



EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Exploratory data analysis

Cleaning and exploring the data

Vladimir Batagelj

IMFM Ljubljana and IAM UP Koper

Master's programme

Applied Statistics with Social Network Analysis

International Laboratory for Applied Network Research

NRU HSE, Moscow 2017

EDA, clean
and explore

V. Batagelj

Cleaning

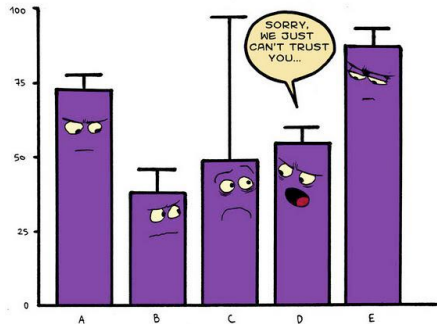
Exploring

Regression

Clustering

Solving the
clustering
problem

- 1 Cleaning
- 2 Exploring
- 3 Regression
- 4 Clustering
- 5 Solving the clustering problem



Vladimir Batagelj: vladimir.batagelj@fmf.uni-lj.si

Current version of slides (November 16, 2017 at 13:42): [slides PDF](#)



Cleaning the data

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

We collected the data in a CSV file. We can inspect them using a text editor or a spreadsheet program. We can also import them into R

```
> wdir <- "C:/Users/batagelj/Documents/papers/2017/Moscow/EDA/te
> setwd(wdir)
> T <- read.csv2("newBooks.csv", stringsAsFactors=FALSE)
> dim(T)
[1] 970 15
> nrow(T)
[1] 970
> ncol(T)
[1] 15
> head(T)
> tail(T)
> T[c(5, 9, 333), 1:8]
      bID      Amazon      bind npag      pub year      lang wid
5      5 0199206651 Hardcover   720   Oxford UP 2010 English 9.8
9      9 1473952123 Paperback   248          SAGE 2017 English 6.7
333 332 1546640010 Paperback    74 CreateSpace 2017 English 6
```



Cleaning and exploring the data

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

An informative view of a data frame is provided by the function `str`

```
> str(T)
'data.frame':   970 obs. of  15 variables:
 $ bID      : chr  "1" "2" "3" "4" ...
 $ Amazon  : chr  "0521840856" "0521387078" "1446247414" "0195379470" ...
 $ bind    : chr  "Hardcover" "Paperback" "Paperback" "Paperback" ...
 $ npag    : int   402 857 304 264 720 207 344 744 248 272 ...
 $ pub     : chr  "Cambridge University Press" "Cambridge University Press" "SAGE Publi
 $ year    : int   2004 1994 2013 2011 2010 2014 2005 2010 2017 2011 ...
 $ lang    : chr  "English" "English" "English" "English" ...
 $ wid     : chr  "6" "6" "7.3" "9.2" ...
 $ thi     : chr  "1.1" "1.5" "0.7" "0.7" ...
 $ hei     : chr  "9" "9" "9.1" "6.1" ...
 $ duni    : chr  "inches" "inches" "inches" "inches" ...
 $ weig    : chr  "1.4" "2.6" "1.4" "12.8" ...
 $ wuni    : chr  "pounds" "pounds" "pounds" "ounces" ...
 $ pric    : chr  "121.52" "52.41" "37.38" "20.75" ...
 $ titl    : chr  "Amazon.com: Generalized Blockmodeling (Structural Analysis in the Sc
```

The data obtained from our scraping program are “messy” – we need to *clean* them to be ready for analysis. This is true for most data obtained from different sources. After cleaning we *explore* the data to “get feeling” and ideas for analyses. Sometimes, if possible, we need to correct our scraping program and repeat the data collection. For larger data collections a test collection of a small sample is advised.

It is useful to preserve a copy of original raw data. Many problems can be resolved by correcting the original data in its copy. From the corrected data we construct a data frame (or some other structure) for analyses.



Cleaning the data

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Typical tasks in data cleaning

- correcting for unexpected values; consider extreme and influential units.
- normalization of values (dates in different formats; weights, money, lengths in different units; recategorization; unification: lower/upper case, nonASCII chars, ’; names (first, last)).
- factorization of ordinal and categorical variables.
- splitting variables (date \rightarrow year, month, day; name \rightarrow first, last).
- combining variables (year, month, day \rightarrow date).
- transforming variables (date \rightarrow day of week; Box-Cox (1, 2)).
- combining, adding data from other sources (geographical coordinates).
- dealing with missing data.



Missing data

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

There are different options to deal with missing data:

- do nothing, mark with NA.
- find the value and insert it.
- remove the unit (in creating clean data frame).
- impute a value (guess, mean value, random, nearest neighbor, interpolation)



Identity (entity resolution) problem

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

In dealing with data extracted from text sources we often encounter the *identity problem*. It has two parts:

- *equivalence* (different words/phrases representing the same term – synonyms); and
- *ambiguity* (same word/phrase representing different terms – homonyms).

When dealing with names of people that include Chinese the “*three Zhang, four Li*” effect can make it to the surface.

The problem can be partially solved using dictionaries, considering context, using tools like stemming and lemmatization, etc.

For cleaning of Amazon data see the [wiki page](#).



Amazon: old books – May 2012

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

```
> help(read.csv)
> getwd()
[1] "C:/Users/Batagelj/test/python/2012/amazon"
> setwd("C:/Users/Batagelj/test/python/2012/amazon")
> dat <- read.csv2("booksT.csv",header=FALSE,stringsAsFactors=FALSE)
> dim(dat)
[1] 16804      23
> names(dat)
 [1] "V1" "V2" "V3" "V4" "V5" "V6" "V7" "V8" "V9" "V10" "V11" "V12" "V13" "V14" "V15" "V16" "V17" "V18" "V19" "V20" "V21" "V22" "V23"
[16] "V16" "V17" "V18" "V19" "V20" "V21" "V22" "V23"
> dat[c(3,7),]
  V1 V2 V3      V4                                     V5
3  3 30 33 1451648537                                     Walter Isaacson          Steve J
7  7 53 60 140123206X Scott Snyder, Jock, Francesco Francavilla Batman: The Black Mir
  V7 V8          V9 V10 V11 V12
3 Simon & Schuster; First Edition ~1st Printing edition 2011 Hardcover 656 35.0 16.8
7                                     DC Comics 2011 Hardcover 304 29.99 16.
  V13
3 Biography/Autobiography$1955-2011$Biography$Businessmen$Computer engineers$Jobs, St
7 Comic books, strips, etc$Graphic novels$Comics & Graphic Novels$Comics & Graphic No
  V14 V15 V16 V17 V18 V19 V20 V21 V22 V23
3  26 27 28 29 30 31 27 32 26 33
7  54 55 56 57 58 59 54 55 56 60
>
-----
V1  index      V4  AmazonID      V7  publisher      V10  pages      V13  subject
V2  lenQ       V5  authors      V8  year          V11  listPrice   V14-V23  neighb
V3  lenK       V6  title        V9  binding      V12  price
-----
> year <- dat$V8
> summary(year)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
    0      2002    2008    1970    2011    2013     17
```




Amazon: data cleaning and exploration

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

```
> year <- dat$V8; pages <- dat$V10; binding <- dat$V9; price <- dat$V12
> isNA <- which(is.na(year)|is.na(pages)|is.na(binding)|is.na(price))
> year <- year[-isNA]; pages <- pages[-isNA]; binding <- binding[-isNA]
> typeof(price)
[1] "character"
> price <- as.numeric(price[-isNA])
> OK <- (0<pages)&(pages<2050) & (1900<year)&(year<2013) & (0<price)&(price<2000)
> table(OK)
OK
FALSE TRUE
1759 15028
> pages <- pages[OK]; binding <- binding[OK]; year <- year[OK]; price <- price[OK]
> bind <- rep(3,length(binding))
> bind[binding %in% c("Paperback", "Perfect Paperback", "Mass Market Paperback")] <-
> bind[binding %in% c("Hardcover", "Bonded Leather", "Leather Bound", "Hardcover-spiral")]
> table(bind)
> plot(density(pages))
> plot(density(year))
> plot(density(price[(0<price)&(price<60)]))
> plot(pages,price,col=c("red","blue","green")[bind],pch=16,cex=0.1)
```



Exploring the data

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Exploration phase of data analysis gives us an initial insight in the data – we get feeling about variables and their relations. It also provides hypotheses for further analyses.

We usually start the exploration by looking at each variable separately (univariate). Besides numerical characteristics we use also visualizations according to the type of variable.

Later we look to relations among variables (multivariate). The two main types of relations are association (regression) and grouping (clustering).



Basic data visualization in R

EDA, clean
and explore

V. Batagelj

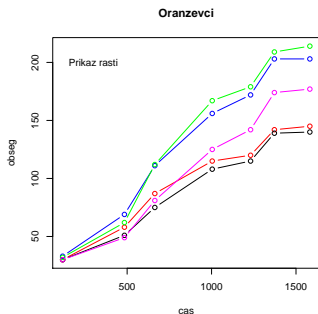
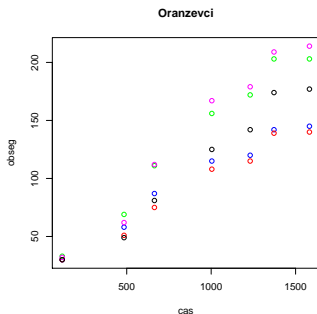
Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem



```
> help(plot)
> (c <- Orange[29:35,2])
[1] 118 484 664 1004 1231 1372 1582
> b <- c("red", "blue", "black", "green", "magenta")
> plot(Orange[,2], Orange[,3], col=b[Orange[,1]], xlab="cas", ylab="obseg",
      main="Oranzevci")
> plot(Orange[,2], Orange[,3], xlab="cas", ylab="obseg", main="Oranzevci",
      type="n")
> for(k in 1:5) {points(c, Orange[(7*k-6):(7*k), 3], col=b[k], type="b") }
> text(300, 200, "Prikaz rasti")
```



Marks

EDA, clean
and explore

V. Batagelj

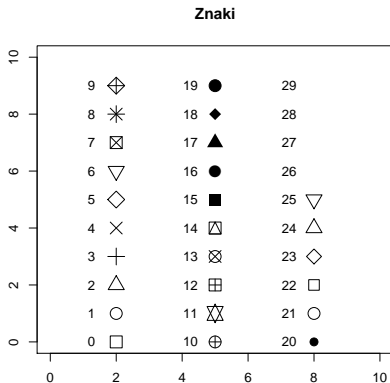
Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem



```
> plot(0:10,0:10,type="n",main="Znaki",xlab="",ylab="")
> k <- -1
> for(i in c(2,5,8)){for(j in 0:9){
  k <- k+1;text(i-0.75,j,k);points(i,j,pch=k,cex=2)}}
```





Colors

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem



Spectral (divergent)

```
> colors()
[1] "white"           "aliceblue"       "antiquewhite"
[655] "yellow3"        "yellow4"         "yellowgreen"
> library(RColorBrewer)
> display.brewer.pal(11, 'Spectral')
> help(rgb); help(palette); help(RColorBrewer)
```

Escaping RGBland



V. Batagelj

EDA, clean and explore



Categorical : numerical

EDA, clean
and explore

V. Batagelj

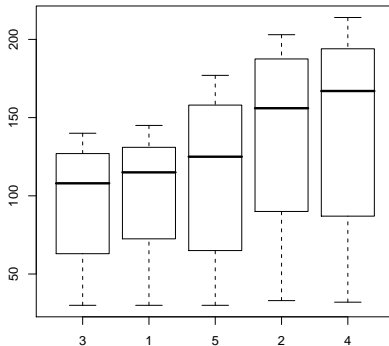
Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem



```
> plot(Orange$Tree, Orange$circumference)
```



Categorical : numerical

EDA, clean
and explore

V. Batagelj

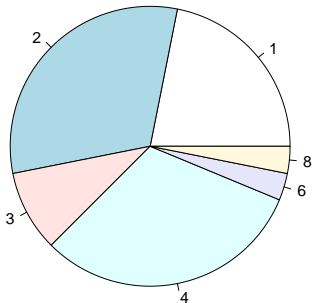
Cleaning

Exploring

Regression

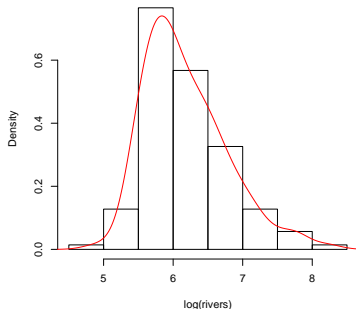
Clustering

Solving the
clustering
problem



```
> table(mtcars$carb)
 1  2  3  4  6  8
10 13 10 10  1  1
> barplot(table(mtcars$carb))
> pie(table(mtcars$carb))
```

Histogram of log(rivers)



```
> dotchart(table(mtcars$carb))
> stripchart(mtcars$carb,method="stack",pch=16)
> hist(log(rivers),prob=TRUE)
> lines(density(log(rivers)),col="red")
```



Different displays

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

```
> attach(faithful)
> hist(waiting)
> summary(waiting)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  43.0   58.0   76.0   70.9   82.0   96.0
> bins <- seq(42,109,by=10)
> bins
[1] 42  52  62  72  82  92 102
> freqs <- table(cut(waiting,bins))
> y <- c(0,freqs,0)
> x <- seq(37,107,by=10)
> plot(x,y,type="l")
> rug(waiting)
> hist(waiting,breaks="Scott",prob=TRUE,ylab="",main="Faithful")
> lines(density(waiting),col="blue",lwd=2)
> boxplot(rivers)
> plot(rev(rivers[order(rivers)]))
> boxplot(rivers)
> f <- fivenum(rivers)
> f
[1] 135  310  425  680 3710
> text(rep(1.3,5),f,labels=c("min","1/4","1/2","3/4","max"))
```




Relations among variables

EDA, clean
and explore

V. Batagelj

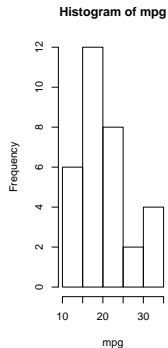
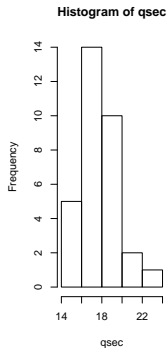
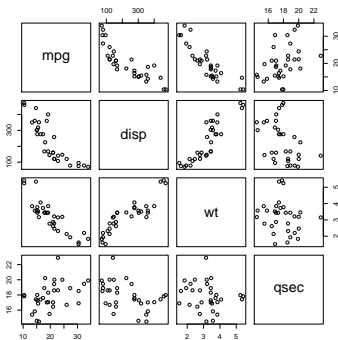
Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem



```
> attach(mtcars)
> pairs(mtcars[,c(1,3,6,7)])
> par(mfrow=c(1,2))
> hist(qsec,breaks="scott")
> hist(mpg,breaks="scott")
> par(mfrow=c(1,1))
```



Distribution using step function

EDA, clean
and explore

V. Batagelj

Cleaning

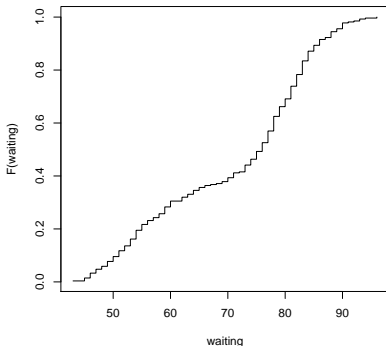
Exploring

Regression

Clustering

Solving the
clustering
problem

Porazdelitev waiting



```
> attach(faithful)
> n <- length(waiting)
> plot(sort(waiting), (1:n)/n, type="s", main="Porazdelitev waiting",
       xlab="waiting", ylab="F(waiting)")
> plot(ecdf(waiting)) # empirical cumulative distribution function
```



Distributions in R

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Most of the standard distributions is available in R as functions. For a distribution *dist* are: `d`*dist* – density $g(x)$, `p`*dist* – cumulative $F(x) = \int_{-\infty}^x g(t)dt$, `q`*dist* – inverse – quantile function $q = F^{-1}(p)$, `r`*dist* – random numbers distributed according to *dist*.

Example use *dist* (use `help`): `unif`, `beta`, `binom`, `cauchy`, `exp`, `chisq`, `f`, `gamma`, `geom`, `hyper`, `lnorm`, `logis`, `nbinom`, `norm`, `pois`, `signrank`, `t`, `weibull`, `wilcox`.

The function `sample` supports random sampling (`replace=TRUE`) from a given set.

EDA, clean and explore

V. Batagelj

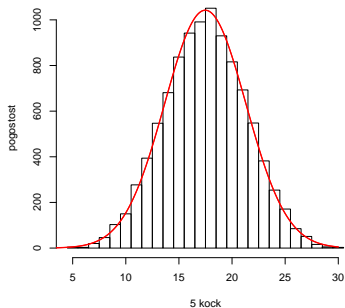
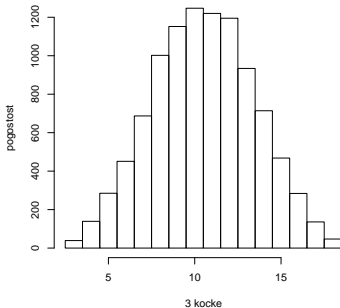
Cleaning

Exploring

Regression

Clustering

Solving the clustering problem



```
> a <- sample(1:6,replace=TRUE,10000); b <- sample(1:6,replace=TRUE,10000)
> c <- sample(1:6,replace=TRUE,10000); s <- a+b+c
> hist(s,breaks=2.5:18.5,xlab="3 kocke",ylab="pogostost",main="")
> d <- sample(1:6,replace=TRUE,10000); e <- sample(1:6,replace=TRUE,10000)
> s <- s+d+e; x <- seq(1,30,0.1)
> hist(s,breaks=4.5:30.5,xlab="5 kock",ylab="pogostost",main="")
> lines(x,dnorm(x,mean(s),sd(s))*10000,lwd=2,col="red")
```



Comparing distributions

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

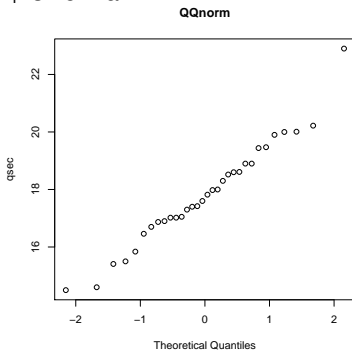
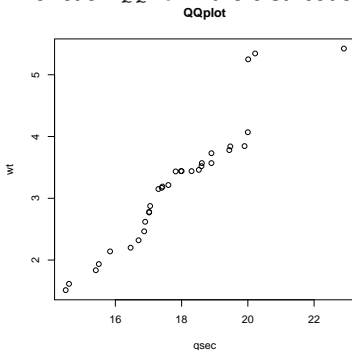
Regression

Clustering

Solving the
clustering
problem

QQplot consists of point (x, y) over the domains of distributions F_1 and F_2 , such that $F_1(x) = F_2(y)$. For equal distributions they lie on the diagonal.

In function `QQnorm` the distribution F_1 is normal.



```
> attach(mtcars)
> qqplot(qsec, wt, main="QQplot")
> qqnorm(qsec, ylab="qsec", main="QQnorm")
```



Models

EDA, clean
and explore

V. Batagelj

Cleaning

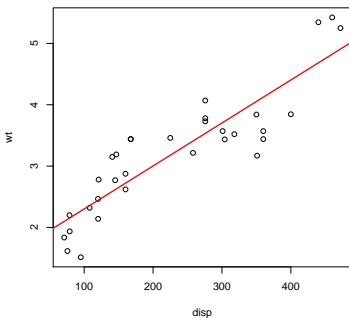
Exploring

Regression

Clustering

Solving the
clustering
problem

With an expression $y \sim f(x_1, x_2, \dots, x_k)$ we describe a *model* – relation between dependent variable and independent variables. There exist some functions that on the basis of data determine (parameters of) the function f optimizing some fit criterion: `lm`, `gam`, `loess`, `lowess`, ... The values of the model function in selected points are obtained using the function `predict`. The simplest model is the *regression* line:



```
> attach(mtcars)
> res <- lm(wt ~ disp)
> res[[1]]
(Intercept)      disp
1.599814597 0.007010325
> plot(wt ~ disp)
> abline(res,col="red",lwd=2)
> predict(res,list(dis=c(410,200)))
      1      2
4.474048 3.001880
```



Fitting the data

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

From the selected class of functions \mathcal{F} we would like to select one that fits the best our data (x_k, y_k) , $k \in I$. Let's denote it with $f(x, a)$. a are parameters. The error in a point (x_k, y_k) is equal to

$$y_k = f(x_k, a) + \varepsilon_k$$

These errors can be combined into a **total error** $E(f)$ in different ways

$$E_1(f) = \sum_k |\varepsilon_k|$$

$$E_2(f) = \sum_k \varepsilon_k^2$$

$$E_3(f) = \max_k |\varepsilon_k|$$

$$E_4(f) = \text{lik}(f) = \prod_k f(x_k, a), \quad f \text{ is a distribution}$$

First three min; E_4 max.





Fitting

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Instead with ε_k we can measure the point error also using some other quantities – *ortogonal error* ϱ_k .

For fitting distributions the *maximum likelihood* (E_4) is usually used..

For general functions the *least squares method* (E_2) is used. In many cases it allows to get the solution analitically. Its main weakness is that it is very sensitive to outliers. Using computers also other, more robust methods became an option.



Weighted fitting

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

$$E(a) = \sum_i w_i \varepsilon_i^2 = \sum_{i=1}^n w_i (f(x_i, a) - y_i)^2$$

Measurements with precision $y_i \pm \sigma_i$; then $\varepsilon'_i = \frac{\varepsilon_i}{\sigma_i}$

$$E'(a) = \sum_i (\varepsilon'_i)^2 = \sum_i \left(\frac{\varepsilon_i}{\sigma_i}\right)^2 = \sum_i \frac{1}{\sigma_i^2} \varepsilon_i^2$$

Therefore $w_i = \frac{1}{\sigma_i^2}$.

Relative error: $y_i = f(x_i)(1 + \delta_i)$

$$\delta_i = \frac{y_i - f(x_i)}{f(x_i)} \approx \frac{y_i - f(x_i)}{y_i} \Rightarrow w_i = \frac{1}{y_i^2}$$



Is there a functional relation between given variables?

EDA, clean and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the clustering problem

Let $p(X) = (p(x_i))_{i=1}^n$ be a discrete probability distribution. Its *entropy* is defined as

$$H(X) = - \sum_{i=1}^n p(x_i) \lg p(x_i)$$

where $\lg \equiv \log_2$ and $p = 0 \Rightarrow p \lg p = 0$.

It holds $0 \leq H(X) \leq \lg n$. For $p(x_k) = 1$; $p(x_i) = 0$, $i \neq k$ we have $H = 0$; and for $p(x_i) = \frac{1}{n}$, $i = 1, \dots, n$ we get $H = \lg n$. The *normalized entropy* $h(X) = \frac{H(X)}{\lg n}$ has values in $[0, 1]$.

For discrete variables X and Y with distributions $p(X)$ and $p(Y)$ and joint probability distribution $p(XY)$ their *information* is

$$I(X, Y) = \sum_{i=1}^n \sum_{j=1}^m p(x_i, y_j) \lg \frac{p(x_i, y_j)}{p(x_i)p(y_j)}$$

Considering $\sum_{j=1}^m p(x_i, y_j) = p(x_i)$ and $\sum_{i=1}^n p(x_i, y_j) = p(y_j)$ we get

$$I(X, Y) = H(X) + H(Y) - H(XY)$$



Raiski's coefficient

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Information $I(X, Y)$ has value 0 iff we have for all pairs $p(x_i, y_j) = p(x_i)p(y_j)$ – X and Y are independent.

The other extreme is attained iff X and Y are functionally related – in each row and each column of the distribution there is at most one nonempty cell, $H(X) = H(Y) = H(XY) = I(X, Y)$.

In 1964 Raiski introduced a coefficient

$$R(X \leftrightarrow Y) = \frac{I(X, Y)}{H(XY)} \quad \text{or in directed version} \quad R(X \rightarrow Y) = \frac{I(X, Y)}{H(Y)}$$

Both take values in $[0, 1]$ and have value 0 when X and Y are independent

$R(X \rightarrow Y) = 1$, when Y is a function of X ; $R(X \leftrightarrow Y) = 1$, when the variables are linked one-to-one.

The Raiski's coefficient is defined for **all types of scales**.



Power law (Zipf, Lotka, Pareto)

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

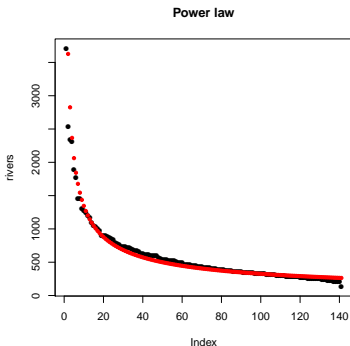
Clustering

Solving the
clustering
problem

The model function is selected in different ways: availability of a tool, simplification, guess – similarity to a curve on the picture, on theoretical basis (laws in the field), etc.

In double-logarithmic scale a *power law* curve is a line. Therefore we can determine its coefficients (little cheating) using the regression line:

```
> plot(rev(sort(rivers)))
> plot(rev(sort(rivers)), log="xy")
> x <- log(1:length(rivers))
> y <- log(rev(sort(rivers)))
> plot(y ~ x)
> rp <- lm(y ~ x)
> (a <- rp[[1]])
(Intercept)          x
 8.6233680    -0.6160568
> abline(rp, col="red", lwd=2)
> plot(rev(sort(rivers)), ylab="rivers",
+      pch=16, main="Power law")
> pow <- function(x){exp(a[1])*x^a[2]}
> x <- 1:length(rivers)
> y <- pow(x)
> points(x, y, pch=20, col="red")
```





Nonparametric smoothing / Boston

EDA, clean
and explore

V. Batagelj

Cleaning

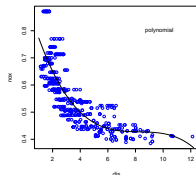
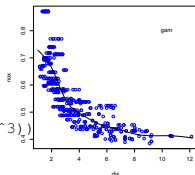
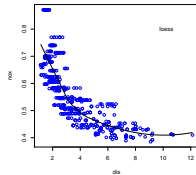
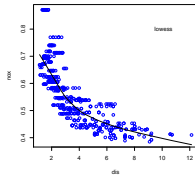
Exploring

Regression

Clustering

Solving the
clustering
problem

```
> library(MASS); attach(Boston)
> pairs(Boston)
> plot(dis,nox); s <- order(dis)
> plot(dis,nox,col="blue")
> lines(dis[s],nox[s])
> par(mfrow=c(2,2),cex=0.5)
> plot(dis,nox,col="blue")
> text(11,0.8,"lowess",pos=2)
> lines(lowess(dis,nox))
> plot(dis,nox,col="blue")
> text(11,0.8,"loess",pos=2)
> model <- loess(nox ~ dis)
> x <- seq(1,12.2,0.05)
> y <- predict(model,data.frame(dis=x))
> lines(x,y)
> plot(dis,nox,col="blue")
> text(11,0.8,"gam",pos=2)
> library(mgcv)
> model <- gam(nox ~ s(dis))
> y <- predict(model,list(dis=x))
> lines(x,y)
> plot(dis,nox,col="blue")
> text(11,0.8,"polynomial",pos=2)
> model <- lm(nox ~ dis+I(dis^2)+I(dis^3))
> y <- predict(model,list(dis=x))
> lines(x,y)
> par(mfrow=c(1,1),cex=1)
```





Fitting OECD data $pcinc \sim agr$

EDA, clean
and explore

V. Batagelj

OECD data

```
> oecd <- read.table("OECD.dat", header=TRUE)
> pairs(oecd); attach(oecd)
> plot(agr, pcinc, pch="+")
> # linear regression
> lin <- lm(pcinc ~ agr)
> abline(lin, col="green")
> lp <- lin$coef[2]*agr + lin$coef[1]
> sum((lp - pcinc)^2)
> # exponential with linear regression
> pi <- log(pcinc); m <- lm(pi ~ agr)
> b <- exp(m$coef[1]); a <- exp(m$coef[2])
> pl <- function(x){b*a^x}
> points(agr, pl(agr), col="red", pch=16)
> # least squares
> f <- function(t, p){a <- p[1]; b <- p[2]; b*a^t}
> E <- function(p){d <- f(agr, p) - pcinc; sum(d^2)}
> p0 <- c(a, b); best <- optim(p0, E)
> E(p0)
> best
> pr <- function(x){f(x, best$par)}
> points(agr, pr(agr), col="blue", pch=16)
> d <- seq(0, 84, 2); lines(spline(d, pr(d)), col="blue")
```



Fitting OECD

EDA, clean and explore

V. Batagelj

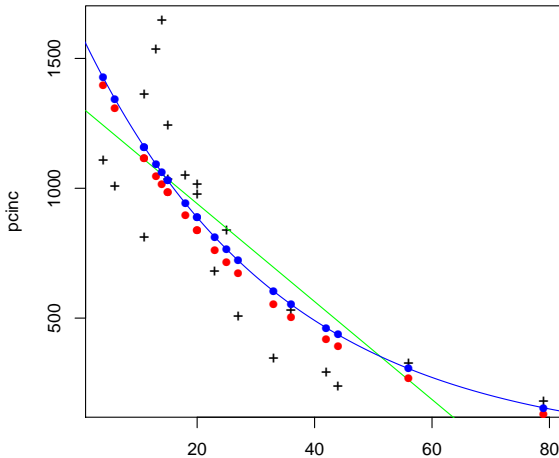
Cleaning

Exploring

Regression

Clustering

Solving the clustering problem



V. Batagelj

EDA, clean and explore



Clustering

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Given a set of units \mathcal{U} the clustering is a process of organizing units into groups – clusters of similar units. In real life clustering problems we have to deal with different their characteristics:

- description of units: vectors (types of measurement scales, number of variables, missing values, . . .) or structured units;
- size of the set of units;
- structure of units "space" (density, shapes of clusters).

A recent survey on clustering is given in [?].



Clustering and optimization

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

We approach the clustering problem as an optimization problem over the set of *feasible* clusterings Φ_k – partitions of units into k clusters. A cluster is a nonempty subset of the set of unit \mathcal{U} . The *criterion function* has the following form

$$P(\mathbf{C}) = \sum_{C \in \mathbf{C}} p(C)$$

The *total error* $P(\mathbf{C})$ of the clustering $\mathbf{C} = \{C_1, C_2, \dots, C_k\}$ is a sum of *cluster errors* $p(C)$.

There are many possibilities how to express the cluster error $p(C)$. Here we shall assume a model in which the error of a cluster is a sum of differences of its units from the cluster's *representative* T

$$p(C, T) = \sum_{X \in C} d(X, T)$$

Note that in general the representative needs not to be from the same "space" (set) as units.



Representatives, dissimilarities

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

The best representative is called a *leader*

$$T_C = \operatorname{argmin}_T p(C, T)$$

Then we define

$$p(C) = p(C, T_C) = \min_T \sum_{X \in C} d(X, T)$$

In most cases we express the cluster error in terms of a *dissimilarity* between units $d(X, Y)$; $d(X, X) = 0$ and $d(X, Y) = d(Y, X)$.

Another example of cluster error is a diameter

$$p(C) = \operatorname{diam}(C) = \max_{X, Y \in C} d(X, Y)$$



Dissimilarities on \mathbb{R}^m / examples 1

EDA, clean and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the clustering problem

n	measure	definition	range	note
1	Euclidean	$\sqrt{\sum_{i=1}^m (x_i - y_i)^2}$	$[0, \infty)$	$M(2)$
2	Sq. Euclidean	$\sum_{i=1}^m (x_i - y_i)^2$	$[0, \infty)$	$M(2)^2$
3	Manhattan	$\sum_{i=1}^m x_i - y_i $	$[0, \infty)$	$M(1)$
4	rook	$\max_{i=1}^m x_i - y_i $	$[0, \infty)$	$M(\infty)$
5	Minkowski	$\sqrt[p]{\sum_{i=1}^m (x_i - y_i)^p}$	$[0, \infty)$	$M(p)$



Dissimilarities on \mathbb{R}^m / examples 2

EDA, clean and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the clustering problem

n	measure	definition	range	note
6	Canberra	$\sum_{i=1}^m \frac{ x_i - y_i }{ x_i + y_i }$	$[0, \infty)$	
7	Heincke	$\sqrt{\sum_{i=1}^m \left(\frac{ x_i - y_i }{ x_i + y_i }\right)^2}$	$[0, \infty)$	
8	Self-balanced	$\sum_{i=1}^m \frac{ x_i - y_i }{\max(x_i, y_i)}$	$[0, \infty)$	
9	Lance-Williams	$\frac{\sum_{i=1}^m x_i - y_i }{\sum_{i=1}^m x_i + y_i}$	$[0, \infty)$	
10	Correlation c.	$\frac{\text{cov}(X, Y)}{\sqrt{\text{var}(X)\text{var}(Y)}}$	$[1, -1]$	



(Dis)similarities on \mathbb{B}^m / examples

EDA, clean and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the clustering problem

Let $\mathbb{B} = \{0, 1\}$. For $X, Y \in \mathbb{B}^m$ we define $a = XY$, $b = X\bar{Y}$, $c = \bar{X}Y$, $d = \bar{X}\bar{Y}$. It holds $a + b + c + d = m$. The counters a, b, c, d are used to define several (dis)similarity measures on binary vectors.

In some cases the definition can yield an indefinite expression $\frac{0}{0}$. In such cases we can restrict the use of the measure, or define the values also for indefinite cases. For example, we extend the values of Jaccard coefficient such that $s_4(X, X) = 1$. And for Kulczynski coefficient, we preserve the relation $T = \frac{1}{s_4} - 1$ by

$$s_4 = \begin{cases} 1 & d = m \\ \frac{a}{a+b+c} & \text{otherwise} \end{cases} \quad s_3^{-1} = T = \begin{cases} 0 & a = 0, d = m \\ \infty & a = 0, d < m \\ \frac{b+c}{a} & \text{otherwise} \end{cases}$$

We transform a similarity s from $[1, 0]$ into dissimilarity d on $[0, 1]$ by $d = 1 - s$.

For details see Batagelj, Bren (1995).



(Dis)similarities on \mathbb{B}^m / examples 1

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

n	measure	definition	range
1	Russel and Rao (1940)	$\frac{a}{m}$	$[1, 0]$
2	Kendall, Sokal-Michener (1958)	$\frac{a+d}{m}$	$[1, 0]$
3	Kulczynski (1927), T^{-1}	$\frac{a}{b+c}$	$[\infty, 0]$
4	Jaccard (1908)	$\frac{a}{a+b+c}$	$[1, 0]$
5	Kulczynski	$\frac{1}{2} \left(\frac{a}{a+b} + \frac{a}{a+c} \right)$	$[1, 0]$
6	Sokal & Sneath (1963), un_4	$\frac{1}{4} \left(\frac{a}{a+b} + \frac{a}{a+c} + \frac{d}{d+b} + \frac{d}{d+c} \right)$	$[1, 0]$
7	Driver & Kroeber (1932)	$\frac{a}{\sqrt{(a+b)(a+c)}}$	$[1, 0]$
8	Sokal & Sneath (1963), un_5	$\frac{ad}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$	$[1, 0]$



(Dis)similarities on \mathbb{B}^m / examples 2

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

n	measure	definition	range
9	Q_0	$\frac{bc}{ad}$	$[0, \infty]$
10	Yule (1927), Q	$\frac{ad-bc}{ad+bc}$	$[1, -1]$
11	Pearson, ϕ	$\frac{ad-bc}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$	$[1, -1]$
12	$-bc -$	$\frac{4bc}{m^2}$	$[0, 1]$
13	Baroni-Urbani, Buser (1976), S^{**}	$\frac{a+\sqrt{ad}}{a+b+c+\sqrt{ad}}$	$[1, 0]$
14	Braun-Blanquet (1932)	$\frac{a}{\max(a+b, a+c)}$	$[1, 0]$
15	Simpson (1943)	$\frac{a}{\min(a+b, a+c)}$	$[1, 0]$
16	Michael (1920)	$\frac{4(ad-bc)}{(a+d)^2+(b+c)^2}$	$[1, -1]$



Dissimilarities between sets

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Let \mathcal{F} be a finite family of subsets of the finite set U ; $A, B \in \mathcal{F}$ and let $A \oplus B = (A \setminus B) \cup (B \setminus A)$ denotes the symmetric difference between A and B .

The 'standard' dissimilarity between sets is the *Hamming distance*:

$$d_H(A, B) := \text{card}(A \oplus B)$$

Usually we normalize it $d_h(A, B) = \frac{1}{M} \text{card}(A \oplus B)$. One normalization is $M = \text{card}(U)$; the other $M = m_1 + m_2$, where m_1 and m_2 are the first and the second largest value in $\{\text{card}(X) : X \in \mathcal{F}\}$.

Other dissimilarities

$$d_s(A, B) = \frac{\text{card}(A \oplus B)}{\text{card}(A) + \text{card}(B)} \quad d_u(A, B) = \frac{\text{card}(A \oplus B)}{\text{card}(A \cup B)}$$

$$d_m(A, B) = \frac{\max(\text{card}(A \setminus B), \text{card}(B \setminus A))}{\max(\text{card}(A), \text{card}(B))}$$

For all these dissimilarities $d(A, B) = 0$ if $A = B = \emptyset$.



Problems with dissimilarities

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Functions in R: `dist`, `cluster/daisy`

What to do in the case of *mixed units* (with variables measured in different types of scales)?

- conversion to a common scale
- compute the dissimilarities on homogeneous parts and combine them (Gower's dissimilarity)

Fairness of dissimilarity – all variables contribute equally.
Approaches: use of normalized variables, analysis of dependencies among variables.



Gower's dissimilarity

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

the Gower dissimilarity coefficient for a mix of variables

$$d_{ij} = \sum_{v=1}^m \frac{\delta_{ijv} d_{ijv}}{\sum_{i=1}^m \delta_{ijv}}$$

where δ_{ijv} is a binary indicator equal to one whenever both observations i and j are nonmissing for variable v , and zero otherwise. Observations with missing values are not included.

For binary and nominal variables v , $d_{ijv} = 0$ if $x_{iv} = x_{jv}$; and $d_{ijv} = 1$ otherwise.

Ordinal variables v are considered as categorical ordinal variables and the values are substituted with the corresponding position index, r_{iv} in the factor levels. These position indexes are transformed in the following manner $z_{iv} = \frac{r_{iv}-1}{\max_k r_{kv}-1}$. These new values, z_{iv} , are treated as observations of an interval scaled variable.

For continuous variables v ,

$$d_{ijv} = \frac{|x_{iv} - x_{jv}|}{\max_k(x_{kv}) - \min_k(x_{kv})}$$

d_{ijv} is set to 0 if $\max_k(x_{kv}) = \min_k(x_{kv})$.

Functions `cluster/daisy` and `StatMatch/gower.dist`.



Solving the clustering problem

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Finite - solution always exists, but in most cases algorithmically hard problem → heuristics.

- hierarchical
 - agglomerative methods (`hclust`, `cluster/agnes`, `amap/hcluster`, `amap/hclusterpar`)
 - divisive methods (`cluster/diana`, `cluster/mona`)
 - adding methods
- local optimization (leaders method) (`kmeans`, `cluster/pam`, `cluster/clara`, `cluster/fanny`)
- linear algebra methods
- graph theory methods
- other methods (`mclust/Mclust`, `fpc/dbscan`, `dbscan/dbscan`, `factoextra/hkmeans`)



Acronyms

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Agnes - Agglomerative Nesting

Diana - Divisive Analysis

PAM - Partitioning around medoids

CLARA - Clustering Large Applications

hkmeans - Hierarchical K-means

FANNY - Fuzzy analysis clustering

Mclust - Model based clustering

DBSCAN - Density-Based Clustering



Leaders method

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Leaders method is a generalization of a popular nonhierarchical clustering k-means method. The idea is to get "optimal" clustering into a pre-specified number of clusters with the following iterative procedure:

determine an initial clustering

repeat

determine leaders of the clusters in the current clustering;
assign each unit to the nearest new leader – producing a
new clustering

until the leaders stabilize.



Hierarchical agglomerative clustering

The *hierarchical agglomerative clustering* procedure is based on a step-by-step merging of the two closest clusters.

each unit forms a cluster: $\mathbf{C}_n = \{\{X\}: X \in \mathcal{U}\}$;

they are at level 0: $h(\{X\}) = 0, X \in \mathcal{U}$;

for $k = n - 1$ **to** 1 **do**

 determine the closest pair of clusters

$(u, v) = \operatorname{argmin}_{i,j: i \neq j} \{D(C_i, C_j) : C_i, C_j \in \mathbf{C}_{k+1}\}$;

 join the closest pair of clusters $C_{(uv)} = C_u \cup C_v$

$\mathbf{C}_k = (\mathbf{C}_{k+1} \setminus \{C_u, C_v\}) \cup \{C_{(uv)}\}$;

$h(C_{(uv)}) = D(C_u, C_v)$

 determine the dissimilarities $D(C_{(uv)}, C_s), C_s \in \mathbf{C}_k$

endfor

\mathbf{C}_k is a partition of the finite set of units \mathcal{U} into k clusters.

The level $h(C)$ of the cluster $C_{(uv)} = C_u \cup C_v$.





Methods

EDA, clean
and explore

V. Batagelj

Cleaning

Exploring

Regression

Clustering

Solving the
clustering
problem

Hierarchical methods differ in selection of a between cluster dissimilarity D :

- **single linkage:** $D(C_p, C_q) = \min_{X \in C_p, Y \in C_q} d(X, Y)$
- **complete linkage:** $D(C_p, C_q) = \max_{X \in C_p, Y \in C_q} d(X, Y)$
- **Ward:** $D(C_p, C_q) = \frac{n_p \cdot n_q}{n_p + n_q} d(T_p, T_q)$
- see `help` and [paper](#)