that’s what “..” means) and then you go down again in the “data” directory

# Talking to R

Unlike a programme like Excel or Word where you do most things by clicking on menus, you interact with R primarily by issuing commands. The most basic way to that is by writing a command in the console window.

print("hello world")

## [1] "hello world"

7/6

## [1] 1.166667

when we print something shows up. You can use this as a pocket calculator

Let’s create some lists

1:10

## [1] 1 2 3 4 5 6 7 8 9 10

runif(10)

## [1] 0.4506879 0.6109290 0.5721838 0.9044007 0.4564178 0.6684749 0.7426940  
## [8] 0.5319009 0.4060820 0.4665630

seq(0,20,2)

## [1] 0 2 4 6 8 10 12 14 16 18 20

sample(1:6,5)

## [1] 3 5 1 4 2

#mydir=paste0(dropbox\_path,'/teaching/data stories') # set this according to what applies on your computer  
mydir=getwd()  
print(mydir)

## [1] "C:/Users/Ralf Martin/Dropbox/datastories/datastorieshub/code"

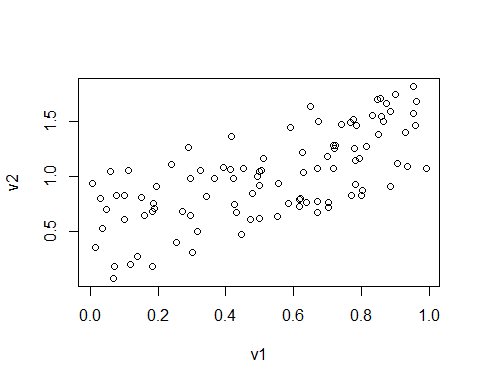
# e.g. mydir="c:/user/Rproject1/"  
setwd(mydir) # This is how you set a working directory in normal R code  
getwd()

## [1] "C:/Users/Ralf Martin/Dropbox/datastories/datastorieshub/code"

knitr::opts\_knit$set(root.dir = getwd(), verbose = TRUE) # This makes sure you keep the same directory in subsequent chuncks of code

# Some playing

v1=runif(100)  
 v2=runif(100)+v1  
   
   
 plot(v1,v2)



# Migrants

Since they gained power in 2010 the UK Tory Junta have led a relentless campaign of scapegoating immigrants for their policy failures. Part of this is the suggestion that increased levels of immigration lead to more crime. Here we explore to what extent there is actually any evidence in the data for this.

## Load the Data

getwd()

## [1] "C:/Users/Ralf Martin/Dropbox/datastories/datastorieshub/code"

library(haven) # make sure libraries such as this are installed. If not go to Tools -> Install Packages  
 setwd("..")  
  
 getwd()

## [1] "C:/Users/Ralf Martin/Dropbox/datastories/datastorieshub"

df=read.csv("../data/foreigners.csv")  
 setwd("../data")  
 getwd()

## [1] "C:/Users/Ralf Martin/Dropbox/datastories/data"

df=read.csv("foreigners.csv")  
   
   
#ralf<-read\_dta("../data/foreigners.dta")  
   
df=read.csv("https://www.dropbox.com/s/g1w75gkw7g91zef/foreigners.csv?dl=1")

some stuff I wrote…

print(3/7)

## [1] 0.4285714

## Prepare Data

newv=runif(20)  
 #df['test']=newv/df$pop11  
 summary(df)

## X crimes11 b\_migr11 pop11   
## Min. : 1.00 Min. : 1134 Min. : 2.241 Min. : 2203   
## 1st Qu.: 87.75 1st Qu.: 107618 1st Qu.: 4.899 1st Qu.: 94263   
## Median :174.50 Median : 160556 Median : 7.603 Median : 125746   
## Mean :174.50 Mean : 236988 Mean :11.226 Mean : 161434   
## 3rd Qu.:261.25 3rd Qu.: 309377 3rd Qu.:12.382 3rd Qu.: 200247   
## Max. :348.00 Max. :1714295 Max. :55.161 Max. :1072372   
## NA's :24 NA's :9 NA's :9   
## area   
## : 9   
## Adur : 1   
## Allerdale : 1   
## Amber Valley: 1   
## Arun : 1   
## Ashfield : 1   
## (Other) :334

summary(df$crimes11)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 1134 107618 160556 236988 309377 1714295 24

summary(df$pop11)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 2203 94263 125746 161434 200247 1072372 9

crimes11=df$crimes11  
   
 summary(crimes11)

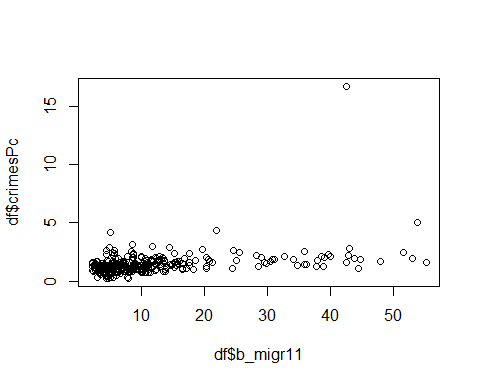
## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 1134 107618 160556 236988 309377 1714295 24

df['crimesPc']=df$crimes11/df$pop11

## Look at data

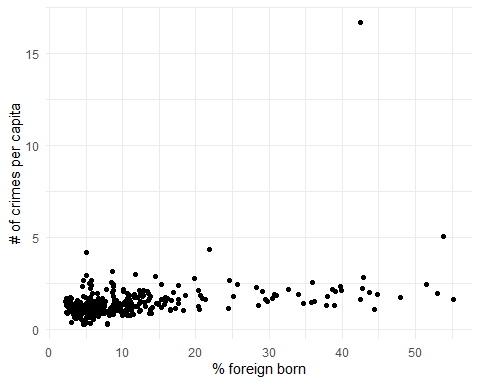
A scatter plot

plot(df$b\_migr11,df$crimesPc)  
  
  
library(ggplot2)



ggplot(df, aes(x = b\_migr11, y = crimesPc)) +  
 geom\_point()+theme\_minimal()+  
 xlab("% foreign born")+ylab("# of crimes per capita")

## Warning: Removed 24 rows containing missing values (geom\_point).



How about leaving the City of London out?

df[df$crimesPc>5,"area"]

## [1] <NA> <NA> <NA> <NA> <NA>   
## [6] <NA> <NA> <NA> <NA> <NA>   
## [11] <NA> <NA> <NA> <NA> <NA>   
## [16] City of London Westminster <NA> <NA> <NA>   
## [21] <NA> <NA> <NA> <NA> <NA>   
## [26] <NA>   
## 340 Levels: Adur Allerdale Amber Valley Arun Ashfield ... York

df[df$crimesPc>4.5,"area"]

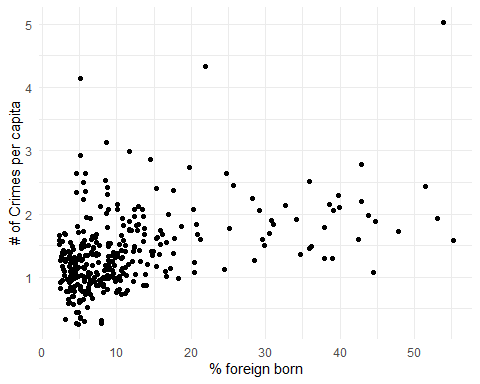
## [1] <NA> <NA> <NA> <NA> <NA>   
## [6] <NA> <NA> <NA> <NA> <NA>   
## [11] <NA> <NA> <NA> <NA> <NA>   
## [16] City of London Westminster <NA> <NA> <NA>   
## [21] <NA> <NA> <NA> <NA> <NA>   
## [26] <NA>   
## 340 Levels: Adur Allerdale Amber Valley Arun Ashfield ... York

df[df$crimesPc>4.5,c("area","crimesPc")]

## area crimesPc  
## NA <NA> NA  
## NA.1 <NA> NA  
## NA.2 <NA> NA  
## NA.3 <NA> NA  
## NA.4 <NA> NA  
## NA.5 <NA> NA  
## NA.6 <NA> NA  
## NA.7 <NA> NA  
## NA.8 <NA> NA  
## NA.9 <NA> NA  
## NA.10 <NA> NA  
## NA.11 <NA> NA  
## NA.12 <NA> NA  
## NA.13 <NA> NA  
## NA.14 <NA> NA  
## 285 City of London 16.690982  
## 317 Westminster 5.030381  
## NA.15 <NA> NA  
## NA.16 <NA> NA  
## NA.17 <NA> NA  
## NA.18 <NA> NA  
## NA.19 <NA> NA  
## NA.20 <NA> NA  
## NA.21 <NA> NA  
## NA.22 <NA> NA  
## NA.23 <NA> NA

dfx = df[df$crimesPc<10,]  
  
  
ggplot(dfx, aes(x = b\_migr11, y = crimesPc)) +  
 geom\_point()+theme\_minimal()+  
 xlab("% foreign born")+ylab("# of Crimes per capita")

## Warning: Removed 24 rows containing missing values (geom\_point).



What about a correlation?

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

#summary(ralf$crimes11)  
  
cor(df[,c("crimes11","crimesPc","b\_migr11")],use="complete.obs")

## crimes11 crimesPc b\_migr11  
## crimes11 1.0000000 0.3101539 0.4874797  
## crimesPc 0.3101539 1.0000000 0.3810661  
## b\_migr11 0.4874797 0.3810661 1.0000000

cor(dfx[,c("crimesPc","b\_migr11")],use="complete.obs")

## crimesPc b\_migr11  
## crimesPc 1.0000000 0.4331594  
## b\_migr11 0.4331594 1.0000000

df %>% select(crimesPc,b\_migr11) %>% cor(use="complete.obs")

## crimesPc b\_migr11  
## crimesPc 1.0000000 0.3810661  
## b\_migr11 0.3810661 1.0000000

library(corrr)  
test=df %>% select(crimesPc,b\_migr11)   
df %>% select(crimesPc,b\_migr11) %>% correlate()

##   
## Correlation method: 'pearson'  
## Missing treated using: 'pairwise.complete.obs'

## # A tibble: 2 x 3  
## rowname crimesPc b\_migr11  
## <chr> <dbl> <dbl>  
## 1 crimesPc NA 0.381  
## 2 b\_migr11 0.381 NA

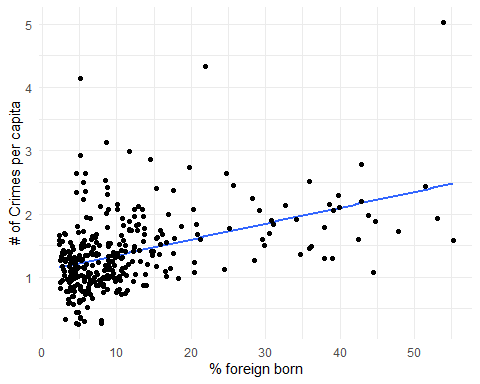
Or trend line?

library(ggplot2)  
ggplot(dfx, aes(x = b\_migr11, y = crimesPc)) +  
 geom\_smooth(method = "lm", se = FALSE) +  
 geom\_point()+theme\_minimal()+  
 xlab("% foreign born")+ylab("# of Crimes per capita")

## `geom\_smooth()` using formula 'y ~ x'

## Warning: Removed 24 rows containing non-finite values (stat\_smooth).

## Warning: Removed 24 rows containing missing values (geom\_point).



reg1=lm(crimesPc~b\_migr11,df)  
  
 reg1\_df=df[-reg1$na.action,] # get rid of missing values  
   
   
 summary(reg1)

##   
## Call:  
## lm(formula = crimesPc ~ b\_migr11, data = df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.5886 -0.3789 -0.1038 0.2046 14.0988   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.992957 0.079387 12.508 < 2e-16 \*\*\*  
## b\_migr11 0.037630 0.005088 7.396 1.23e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9674 on 322 degrees of freedom  
## (24 observations deleted due to missingness)  
## Multiple R-squared: 0.1452, Adjusted R-squared: 0.1426   
## F-statistic: 54.7 on 1 and 322 DF, p-value: 1.226e-12

summary(df$b\_migr11)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 2.241 4.899 7.603 11.226 12.382 55.161 9

#summary(lm(crimesPc~b\_migr11,dfx))

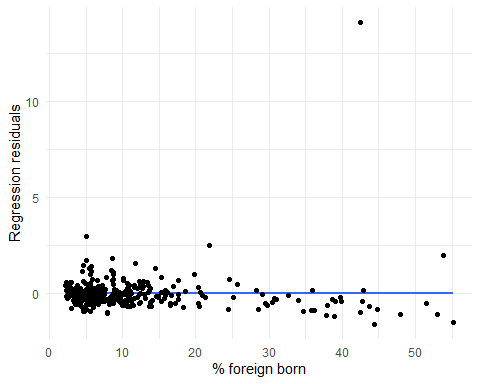
Notice that the residulas from a regression are not correlated

reg1\_df['residuals']= reg1$residuals  
   
  
 cor(reg1\_df[,c("residuals","b\_migr11")],use="complete.obs")

## residuals b\_migr11  
## residuals 1.000000e+00 -2.474525e-17  
## b\_migr11 -2.474525e-17 1.000000e+00

ggplot(reg1\_df, aes(x = b\_migr11, y = residuals)) +  
 geom\_smooth(method = "lm", se = FALSE) +  
 geom\_point()+theme\_minimal()+  
 xlab("% foreign born")+ylab("Regression residuals")

## `geom\_smooth()` using formula 'y ~ x'



# Some further perspectives on the data

Just some basic numbers

#library(psych)  
#describe(df$crimesPc)  
#describe(df$b\_migr11)  
  
library(Hmisc)

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

##   
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':  
##   
## src, summarize

## The following objects are masked from 'package:base':  
##   
## format.pval, units

Hmisc::describe(df[,c("b\_migr11","crimesPc")])

## df[, c("b\_migr11", "crimesPc")]   
##   
## 2 Variables 348 Observations  
## --------------------------------------------------------------------------------  
## b\_migr11   
## n missing distinct Info Mean Gmd .05 .10   
## 339 9 339 1 11.23 9.593 3.013 3.687   
## .25 .50 .75 .90 .95   
## 4.899 7.603 12.382 25.201 38.024   
##   
## lowest : 2.240523 2.279650 2.342227 2.361519 2.373362  
## highest: 47.886591 51.521316 53.043116 53.820812 55.161434  
## --------------------------------------------------------------------------------  
## crimesPc   
## n missing distinct Info Mean Gmd .05 .10   
## 324 24 324 1 1.425 0.7209 0.6406 0.7759   
## .25 .50 .75 .90 .95   
## 0.9905 1.2901 1.6582 2.1164 2.4465   
##   
## lowest : 0.2544352 0.2669489 0.2685578 0.3042361 0.3114384  
## highest: 3.1323595 4.1454223 4.3277037 5.0303810 16.6909820  
##   
## Value 0.2 0.4 0.6 0.8 1.0 1.2 1.4 1.6 1.8 2.0 2.2  
## Frequency 3 7 12 32 64 46 49 38 22 17 10  
## Proportion 0.009 0.022 0.037 0.099 0.198 0.142 0.151 0.117 0.068 0.052 0.031  
##   
## Value 2.4 2.6 2.8 3.0 3.2 4.2 4.4 5.0 16.6  
## Frequency 9 5 3 2 1 1 1 1 1  
## Proportion 0.028 0.015 0.009 0.006 0.003 0.003 0.003 0.003 0.003  
##   
## For the frequency table, variable is rounded to the nearest 0.2  
## --------------------------------------------------------------------------------

summary(df$b\_migr11)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 2.241 4.899 7.603 11.226 12.382 55.161 9

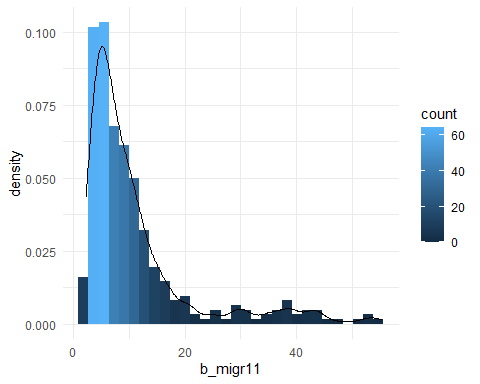
summary(df$crimesPc)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 0.2544 0.9905 1.2901 1.4251 1.6582 16.6910 24

library(scales)  
library(magrittr)  
  
  
  
ggplot(data = df, aes(b\_migr11)) +   
 geom\_histogram( bins=30, aes(y = ..density.., fill=..count..)) +  
 geom\_density()+theme\_minimal()

## Warning: Removed 9 rows containing non-finite values (stat\_bin).

## Warning: Removed 9 rows containing non-finite values (stat\_density).



ggplot(data = df, aes(crimesPc)) +   
 geom\_histogram( bins=30, aes(y = ..density.., fill=..count..)) +  
 geom\_density()+theme\_minimal()

## Warning: Removed 24 rows containing non-finite values (stat\_bin).

## Warning: Removed 24 rows containing non-finite values (stat\_density).

