EUROPEAN COMMISSION
Community Research

SEVENTH FRAMEWORK PROGRAMME

AMELI

Advanced Methodology for European Laeken Indicators

Deliverable 8.2

Visualisation Tools

Version: 2011

Matthias Templ, Beat Hulliger, Andreas Alfons, Daniela Lussmann, Peter Filzmoser

The project **FP7–SSH–2007–217322 AMELI** is supported by European Commission funding from the Seventh Framework Programme for Research.

http://ameli.surveystatistics.net/

Contributors to Deliverable 10.3

Chapter 1: M. Templ

Chapter 2: D. Lussmann, B. Hulliger, M. Templ

Chapter 3: B. Hulliger, S. Zechner, M. Templ, P. Filzmoser

Chapter 4: A. Alfons, M. Templ

Chapter 6: A. Kowarik, B. Meindl, M. Templ, S. Zechner

Chapter 7: A. Alfons

Chapter 8: M. Templ, A. Alfons, P. Filzmoser

Main Responsibility

Matthias Templ (Workpackage leader)
Department of Statistics and Probability Theory
Vienna University of Technology
Wiedner Hauptstr. 8–10, 1040 Vienna, Austria
E-Mail: templ@statistik.tuwien.ac.at
Telephone: +43 58801 10715

Evaluators

Timo Alanko (*Statistics Finland*)

Aims and Objectives of Deliverable 8.2

Visualization of data and exploratory data analysis become more and more important as a tool for analysing micro data, tables or indicators. The aim is to research and implement visualization tools

- for visualising indicators in order to support policy decisions,
- for highlighting selected/special data (e.g. outlying and influential observations, non-response, imputed values),
- for the visualisation of simulation results,
- for visualizing regional indicators in maps, as well as
- for visualisation for a better understanding by the end user of the indicator values including their quality.

Contents

Chapter 1

Introduction

The aim of work package 8 of the AMELI project is the development of visualization tools. Basically, there are three main directions for the development of visualization methods:

- *Visualization of indicators:* This plays a central role in the AMELI project. A quick visual unterstanding of indicators and their development (e.g. over time) is an important step in supporting policy makers (see also Münnich et al., 2011).

- *Visualization of data quality:* The data quality of microdata is influential to all derived statistical summaries. Accordingly, the visualization of measures of data quality is an important first step in the analysis of microdata. Poor data quality may occur by data outliers, erroneous data or missing values. Identifying and highlighting outliers and suspect observations thus belongs to the goal of these visualization methods. A visualization of the structure of missing values is another purpose of the developed tools.

- *Visualization of simulation results:* Statistical simulation under various scenarios formed a major part of the AMELI project (see Hulliger et al., 2011; Alfons et al., 2011). Summarizing the simulation results by means of appropriate graphics thus needs the development of such visualization methods.

Structure of the Paper: Section 2 shows the state-of-the-art in visualising indicators. Various plotting techniques are discussed, focusing on the visualization of the univariate and the multivariate information. In Section 3, the so called evaluation plot is revisited and modified. This plot evaluates the values of a given time series if they follow a specified trend. Policy maker thus can immediately see if the development of an indicator over time follows the expected trend, or if it deviates significantly. Some problems in mapping are discussed in Section 4 as well as estimating correlations of short time series and presenting the information in maps is discussed. The combination of thematic maps and traditional representations is presented as the newly developed checkerplot in Section 5. Section 6 shows the application of package `sparkTable` (Kowarik et al., 2010) to social inclusion indicators to produce graphical tables.

While the previous sections were focused on visualizing indicators, Section 7 is on visualizing indicators estimated within a simulation study. This section shows how simulation results from the simFrame R-package (Alfons et al., 2010) are visualised by selecting automatically suitable plot methods depending on the setup of the simulation study.

The final section, Section 8 shows R-package VIM (Templ et al., 2010; Templ and Filzmoser, 2008) which is on visualizing missing values in microdata. In addition, the visualisation of imputed values is discussed and examples are given.

Bibliography

Alfons, A., **Burgard, J. P.**, **Filzmoser, P.**, **Hulliger, B.**, **Kolb, J.-P.**, **Kraft, S.**, **Münnich, R.**, **Schoch, T.** and **Templ, M.** (**2011**): *The AMELI Simulation Study.* Research Project Report WP6 – D6.1, FP7-SSH-2007-217322 AMELI.
URL http://ameli.surveystatistics.net

Alfons, A., **Templ, M.** and **Filzmoser, P.** (**2010**): *An Object-Oriented Framework for Statistical Simulation: The R Package simFrame.* Journal of Statistical Software, 37 (3), pp. 1–36.
URL http://www.jstatsoft.org/v37/i03/

Hulliger, B., **Alfons, A.**, **Bruch, C.**, **Filzmoser, P.**, **Graf, M.**, **Kolb, J.-P.**, **Lehtonen, R.**, **Lussmann, D.**, **Meraner, A.**, **Münnich, R.**, **Nedyalkova, D.**, **Schoch, T.**, **Templ, M.**, **Valaste, M.**, **Veijanen, A.** and **Zins, S.** (**2011**): *Report on the Simulation Results.* Research Project Report WP7 – D7.1, FP7-SSH-2007-217322 AMELI.
URL http://ameli.surveystatistics.net

Kowarik, A., **Meindl, B.** and **Zechner, S.** (**2010**): sparkTable: Sparklines and graphical tables for tex and html. R package version 0.1.3.
URL http://CRAN.R-project.org/package=sparkTable

Münnich, R., **Alfons, A.**, **Bruch, C.**, **Filzmoser, P.**, **Graf, M.**, **Hulliger, B.**, **Kolb, J.-P.**, **Lehtonen, R.**, **Lussmann, D.**, **Meraner, A.**, **Myrskylä, M.**, **Nedyalkova, D.**, **Schoch, T.**, **Templ, M.**, **Valaste, M.**, **Veijanen, A.** and **Zins, S.** (**2011**): *Policy Recommendations and Methodological Report.* Research Project Report WP10 – D10.1/D10.2, FP7-SSH-2007-217322 AMELI.
URL http://ameli.surveystatistics.net

Templ, M., **Alfons, A.** and **Kowarik, A.** (**2010**): **VIM**: Visualization and Imputation of Missing Values. R package version 1.4.
URL http://cran.r-project.org/package=VIM

Templ, M. and **Filzmoser, P.** (**2008**): *Visualization of missing values using the R-package VIM*. Research report CS-2008-1, Department of Statistics and Probability Theory, Vienna University of Technology.
URL http://www.statistik.tuwien.ac.at/forschung/CS/CS-2008-1complete.pdf

Chapter 2

State-of-the-art in Visualisation of Indicators

Introduction

Indicators have different meaning for different groups of people like statistician, politician or 'normal' inhabitants. The name 'indicator' already expresses the objective, namely to indicate relevant information out of a complex data set, which can appear in ecology (e.g. ozone), economics (e.g. inflation), sociology (e.g. poverty rate) or any other field of science.

An indicator can tell us something about the current state or how far away a value is from a defined target. It is a small peace of information, which somehow summarize some characteristics of a complex data sets. Indicators are not made to tell us everything, but they help for guidance policy decisions.

Often, indicators are changing over time, so they should be visualized by numbers or graphics, which are made understandable for most people, who do not have the background of scientists. In other words, people should see the big picture by looking on a small part of the information.

The *International Institute for Sustainable Development* (IISD)[1] defines an indicator as follows:

> *An indicator quantifies and simplifies phenomena and helps us understand complex realities. Indicators are aggregates of raw and processed data but they can be further aggregated to form complex indices.*

Another definition is given by the *Organisation for Economic Co-operation and Development* (OECD)[2]

[1] www.iisd.org

[2] www.oecd.org

> *An indicator is a parameter, or a value derived from parameters, which points to, provides information about or describes the state of a phenomenon/environment/area, with a significance extending beyond that directly associated with a parameter value.*

In most countries institutions such as the national statistical offices (e.g. Statistics Austria), different ministries or others are responsible for collecting data and publishing indicators from these data in both ways, as tables and as graphics. Nowadays institutions of different countries are working together in international programs. These bilateral relationships make it easier to develop new methods and to standardize them. Visualisation then may show the differences of estimates between countries.

Selecting indicators: Indicators can be used on global, international or national levels. At the international level different organisations collect data from various countries, like the OECD or the EU. At national levels the statistical offices are doing the same within their country.

Some points which should be considered if choosing an indicator are now explained:

- Indicators should provide easy understandable information about the condition and result of the measured data. For example, if the wanted result is a reduction of poverty, achievement would be best measured by an outcome indicator, such as the at-risk-of-poverty rate.
- Over time the definition of the indicator should stay the same, and so should the process of collecting data. Decision makers have to be certain that the data they are looking at did not change over time. The data has to be collected frequently enough to be useful, most data is available on an annual base. Can new methods be developed to collect those data more cost effective?
- Is this indicator important to most people? Published indicators should have a high credibility. They are providing information which is easy to understand and accepted by decision makers. Highly technical indicators which require numerous explanations may not be useful for most people, but only for those who are working closer in the program for which the indicator is needed.
- Wrong or poorly measured indicators can lead decision makers down to a wrong path, faster than they would be without indicators. Indicators only represent a part of the picture, that should be kept in mind.
- The data quality is very important. Without good data the calculation or estimation of indicators may be biased, which could lead to wrong decisions.

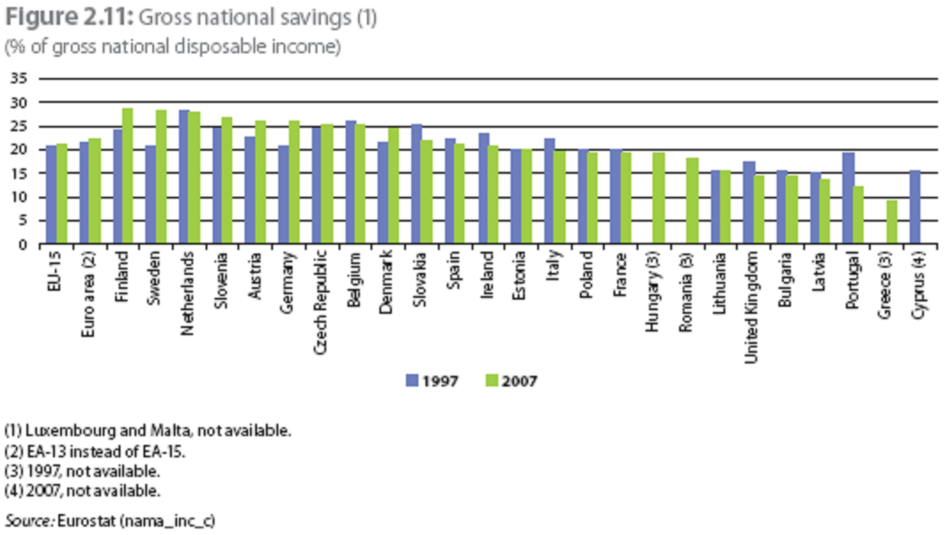


Figure 2.11: Gross national savings (1)
(% of gross national disposable income)

(1) Luxembourg and Malta, not available.
(2) EA-13 instead of EA-15.
(3) 1997, not available.
(4) 2007, not available.

Source: Eurostat (nama_inc_c)

Figure 2.1: Standard double bar chart from EUROSTAT (2009)

Visualizing Indicators: After the indicators are estimated, the final step is the presentation of the results to the general public, in various media like newspapers, or to policy decision makers.

The indicators should be visualized clearly to be understandable for everyone, awake interest and be graphically appealing. Clear to understand can also be expressed as simplicity, technical symbols or too many details will most likely confuse the audience. In which direction the chosen indicator is heading should also be clear.

But why do we want to visualize indicators?

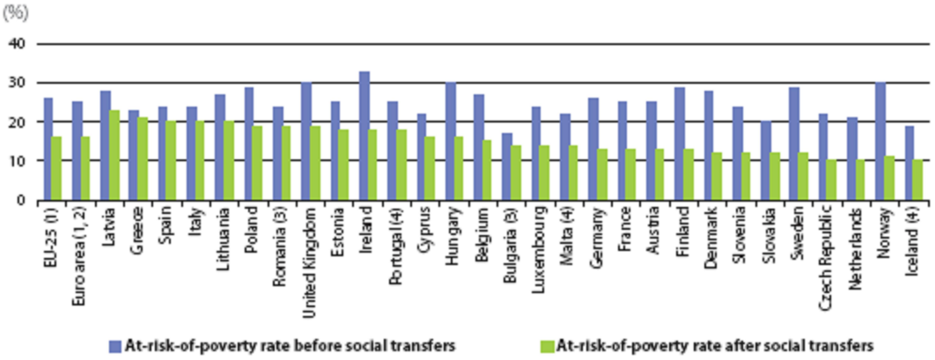
The most important point is comparability of indicators between regions. The globalization has lead to a standardization of most indicators. This makes indicators comparable with each other in an international context.

The aim of this chapter is on the one hand to give an overview about the existing graphics in statistical publications and on the other hand to develop a suggestion for visualizing social indicators for a broad public - first of all the decision-makers, namely the politicians. In the first part, commonly used visualization techniques, as well as some other visualization ideas are presented and discussed. Besides, it will be tried to review the diagrams from a non-statistician's perspective. The second part develops a suggestion to display Laeken indicators with the aim to present data understandable for non-statisticians.

2.1 Bar plot

Figure 2.1 is a double bar plot which presents the share of gross national disposal income for the EU-countries in the years 1997 and 2007. The plot is a standard display from the

Figure 6.4: At-risk-of-poverty rate, 2006



(1) Eurostat estimates based on population-weighted averages of national data.
(2) EA-13 instead of EA-15.
(3) National HBS.
(4) Provisional.

Source: Eurostat (tsisc020 and tsisc030)

Figure 2.2: Standard double bar chart from EUROSTAT (2009)

.

Eurostat web-site.

Bar plots are well known, because frequently used to show statistical data to a broad public, so most people will be able to read and understand this diagram.

The vertical labelling of the categorical axis makes it difficult to read the labels. Further, the country names are written in full - which takes much space.

When a viewer looks at this graph, maybe he will first search his country and than compare with other countries. He will probably not notice the development between 1997 and 2007 for his country. For example a Dutch person will perhaps observe that his country is at present among the countries with the highest share of gross national disposal income. But he will probably less evaluate that this value has decreased since 1997 and that there are several countries like Finland, Sweden, Germany etc. which had to register a large increase since 1997. Thus there are conflicting objectives in this graph: displaying the state and the evolution. Here clearly the state is prioritsed.

Figure 2.2 shows two indicators in a bar plot - the At-risk-of-poverty rate before and after social transfers. Basically it makes sense to show these two indicators together because one depends on the other. But like in the previous diagram, the user cannot really observe the difference between the first and the second variable. For the viewer will not only be of interest to know how important is the At-risk-of-poverty rate before or after social transfers but also what is the difference between the At-risk-of-poverty rate before and after social transfers. The graph is a bit overcharged because of the country labels which are written in full. The bars are ordered by descending values of the variable At-risk-of-poverty rate after social transfers. We note that this makes it more difficult to find a particular country.

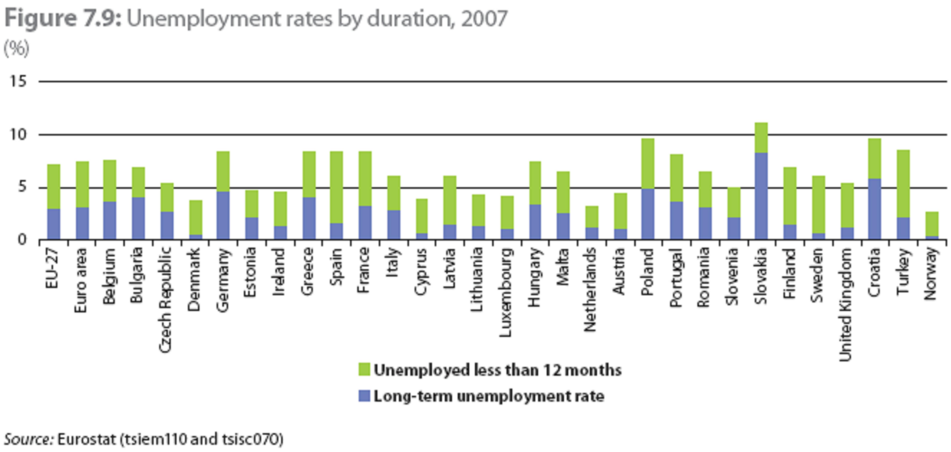


Figure 2.3: Stacked bar chart from EUROSTAT (2009)

.

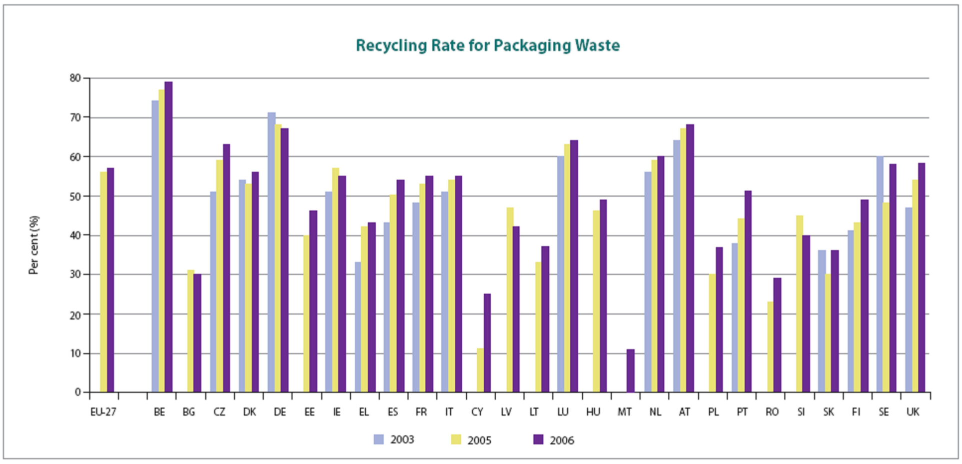


Figure 2.4: Multi bar plot from EUROSTAT (2008)

Figure 2.3 shows the unemployment rate less than 12 months and the long-term unemployment rate stacked in the same column. The ordering is alphabetically but according to the official names of country and not by english names which are given in the labels. It makes it difficult to find a certain country. The viewer can determine the countries with a high or low global unemployment rate. But he would not be able to determine or compare the values of the unemployment rate less than 12 month. The same problem with the vertical oriented country names like in the two previous graphs.

The multi bar plot displayed in Figure 2.4 shows that it is a real challenge to handle three variables (Recycling rate for the years 2003, 2005 and 2006) for more than 25 observations in such a small space.

For the categories labels, country codes (two characters) have been chosen. It needs less space than the full country names and thus labels can be positioned horizontally. Like in the previous graph (Figure 2.3) in this bar plot the ordering is as well alphabetically

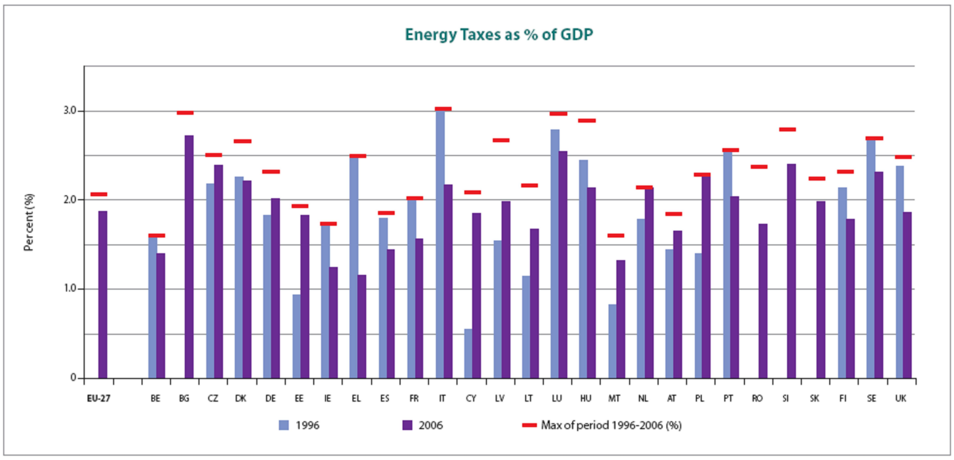
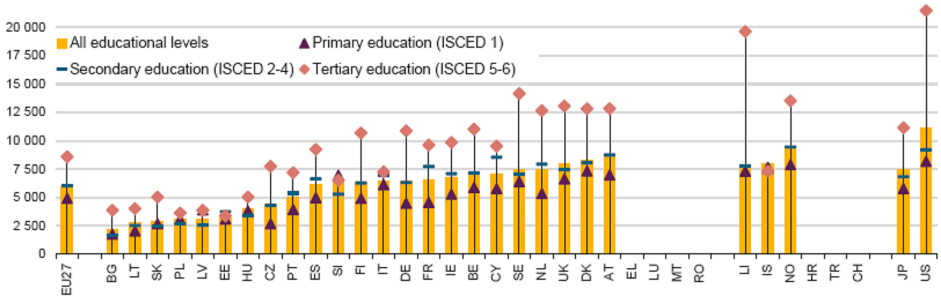


Figure 2.5: Double bar plot with target line from EUROSTAT (2008)

Figure 2: Expenditure on public and private institutions per full time equivalent pupil/student in EUR PPS – 2006



Source: Eurostat, Education statistics, UOE data collection (educ_fitotin)

Figure 2.6: Bar plot with data points from EUROSTAT (Reis and Hirmo, 2009)

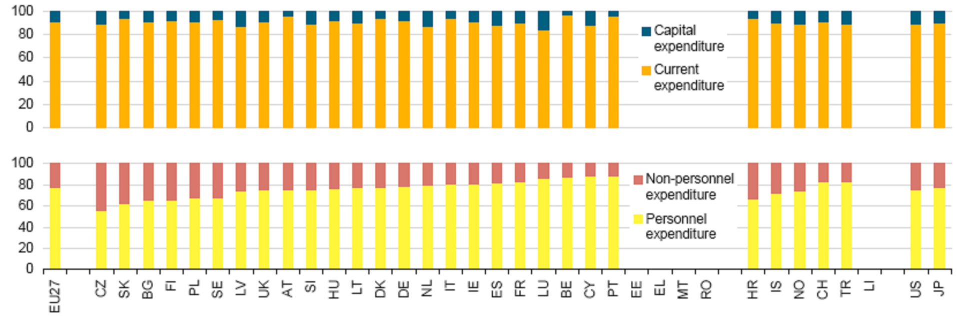
according to the official names of country names and does not show the categories in a ascending order or by alphabetically order of the labels. In addition the viewer's orientation is constrained by the presentation of the tree years. The selection of the years 2003, 2005, 2006 (there is not the same interval between the years) could cause misinterpretation. Above all some missing values (see CY, LV, MT etc.) disturb the reading of the diagram.

Figure 2.5, a double bar plot, reveals the percent of energy taxes of GDP by EU-countries for the years 1996 and 2006. Additionally to the vertical bars the maximum value of the period from 1996 to 2006 is indicated by a horizontal red bar.

There is unordered information because of the ordering by according to the official names of country and not to the English names which are given in the labels. Besides some missing values complicate the reading of the plot. Also the red bars confuse the comprehensibility of the graph.

The experiment for showing four variables and more than 25 observations in one bar plot has failed in Figure 2.6. The viewer has to observe, understand and compare four variables. The four variables for one observation are represented in a same column. If

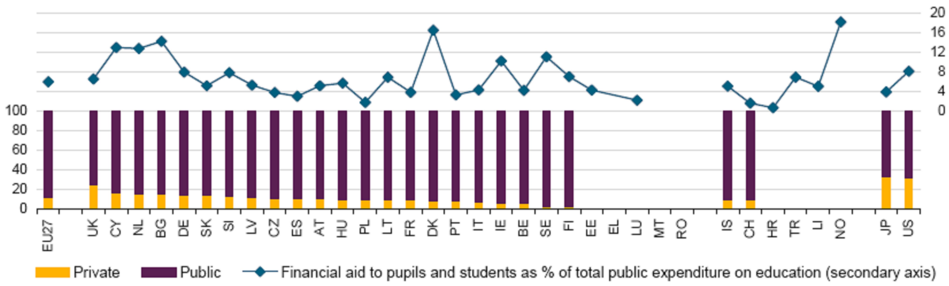
Figure 3: Distribution of expenditure of public institutions by type of expenditure (in %) - 2006



Source: Eurostat, Education statistics, UOE data collection (educ_fipubin)

Figure 2.7: Double stacked bar plot from EUROSTAT (Reis and Hirmo, 2009)

Figure 4: Distribution of educational institutions expenditure by source of funds - 2006



Source: Eurostat, Education statistics, UOE data collection (educ_fiaid, educ_fifunds)

Figure 2.8: Stacked diagram from EUROSTAT (Reis and Hirmo, 2009)

two variables have more or less the same value, you can hardly distinguish the different values because they overlap. Furthermore, there are a lot of missing values (EL, LU etc.) that take up precious space for non-information.

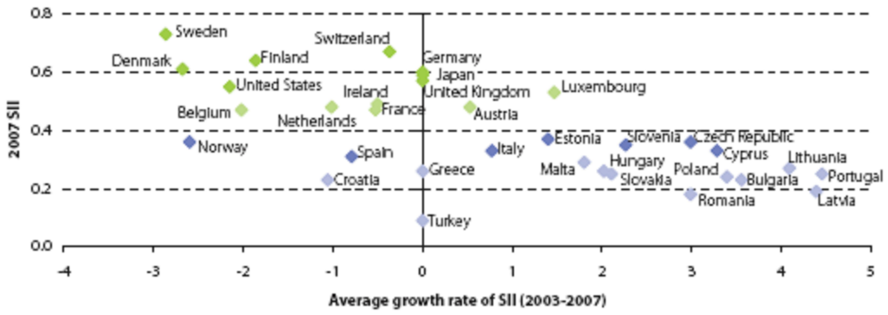
At least, the values are presented in a ascending order by the global variable "all educational levels". The categories have been grouped in EU-countries, non EU-countries (e.g. LI, NO, CH) and non European countries (JP, US).

Figure 2.7 is another attempt to show several variables within a small space. The figure shows two bar graphs with the same scale one upon the other. Each graph contains two variables which are presented in stacked bars.

To save space the break in place of the missing values has been used for the legend. It is not possible to compare the values of the first graph with the second graph because there are only relative values. So it makes not a lot of sense to show these two charts so close in one image. Maybe the viewer wants to compare the lengths of the bars of the first plot with these of the second plot. This would cause misinterpretation.

The next image (Figure 2.8) displays a stacked diagram consisting of a stacked bar plot and a line graph. For both graphs we have the same categorical axis (country names) but not the same scale on the value axis.

Figure 1.19: Evolution of the Summary Innovation Index (SII)



Source: European Commission, Directorate-General for Enterprise and Industry (European Innovation Scoreboard, 2007 - Comparative analysis of innovation performance)

Figure 2.9: Scatter plot with reference lines from EUROSTAT (2009)

The observations on the categorical axis are ordered by descending values of the variable "private" of the bar plot. The EU and non-EU countries are grouped. In the line graph there is no direct connection leading from one point to the other. In other words the line has no motivation like in a time plot and therefore the line is misleading. Another problem in this presentation is that the line graph and the bar plot are representing two different things. It makes no sense to present this two graphs as a composed graph because the variable of the line graph is not directly correlated with the variables of the bar graph. At least the correlation would not be visible because of the scale of the bar chart, which hides the difference in the proportion of the lowest stack.

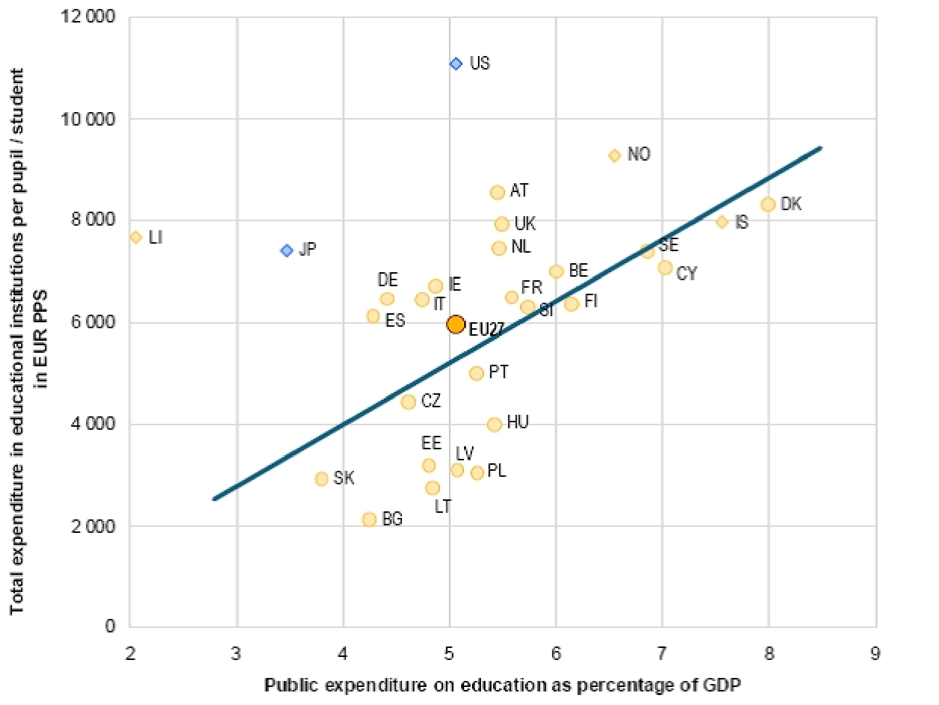
2.2 Scatter plot

The scatter plot (Figure 2.9) allows quite good to present two variables with many values. The dots, representing the observations (countries), are coloured in function of the y-axis values. High values are green coloured, low values are blue coloured. The dots are labelled by full label names (country names). Thus the diagram is overcharged. What is more, the labels aren't clearly assignable to the respective dots. The horizontal reference lines function as guide lines which redundantize (or could replace) the different dot-colours.

In contrast to Figure 2.9, the following scatterplot (Figure 2.10) is less charged because the dots are labeled with only two characters and not with the full country names. But the dots are too bright compared to the labels. The viewer run the risk of perceive the labels as the real data point. Labels are not always positioned in the same way (once the label is on the right of the data point, another time the label lies above or below the data point).

Figure 2.11 shows the relationship between two variables (property crime and violent crime) by state for the indicated year 1973. The states are represented in form of a

Figure 1: Public education expenditure as a percentage of GDP and total expenditure in Euro PPS per pupil / student – 2006



Notes: Regression line based on EU countries only.
The particularly low value of public expenditure on education as percentage of GDP for Liechtenstein is the result of the combination of a relatively small student population and a relatively high GDP.

Source: Eurostat, Education statistics, UOE data collection (educ_figdp, educ_fitotin)

Figure 2.10: Scatterplot with regression line from EUROSTAT (Reis and Hirmo, 2009)

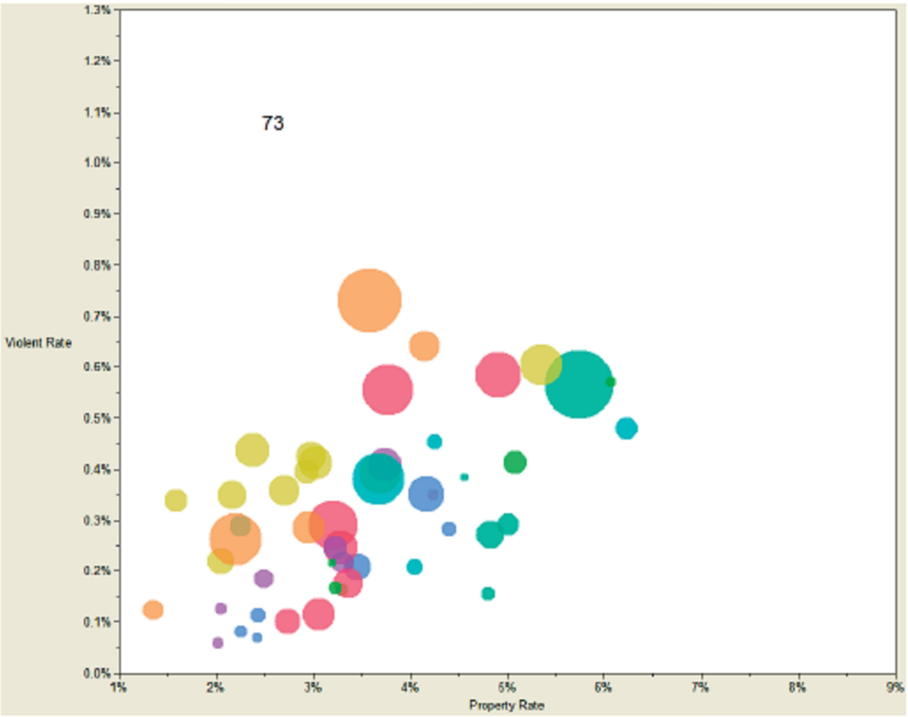


Figure 2.11: Bubble plot by Stephen Few (2007a)

bubble whose seize represent the population of the relative state. Showing the bubbles in comparative amplitude of the population allows to display a third variable additionally to the property crime and the violent crime variable. The colours mean the belonging to a geographical region which allows to compare not only states but also regions (e.g. we can observe that the green coloured region has a higher Property crime rate than other regions). However, the bubbles are not labelled, which makes it impossible to spot a particular country. This may not be necessary if the main point is rather to show the corrleation.

In the next dot plot (Figure 2.12) the categorical axis is the y-axis and the values are on the x-axis. Probably the maker of this graph wanted that the company names would be readable in a usual manner (horizontal). But for all that the names are too small and too close. Furthermore it is difficult to assign a dot to the respective company name. You can only observe that there are some companies with a huge revenue (above 100 billion Dollars). And the other companies are located into more or less the same range.

2.3 Trellis plot

A trellis plot displays multiple (scatter) plots for pairs of variables. The displayed example (Figure 2.13) shows the percent with incomes over 50K and 75K for various ethnic groups in New Jersey. Each plot of this multi-panel dot chart presents the values for a county.

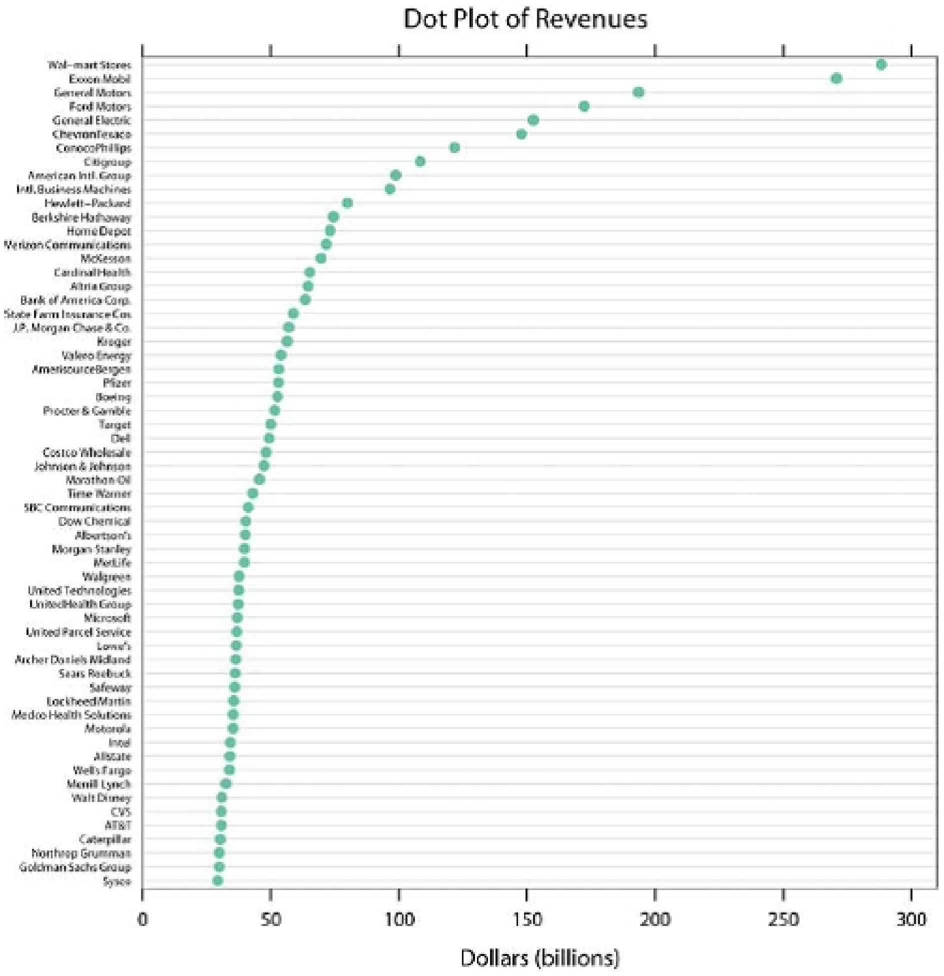


Figure 1: This dot plot shows the revenues of the top 60 companies from the Fortune 1000 list.

Figure 2.12: Dot plot of revenues (Robbins, 2006)

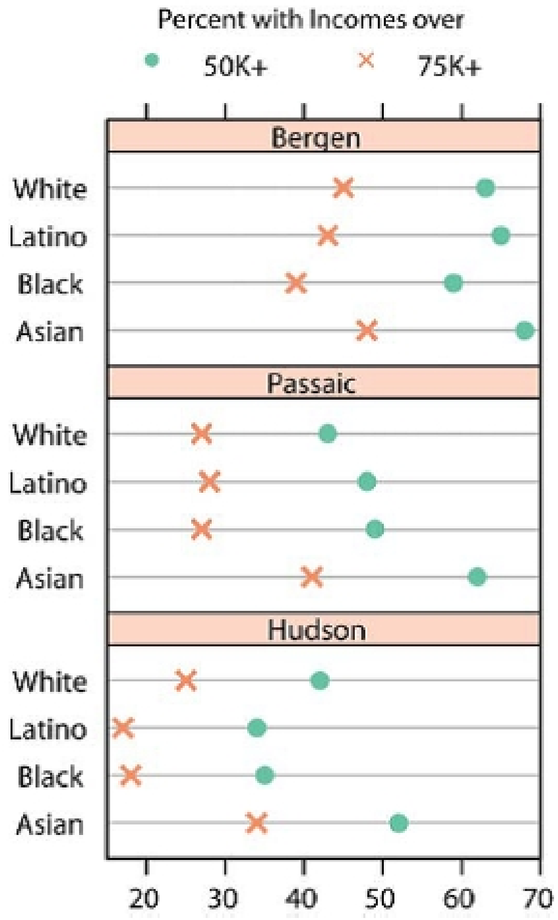


Figure 6: This shows the data of Figure 5 in a multi-panel dot chart.

Figure 2.13: Trellis plot (ROBBINS, 2006)

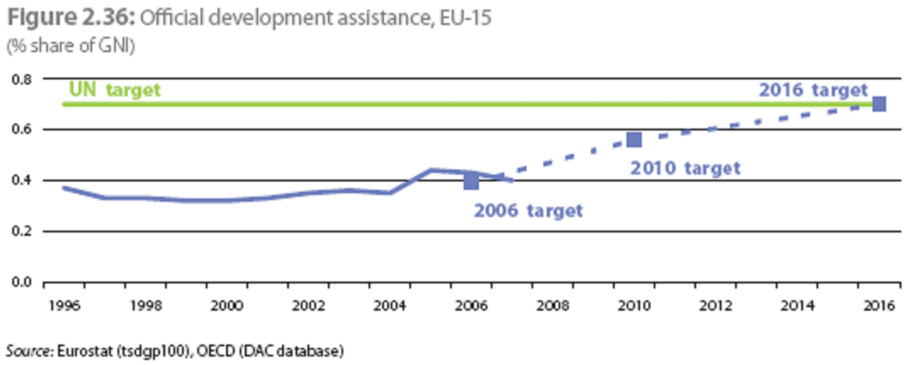


Figure 2.14: Line graph with target line from EUROSTAT (2009)

This presentation allows quite good to compare different counties one to another. The table is rich in information but even so it do not need much time to understand the graphs.

2.4 Line graph

Line graphs are essentially applied to show temporal evolution of one ore several variables. Figure 2.14 shows the evolution of the share of official development assistance in the EU. The graph also contains a target line.

This kind of graph shows good the evolution from past to present as well as a determined evolution in the future to achieve the target value. In this kind of graph you only can present the values for one or just a couple of countries. It would though not be possible to display the values for each of the EU countries. Otherwise you would have a muddle of near 30 lines in the same graph. It would make it impossible to distinguish the single lines and to receive a clear information.

Figure 2.15 is a line graph which shows the evolution of import of electricity in TWh for the EU-27, the top 5 exporting countries and the top 5 importing countries for the period 1996 to 2006. The graph reveals information about the evolution with all fluctuations of the EU-27 electricity imports. The marks at the measured value are not necessary and clutter the image. The colour serves already to distinguish the three lines. If a black-and white capable plot is the objective, then a better way to distinguish the lines is by the line structure (solid, dashed, dotted etc.).

Figure 2.16 shows the time evolution of salary expenses from July to December 2004 of eight variables, each presented in another slot of the graph. The eight graphs are arranged side by side, each one using the same value axis.

The graph contains (too) much information. The tree lines per graph are showing “exempt”, “non-exempt” and “total”. The “total”-line only resume the values of “exempt” and “non-exempt”. So it is not absolutely needed to show the “total” further to the two

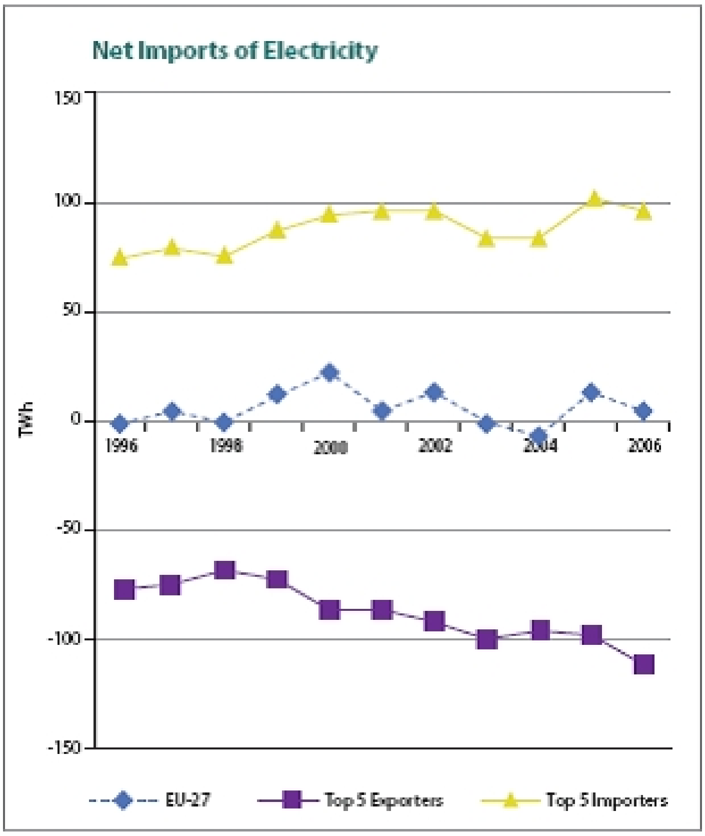


Figure 2.15: Time series line graph from EUROSTAT (2008)

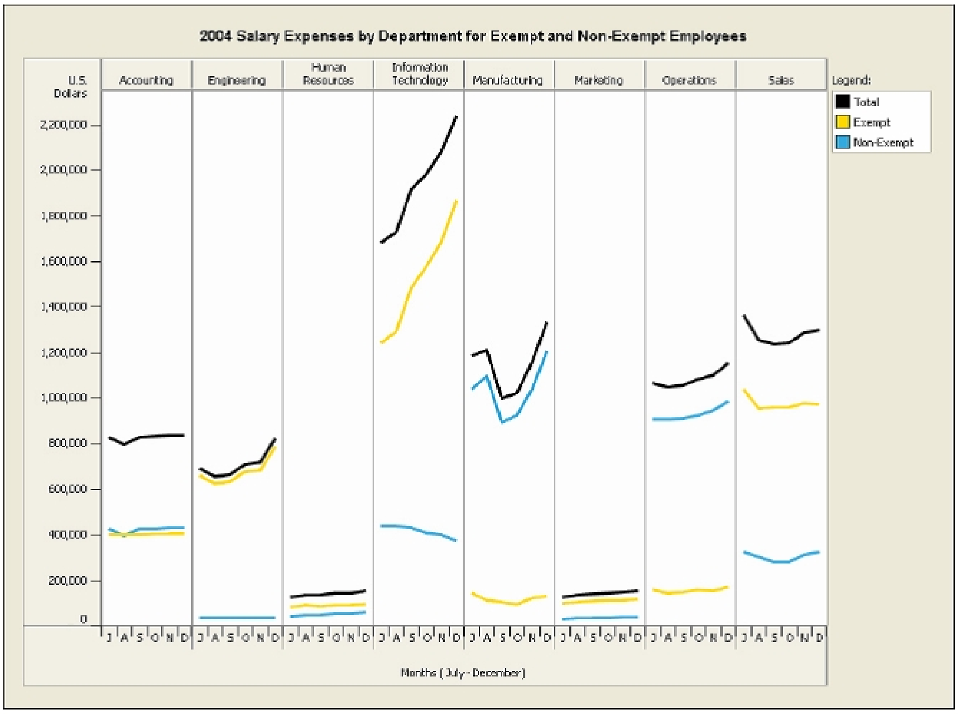


Figure 1: Winning Solution, submitted by Tableau Software

Figure 2.16: Visualizing multidimensional data through time (FEW, 2005a)

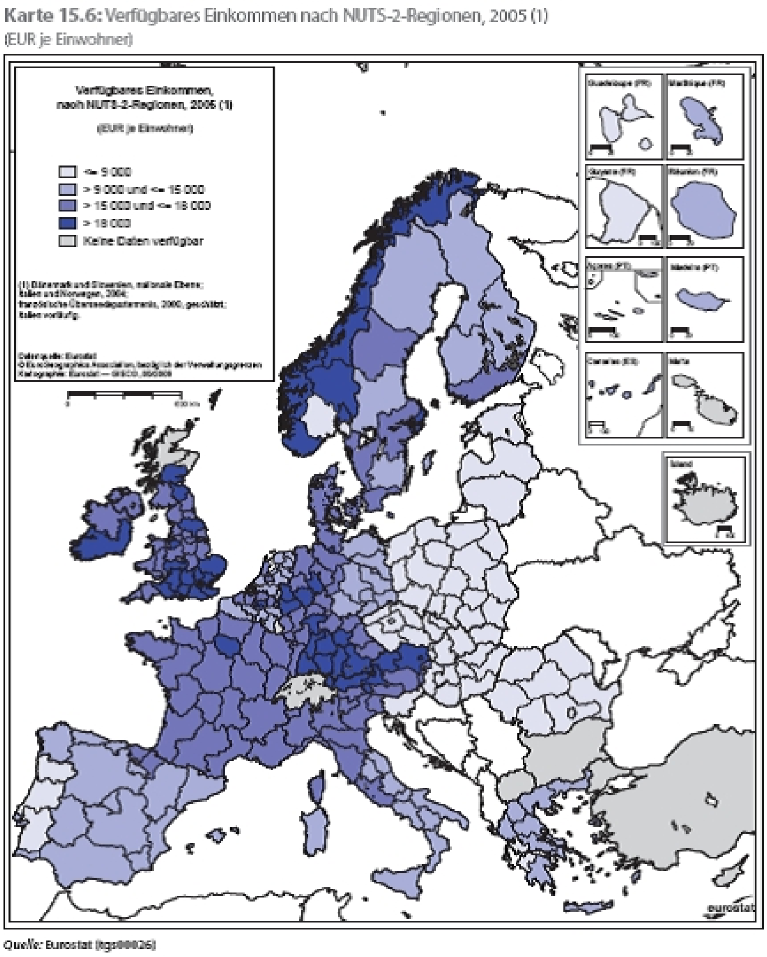

Karte 15.6: Verfügbares Einkommen nach NUTS-2-Regionen, 2005 (1)
(EUR je Einwohner)
Verfügbares Einkommen, nach NUTS-2-Regionen, 2005 (1)
(EUR je Einwohner)
<= 9 000
> 9 000 und <= 15 000
> 15 000 und <= 18 000
> 18 000
Keine Daten verfügbar
(1) Dänemark und Slowenien, nationale Ebene; Italien und Norwegen, 2004; französische Überseedepartements, 2000, geschätzt; Italien vorläufig.
Datenquelle: Eurostat
© EuroGeographics Association, bezüglich der Verwaltungsgrenzen
Kartographie: Eurostat — GISCO, 00/2008
0
600 km
Guadeloupe (FR)
Martinique (FR)
Guyane (FR)
Réunion (FR)
Açores (PT)
Madeira (PT)
Canarias (ES)
Malta
Island
eurostat
Quelle: Eurostat (tgs00026)

Figure 2.17: Geographical map from EUROSTAT (2009)

other variables.

2.5 Geographic visualization

Geographic or thematic maps are often used in publications of statistical offices to show the spatial variations (e.g. between countries or a regions) of a certain topic. This geographical map (Figure 2.17) presents the disposal income by NUTS[3]-II regions. The darker the blue colour the higher is the disposal income in the respective region. Since the map of Europe is a well known picture and European citizens are well trained in finding their country the map is a good instrument for locating countries. Whether a European citizen knows the shape and location (within his country) of the NUTS-II region he is living in, seems to be less likely, though.

The values for the regions are categorical instead of metric values. A colour represents thus only a range of values. So the information content of the map is comparatively poor.

[3] Nomenclature of Territorial Units for Statistics

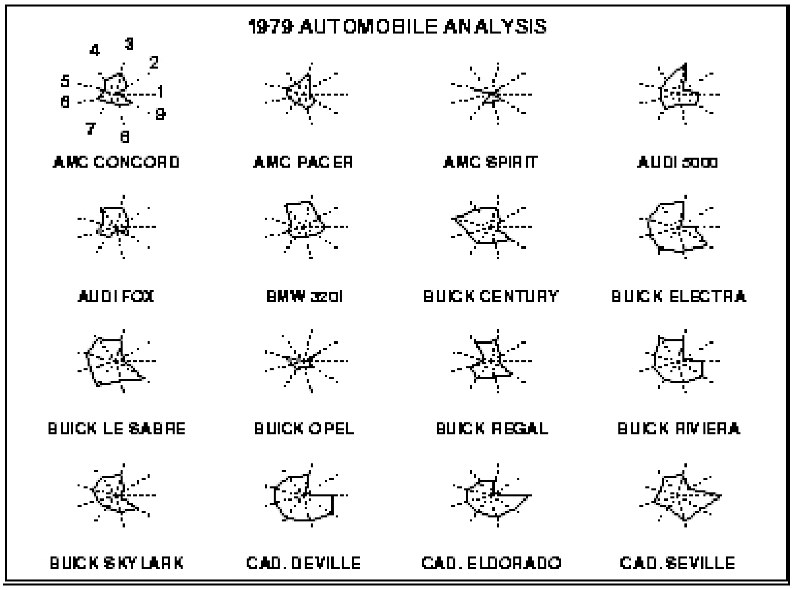


Figure 2.18: Starplot seen in Engeneering Statistics Handbook (NIST/SEMATECH, 2010)

The map needs a lot of space for relative few information. At this place we could cite Edward Tufte, who mentioned, that a large share of ink on a graphic should present data-information (Tufte, 2001). But in a geographic map the ratio between the ink using for information and total ink used to print the graphic is quite low. Besides in our example another ambiguity comes up: it is not evident to show the disposal income for regions with a low population density (see e.g. Norway). It could cause misinterpretation. For this map different blue tones have been chosen for the classes. Above all the colours for "no data available" and the class "<= 9000" look alike. The two classes can easily be confused.

2.6 Star plot

The aim is to visualize more than one kind of indicator, for example, on domains.

Figure 2.18 shows a star plot matrix for several automobile models of the year 1979. Each star represents an observation (a car model). Each ray (axis line) of the star stands for an other variable and the length corresponds to a value. The axis don't need to represent dependent measures. It could be interesting to find and group observations which are similar. A star plot allows to present a lot of information in a small space, but the shown information is consisting of only relative values.

Table 2.1: Conditions of poverty in households (HH) for risk groups

	no probl.	small probl.	medium probl.	apperant poverty
smallest children 4-6 years	76	9	8	7
HH with nationalized persons	67	10	15	8
MPH, greater eq. 3 childrens	71	14	10	5
single woman in HH without rent	66	12	12	10
HH with one person handicaped	62	7	21	10
HH with only one parent	51	18	17	14
HH with immigrands	60	11	14	15
single woman in HH with rent	62	14	12	12
HH with social benefits	42	24	18	16
HH with perm. unempl. pers.	43	13	16	28
total	78	7	10	5

2.6.1 An example of a starplots using EU-SILC

With starplots one is able to visualize multivariate information, e.g. values of several indicators splitted by years and/or domains which is done here. As mentioned abobe each star plot or segment diagram represents one row of the input x. In case of indicators this input row consists of the values of the different indicators obtained from one domain. An aritifical example of an input might be:

domain	ARPR	QSR	GINI
A-Vienna	0.11	0.20	0.22
A-Styria	0.13	0.25	0.26
A-Carinthia	0.14	0.28	0.30
A-UpperAustria	0.10	0.19	0.21

The different indicators start on the right and wind counterclockwise around the circle. The size of the (scaled) column is shown by the distance from the center to the point on the star.

Figure 2.19 shows a stars plot of the condition of risk groups obtained from the A-SILC 2007 survey, see Table 2.1.

With this represenation the domains with similar or dissimilar characteristics could be identified.

It is easy to see, that single womens with or without rent do have a quite similar behaviour. It is also interesting to see, that the other risk groups are quite different.

The basic function (`stars()`) to produce such a plot can be found in the R base package.

2.6.2 An example of a starplot with indicators

A starplot which was presented in the AMELI session at the NTTS 2011 conference is shown in Figure 2.20. This starplot (also called spiderplot) shows the values of six

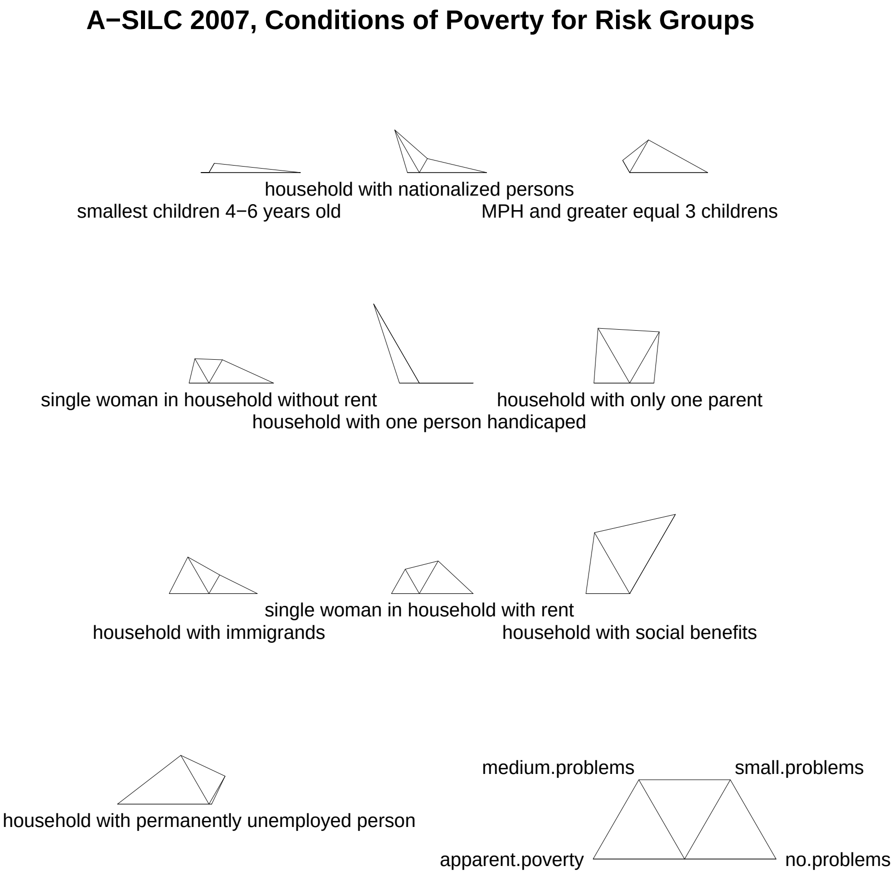


Figure 2.19: Stars plot of poverty risk groups.

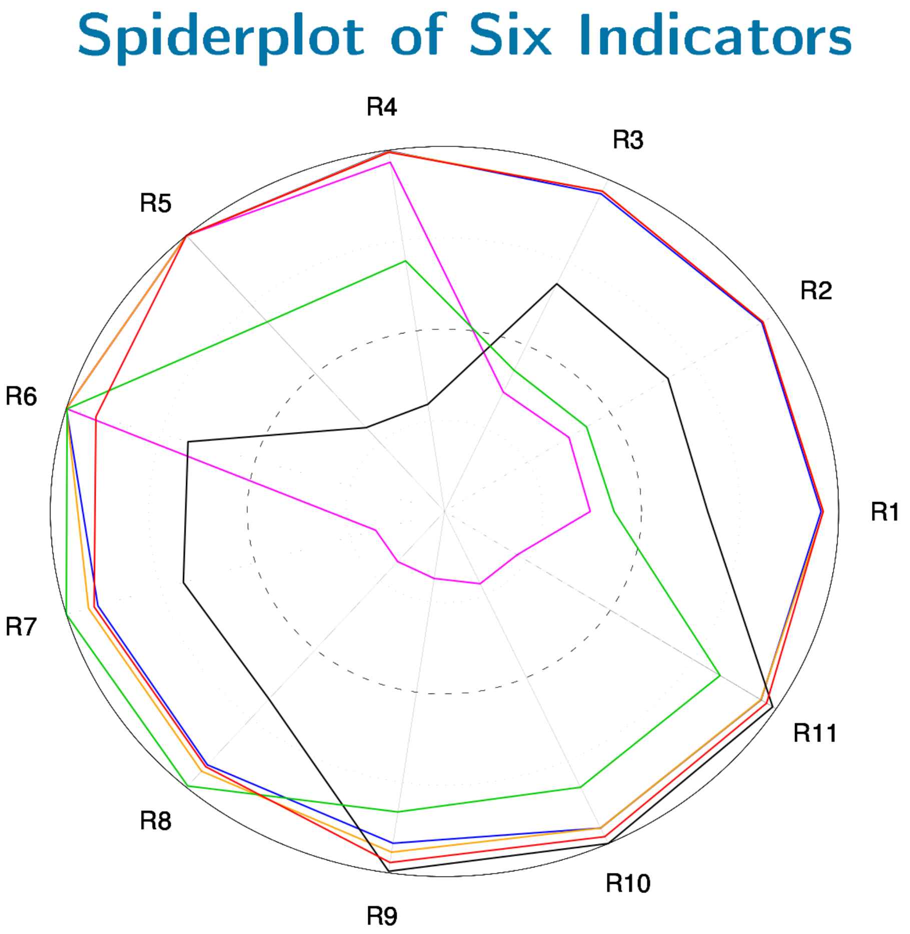
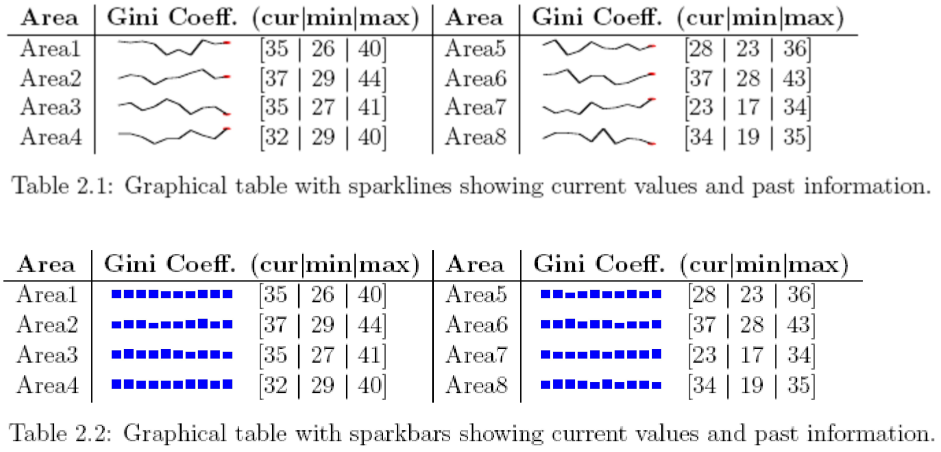


Figure 2.20: A star plot (spiderplot) which represents six different indicators.



<table>
<tr><th>Area</th><th colspan="2">Gini Coeff. (cur|min|max)</th><th>Area</th><th colspan="2">Gini Coeff. (cur|min|max)</th></tr>
<tr><td>Area1</td><td></td><td>[35 | 26 | 40]</td><td>Area5</td><td></td><td>[28 | 23 | 36]</td></tr>
<tr><td>Area2</td><td></td><td>[37 | 29 | 44]</td><td>Area6</td><td></td><td>[37 | 28 | 43]</td></tr>
<tr><td>Area3</td><td></td><td>[35 | 27 | 41]</td><td>Area7</td><td></td><td>[23 | 17 | 34]</td></tr>
<tr><td>Area4</td><td></td><td>[32 | 29 | 40]</td><td>Area8</td><td></td><td>[34 | 19 | 35]</td></tr>
</table>

Table 2.1: Graphical table with sparklines showing current values and past information.

<table>
<tr><th>Area</th><th colspan="2">Gini Coeff. (cur|min|max)</th><th>Area</th><th colspan="2">Gini Coeff. (cur|min|max)</th></tr>
<tr><td>Area1</td><td></td><td>[35 | 26 | 40]</td><td>Area5</td><td></td><td>[28 | 23 | 36]</td></tr>
<tr><td>Area2</td><td></td><td>[37 | 29 | 44]</td><td>Area6</td><td></td><td>[37 | 28 | 43]</td></tr>
<tr><td>Area3</td><td></td><td>[35 | 27 | 41]</td><td>Area7</td><td></td><td>[23 | 17 | 34]</td></tr>
<tr><td>Area4</td><td></td><td>[32 | 29 | 40]</td><td>Area8</td><td></td><td>[34 | 19 | 35]</td></tr>
</table>

Table 2.2: Graphical table with sparkbars showing current values and past information.

Figure 2.21: Sparklines and sparkbars (Alfons et al., 2009)

indicators regarding eleven countries. It is easy to see that the black, the green and the magenta coloured indicators behaves quite different while the blue, red and orange coloured indicators show quite similar behaviour.

2.7 Spark lines/spark bars

In the first of the two tables of Figure 2.21 sparklines are showing the evolution of the Gini Coefficient for several areas. The current value is shown in red. In the same row after the sparkline, the current value, the minimum and the maximum of the passed years are displayed. The sparkline aid to see how the Gini coefficient has evolved during the last few years, but you don't have exact values, only trends. An advantage of sparklines is indeed, that they could directly be included in texts (Alfons et al., 2009).

The second table shows sparkbars in place of sparklines. The sparkbars table also show current values and past information. In contrary to the sparklines, sparkbars don't let observe any trends. The bars are too thick to compare one with another.

2.8 Mosaic plot

In the mosaic plot shown in Figure 2.22 each column represent a hair colour. The columns are divided in function of the conditional frequency of hair colour. The same was made for the rows: the rows have been divided in function of the conditional frequency of eye colour. The area of each cell is proportional to the frequency in this cell. This graph is not intuitively comprehensible. A person who doesn't know this kind of graphs needs an explication to read and understand a mosaic plot. Even for experienced persons it needs some time to analyse the graph. On the other hand, a mosaic plot is maybe the best representation of the complex information in a contingency table.

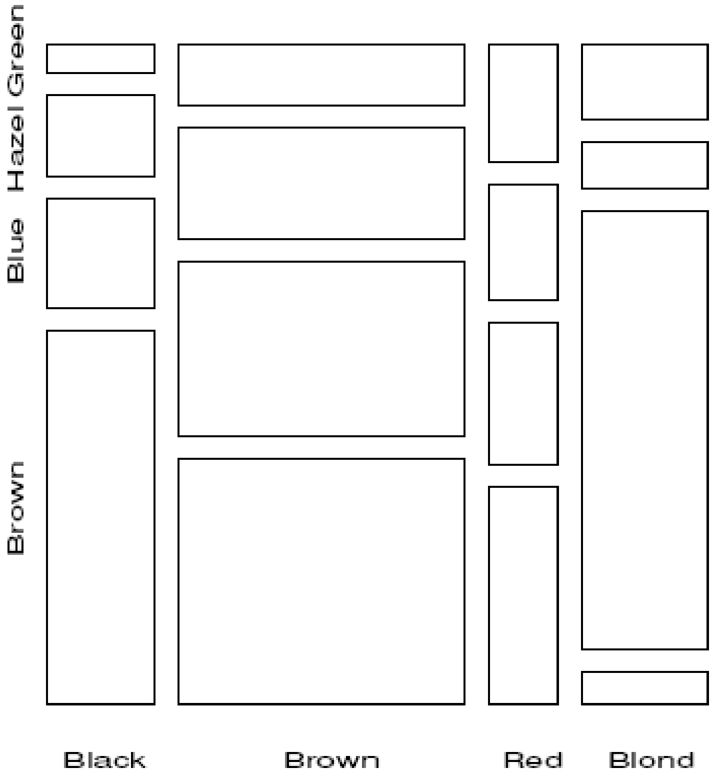


Figure 2.22: Mosaic plot (Friendly, 2003)

The common representation of the data given in Table 2.1 is a barchart (see, e.g., Austria (2009)). However the relative size of the segments cannot be represented in such barcharts. Mosaicplots is an extension of a barchart whereas the relative size of the cells is considered. Mosaic displays have been suggested in the statistical literature by Hartigan and Kleiner (1984) and have been extended by Friendly (1994).

2.8.1 Example of a mosaicplot for poverty-at-risk groups

Figure 2.23 represents another mosaicplot. This extended mosaic plot visualizes additionally the fit of a log-linear model. To provide additional information, the residuals are colour coded to visualize their sign, size and possibly significance. This mosaicplot shows the proportions of poverty risk groups on smallest children 4-6 years old (4–6), household with nationalized persons (NP), MPH and greater equal 3 childrens (MPH3), single woman in household without rent (SW), household with one person handicaped (HC), household with only one parent (OP), household with immigrands (I), single woman in household with rent (SWR), household with social benefits (SB), household with permanently unemployed person (UP), total (tot) and the proportions of their finanzial situation indicated by no problems (NP), small problems (SP), medium problems (MP) and apparent poverty (AP) for the year 2007 A-SILC data. It can be easily seen that households

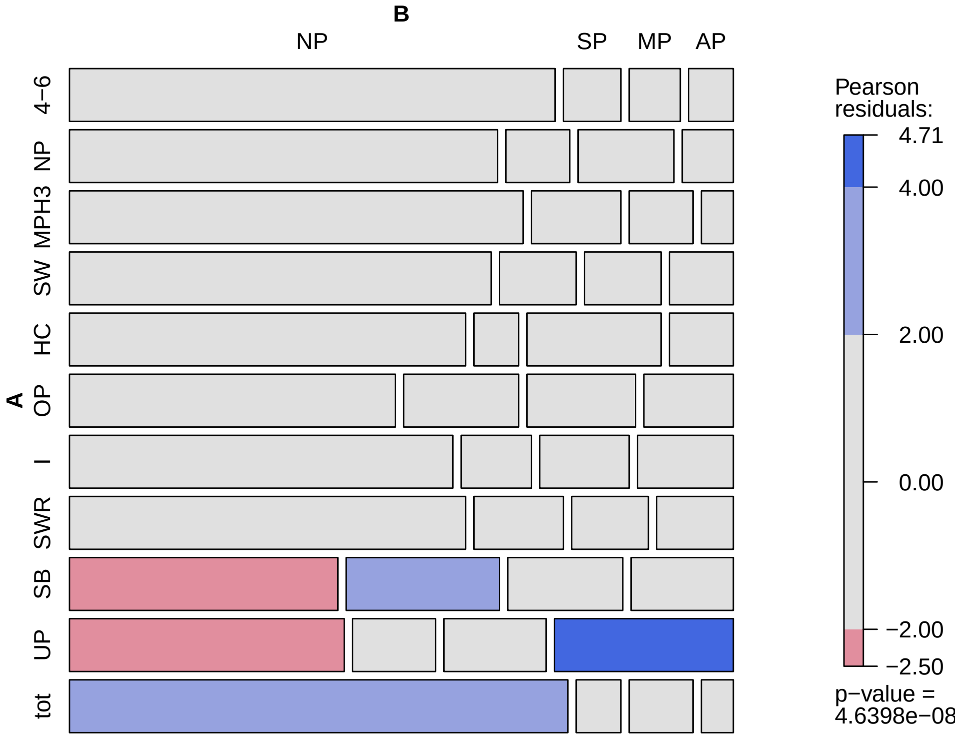


Figure 2.23: Mosaicplot of poverty risk groups.

with unemployed persons includes a much higher rate of poverty as the other household. But the behaviour of households including a children between 4–6 years old is similar to the average of all households.

The basic function to produce such a plot (`mosaic()`) can be found in the R-package **vcd** (Meyer et al., 2009).

2.9 Funnel plot

The thick curves of the funnel plot in Figure 2.24 show change in time (1996 to 2000) of Research and Development expenditure of industry in % of GDP for Japan and Switzerland. The green funnels represent the confidence intervals at 2000 for the respective change. This graph let observe the evolution of an indicator between two years. It presents also the uncertainty of the change. But this graph do not show how the indicator proceeded between 1996 and 2000.

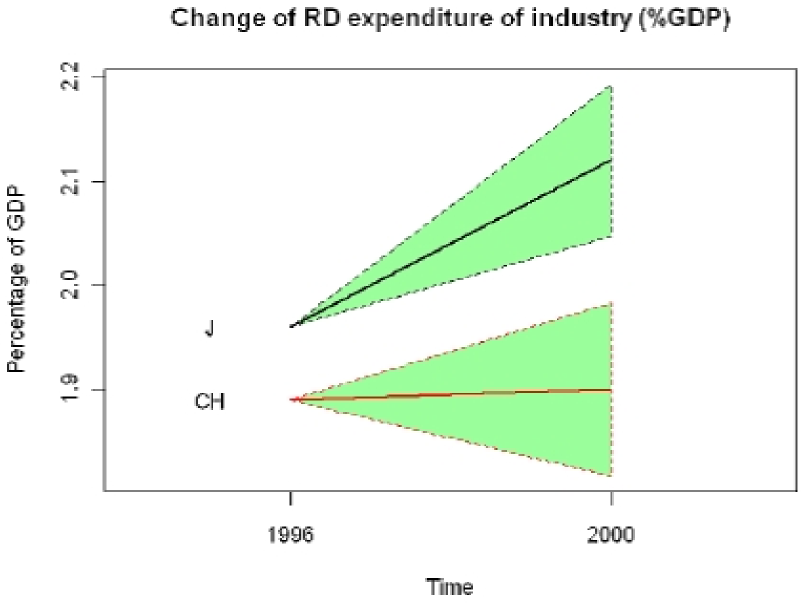


Figure 1: Change in time plot.

Figure 2.24: Funnel plot (Hulliger and Pooda, 2010)

2.9.1 Application in change over time in the AMELI project

Figure 2.25 shows an application of a funnel plot to show if a change over time of an indicator is significant or not. The corresponding confidence intervals of an indicator is shown in the funnels. It is easy to see that no significance change over time happen in any of this indicators from year 2005 until year 2008, because the point estimate in the following year is covered by the confidence intervall of the change.

2.10 Scorecard

The scorecard (Figure 2.26) from the European Environmental Agency presents the results of several environmental indicators for European states. The different tones of blue stand for top 25%, middle 50% and lowest 25% of indicator values. The blue coloured rectangles with white bolts stand for the level of progress between 1992 and 2002/03. The red and green rectangles indicate the distance to target (on track/not on track). The countries are grouped by similar results, socio-economic and geographical factors (EEA, 2005), but in spite of this arrangement a comparison with other countries is only possible in a limited way.

This table needs much space (about three pages) for relative poor information. Per indicator only three classes are used to represent the status for the last year, the progress over time and the distance to target (see the legend between the first and the second part

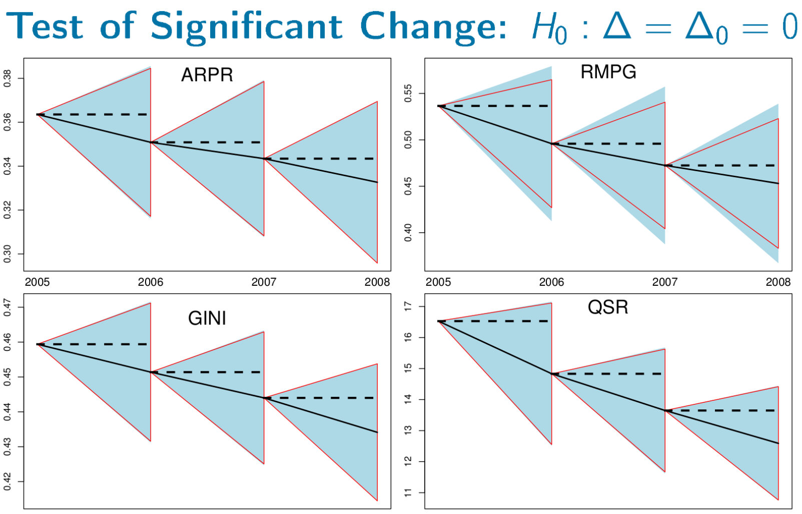
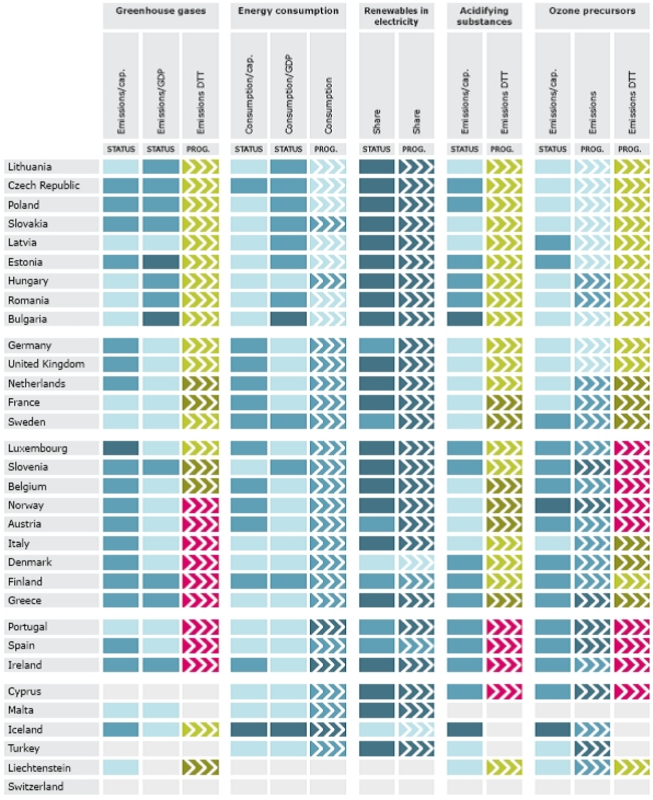
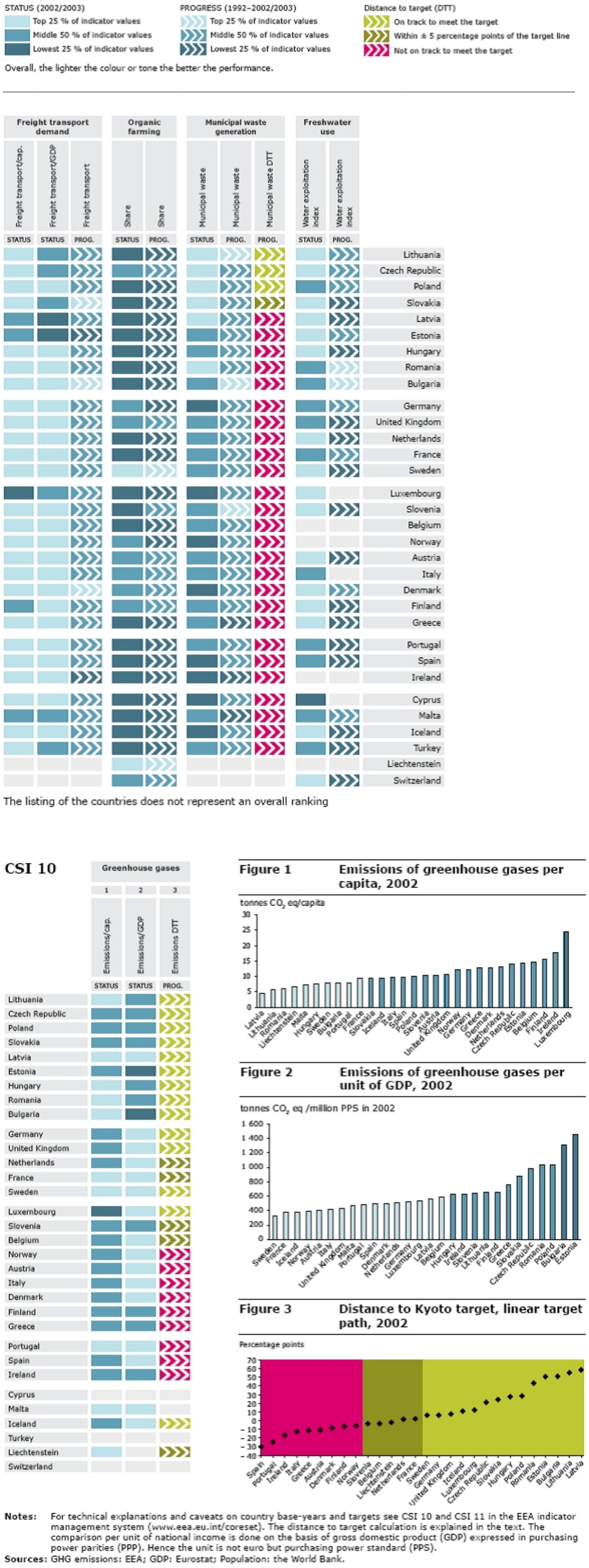


Figure 2.25: Funnel plot shown at the AMELI session at the NTTS 2011 conference.



The listing of the countries does not represent an overall ranking



STATUS (2002/2003)
Top 25 % of indicator values
Middle 50 % of indicator values
Lowest 25 % of indicator values

PROGRESS (1992–2002/2003)
Top 25 % of indicator values
Middle 50 % of indicator values
Lowest 25 % of indicator values

Distance to target (DTT)
On track to meet the target
Within ± 5 percentage points of the target line
Not on track to meet the target

Overall, the lighter the colour or tone the better the performance.

The listing of the countries does not represent an overall ranking

Figure 1 Emissions of greenhouse gases per capita, 2002

Figure 2 Emissions of greenhouse gases per unit of GDP, 2002

Figure 3 Distance to Kyoto target, linear target path, 2002

Notes: For technical explanations and caveats on country base-years and targets see CSI 10 and CSI 11 in the EEA indicator management system (www.eea.eu.int/coreset). The distance to target calculation is explained in the text. The comparison per unit of national income is done on the basis of gross domestic product (GDP) expressed in purchasing power parities (PPP). Hence the unit is not euro but purchasing power standard (PPS).

Sources: GHG emissions: EEA; GDP: Eurostat; Population: the World Bank.

Figure 2.26: Scorecard from the European Environment Agency (EEA, 2005)

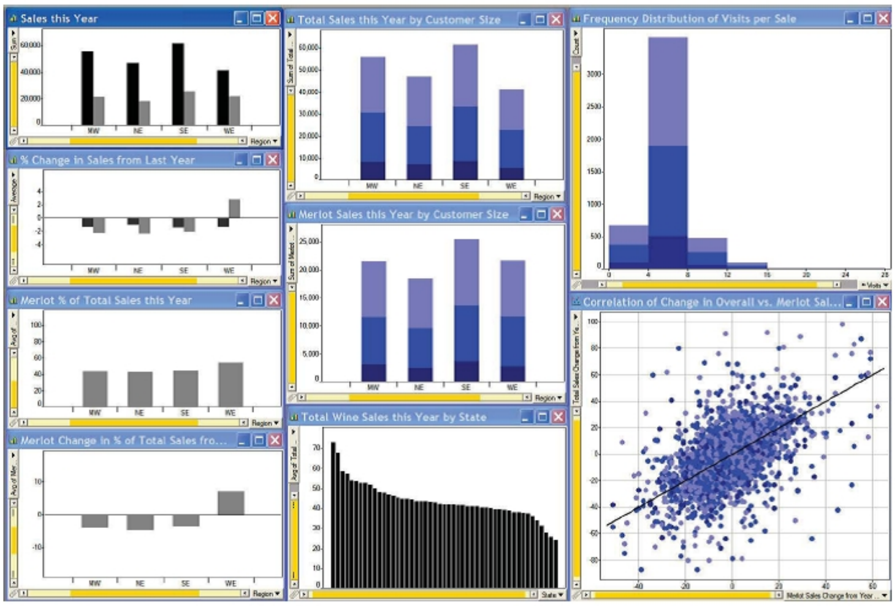

Sales this Year
% Change in Sales from Last Year
Merlot % of Total Sales this Year
Merlot Change in % of Total Sales fro...
Total Sales this Year by Customer Size
Merlot Sales this Year by Customer Size
Total Wine Sales this Year by State
Frequency Distribution of Visits per Sale
Correlation of Change in Overall vs. Merlot Sal...
MW
NE
SE
WE
Region

Figure 1: Example of a faceted analytical display created using Spotfire DecisionSite.

Figure 2.27: Dashboard: wine sales, seen by Few (2007b)

of the scorecard).

2.11 Dashboard

Figure 2.27 presents a dashboard which contains nine graphs based on a common set of wine market data. There are simple and double bar graphs, staked bar graphs as well as a scatter plot. The dashboard contains much information and give an overview about different aspects of wine sales, but to study and understand the whole board demands time.

The next dashboard (Figure 2.28) reveals also different aspects of wine sales. The table is very rich in information, nevertheless the dashboard is still comprehensible. That's because the same notation (bar plot) has been kept for the whole dashboard. The raw data have been enriched by inserting evaluation. The colours green, pink and green indicate if the respective values are estimated as good, satisfactory or poor.

2.12 Crosstab arrangement

Figure 2.29 presents a series of bar graphs. The graphs are arranged as a crosstab. Each of the four columns represent a geographical region and each row a sort of coffee or tea. All bar graphs show the profits of the respective product per region and per quarter. The regions have been arranged in the usual way of reading west pointing leftmost, followed by central, east and south.

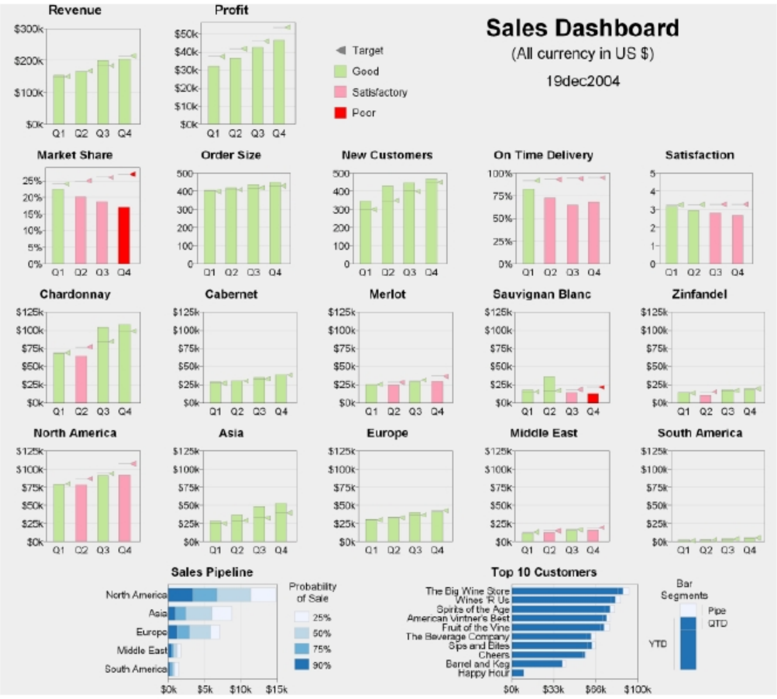


Figure 1: Winning sales dashboard submitted by Robert Allison of SAS.

Figure 2.28: Sales Dashboard seen by Few (2005)

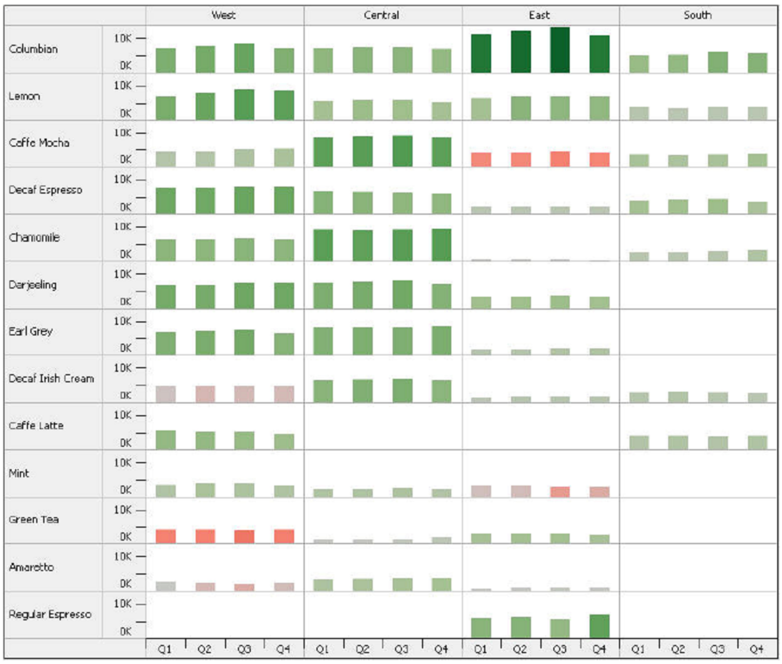


Figure 1: Example of a crosstab arrangement of small multiples, created with Tableau Software.

Figure 2.29: Crosstab arrangement seen by Few (2006f)

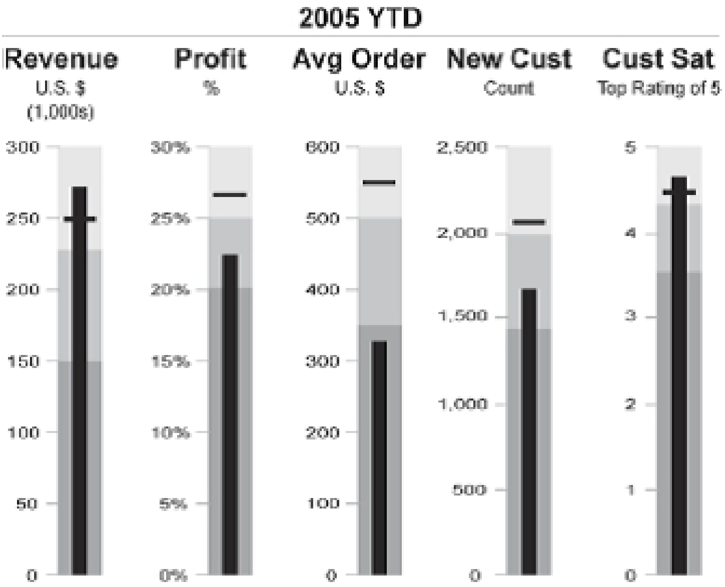


Figure 2.30: Bullet graphs arranged vertically, seen by FEW (2009a)

The crosstab arrangement allows to present several bar graphs for a whole list of observations. There is a lot of information in a small space. Compared to a dashboard, which is also rich in information, the crosstab arrangement is quickly comprehensible (because it contains only one type of graph). Linecharts instead of barcharts would depict the evolution better. It would also be a more ink-saving manner to show same data so that the graph would seem less overloaded. However the size of changes is depressed by the small vertical extension allowed for due to the many lines of the crosstab display.

2.13 Bullet graph

Figure 2.30 is a multiple bullet graph representing five variables. Each of the five graphs is using an other scale. The black bar in each graph represents the current value for the year 2005. The background fill colours encode defined qualitative ranges to declare if the current value has to be evaluated as “bad”, “satisfactory” or “good”. The horizontal bar indicate a given target. It is a space saving form to show different measures and evaluations belonging to a same theme. But the measures are not comparable one with another because each graph use a different scale.

2.14 Multivariate heatmap matrix

A multivariate heatmap matrix presents the values of multiple variables for a list of observations. In this example (Figure 2.31) several variables like price, revenue and profit

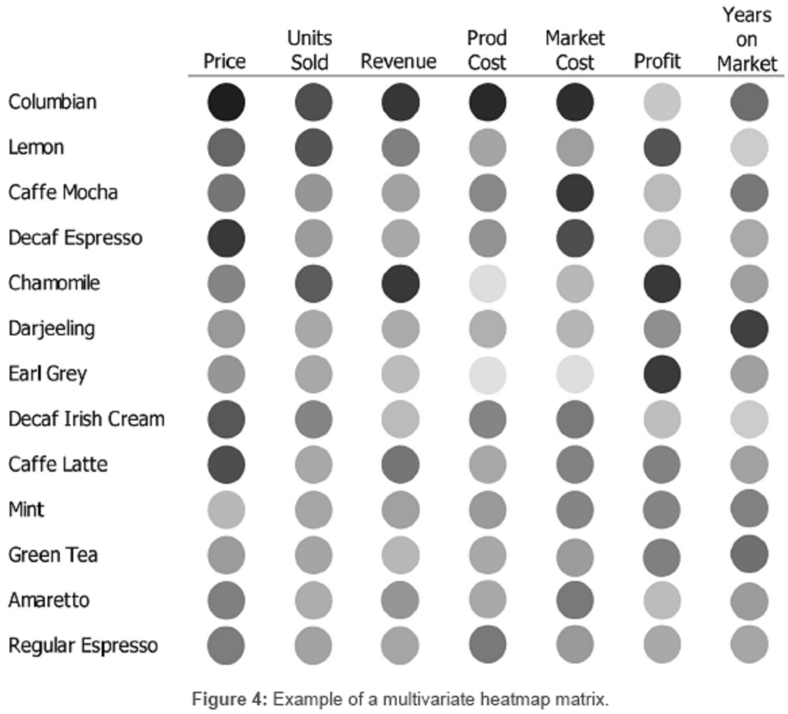


Figure 2.31: Multivariate heatmap matrix seen by Few (2009)

are represented for a set of coffees and teas. The values of the variable have been divided into classes. The classes are represented by different grey levels. The darker the circle, the higher the value of the respective variable - the brighter the circle the lower the value.

The diagram is quite condensed in information because two nominal variables and one ordered categorical are displayed.

Bibliography

Alfons, A. et al. (**2009**): *State-of-the-art in visualization of indicators in survey statistics*. Technical report, Vienna University of Technology.

Austria, S. (**2009**): *EINKOMMEN, ARMUT UND LEBENSBEDINGUNGEN: Ergebnisse aus EU-SILC 2007*. Official document, Statistics Austria.
URL http://www.statistik.at/web_de/static/einkommen_armut_und_lebensbedingungen_2007_035744.pdf

EEA (**2005**): *The European environment - State and outlook 2005*. Technical report, European Environment Agency, Kopenhagen, methodology: Methodology and main decision points, page 498.

EUROSTAT (**2008**): Energy, transport and environment indicators. Eurostat.

URL http://epp.eurostat.ec.europa.eu/cache/ITY_OFFPUB/KS-DK-08-001/EN/KS-DK-08-001-EN.PDF

EUROSTAT (**2009**): Europe in Figures. Eurostat Yearbook 2009. EUROSTAT.
URL http://epp.eurostat.ec.europa.eu/cache/ITY_OFFPUB/KS-CD-09-001/EN/KS-CD-09-001-EN.PDF

Few, S. (**2005**): *Intelligent Dashboard Design.*
URL http://www.perceptualedge.com/articles/dmreview/intelligent_dashboard.pdf

Few, S. (**2005a**): *Visualizing Multidimensional Data Through Time.*
URL http://www.perceptualedge.com/articles/dmreview/data_through_time.pdf

Few, S. (**2006f**): *An Introduction to Visual Multivariate Analysis.*
URL http://www.perceptualedge.com/articles/b-eye/visual_multivariate_analysis.pdf

Few, S. (**2007a**): *Visualizing Change. An Innovation in Time-Series Analysis.*
URL http://www.perceptualedge.com/articles/visual_business_intelligence/visualizing_change.pdf

Few, S. (**2007b**): *Dashboard Confusion Revisited.*
URL http://www.perceptualedge.com/articles/03-22-07.pdf

Few, S. (**2009**): *Introduction to Geographical Data Visualization.*
URL http://www.perceptualedge.com/articles/visual_business_intelligence/geographical_data_visualization.pdf

Few, S. (**2009a**): *Bullet Graph Design Specification.*
URL http://www.perceptualedge.com/articles/misc/Bullet_Graph_Design_Spec.pdf

Friendly, M. (**1994**): *Mosaic displays for multi-way contingency tables.* Journal of the American Statistical Association, 89, pp. 190–200.

Friendly, M. (**2003**): *Categorical Data Analysis with Graphics.*
URL http://www.math.yorku.ca/SCS/Courses/grcat/grcat.pdf

Hartigan, J. and **Kleiner, B.** (**1984**): *A mosaic of television ratings.* The American Statistician, 38, pp. 32–35.

Hulliger, B. and **Pooda, D. L.** (**2010**): *Assessment of Sustainable Development and Environmental Indicators. Report commissioned by the Federal Statistical Office, Section Environment, Sustainable Development, Agriculture.* Technical report, School of Business, University of Applied Sciences Northwestern Switzerland (FHNW).

Meyer, D., **Zeileis, A.** and **Hornik, K.** (**2009**): vcd: Visualizing Categorical HData. R package version 1.2-4.

NIST/SEMATECH (**2010**): *e-Handbook of Statistical Methods, http://www.itl.nist.gov/div898/handbook/ (02/06/2010).*
URL http://www.itl.nist.gov/div898/handbook/eda/section3/starplot.htm

Reis, F. and **Hirmo, R.** (**2009**): *Population and social conditions: Indicators on education expenditure ï£¡ 2006.*
URL http://epp.eurostat.ec.europa.eu/cache/ITY_OFFPUB/KS-QA-09-036/EN/KS-QA-09-036-EN.PDF

Robbins, N. B. (**2006**): *Dot Plots: A Useful Alternative to Bar Charts.*
URL http://www.perceptualedge.com/articles/b-eye/dot_plots.pdf

Tufte, E. (**2001**): The Visual Display of Quantitative Information. Graphics Press LLC.

Chapter 3

Evaluation Plots: Revision and Extensions

This section and Section 3.1 consists of parts of the master thesis of Zechner (2010) (which was written within the AMELI project) and Hulliger and Lussmann (2008). Section 3.2 is a more generalized reformulation of the ideas of Hulliger and Lussmann (2008). It then continuous to enhance the evaluation plots by non-linear estimations of the trend.

Different graphics are typically used for the evaluation of indicators, such as different smileys (from happy to sad) or colored symbols showing the tendency of an indicator. Moreover, different ways exist to evaluate an indicator, like evaluation of the

- current status,
- absolute change over time,
- change over time relative to a past value, or
- change over time relative to a target value.

Often a target value is missing and has to be determined by a political authority. As in every other statistical evaluation both, the variance and the accuracy of the indicator has to be assessed.

The quality of the indicators can be divided into thee categories:

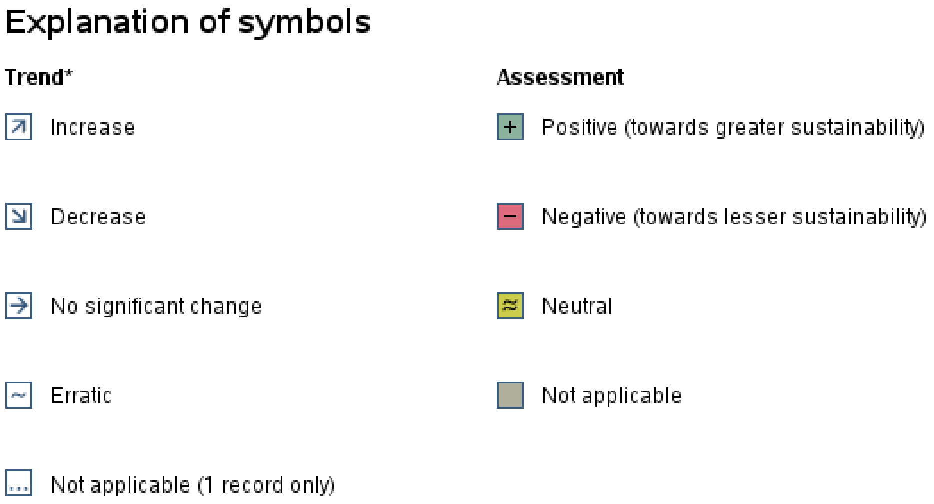
Relevancy refers to the closeness of the indicator, also the chosen methodology and the relevancy of the breakdown published.

Overall accuracy represents issues such as comparability of the data, reliability of data sources and the used methodology, and if the result can be validated (e.g. by sensitivity analysis).

Comparability over time takes a look at the completeness of time series and the consistency of the used methodology over time.

In Agency (1999) a semaphore code (traffic light colors) has been defined for indicators as a guideline by a scientific advisory group, consisting of 2300 European environment experts from all over the EU.

Another example is given by the Federal Statistical Office. The Federal Office for the Environment and the Swiss Federal Office for Spatial Development developed the indicator system *MONET* (German abbreviation for Monitoring Sustainable Development). This indicator system is monitoring the sustainable development, and its aims are to provide information about the current situation and trends in social, economic and environmental aspects of sustainable development. It also allows the comparison to other countries and is designed as an information source for the public, politicians and the Swiss Federal Government.



Explanation of symbols

Trend*	Assessment
↗ Increase	+ Positive (towards greater sustainability)
↘ Decrease	− Negative (towards lesser sustainability)
→ No significant change	≈ Neutral
~ Erratic	▨ Not applicable
... Not applicable (1 record only)	

Figure 3.1: The *MONET* indicator system.

Figure 3.1 presents the *MONET* indicator system. The sustainable development can be directly seen by the trend and the assessment. An increasing trend of the at-risk-of-poverty rate for example will lead to a negative assessment.

In this chapter discusses how to evaluate changes over time.

3.1 Example: Greenhouse gas

In the European Environment Agency (2005) another example is given for the evaluation of indicators. Every indicator is shown with an inaccuracy rate. The following example represents the measurement of greenhouse gases (short: GHG). The absolute values have an inaccuracy rate of ±20%, while the change to the past can be calculated with a variance of ±8%.

This indicator illustrates current trends in anthropogenic GHG emissions in relation to the EU and Member State targets. Emissions are presented by type of gas and weighted by their global warming potentials. There is a growing evidence that emissions of greenhouse gases are causing the global surface air temperatures to increase, resulting in a climate change. The potential consequences are fatal, rising sea levels which will lead to an increased frequency and intensity of floods, global warming will also lead to more droughts. Efforts to reduce or limit the effects of climate change are focused on limiting the emissions of all greenhouse gases covered by the Kyoto Protocol. This indicator supports the Commission's annual evaluation of progress in reducing emissions in the EU and the individual Member States to achieve the Kyoto Protocol targets.

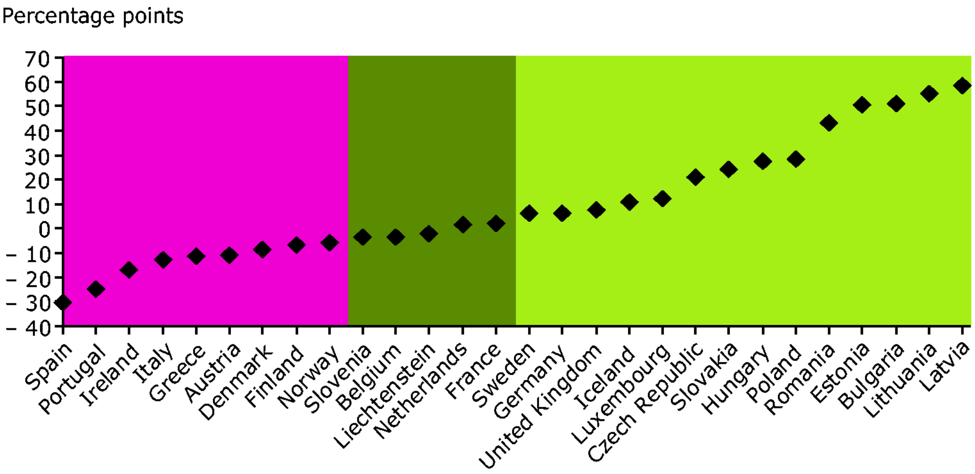


Figure 3.2: Emissions of ozone precursors, distance to NECD targets.

Figure 3.2 shows the information of GHG emissions of European countries. Those which are on the good side of the target-path are in the light-green section, countries with the indicator within a ±5% interval are dark-green and countries which are clearly on the bad side of the target-path are pink. Colors play an important role for visualization (for a detailed discussion about color and color space, see the master thesis of Zechner, 2010, which was written within the AMELI project). This is one of the bad examples of how to choose colors because the colors are gaudy and the saturation differs too much between the colors.

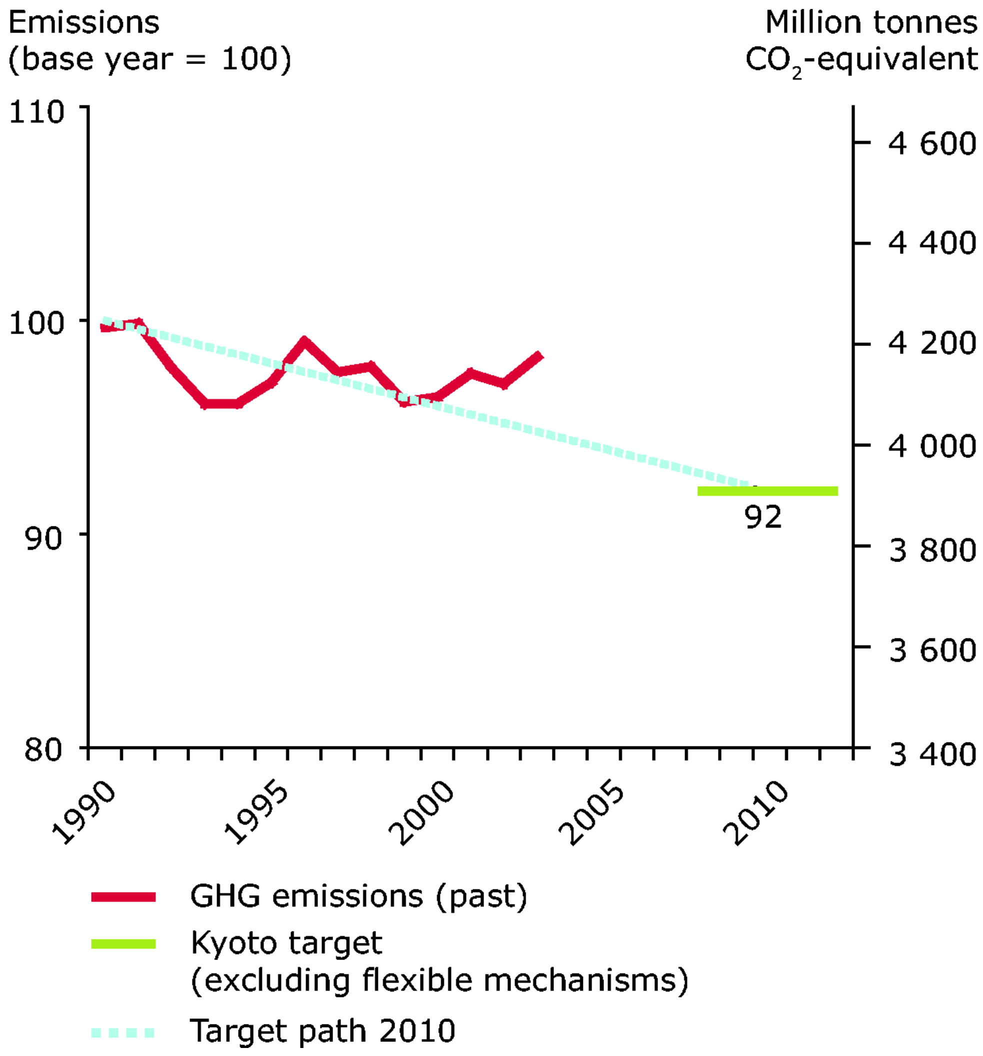


Figure 3.3: Development of EU-15 greenhouse gas emissions.

Figure 3.3[1] shows an example for the visualization of an indicator when a target value is given. From a base value of 100 in the year 1990 the target path aims at the Kyoto target of 92 in the year 2010. Or in other words the CO_2 emissions should decrease by 8% within 20 years.

3.2 Evaluation plots

To present indicators with symbols two different problems have to be faced. The evaluation and the visualization of the indicator. For graphical solutions there exist already different systems, like the one in the greenhouse gas example mentioned above.

The type of visualization depends also on the context. Are there different indicators in one concept, is there one indicator for different regions, or are we looking at one (or

[1] source: http://www.eea.europa.eu/

more) indicators over time?

The main idea behind this evaluation method is based on a paper by Hulliger and Lussmann (2008). This chapter generalizes and expands their approach.

The assumption is that there are three different evaluations for indicators:

- good
- neutral
- bad

The middle category is maybe better described as critical as it may drop in the bad zone. The definition of the parameter is a political process. The way from parameters to the evaluation should be transparent and should support political discussions.

3.2.1 Notation

An indicator is defined as a real-valued time series x_t. The indicator has been monitored from time t_1 until the current time t_N, or written in mathematical notation $t_1 \leq t \leq t_N$. The time t_S ($t_S \leq t_1$) is defined as a reference point or a start point in the past, while t_T ($t_T \geq t_N$) refers to a time in the future. Furthermore we assume without loss of generality to monitor the indicator once per year, so t usually represents the different years.

Reference values are denoted by x^*. The value at a given start time t_S is defined as x^*_S, while a target value at the time t_T is written as x^*_T.

3.2.2 Evaluation

As defined above the evaluation is working with three categories: good, neutral and bad. The evaluation of an indicator is defined as $Eval(x_t)$ with the following values

$$Eval(x_t) = \begin{cases} 1 & \text{good} \\ 0 & \text{neutral} \\ -1 & \text{bad} \end{cases} \tag{3.1}$$

Later on, the evaluation of an indicator will also depend on other quantities than on x_t. An increasing indicator can either be good or bad. An increasing indicator is assumed to be evaluated as negative, because for most Laeken indicators like the Gini coefficient, at-risk-of-poverty rate or the Quintile Share Ratio perform better if they have a low value.

Now different options of the evaluation of an indicator will be discussed, depending if the deviation from an initial value, a target value or from a given course is of interest.

3.2.3 Deviation from an initial value

The difference from the initial value x^*_S is defined as

$$\delta_t(S) = x_t - x^*_S \quad . \tag{3.2}$$

The initial value is either the mean value over a few years or can be defined by other criteria. The aim is an improvement of the indicator over time, and by the definition from above this refers to a decreasing value of the indicator.

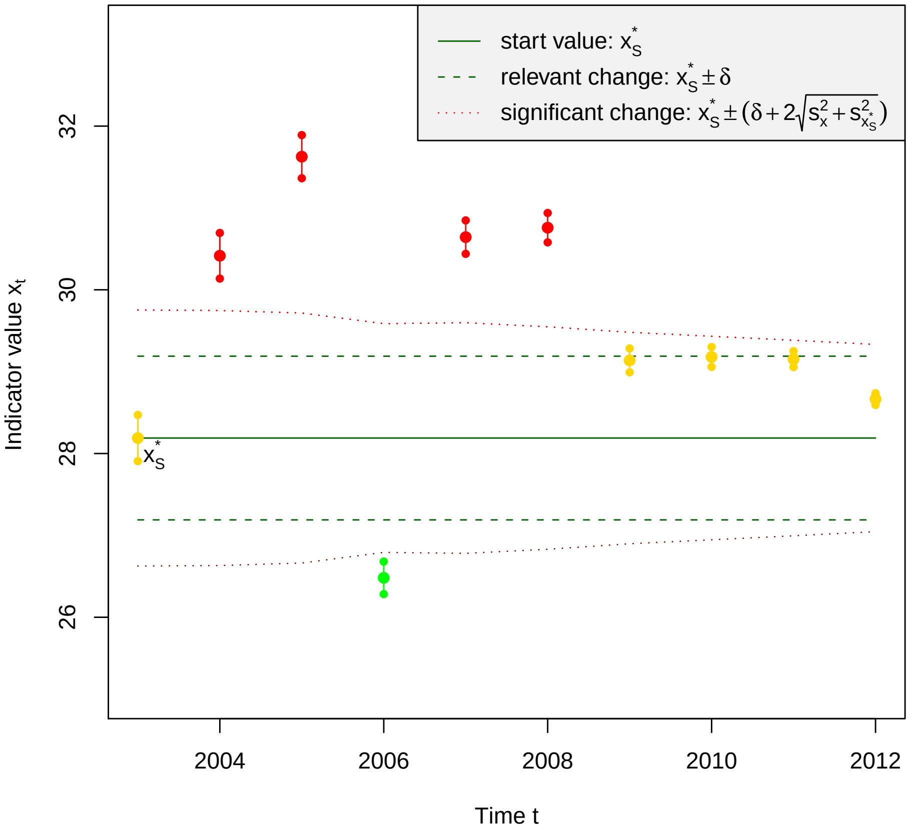


Figure 3.4: The evaluation plot for a given initial value.

For the evaluation of the indicator a reference value δ (the two dashed lines in Figure 3.5) is needed, which defines whether a change is relevant or not. This is one of the difficult decisions for the evaluation of an indicator, because it is mainly based on the knowledge of experts. However, the question arises how a relevant change is defined? It might be possible to estimate s_x, the standard deviation of x_t. Furthermore it might also be possible to estimate $s_{x_S^*}$, the standard deviation of x_S^*. However, if the standard deviation $s_{x_S^*}$ cannot be estimated adequately, and experts have to determine it. If s_x cannot be estimated, s_x has to be set at zero ($s_x = 0$) to prevent wrong evaluations. Note, it is possible that a relation between δ and $s_{x_S^*}$ exists. Experts would not choose a δ smaller than $s_{x_S^*}$ for example. Now 'relevant changes' have to be defined. It does not make any sense to name changes smaller than $s_{x_S^*}$ or even better, smaller than $2 \cdot s_{x_S^*}$ as relevant. This follows from the fact that if a data distribution is approximately normal then approximately 95% of the data values are within mean plus/minus two standard deviations of the mean.

The relevant deviation δ can also be seen as the real variability of the phenomena which got observed. The standard deviation s_δ could be seen as a measurement variability, which will be added to the real variability and the measurement variability may change over time. In Figure 3.5 the measurement variability is visualized via vertical lines at the indicator values. However, in practice the two variabilities are often hard or even impossible to differ.

The variance of $\delta_t(S)$ is defined as $s^2_{\delta(S)} = s^2_x + s^2_{x_S^*}$ if the covariance between x_t and x_S^* is zero, so $cov(x_t, x_S^*) = 0$. The 95% confidence interval for the deviation $\delta_t(S)$ is then approximately $[\delta_t(S) - 2 \cdot s_{\delta(S)}, \delta_t(S) + 2 \cdot s_{\delta(S)}]$. The dotted lines in Figure 3.5 illustrate this confidence interval.

If zero is not included in this confidence interval the deviation differs relevant from zero. If the confidence interval does not overlap with any part of the interval $[-\delta, \delta]$, then the deviation differs not only relevant from zero, but it is also significant.

In the case that high values of the indicator are bad, following valuation can be used:

$$Eval(x_t, x_S^*, \delta, s_{\delta(S)}) = \begin{cases} -1 & x_t > x_S^* + \delta + 2 \cdot s_{\delta(S)} \\ 1 & x_t < x_S^* - \delta - 2 \cdot s_{\delta(S)} \\ 0 & \text{else} \end{cases} \tag{3.3}$$

3.2.4 Deviation from a target value

A target value is mostly defined by politicians, but it could also be a natural value. Now the difference from the actual value to the target value can be evaluated. The target value at time t_T is defined as x_T^*, and the difference between that target value and the time series is defined as

$$\delta_t(T) = x_t - x_T^* \tag{3.4}$$

As mentioned in the previous section a reference value δ is needed which defines the relevant distance. $\delta_t(T)$ has again variance greater zero, and if x_T^* is a constant value without a zero variance then $s_{\delta(T)}^2 = s_x^2$. The evaluation is the same as in the section before:

$$Eval(x_t, x_T^*, \delta, s_{\delta(T)}) = \begin{cases} -1 & x_t > x_T^* + \delta + 2 \cdot s_{\delta(T)} \\ 1 & x_t < x_T^* - \delta - 2 \cdot s_{\delta(T)} \\ 0 & \text{else} \end{cases} \tag{3.5}$$

So an indicator is defined as 'bad', if its value is above a threshold, where a positive deviance from the target value (in the 'bad' direction of the indicator) is significant.

3.2.5 Deviation from a given course

In this section starting value x_S^* and a given target value x_T^* is given, together with a period of time from t_S to t_T in which the target value should be reached. The interesting part is the distance between the indicator and the course which leads from the starting value to the target value.

The linear course from x_S^* to x_T^* for $t_S \leq t \leq t_T$ is

$$x_t^* = x_S^* + \frac{x_T^* - x_S^*}{t_T - t_S} \cdot (t - t_S) \tag{3.6}$$

Other courses are also possible, like asymptotic ones (e.g. halving of the distance every year), but here the focus is on the linear one (one possible non-linear course will be discussed later). The quotient

$$\beta = \frac{x_T^* - x_S^*}{t_T - t_S} \tag{3.7}$$

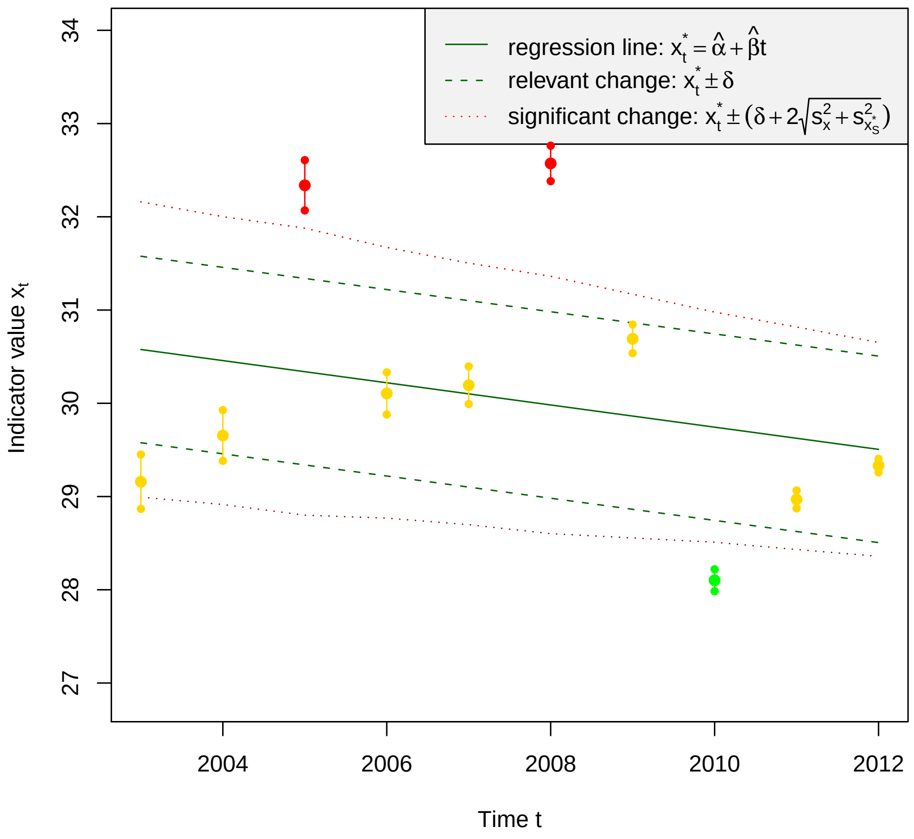


Figure 3.5: The evaluation plot for a given course.

is the slope of the line and describes the desired change in a time period. For every point in time a target value is given as a reference point.

The evaluation is calculated as follows:

$$Eval(x_t, x_S^*, \delta, \beta, s_\delta^2) = \begin{cases} -1 & x_t > x_S^* + \delta + \beta(t - t_S) + 2 \cdot s_\delta^2 \\ 1 & x_t < x_S^* - \delta + \beta(t - t_S) - 2 \cdot s_\delta^2 \\ 0 & \text{else} \end{cases} \tag{3.8}$$

If and only if there is an significant deviation from the course, the evaluation can be considered as good or bad. In the middle band around the course the evaluation is neutral, as the evaluation is neither significantly worse than the course, nor is it on the good side of the course.

3.2.6 Calculating the linear trend of an indicator

Not for every indicator or every region a course or target value is given. In order to get at least a better picture of the indicator and its trend a linear regression is applied. One possibility is to estimate the trend $\hat{\beta}$ and the intercept $\hat{\alpha}$ of an indicator by least squares estimation. Let $\hat{\beta}$ be the trend estimated by minimizing the sum of squared distances between the observed indicator x_t and the regression line x_t^*, this distances are called residuals.

$$x_t = \hat{\alpha} + t \cdot \hat{\beta} + \varepsilon, \qquad t_1 \leq t \leq t_N \tag{3.9}$$

where ε is random variable (errors) which accounts for the discrepancy between the actually observed responses x_t and the predicted outcomes $\hat{\alpha} + t \cdot \hat{\beta}$.

The evaluation is calculated as follows:

$$Eval(x_t, \delta, s_\delta^2) = \begin{cases} -1 & x_t > \hat{\alpha} + t \cdot \hat{\beta} + \delta + 2 \cdot s_\delta^2 \\ 1 & x_t < \hat{\alpha} + t \cdot \hat{\beta} - \delta - 2 \cdot s_\delta^2 \\ 0 & \text{else} \end{cases} \tag{3.10}$$

This approach has one great disadvantage, its non-robustness. Robustness is needed to face outliers. Outliers are observations that severely deviate from the linear data trend. The aim is to apply a regression method which is not sensitive if outliers occur. For this reason the MM-estimator got chosen for the robust regression. For more information on robust regression take a look at YOHAI (1987).

Figure 3.6 presents the difference between robust and non-robust regression methods. Outliers can have a strong influence on the OLS regression line.

3.2.7 Calculating a non-linear trend of an indicator

For a nonlinear regression the LOESS model is used. LOESS (locally weighted scatterplot smoothing) was proposed by CLEVELAND (1979) as a model which uses locally weighted polynomial regression. For every point in the data a polynomial of low degree is fit to a subset of the data. The observations closer to the point have a higher weight than points further away. The default weight function for LOESS is usually a tri-cube weight function:

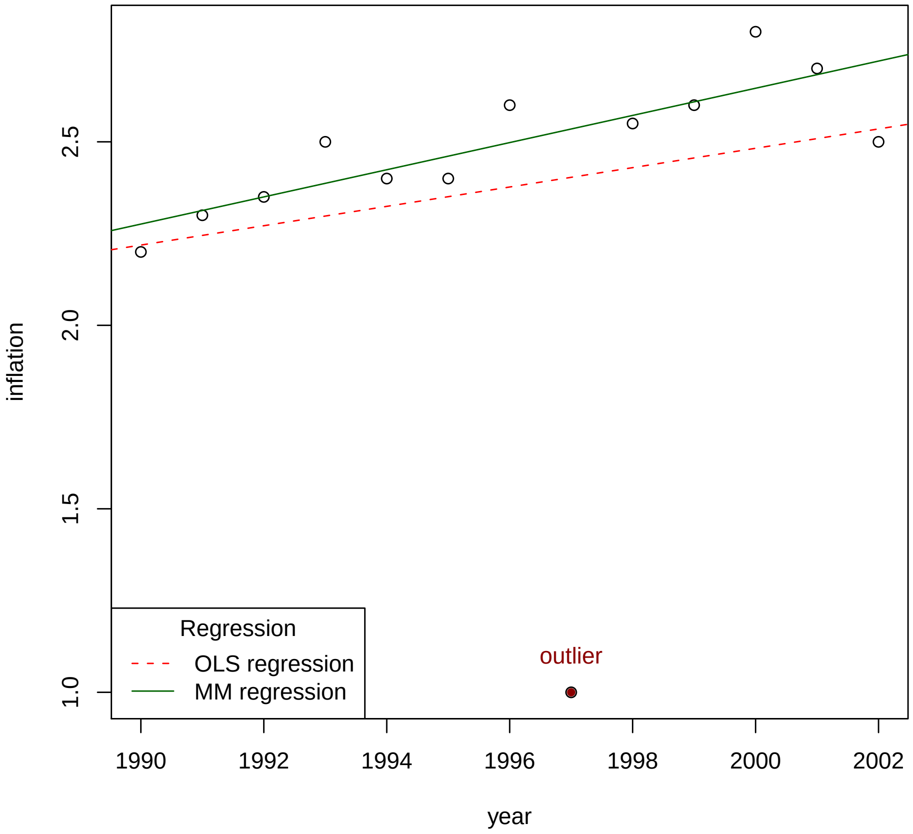


Figure 3.6: Comparison between robust and non-robust regression. The point in dark red is an outlier.

$$w(x) = \begin{cases} (1 - |x|^3)^3 & \text{for } |x| < 1 \\ 0 & \text{for } |x| \geq 1 \end{cases} .$$

For every point in the data x a polynomial of low degree is fit to a subset of the data. Points closer to the estimated response get more weight than points farther away. The degree of that polynomial as well as a smoothing parameter can be chosen by the user. How this works in practice can be seen in Figure 3.9.

3.3 Implementation in R

Based on a function written by Hulliger and Lussmann (2008, Appendix B) an extended function was written to evaluate and visualize indicators. The input for this function *indEval()* can either be a time series or a vector with values and an additional time vector.

For the following visualizations the Gini coefficient of the equivalized income for Austria[2] in the years 1995-2008 got used. The indicator for 2002 is missing, for the visualization the value was imputed by the median of the time series, and for the years 2004-2008 the results from the *EU-SILC* data are used.

Table 3.1: Gini coefficient for Austria in the years 1995-2008

year	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008
Gini	27.0	26.0	25.0	24.0	26.0	24.0	24.0	26.0	27.7	25.8	26.1	25.3	26.1	26.2

This is the *indEval()* function including all the parameters which can be set by the user:

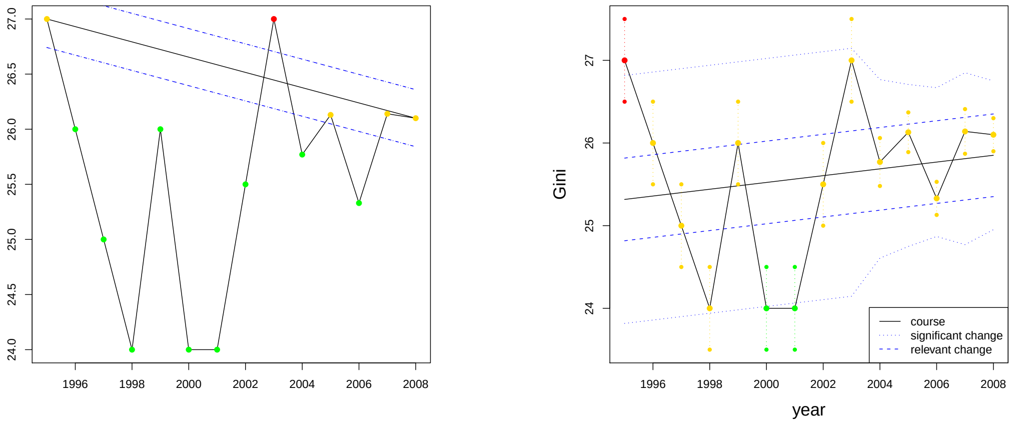
```
`indEval` <-
function(x, x_time=1:length(x), sderr=0, dr=0.01*median(x),
x0=x[1], t0=x_time[1], x1=x[length(x)], t1=x_time[length(x_time)],
betax, eval.labels=TRUE, good.ind="low", ind.size=1,
Legend=FALSE, placeLegend="topleft",
show.axes=TRUE, x.lab="Time", y.lab="Indicator",
parList=list(cex.lab=1.5, pch=19), sparkline=FALSE, regression="none",
plot.title=title(main = NULL, sub = NULL, xlab = NULL, ylab = NULL),
...)
```

The variable x is the data, which can either be a vector or a time series. The other default settings are sensitive to changes.

[2] http://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=ilc_sic2&lang=en

3.3.1 Default visualization

With the above data given in Table 3.1 the default plot of the time series in Figure 3.7(a) looks only slightly different from a normal *plot* function. Additional information in the default plot of the *indEval* function is represented as the dotted lines and the different colors of the Gini coefficient. The dotted lines determine the region where the indicator will be called 'neutral', it is the region around the line reaching from the first value to the last value in our time series. By default this marks a 1% margin of the median of the indicator around the course. Outside this corridor an indicator is called 'bad' if it has a higher value, or 'good' if it has a lower value.



(a) Default visualization

(b) Visualization of the robust regression and standard deviation of the indicator

Figure 3.7: Two plots with the *indEval* function of the Gini coefficient of Austria in the years 1995-2008.

There are other indicators of interest which are called 'better' if they have a high value, or are staying within/outside a corridor. This can be changed via the option for the parameter `good.ind`:

- `"high"`
- `"in.funnel"`
- `"out.funnel"`

Figure 3.7(b) represents an advanced plot of the *indEval()* function, using a robust regression. There are three different settings available for the `regression` parameter:

- `"none"` - linear trend from the first to the last value
- `"LM"` - robust linear regression

- `"LOESS"` - non-linear regression

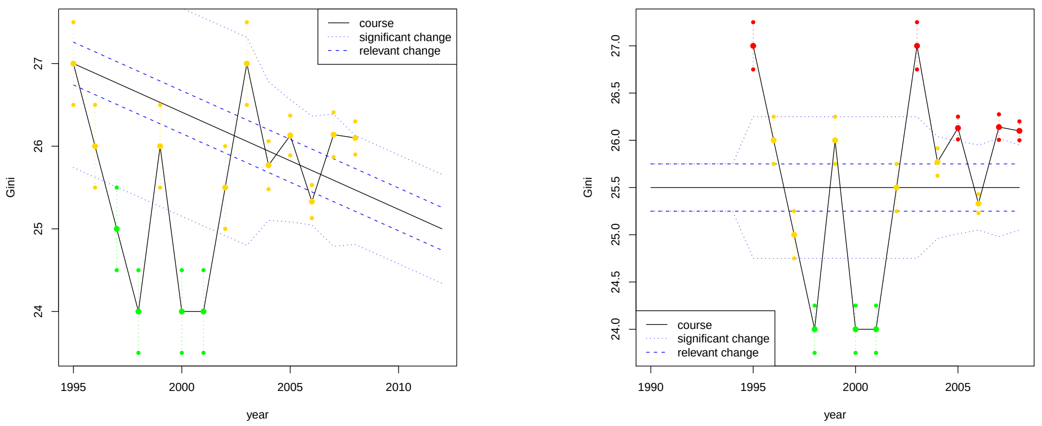
Additionally the measured inaccuracy of the indicators are added to the plot, which have been calculated with bootstrap resampling (take a look at Section 6.3 for details) More graphical parameters for the *indEval()* function will be explained in the following subsections.

3.3.2 Visualization of a target value

This is probably the most used way to visualize indicators. Target values are used in almost all parts of science. There are several examples for this case:

Ecologists use target values for greenhouse gas, which should decrease over time and reach a small value in future, like shown in Figure 3.3. Economists use them for economic growth, the higher the better, on the other side a low value for the unemployment rate is aimed at.

The Gini coefficients in Table 3.1 should be as low as possible, since a low Gini coefficient indicates a more equal income distribution. An artificial long term target value x_T^* of the Gini coefficient of 25 in the year $t_T = 2012$ would lead to following visualization in Figure 3.8(a).



(a) Target value plot

(b) Initial value plot

Figure 3.8: Gini coefficient with a specified target value in the future (a) and a given initial value (b)

3.3.3 Visualization of a given initial value

If x_S^* is given the visualization of the development since that time can be made. Assume that politicians have defined a target for the Gini coefficient in the year 1990 of 25.5 for the following years. The plot shown in Figure 3.8(b) would present the outcome of such a setting. Note that in this plot the measuring inaccuracy (standard deviation of the estimated Gini coefficient) was halved. If the first known value is too far away in the past from the time series, the figure could look dispersed. In that case the visualization could lead to wrong conclusions.

3.3.4 Nonlinear Regression

Taking a look at the last option for the `regression` parameter for the *indEval()* function: `LOESS`. The LOESS function mentioned in Section 3.2.7 performs a nonlinear regression.

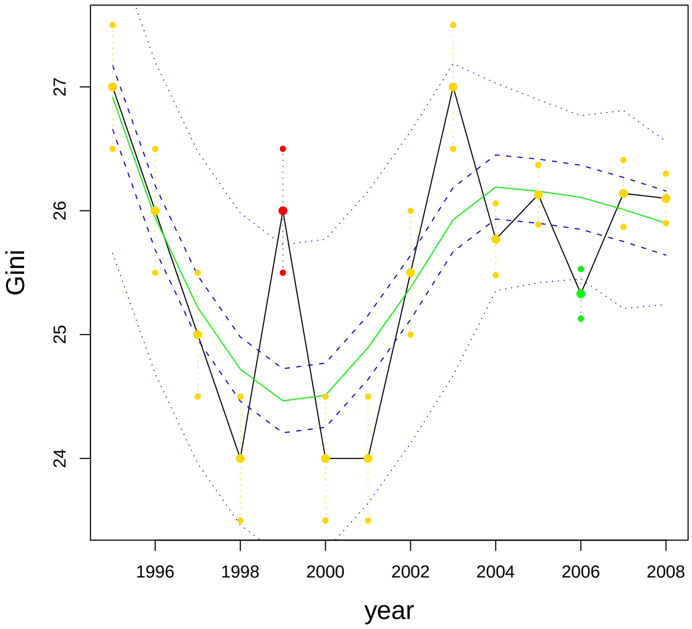


Figure 3.9: LOESS regression.

Figure 3.9 presents the outcome of the LOESS regression for our given data. With the non-linear regression it is possible to see some kind of cycle which could be connected to the business cycle for example. However, the focus is on the visualization the data, the interpretation of the results should be left over for experts.

Bibliography

Agency, E. E. (**1999**): *Towards environmental pressure indicators for the EU*. Technical report, European Commission, Office for Official Publications of the European Communities, Luxembourg, [Online; accessed November-2009].
URL esl.jrc.it/envind/tepi99rp.pdf

Cleveland, W. S. (**1979**): *Robust Locally Weighted Regression and Smoothing Scatterplots*. Journal of the American Statistical Association, 74, pp. 829–836.

European Environment Agency (**2005**): *The European environment - State and outlook 2005*. Technical report, BFS/BUWAL/ARE.

Hulliger, B. and **Lussmann, D.** (**2008**): *Bewertung der Nachhaltigkeits- und Umwelt-Indikatoren*. Technical report, Institute for Competitiveness and Communication, University of Applied Sciences, Northwestern Switzerland.

Yohai, V. (**1987**): *High breakdown-point and high efficiency estimates for regresssion*. The Annals of Statistics, 15, pp. 642–656.

Zechner, S. (**2010**): *Visualization of indicators in R with application to EU-SILC*. Supervisors: P. Filzmoser, M. Templ.

Chapter 4

Some Issues in Mapping: Projections and Correlations of Indicators in Time

4.1 Projections

The map of Europe that has been provided by Eurostat to the AMELI consortium is given in long/lat coordinates which is fine when visualising the map of Europe. However, when extracting and visualizing smaller regions such as countries leads to distorted representations of the corresponding countries when using lat/long representation.

Thus, projection into other coordinate systems, which are better suited to present smaller regions, is absolutely necessary. The aim is that projections should be made interactively when selecting countries for plotting by a mouse click on the underlying figure. However, one major problem is that every selected region needs other parameters for a suitable projection. This is also true if the map of Europe is given in another coordinate system - each subregion needs other parameters for a fine representation in a map. For example, the reference coordinates for the Labert Area Level projection must be provided which is estimated by the center of the outer borders of corresponding subregion.

Figure 4.1 shows the map of Europe in lat/long representation. When selecting one country, e.g. Austria, and presenting this subregion in a map, the resulting map looks quite distorted. However, if a projection is made with all the polygon lines of Austria, the map improves. This is shown in Figure 4.2(b). Another example is given in Figure refirelandF (distorted map in lat/long) and Figure 4.3(b) (projected map to Area Lambert with certain parameters).

Functions which are already developed for mapping:

- `subsetNUTS()`: selects regions.
- `projection()`: does the projection, after
- `plot.subsetNUTS()` is applied.

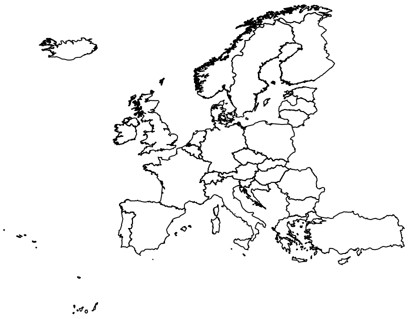


Figure 4.1: Map of Europe in long/lat.

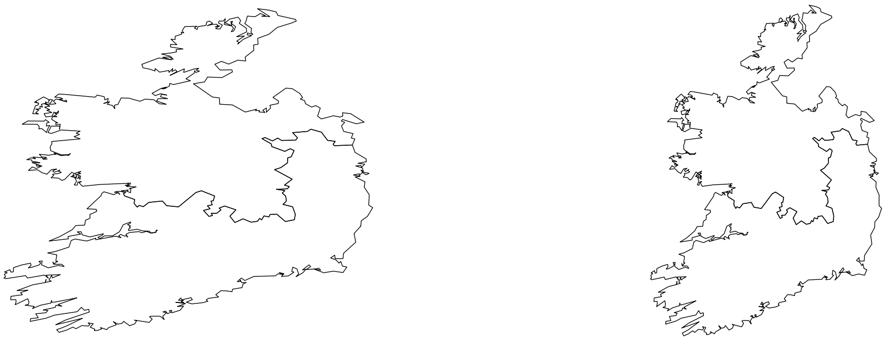
(a) Subset of the map of Europe (Austria) in long-lat.

(b) Subset of the map of Europe (Austria) after projection into Lambert Area Level coordinates using an automatic calculation of necessary parameters for the projection.

Figure 4.2: Representation of Austria in different coordinate systems.

- plus functions for interactive selection, maps in continuous color scale, etc. are already developed.

This functions allows to extract information of a spatial data frame, which has a special structure in R. A spatial data frame is typically a S4 class object consisting of list of list of lists (see Lewin-Koh et al., 2011). The developed functions therefore turns out to be very useful when representing statistical information on a subset of the map of Europe.



(a) Subset of the map of Europe (Ireland) in long-lat.

(b) Subset of the map of Europe (Ireland) after projection into Lambert Area Level coordinates using an automatic calculation of necessary parameters for the projection.

Figure 4.3: Representation of Ireland in different coordinate systems.

Bibliography

Lewin-Koh, N., **Bivand, R.** and **various contributors** (**2011**): maptools: Tools for reading and handling spatial objects. R package version 0.8-3.
URL http://CRAN.R-project.org/package=maptools

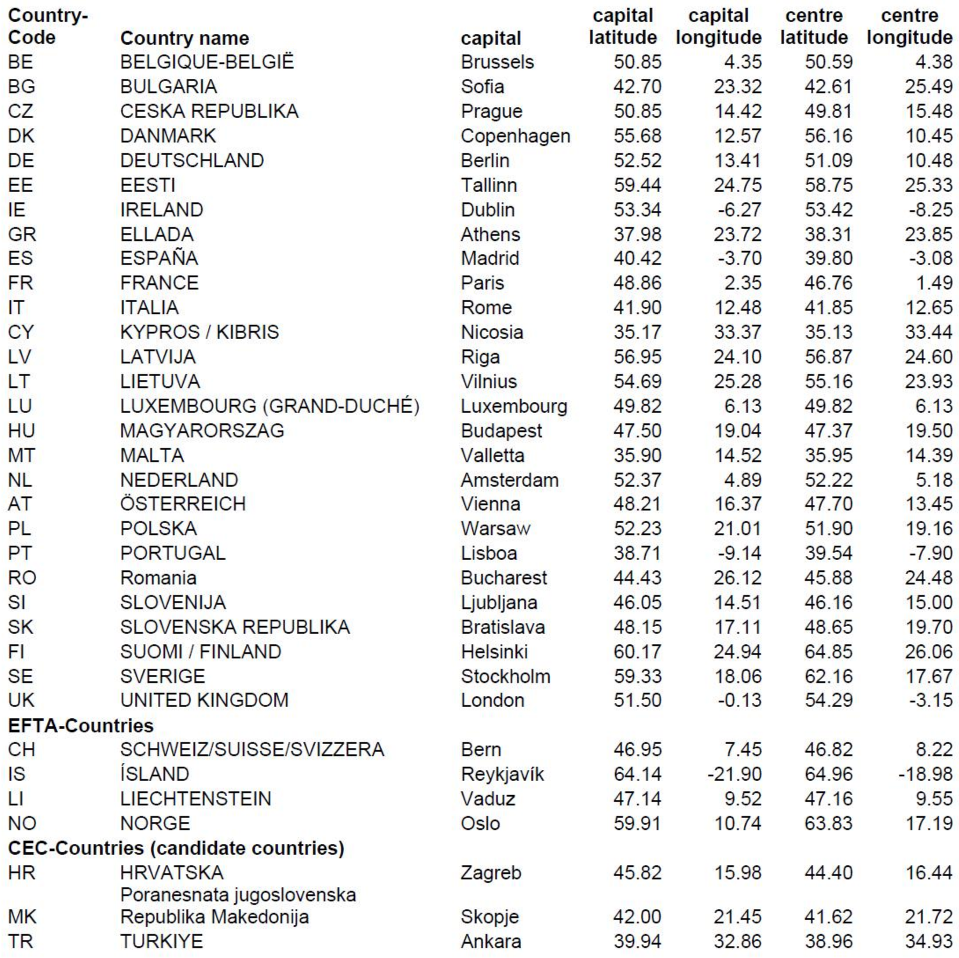
Chapter 5

The Checkerplot

The initial question was, how to display social indicators for several countries in one plot and how to present these data in a relative simple and comprehensible manner. The idea for the subsequent plot has been inspired by thematic maps. Thematic maps are, as presented above, geographical maps which usually display categorical values. Positive is by presenting statistical data in a thematic map, that countries are rapidly recognisable through its typical contour and position. But there is unequal space for each country to present data and it is only possible to show simple statistics. Thus the idea was to create a stylised map in a rectangular form, in which position and neighbourhood are approximately maintained and where exists same plotting region for each country. To display this stylised map, it is convenient to use a multi-panel chart, a so called trellis plot. We will in succession call it ”checker plot”. The checker board layout provides a way to display graphics on countries (or any other region of the World) in a way that each country obtains the same rectangular plotting region while maintaining as much as possible the position of the country. Each panel of this checker plot will be used to present data for one country.

5.1 Presentation and order of the countries in a checker plot

To present the countries in a multi-panel chart we have first to determine an order of presentation. Currently the EU includes 27 member states. There are three candidate countries and, with four other countries the EU has agreements. So there are in total 34 states for which we have to present data. For this a grid of 6 x 6 appears reasonable. Our objective is to find a checker board type of arrangement of plot frames where the position of a particular field of the checker board would reflect as close as possible the geographic position of the country depicted in the field. The aim is to combine the strength of the familiar map of Europe, easy spotting of a particular country, and of a Trellis display where each country obtains the same space for display and thus it is not just the geographical surface of a country which determines the size of the plot per country.



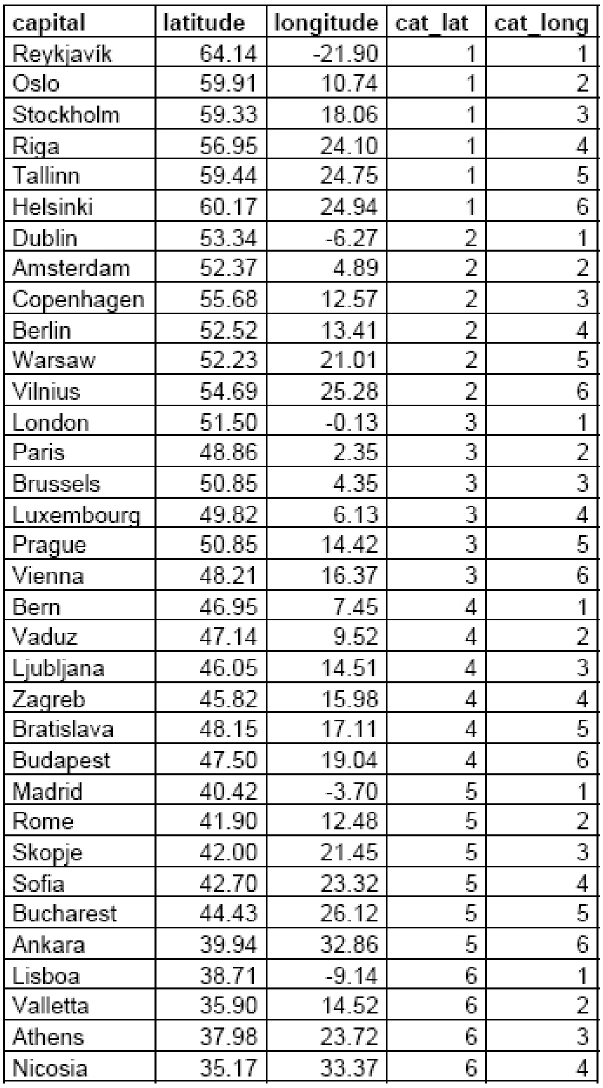
Country- Code	Country name	capital	capital latitude	capital longitude	centre latitude	centre longitude
BE	BELGIQUE-BELGIË	Brussels	50.85	4.35	50.59	4.38
BG	BULGARIA	Sofia	42.70	23.32	42.61	25.49
CZ	CESKA REPUBLIKA	Prague	50.85	14.42	49.81	15.48
DK	DANMARK	Copenhagen	55.68	12.57	56.16	10.45
DE	DEUTSCHLAND	Berlin	52.52	13.41	51.09	10.48
EE	EESTI	Tallinn	59.44	24.75	58.75	25.33
IE	IRELAND	Dublin	53.34	-6.27	53.42	-8.25
GR	ELLADA	Athens	37.98	23.72	38.31	23.85
ES	ESPAÑA	Madrid	40.42	-3.70	39.80	-3.08
FR	FRANCE	Paris	48.86	2.35	46.76	1.49
IT	ITALIA	Rome	41.90	12.48	41.85	12.65
CY	KYPROS / KIBRIS	Nicosia	35.17	33.37	35.13	33.44
LV	LATVIJA	Riga	56.95	24.10	56.87	24.60
LT	LIETUVA	Vilnius	54.69	25.28	55.16	23.93
LU	LUXEMBOURG (GRAND-DUCHÉ)	Luxembourg	49.82	6.13	49.82	6.13
HU	MAGYARORSZAG	Budapest	47.50	19.04	47.37	19.50
MT	MALTA	Valletta	35.90	14.52	35.95	14.39
NL	NEDERLAND	Amsterdam	52.37	4.89	52.22	5.18
AT	ÖSTERREICH	Vienna	48.21	16.37	47.70	13.45
PL	POLSKA	Warsaw	52.23	21.01	51.90	19.16
PT	PORTUGAL	Lisboa	38.71	-9.14	39.54	-7.90
RO	Romania	Bucharest	44.43	26.12	45.88	24.48
SI	SLOVENIJA	Ljubljana	46.05	14.51	46.16	15.00
SK	SLOVENSKA REPUBLIKA	Bratislava	48.15	17.11	48.65	19.70
FI	SUOMI / FINLAND	Helsinki	60.17	24.94	64.85	26.06
SE	SVERIGE	Stockholm	59.33	18.06	62.16	17.67
UK	UNITED KINGDOM	London	51.50	-0.13	54.29	-3.15
EFTA-Countries						
CH	SCHWEIZ/SUISSE/SVIZZERA	Bern	46.95	7.45	46.82	8.22
IS	ÍSLAND	Reykjavík	64.14	-21.90	64.96	-18.98
LI	LIECHTENSTEIN	Vaduz	47.14	9.52	47.16	9.55
NO	NORGE	Oslo	59.91	10.74	63.83	17.19
CEC-Countries (candidate countries)						
HR	HRVATSKA	Zagreb	45.82	15.98	44.40	16.44
MK	Poranesnata jugoslovenska Republika Makedonija	Skopje	42.00	21.45	41.62	21.72
TR	TURKIYE	Ankara	39.94	32.86	38.96	34.93

Table 5.1: Geographical coordinates of capitals and geographical centres of European states

The latter is a problem in particular for small countries like the Benelux countries or the Balkan countries.

First of all we have to find out an satisfiable presentation order. So we check some options: For the whole list of the EU-27 as well as of the candidate countries and the non-EU-countries like Island, Norway and Switzerland - we have listed the geographical coordinates of their capitals as well as their geographical centres in Table 5.1. [1] The geographical centres have to be understood as the centre of gravity of the respective country.

[1] The coordinates of geographical centres we adopted from the website http://www.earthtools.org and the coordinates of the capital cities are from on the site http://www.vivid-planet.com/sandkiste/google_maps_koordinaten.



capital	latitude	longitude	cat_lat	cat_long
Reykjavík	64.14	-21.90	1	1
Oslo	59.91	10.74	1	2
Stockholm	59.33	18.06	1	3
Riga	56.95	24.10	1	4
Tallinn	59.44	24.75	1	5
Helsinki	60.17	24.94	1	6
Dublin	53.34	-6.27	2	1
Amsterdam	52.37	4.89	2	2
Copenhagen	55.68	12.57	2	3
Berlin	52.52	13.41	2	4
Warsaw	52.23	21.01	2	5
Vilnius	54.69	25.28	2	6
London	51.50	-0.13	3	1
Paris	48.86	2.35	3	2
Brussels	50.85	4.35	3	3
Luxembourg	49.82	6.13	3	4
Prague	50.85	14.42	3	5
Vienna	48.21	16.37	3	6
Bern	46.95	7.45	4	1
Vaduz	47.14	9.52	4	2
Ljubljana	46.05	14.51	4	3
Zagreb	45.82	15.98	4	4
Bratislava	48.15	17.11	4	5
Budapest	47.50	19.04	4	6
Madrid	40.42	-3.70	5	1
Rome	41.90	12.48	5	2
Skopje	42.00	21.45	5	3
Sofia	42.70	23.32	5	4
Bucharest	44.43	26.12	5	5
Ankara	39.94	32.86	5	6
Lisboa	38.71	-9.14	6	1
Valletta	35.90	14.52	6	2
Athens	37.98	23.72	6	3
Nicosia	35.17	33.37	6	4

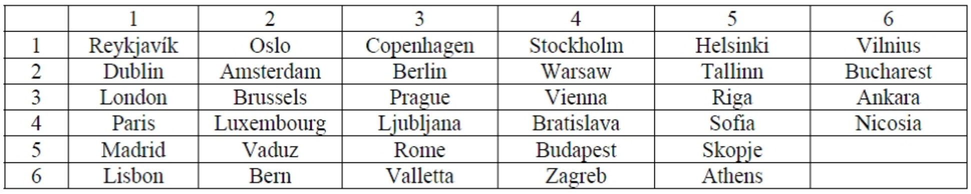
Table 5.2: Geographical latitudes and longitudes of capital cities, ranked by latitudes then by longitudes

5.2 Center of a country and heuristic arrangement

One reference point is needed for each country. This can be either the geographical coordinates of capital cities is chosen or the geographical center.

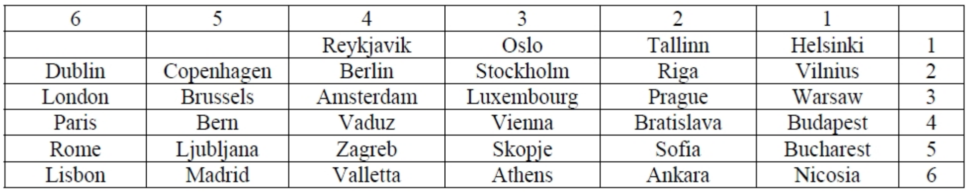
Table 5.2 shows the coordinates of the capitals and the geographical centres of the European states. This information is used in the following when determining the optimal arrangement of these countries in a grid.

For the second option (Table 5.3) the longitudes of the capitals have first been sorted by ascending values. Then the longitudes have been ranked in six categories. The first six longitudinal values belong to the category 1 and so on. The first five categories contain each six capitals and the sixth category four capitals (that's why we only have 34 values). Then in each longitude-category, the capitals have been sorted by descending latitude



	1	2	3	4	5	6
1	Reykjavík	Oslo	Copenhagen	Stockholm	Helsinki	Vilnius
2	Dublin	Amsterdam	Berlin	Warsaw	Tallinn	Bucharest
3	London	Brussels	Prague	Vienna	Riga	Ankara
4	Paris	Luxembourg	Ljubljana	Bratislava	Sofia	Nicosia
5	Madrid	Vaduz	Rome	Budapest	Skopje	
6	Lisbon	Bern	Valletta	Zagreb	Athens	

Table 5.3: Grid order option 2 (by coordinates of capital cities)



6	5	4	3	2	1	
		Reykjavik	Oslo	Tallinn	Helsinki	1
Dublin	Copenhagen	Berlin	Stockholm	Riga	Vilnius	2
London	Brussels	Amsterdam	Luxembourg	Prague	Warsaw	3
Paris	Bern	Vaduz	Vienna	Bratislava	Budapest	4
Rome	Ljubljana	Zagreb	Skopje	Sofia	Bucharest	5
Lisbon	Madrid	Valletta	Athens	Ankara	Nicosia	6

Table 5.4: Grid order option 3 (by coordinates of capital cities)

values and inside of each category ranked as well from 1 to 6.

In arrangement option 3 the capitals have first been sorted by ascending values of latitudes. Like in the previous options, six categories have been built. Then the categories itself have been sorted by descending longitudinal values and ranked from 1 to 6. In contrast to the options 1 and 2, the grid boxes have been filled up by beginning with the lower right box. So the upper two panels on the left will remain blank (Table 5.4).

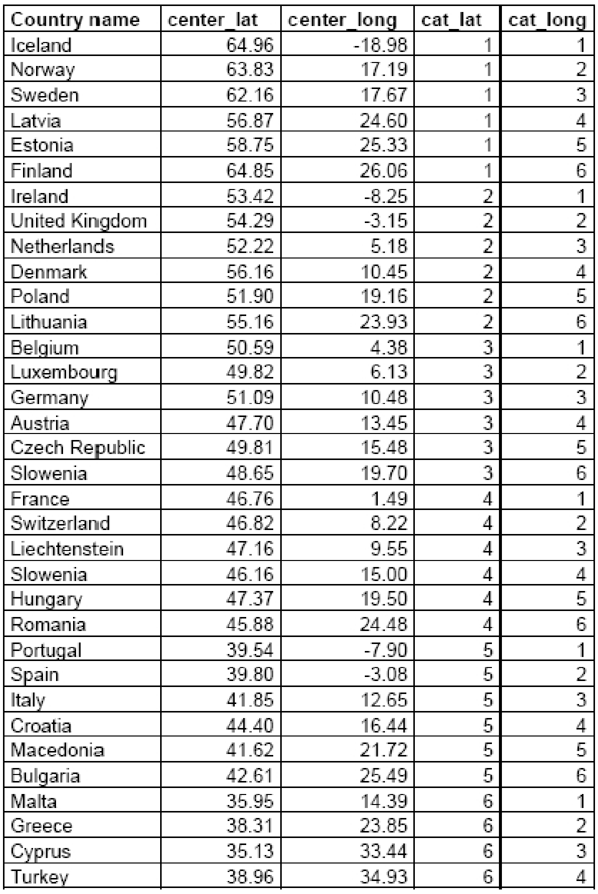
We did the same procedure as described above also with geographical centres of the countries. In arrangement option 4 (Table 5.5) the latitudes have first been sorted by descending values. Then by ascending longitude values (Table 5.5).

For option 5 the geographical centers have been sorted by ascending values of longitudes. Then in the respective categories sorted by descending latitude values (Tab 5.7).

In the sixth and last tested arrangement option geographical centers of the countries have first been sorted by ascending values of latitudes. Then sorted by descending longitude values (Table 5.8).

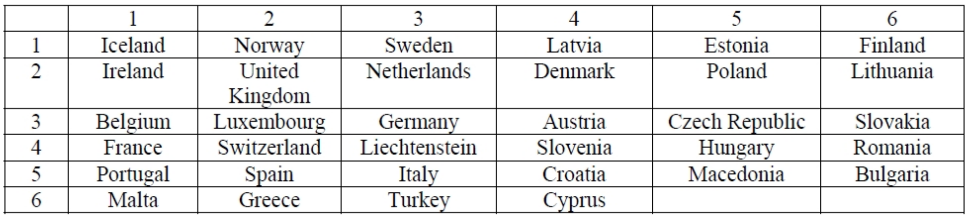
5.3 Determination of arrangement order

Obviously some neighbourhood relations are cut because the arrangement over rows or columns condenses or expands the true extensions. The optimal display thus would have minimal distance to the true geographic position and minimal distance to the neighbours. When we compare all options based on heuristic approaches, we tend to option 1 (for the optimal checker, see Section 5.4. It is a table with strong affinity to the real geographical position of the capitals and countries respectively. The presentation where the latitudinal values have been prioritized seems a more adequate option. Thus we decided in favor for a version where neighbourhoods between countries are rather maintained in horizontal



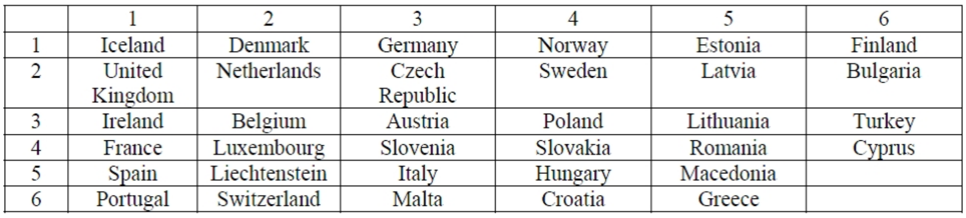
Country name	center_lat	center_long	cat_lat	cat_long
Iceland	64.96	-18.98	1	1
Norway	63.83	17.19	1	2
Sweden	62.16	17.67	1	3
Latvia	56.87	24.60	1	4
Estonia	58.75	25.33	1	5
Finland	64.85	26.06	1	6
Ireland	53.42	-8.25	2	1
United Kingdom	54.29	-3.15	2	2
Netherlands	52.22	5.18	2	3
Denmark	56.16	10.45	2	4
Poland	51.90	19.16	2	5
Lithuania	55.16	23.93	2	6
Belgium	50.59	4.38	3	1
Luxembourg	49.82	6.13	3	2
Germany	51.09	10.48	3	3
Austria	47.70	13.45	3	4
Czech Republic	49.81	15.48	3	5
Slowenia	48.65	19.70	3	6
France	46.76	1.49	4	1
Switzerland	46.82	8.22	4	2
Liechtenstein	47.16	9.55	4	3
Slowenia	46.16	15.00	4	4
Hungary	47.37	19.50	4	5
Romania	45.88	24.48	4	6
Portugal	39.54	-7.90	5	1
Spain	39.80	-3.08	5	2
Italy	41.85	12.65	5	3
Croatia	44.40	16.44	5	4
Macedonia	41.62	21.72	5	5
Bulgaria	42.61	25.49	5	6
Malta	35.95	14.39	6	1
Greece	38.31	23.85	6	2
Cyprus	35.13	33.44	6	3
Turkey	38.96	34.93	6	4

Table 5.5: Geographical latitudes and longitudes of geographical centres, ranked by latitudes then by longitudes



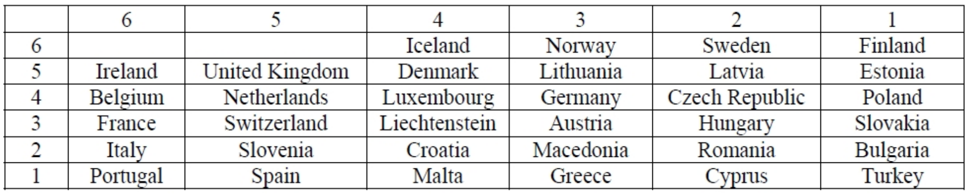
	1	2	3	4	5	6
1	Iceland	Norway	Sweden	Latvia	Estonia	Finland
2	Ireland	United Kingdom	Netherlands	Denmark	Poland	Lithuania
3	Belgium	Luxembourg	Germany	Austria	Czech Republic	Slovakia
4	France	Switzerland	Liechtenstein	Slovenia	Hungary	Romania
5	Portugal	Spain	Italy	Croatia	Macedonia	Bulgaria
6	Malta	Greece	Turkey	Cyprus		

Table 5.6: Grid order option 4 (by coordinates of geographic centres of countries)



	1	2	3	4	5	6
1	Iceland	Denmark	Germany	Norway	Estonia	Finland
2	United Kingdom	Netherlands	Czech Republic	Sweden	Latvia	Bulgaria
3	Ireland	Belgium	Austria	Poland	Lithuania	Turkey
4	France	Luxembourg	Slovenia	Slovakia	Romania	Cyprus
5	Spain	Liechtenstein	Italy	Hungary	Macedonia	
6	Portugal	Switzerland	Malta	Croatia	Greece	

Table 5.7: Grid order option 5 (by coordinates of geographic centres of countries)



	6	5	4	3	2	1
6			Iceland	Norway	Sweden	Finland
5	Ireland	United Kingdom	Denmark	Lithuania	Latvia	Estonia
4	Belgium	Netherlands	Luxembourg	Germany	Czech Republic	Poland
3	France	Switzerland	Liechtenstein	Austria	Hungary	Slovakia
2	Italy	Slovenia	Croatia	Macedonia	Romania	Bulgaria
1	Portugal	Spain	Malta	Greece	Cyprus	Turkey

Table 5.8: Grid order option 6 (by coordinates of geographic centres of countries)

way than in vertical way. That's because we are used to read rows from left to right rather than top down or backwards. The two vacant boxes at the lower right corner can now be used for EU-27 values and eventually for the legend. We did prefer the options with the coordinates of the capitals rather than the geographical centers of the respective countries, because capitals are well known and function besides the political importance as (geographical) reference point for each country. In addition often the capitals are the most important or at least one of the most populated cities in the respective state. Furthermore the centre of gravity could theoretically lie outside of a country (i.e. the centre of gravity could lie in a neighboured country or in the sea) or at least very near of a frontier line.

In spite of the reasoning for grid order option 1, in the end the determination of an arrangement order remains though a subjective selection, which could have been another by considering other argumentation and perspectives. This practical procedure for grid arrangement could also be adhered by a theoretical derivation. A mathematical formulation of the optimal checker board will be introduced in the next chapter.

5.3.1 Grid arrangement with Lattice

To produce checker plots we use the R-package "Lattice" (SARKAR, 2008). Lattice plots the panels in following order: when you want to print a 6x6-grid the first panel of the grid will be plotted in the lower left corner. Lattice prints the panels from left to right and from bottom to top for each grid (see Table 5.9). When you e.g. just have 34 categories the boxes 35 and 36 will remain blank.

5.3.2 European structural indicators - Download data

For our checker plots we look for data on the Eurostat Website [2]. Eurostat provides for download statistical data of the structural indicators. We use the social cohesion and the employment indicators. We downloaded the different indicators with the format Excel. These single files could then be imported to R and merged to one data frame.

The data contains a variable "country codes", which will be the conditioning variable

[2] http://epp.eurostat.ec.europa.eu/portal/page/portal/structural_indicators/indicators (03/11/2009)

31	32	33	34	35	36
25	26	27	28	29	30
19	20	21	22	23	24
13	14	15	16	17	18
7	8	9	10	11	12
1	2	3	4	5	6

Table 5.9: Lattice panel print order

for our checker plots. Conditioning variable means, that several plots will be generated, showing the same variables for each level of the conditioning variable. In other words: lattice will produce a plot for each country showing the value of the selected variable(s).

The country codes follow the Nomenclature of Territorial Units for Statistics (NUTS)[3]. We have also to add a new variable which defines the position of each country in the checker plot. This variable is named ”gridorder” and the values from 1 to 36 are assigned to the country codes according the chosen grid order option (see Table 5.10). E.g. the position of Italy is in the panel 8, according to this, the gridorder value 8 has to be allocated to Italy. Position 5, where is a blank panel, will be occupied by EU-27 data. For the position 6, where is also a blank panel, we have to add a new case in the data set, which value of variable gridorder is 6, and all other variables are missing values. This new case will only function as placeholder.

5.4 Optimal arrangement in the checkerplot

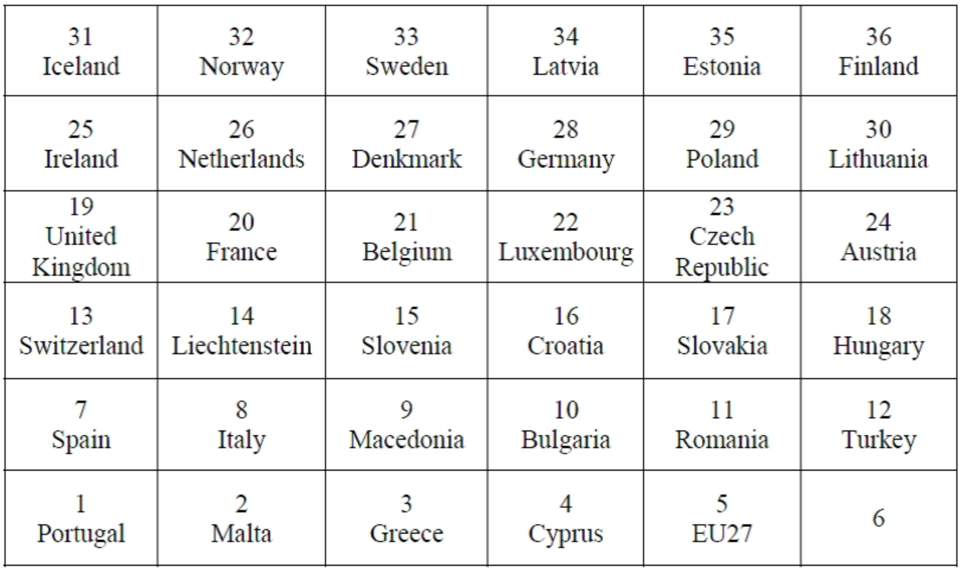
The number of different possible arrangement of 34 countries in a 6 × 6 grid is almost infinite. The optimal arrangement of this optimisation problem is very difficult. It would either a search over all possible allocations of countries to panels which is computational not feasible or needs a type of integer programming algorithm.

5.4.1 Motivating the optimization problem

The problem consists of to assign a set of given coordinate points to another set of coordinates of equal size, whereas the points are assigned in optimal manner.

Figure 5.1(a) shows the grid in the checkerplot where the countries should be plotted.

[3] http://ec.europa.eu/eurostat/ramon/nuts/splash_regions.html (03/11/2009)



31 Iceland	32 Norway	33 Sweden	34 Latvia	35 Estonia	36 Finland
25 Ireland	26 Netherlands	27 Denkmark	28 Germany	29 Poland	30 Lithuania
19 United Kingdom	20 France	21 Belgium	22 Luxembourg	23 Czech Republic	24 Austria
13 Switzerland	14 Liechtenstein	15 Slovenia	16 Croatia	17 Slovakia	18 Hungary
7 Spain	8 Italy	9 Macedonia	10 Bulgaria	11 Romania	12 Turkey
1 Portugal	2 Malta	3 Greece	4 Cyprus	5 EU27	6

Table 5.10: Arrangement (and grid order value) of the countries in the checker plot

Figure 5.1(b) highlights the countries with their acronyms in their original (scaled) position in the map of Europe in lat/long representation.

The distance of the given coordinates in Figure 5.1(a) to the other set of coordinates in Figure 5.1(b) should be minimized.

It exists approx. 36! possiblities to arrage the European countries in a checkerplot. Because of this high number, to try every solution and to evaluate the distance from the grid points to the coordinates of the countries is not possible.

We therefore formulate this problem as a linear programming problem in the next two sections.

5.4.2 The linear program based on 1:1 mapping of points

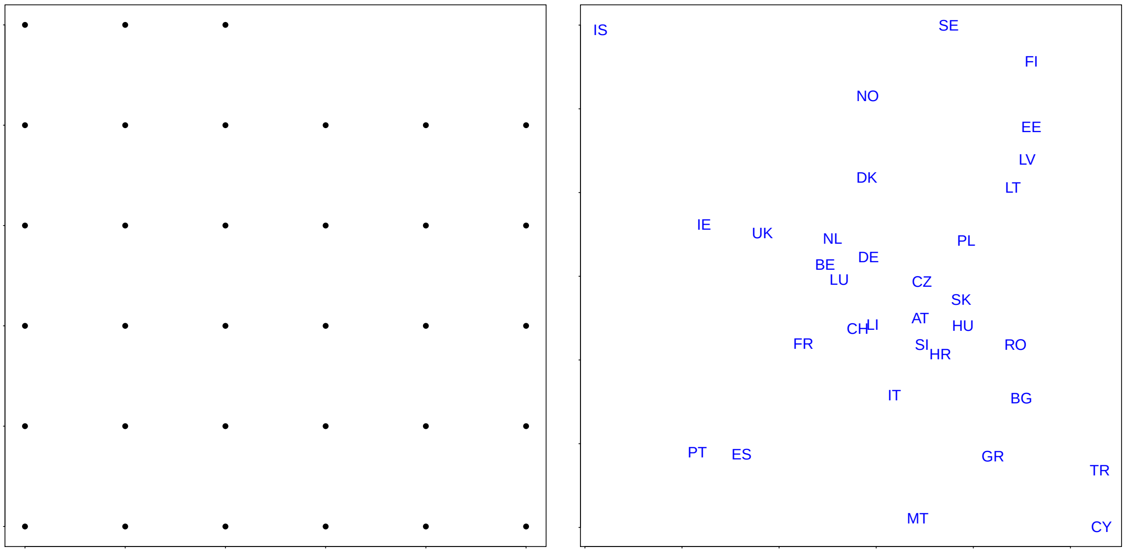
Let both X and Y be a two-dimensional data set with n observations.

First the pairwise distances between X and Y are calculated by

$$d_{ij} = ||x_i - x_j||_2^2 \quad . \tag{5.1}$$

The objective function in the optimization problem is then given by these distances, sticked rowwise together and resulting in a vector of length n^2. Let this vector denoted by

$$z = (d_{11}d_{12}\dots d_{1n}d_{21}d_{22}\dots d_{2n}\dots d_{i1}d_{i2}\dots d_{in}\dots d_{nn}) \quad . \tag{5.2}$$



(a) The position of the countries in a checkerplot on a rectangular grid.

(b) The position of the countries in real (lat/long representation).

Figure 5.1: Arrangement problem from position of countries (b) to the given grid (a)

.

The vector b is a binary vector of length n^2. The subset b_k, with

$$k \in \{(i-1)n+1, (i-1)n+2, (i-1)n+3, \ldots, (i-1)n+n\} \quad ,$$

belongs to the i-th observation in X and defines the closest observation in Y. Only one component of this subset is unqeual 0, the index of the non-zero entry $(i-1)n+j$ defines the closest observation j in Y.

The objective function that has to be minimized is then given by

$$\min \quad zb^T \quad , \tag{5.3}$$

under the contraints

$$Ab^T \; = \mathbf{1} \quad , \tag{5.4}$$

$$\tag{5.5}$$

with

$$a_{km} \in \{0,1\} \ , \quad k = 1, \ldots, 2n \ , \quad m = 1, \ldots, n^2 \quad . \tag{5.6}$$

For $k \in \{1, 2, \dots, n\}$, the elements are defined by

$$a_{km} = \begin{cases} 1 & \text{if} \quad k = \lceil \frac{m}{n} \rceil \\ 0 & \text{else} \quad . \end{cases} \tag{5.7}$$

E.g.

$$a_{1m} = (\underbrace{1, \dots, 1}_{n}, \underbrace{0, \dots, 0}_{n^2-n}) \tag{5.8}$$

$$a_{2m} = (\underbrace{0, \dots, 0}_{n}, \underbrace{1, \dots, 1}_{n}, \underbrace{0, \dots, 0}_{n^2-2n}) \tag{5.9}$$

$$a_{3m} = (\underbrace{0, \dots, 0}_{2n}, \underbrace{1, \dots, 1}_{n}, \underbrace{0, \dots, 0}_{n^2-3n}) \tag{5.10}$$

$$\dots = \dots \tag{5.11}$$

$$a_{nm} = (\underbrace{0, \dots, 0}_{n^2-n}, \underbrace{1, \dots, 1}_{n}) \tag{5.12}$$

$$\tag{5.13}$$

For $k \in \{n+1, n+2, \dots, 2n\}$, the elements are defined by

$$a_{km} = \begin{cases} 1 & \text{if} \quad k - n = m - n(\lceil \frac{m}{n} \rceil - 1) \\ 0 & \text{else} \quad . \end{cases} \tag{5.14}$$

E.g.

$$a_{(n+1)m} = (1, \underbrace{0, \dots, 0}_{n-1}, 1, \underbrace{0 \dots, 0}_{n-1}, \dots, 1, \underbrace{0 \dots, 0}_{n-1}) \tag{5.15}$$

$$\tag{5.16}$$

The linear programming problem is solved with the function lp in the R-package lpSolve (Berkelaar et al., 2010).

5.4.3 The linear program based on different sizes

In contradiction to the previous section, n_2 points are assigned to n_1 points in an optimal manner, with $n_1 > n_2$. We assume therefore that some regions in a rectangular grid left free. Comparing that to Figure 5.1(a) this means that also additionally three points in the upper right part of this grid can be assigned.

Let both X and Y be a two-dimensional data set with n_1 and n_2 observations.

First the pairwise distances between X and Y are calculated by

$$d_{ij} = ||x_i - x_j||_2^2 \quad . \tag{5.17}$$

The objective function in the optimization problem is then given by these distances, sticked row-wise together and resulting in a vector of length $n_1 * n_2$. Let this vector denoted by

$$z = (d_{11}d_{12} \dots d_{1n_2}d_{21}d_{22} \dots d_{2n_2} \dots d_{i1}d_{i2} \dots d_{in_2} \dots d_{n_1n_2}) \quad . \tag{5.18}$$

The vector b is a binary vector of length $n_1 * n_2$. The subset b_k, with

$$k \in \{(i-1)n_1 + 1, (i-1)n_1 + 2, (i-1)n_1 + 3, \dots, (i-1)n_1 + n_2\} \quad ,$$

belongs to the i-th observation in X and defines the closest observation in Y. Only one component of this subset is unequal 0, the index of the non-zero entry $(i-1)n_1 + j$ defines the closest observation j in Y.

The objective function that has to be minimized is then given by

$$\min \quad zb^T \quad , \tag{5.19}$$

under the contraints

$$Ab^T = \mathbf{1} \quad , \tag{5.20}$$

$$Cb^T \leq \mathbf{1} \quad , \tag{5.21}$$

$$\tag{5.22}$$

with

$$a_{km} \in \{0,1\} \ , \quad k = 1, \dots, n_2 \ , \quad m = 1, \dots, n_1n_2 \tag{5.23}$$

$$c_{km} \in \{0,1\} \ , \quad k = 1, \dots, n_1 \ , \quad m = 1, \dots, n_1n_2 \quad . \tag{5.24}$$

The elements of A are defined by

$$a_{km} = \begin{cases} 1 & \text{if} \quad k = \lceil \frac{m}{n_1} \rceil \\ 0 & \text{else} \end{cases} \tag{5.25}$$

, e.g.

$$a_{1m} = (\underbrace{1, \dots, 1}_{n_1}, \underbrace{0, \dots, 0}_{n_1 n_2 - n_1}) \tag{5.26}$$

$$a_{n_2 m} = (\underbrace{0, \dots, 0}_{n_1 n_2 - n_1}, \underbrace{1, \dots, 1}_{n_1}) \tag{5.27}$$

$$\tag{5.28}$$

and the elements of C are defined by

$$c_{km} = \begin{cases} 1 & \text{if} \quad k = m - n_1(\lceil \frac{m}{n_1} \rceil - 1) \\ 0 & \text{else} \end{cases} \tag{5.29}$$

, e.g.

$$c_{1m} = (1, \underbrace{0, \dots, 0}_{n_1 - 1}, 1, \underbrace{0 \dots, 0}_{n_1 - 1}, \dots, 1, \underbrace{0 \dots, 0}_{n_1 - 1}) \quad . \tag{5.30}$$

$$\tag{5.31}$$

The linear programming problem is solved with the function `Rglpk_solve_LP` in the R-package Rglpk (Theussl and Hornik, 2010).

5.4.4 Application to the checkerplot using EU-SILC and European NUTS 1 data

First, the means values of the bounding box around each country has to be calculated. Let Y the matrix which contains the middle points of the bounding box of each country. This is in fact, the coordinates of the center of each country. Then, the lat/long coordinates has to be transformed to the same scale as the grid (or vice versa). This can easily be done by

$$y_{ij} \leftarrow \frac{y_{ij} - \min(y_{.j})}{\max(y_{ij} - \min(y_{.j}))} \cdot 6 \quad , \tag{5.32}$$

where

$$y_{.j} = \begin{pmatrix} y_{ij} \\ y_{2j} \\ \vdots \\ y_{nj} \end{pmatrix} \quad .$$

The grid is defined as a data set consisting all combinations two supplied vectors, both $(1, 2, 3, 4, 5, 6)$, i.e all possible 6^2 combinations of the numbers 1 to 6.

Figure 5.2 shows the result of the optimization from equal number of points which have to be assigned (see Section 5.4.2). The countries are assigned to the grid in an optimal manner.

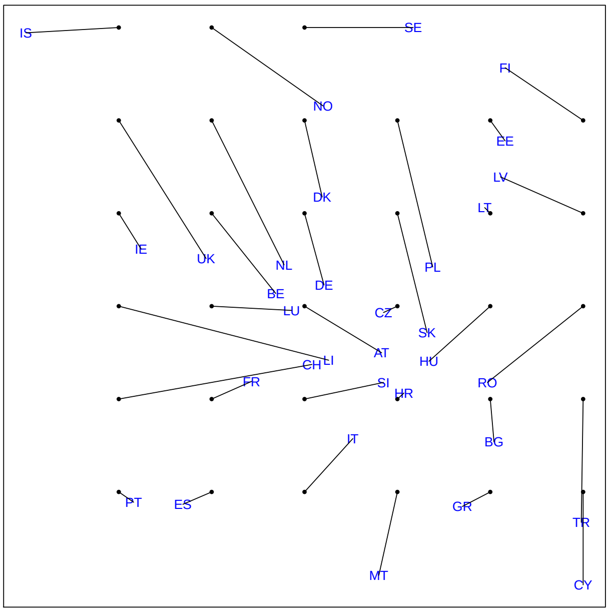


Figure 5.2: The optimal allocation of the countries to the grid.

The checkerplot looks then as presented in Figure 5.3.

The solution where all points in the grid are possible canditates for assignment, the checkerplot becomes as presented in Figure 5.4, right graphic. For comparison reasons, also the 1:1 assignment solution is visualized (left graphic of Figure 5.4).

Remark: A solution for the assignment problems discussed in Section 5.4.2 and 5.4.3 are presented. We showed that this solution is optimal and no heuristic must be chosen. The linear program is implemented in a generalized manner so that it can be applied for any arrangement problem, such as the allocation of US-states or the allocation of subregions in countries.

Each country is placed optimally in the grid so that everybody is able to easily find the countries in the checkerplot. The countries are placed in a rectangular grid as close to their position in their lat/long representation in real.

Note that this is a serious improvement to the previous solution made by hand without solving the optimization problem.

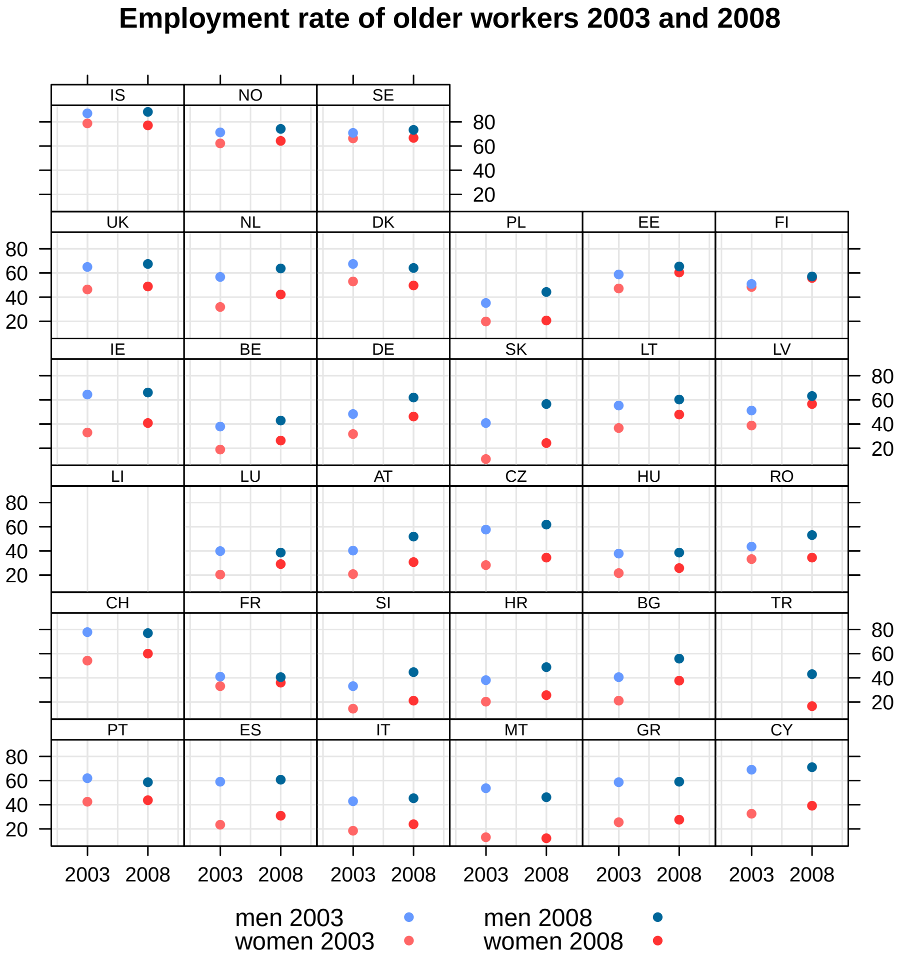


Figure 5.3: The optimal allocation of the countries to the grid.

5.5 Checker plots

Subsequent some checker plot examples will be introduced and discussed.

5.5.1 Bar plots

Figure 5.5 reveals the employment rate of older workers in 2008. The bars show the female, the male as well as the total rate for each country. As this chart shows, lattice presents by default the bars horizontally. This notation (bar plot) needs much colour for relative poor information. Besides this notation let the diagram appear ponderous and overloaded. In addition the horizontal bars are not well comparable between the panels. It is difficult to compare the length of the bars. Above each panel, there is a string which contains the country label. The labels correspond to the official two-letter country codes.

The next example (Figure 5.6) shows vertical bars, but the bars take as well much

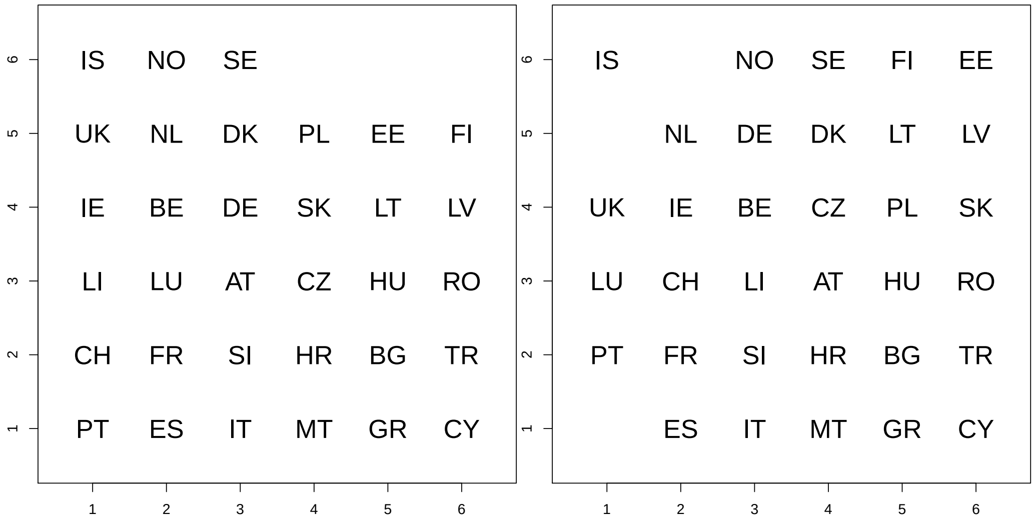


Figure 5.4: The optimal allocation of the countries to the grid with 1:1 assignment (left graphic) and different sizes of points for assignment (right graphic).

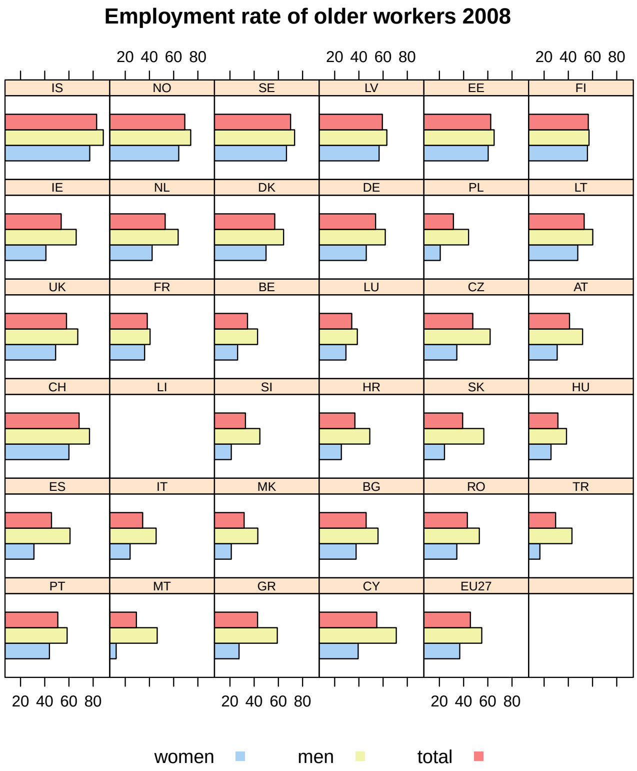


Figure 5.5: Barplot 1: with horizontal bars

space for relative small information. The comparability between the panels is now better. In comparison with horizontal bars, vertical bars allow better to find out countries with high or low employment rate of older workers. Also it is easier to catch varieties between female and male employment rate inside of a panel.

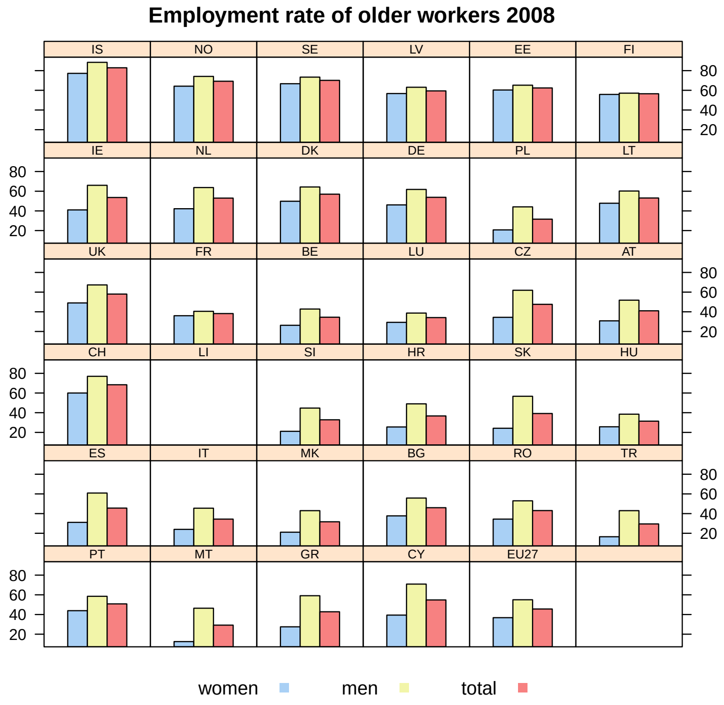


Figure 5.6: Barplot 2: with vertical bars

In the following bar chart (Figure 5.7) the at-risk-of-poverty rate before and after social transfers for the year 2007 have been plotted for women, men and also the total rate. Evidently there is too much information. The chart is overcharged and the viewer needs too much time to read all these information.

Figure 5.8 displays a bar chart which bars stand for the at-risk-of-poverty rate before and after social transfers segmented by gender. Comparing to the previous bar chart (Figure 5.7), it is at least possible to compare the values within the respective panel. Comparisons between the different panels are possible but still suboptimal. First of all the comparability between the different rows is difficult. Besides the problem of comparability, an other problem arises by using bar charts: the usage of adequate colours. A problem which has not been solved in this chart...

5.5.2 Scatter plots

Figure 5.9 shows a scatter plot where the at-risk-of-poverty rate for the year 2007 lies on the x-axis and the income quintile share ratio (QSR) for 2007 on the y-axis. The dots denote the male respectively female at-risk-of-poverty rate in relation to the QSR of the respective country (QSR has not be designated by gender, we only have total values).

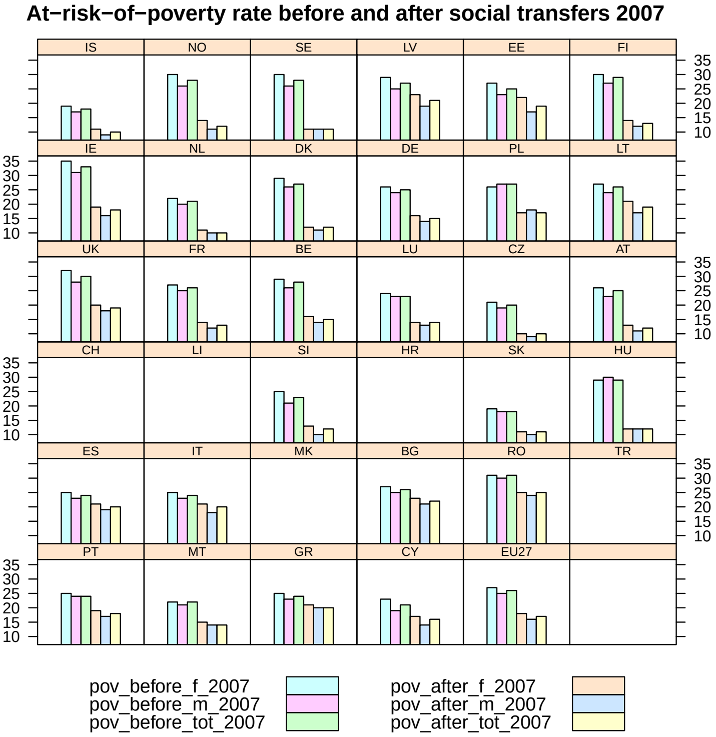


Figure 5.7: Barplot 3: showing the at-risk-of-poverty rate before and after social transfers

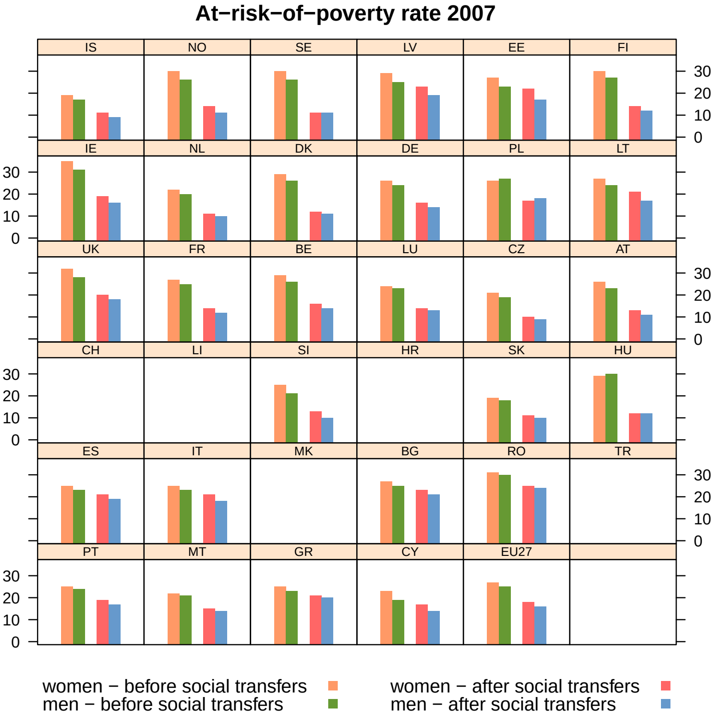


Figure 5.8: Barplot 4: showing the at-risk-of-poverty rate of men and women

Grid lines are helpful for reading of data values, respectively for a better comparability between the different panels.

At first view, Figure 5.9 would probably not be comprehensible for laymen. So this table should be presented with a short reading instruction.

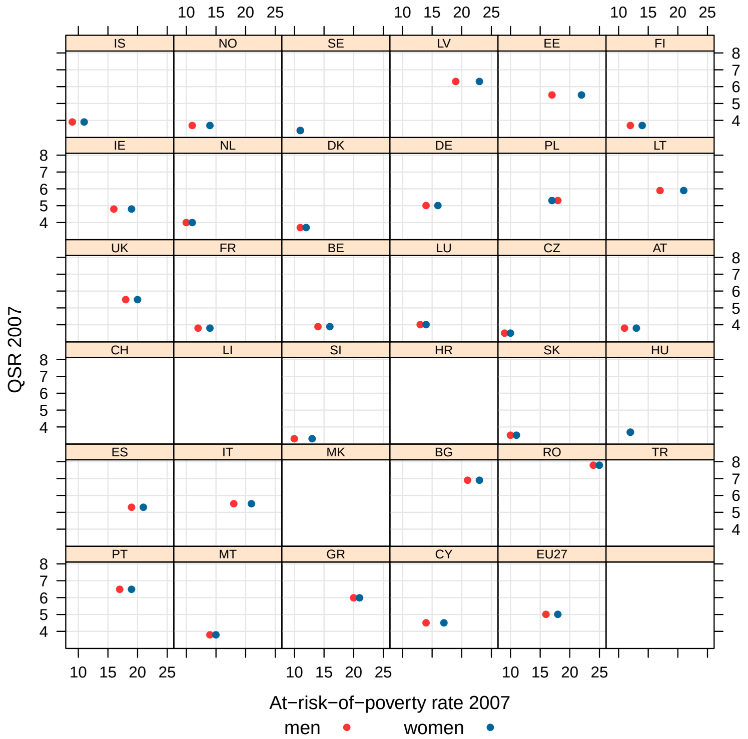


Figure 5.9: Scatter plot 1: showing at-risk-of-poverty rate and QSR

Figure 5.10 reveals the same data as the previous scatter plot (Figure 5.9) but the axis have been transposed, so that the QSR is now on the x-axis and the at-risk-of-poverty rate on the y-axis.

We think this version of scatter plot is the better kind of presentation. In our view comparisons between female and male values could be done easier, as well as comparisons between the different panels, than in the first scatter plot version.

5.5.3 Dot plots

Figure 5.11 shows the same data as the already introduced bar charts (Figure 5.5 and 5.6). Instead of bars, which are taking up much space, this dot plot is a space and ink saving form of showing same data. Differences between male and female employment rate in each panel, as well as differences between the panels are rather good visible (at least between panels in the same row). The gridlines are quite helpful to detect differences. Though, the comparability between different rows is constrained because the strips with the country labels form a break. This is unfortunately a general problem of trellis plots.

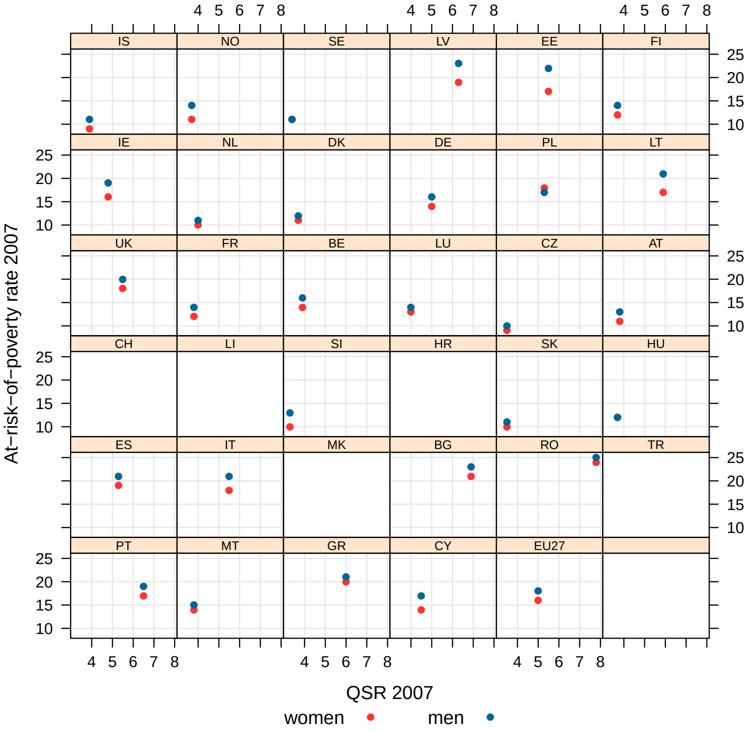


Figure 5.10: Scatter plot 2: showing QSR and at-risk-of-poverty rate

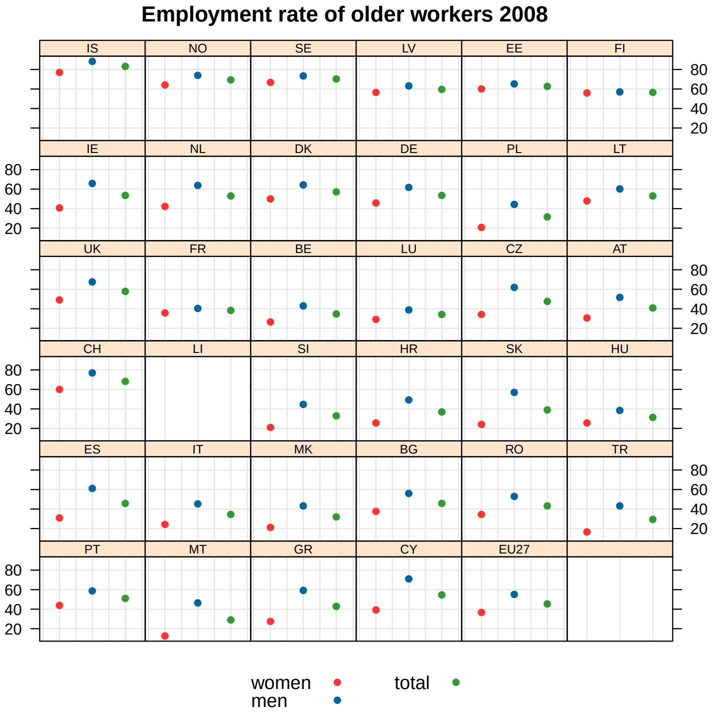


Figure 5.11: Dotplot 1: showing employment rate of older workers

The next dotplot (Figure 5.12) presents the same data as the precedent diagram (Figure 5.11). The male and female values are now presented one upon the other and not side by side as in the previous dot plot. This notation makes it clearer to understand that the female and male date belong together, respectively that they are segments of the total value. One problem by displaying data points for male and female in this way is, that they could overlap when they have similar values or they could even be congruent.

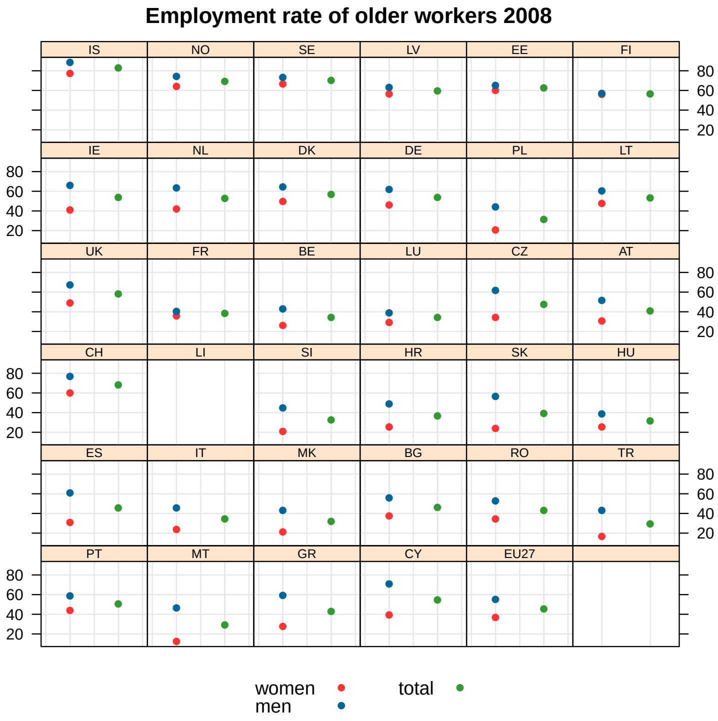


Figure 5.12: Dotplot 2: showing employment rate of older workers 2008

A further dot plot (Figure 5.13) displays the employment rate of older workers segmented by gender for the years 2003 and 2008. It is an attempt to show the evolution between two years. The strip background in this example is not coloured. The intention was to weaken the strong break between the rows. But the main problem persists: it is difficult to compare panels in the vertical sense.

Figure 5.14 shows an example in which the strips are vertical oriented. In this way we can facilitate the vertical comparability. But finally there is not a big gain, because now the comparability in the horizontal sense is constrained. Furthermore the legibility of the vertical aligned labels is not good.

The next two figures (5.15 and 5.16) do not show any strips. The labels are directely inserted in the panels to weaken the break effect of the strings. In figure 5.15 the labels are placed in the upper right corner of each panel, in Figure 5.16 the labels were printed in the upper left corner. Because the strips have been left out, these presentation looks

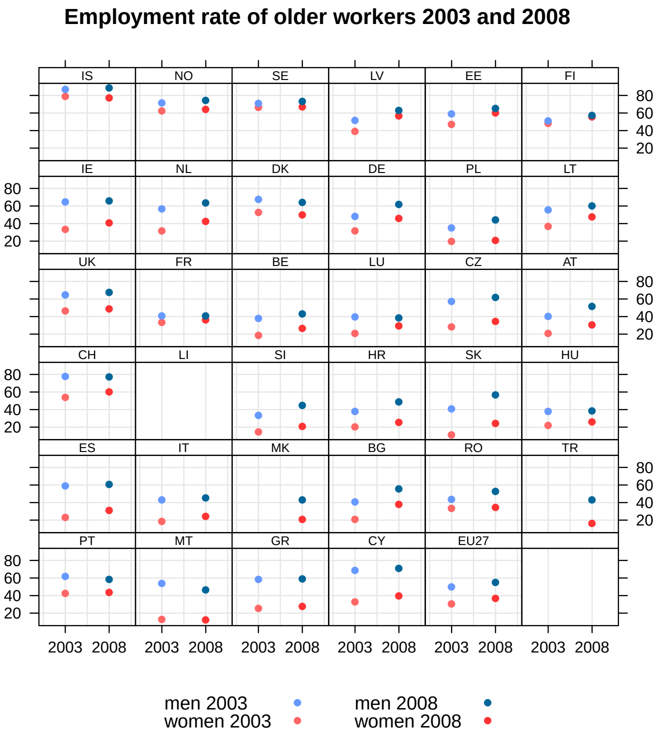


Figure 5.13: Dotplot 3: showing employment rate of older workers 2003 and 2008

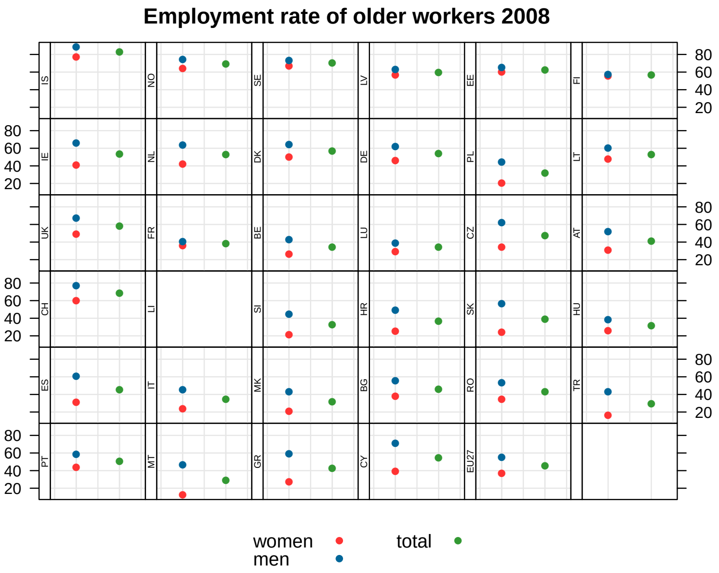


Figure 5.14: Dotplot 4: with vertically aligned labels

the most like a geographic map.

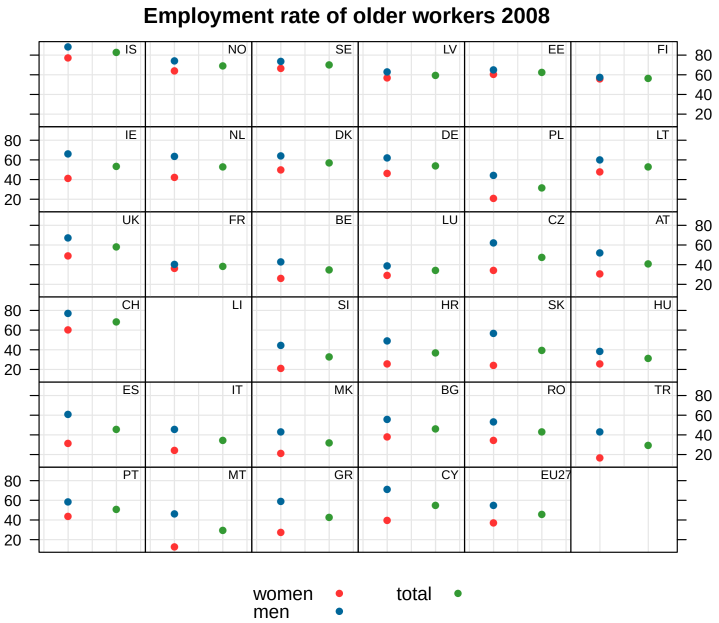


Figure 5.15: Dotplot 5: with labels in the upper right corner of the panels

5.5.4 Sparklines

Finally, Figure 5.17 shows a sparkline representation of indicators over time. The trend is most clearly visible using this visualisation.

Bibliography

Berkelaar, M. et al. (**2010**): lpSolve: Interface to Lp_solve v. 5.5 to solve linear/integer programs. R package version 5.6.5.
URL http://CRAN.R-project.org/package=lpSolve

Sarkar, D. (**2008**): Lattice: Multivariate Data Visualization with R. New York: Springer, iSBN 978-0-387-75968-5.
URL http://lmdvr.r-forge.r-project.org

Theussl, S. and **Hornik, K.** (**2010**): Rglpk: R/GNU Linear Programming Kit Interface. R package version 0.3.5.
URL http://CRAN.R-project.org/package=Rglpk

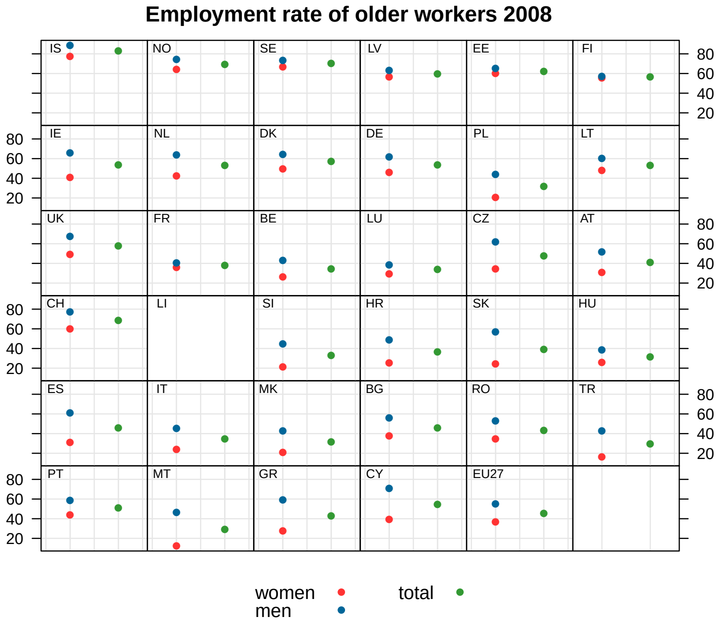


Figure 5.16: Dotplot 6: with labels in the upper left corner of the panels

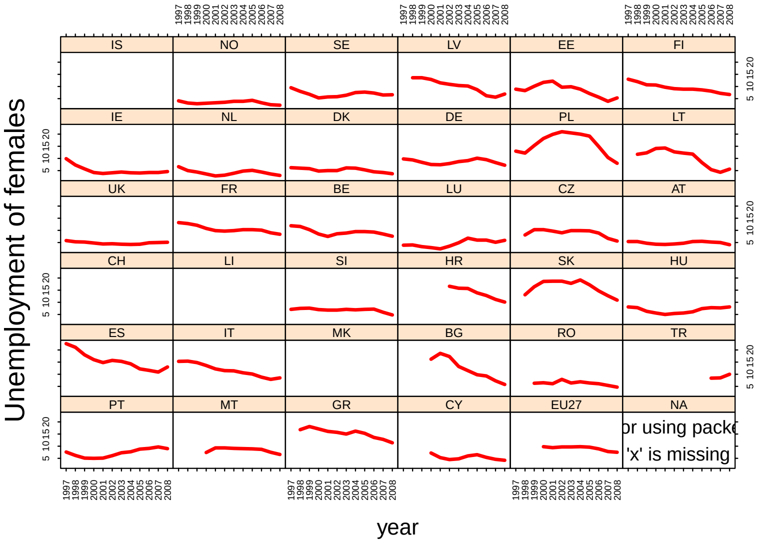
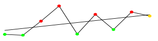


Figure 5.17: Polygon lines representation of an indicator over time.

Chapter 6

Application of Sparktables

6.1 Sparklines

The term sparkline was developed by Edward R. Tufte (Tufte, 2006). A sparkline is a graphic of small size (which fits in a normal text) but containing a lot of information. The idea behind sparklines is the placement of the graphic in the text, for example, 'The Gini coefficient has increased the last years '.

Sparklines are especially useful in tables, if as much information as possible should be visible. The majority of sparklines consists either of polygons like in Table 6.1 or barplots. Last but not least, Alfons et al. (2009) proposed such tables to fit more information about indicators within a table.

Concerning sparklines, Tufte (2006) mentions another important issue, the *aspect ratio*. The *aspect ratio* is the ratio between width and length of a sparkline. Sparklines have a short dimension (the width is mostly specified by the font size of the text beside the sparklines) and a long dimension. In general, sparklines should have a median slope of 45°. This technique is called *banking to 45 degree* and got implemented by Cleveland (1993). For a given time series $x_t, t = 1, \ldots, T$, the median slope of the sparklines

$$median(|\alpha_t|) = median(|arctan(x_t - x_{t-1})|) = 45^\circ \tag{6.1}$$

where α_t denotes the slope of the sparkline between two points x_t and x_{t-1} for $t = 1, \ldots, T$. The shape varies a lot when the aspect ratio gets changed. In practice the height of the sparkline is fixed, and the horizontal length of the sparkline is adjusted by considering the 45° rule. However, in tables this rule is not that easy to implement because the 45° rule will lead to sparklines of different length. Therefor, a fixed aspect ratio of 5:1 is often used especially for sparklines in financial applications.

Table 6.1 shows a practical example for the usage of sparklines in tables. For the purpose to have a user friendly way to produce tables like Table 6.1 the function *indTable()* got developed. It is an **R** function which generates a LaTeX table.

One can easily generate sparklines out of a given data set. Table 6.1 shows an example with artificial data. The sparkline can either be a normal plot or one of the *indEval()*-

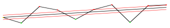
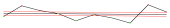
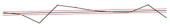
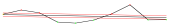
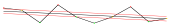
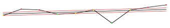
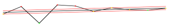
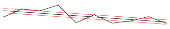
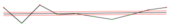
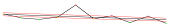
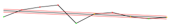
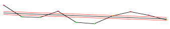
	sparkline	trend	min	max	current	median	mean
AT		0.27	25.56	34.28	34.22	32.00	30.90
BE		0.03	22.51	31.45	27.97	27.19	27.06
CY		-0.03	24.50	33.70	27.22	29.94	29.71
CZ		-0.12	25.67	35.22	27.42	27.42	29.24
DE		-0.11	26.88	32.78	28.37	28.89	29.44
DK		0.20	23.22	32.84	32.84	30.19	29.39
EE		0.01	23.71	32.97	31.27	31.27	30.15
ES		-0.16	27.17	33.34	27.17	29.36	29.49
FI		0.10	21.43	32.15	30.72	27.04	27.46
FR		-0.15	27.18	37.68	27.18	30.27	30.74
GR		-0.29	21.71	36.96	26.77	27.17	29.04
HU		-0.16	24.58	37.32	27.20	29.36	29.84

Table 6.1: Table of sparklines with the usage of the 'indTable' function.

function plots like in Figure 3.7(b) or 3.9. The user can select one or more of these columns: min, max, current, median, mean, trend, sparkline as well as the order of the columns.

An alternative way is to set the option `output="plot"` which generates a single PDF file. However, this function allows only one plot type which gets applied to the whole table. That this functions are not modifiable is a disadvantage, but the advantage is clearly the easy usability. A more complex approach is given in the next section, the *sparkTable* package for **R**.

6.2 The sparkTable package

sparkTable (Kowarik et al., 2010) is a package for **R**. It is an extension of the *sparkTable()*-function and includes various features. While the first two authors of the package did most of the programming part, my job was testing the **R** functions, finding possible errors and to propose modifications. The presentation of this package took place in Vienna at the *Workshop on Exploratory Data Analysis and Visualisation* in May 2010. Within the *indTable()* function the parameters are fixed for every column. The *sparkTable* package completely changes this approach by having various parameter settings for every single cell. Colors, width and length of a specific cell of a n-dimensional table can be changed in an easy manner, leaving all other cells unchanged. This gets especially handy when facing complex tables.

6.2.1 Aims of the package

The following points are considered:

- Providing quick access to additional insights by the use of *graphical tables.*
- Presentation of numerous data in a well-arranged way.
- Improving data density by using spark-graphs.
- Results should be easy to modify.
- Development of a tool to create graphical tables easily.
- Sparktables for multidimensional data.

Based on these ideas and aims the work started on the package, and some of the functions will be described below:

- spark_init(): This functions allows to set a list of properties depending on the type of the desired spark-graph for any given numerical input data vector.
- sparkTable_Config(): Using this function, meta-objects are created in order to generate multiple spark-graphs for multidimensional input data. This function produces the base frame of the table.
- setPara(): Gives the possibility to set or change (graphical) parameters for data objects generated with sparkTable_Config().
- print.sparkTable(): Output of graphical tables in *EPS* or *HTML*-format of given objects generated with sparkTable_Config().

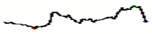
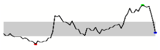
As mentioned above the output format can be chosen between *EPS* or *HTML*, and afterwards included on web pages or documentations. In the current version there are three types of plots to choose from, the time series plot, the bar plot and the boxplot.

Indexing of the meta object

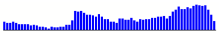
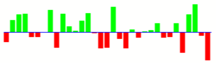
Tables often consist of higher dimensions. For example, k indicators for different countries got measured for T years, which denotes a *three way* table. This example leads to a table with $n \cdot k \cdot T$ cells, which can either be displayed as a k x T table for every group, tables of dimension k x n for every year, etc. The data preparation should construct a table like Table 6.2, which shows the first few lines of the `gini` data set available in the *sparkTable* package. The *sparkTable* package can currently handle *3-dimensional* tables, whereas such a table is structured within the package in a special manner. Further development is aiming for the usage of higher dimensions.

The *sparkTable* package formats three-way tables into a list of lists after calling the *sparkTable_config()* function. First of all it splits the meta-object into two parts, a list containing the information (`metaInfo`) and a list containing the data (`metaData`). The first one consists of a list (i.e. it is a list of a list) with information about which variables are used, what should be calculated with them, what groups have been chosen and so on. The `metaData` list contains information about each cell of the data which gets displayed in the table. The first index represents a list containing the rows of the table, the second index is a list within the previous list representing the columns (for example *metaData$[[1]][[3]]* is the third cell in the first row). The user is able to specify the different columns of the table by an object called `typeNumeric`, which can either be a list or a vector. This list or vector can contain every function that returns a single value, like the mean or the maximum. The order of this object defines the column number (element of the second list in `metaData`). All information of the sparkline is added to this list. Note that the names of the rows and columns are excluded from the indexing. If the chosen list element is a sparkline, the element consists of another list of parameters with data and all necessary graphical parameters for the sparkline. These parameters can be changed by the user. An application example is given in Section 6.2.3.

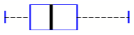
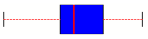
Time series plots

The time series plots  provided bythe *sparkTable* package are common time series plots with certain additional options. It is possible to highlight specific values (like the minimum, maximum, mean) in different colors, sizes or symbols. In addition, the 50% of the inner data points, determined by the interquartile range (IQR), can be additionally visualized in the plots, see .

Bar plots

These are classical bar plots  with the possibility to change the bar widths, heights, borders or colors . This kind of bar plot is mostly used to visualize differences or changes over time.

Boxplots

The visualization of outliers in the boxplots  can be deactivated if the data are too distorted, and different colors of the boxplots can be chosen.  .

6.2.2 An application of sparkTable with EU-SILC data

Gini data set

Table 6.2 presents the first few entries of the Gini data set. For most countries data from 2004-2007 is given, for some countries the data for 2004 are missing because they were not present in the EU-SILC data set for that year.

	year	country	gini
1	2004	AT	25.77
2	2005	AT	26.13
3	2006	AT	25.33
4	2007	AT	26.15
5	2004	BE	27.01
6	2005	BE	28.53

Table 6.2: First 6 observations of the Gini data set, which is available in the sparkTable package.

The Gini data set is available in the *sparkTable* package, and can be loaded with `data(gini)`. The values have been calculated within the *AMELI* project.

6.2.3 A guided tour

Table 6.2 shows a part of the data set which is used for demonstrating the package in the following. First of all, a meta object with the *sparkTable_config()* function is produced by the following code:

```
meta <- sparkTable_Config(gini,
groups=c("AT","DE","IT"), groupVar="country", vars=c("gini"),
     typeNumeric=c("1--4","mean", "max"),
     typePlot="line", output="eps")
```

Austria, Germany and Italy have been chosen, which act as our groups, and the group variable is called `"country"`. Furthermore we want to take a look at the variable `"gini"` which is the only variable in this data set. From the value of this variable `gini` the mean and the maximum is of interest over the years, as well as the values of every year. These values are displayed by the `"1-4"` code. The output should be a time series plot `typePlot="line"` and it should be exported in `"eps"` format, to be easily included it into this diploma thesis. The row names look better in the output if the full names of the countries are used.

```
rowVec <- c("Austria", "Germany", "Italy")
colVec <- c("Mean", "Maximum","Line-Plot")
```

The column names are modified similarly (`colVec <- c("...")`). Now only a final step is needed which produces Table 6.3:

```
eps.text <- print.sparkTable(meta, outdir="examples",
                             rowVec=rowVec, colVec=colVec, outfile=NULL)
```

With the last command the code generating Table 6.3 gets printed in **R**, and the result just needs to transfered into LaTeX. In addition, if also an `outfile` is defined in the previous code line, a standard LaTeX-file with the table is produced.

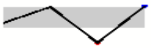
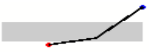
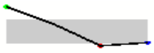
	Sparkline	2004	2005	2006	2007	Mean	Maximum
Austria		25.77	26.13	25.33	26.15	25.84	26.15
Germany		*NA*	26.26	27.03	30.63	27.97	30.63
Italy		33.24	32.75	32.13	32.22	32.59	33.24

Table 6.3: Table of Gini coefficients for Austria, Germany and Italy, produced by the sparkTable package.

Figure 6.3 indicates that no data is available for Germany in the year 2004. Therefore, the sparkline for Germany is starting one year later as Austria or Italy.

Assume that the representation of Table 6.3 does not look like properly, because of the small size of the points. By changing the parameter `pointWidth` in the function `setPara()`, the size of the points can be increased.

```
meta$metaData <- setPara(meta$metaData, "pointWidth",3 )
```

If this is not satisfying, for example, if the point size specifying the Gini coefficient for Germany is too small, the settings for one specific sparkline can be modified. In this example the IQR box will be removed and the color of the last observation is changed to red. The **R**-code for this changes looks like this:

```
meta$metaData[[2]][[7]]$showIQR <- FALSE
meta$metaData[[2]][[7]]$colVals <- c("#f00","#0f0","#f00")
```

`meta$metaData[[2]][[7]]` are the 'coordinates' for the German sparkline. `[[2]]` represents the second row of the table, and `[[7]]` is the column of the sparkline. Colors in the *sparkTable* package are encoded as hexadecimal. To get a picture of all possible sparkline settings take a look at the help files of the package (`help("setPara")`). Now just call

```
eps.text <- print.sparkTable(meta, outdir="examples",
rowVec=rowVec, colVec=colVec)
```

again. The result is shown in Table 6.4.

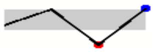
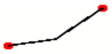
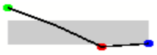
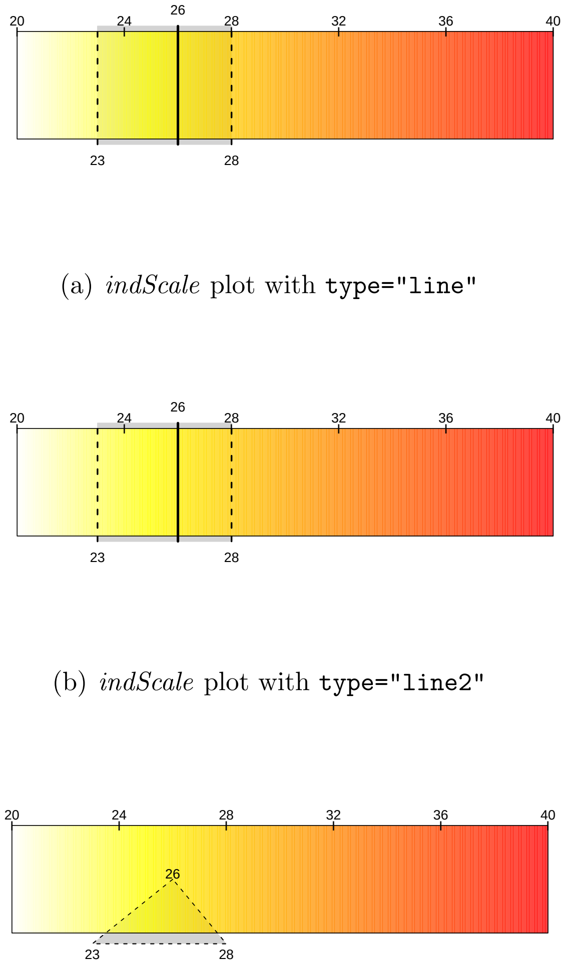
	Sparkline	2004	2005	2006	2007	Mean	Maximum
Austria		25.77	26.13	25.33	26.15	25.84	26.15
Germany		*NA*	26.26	27.03	30.63	27.97	30.63
Italy		33.24	32.75	32.13	32.22	32.59	33.24

Table 6.4: Modified Table of the Gini coefficients for Austria, Germany and Italy.

This is just a very basic example, various more complex tables can be made with this package, which is still under development.

6.3 Confidence Intervals

Various graphical methods have been described so far to visualize indicators. Now the disadvantages of some of these methods are described. The main drawback is that only the point estimates are visualized in the previous figures (like in Figure 3.8(a)), but their uncertainty is not reported. From a statistical point of view this is not enough, a way to visualize confidence intervals has to be found.



(a) *indScale* plot with `type="line"`

(b) *indScale* plot with `type="line2"`

(c) *indScale* plot with `type="tri"`

Figure 6.1: Different plot methods for the *indScale* function to visualize confidence intervals for indicators.

Figure 6.3 shows different ways of visualization, the point estimate of an indicator value of 26, and the following confidence interval of $[23, 28]$. This is just an example for illustration and our estimations of the Laeken indicators have significantly smaller confidence intervals. In order to be still able to see the intervals the scale got changed.

Figure 6.1(a) shows a shaded confidence interval on the scale. This kind of visualization is handy for plots which get scaled-down. This option is used for the sparklines in Table 6.5. Figure 6.1(b) illustrates the confidence intervals above and below the scale, but not within the scale like in Figure 6.1(a). This way the color range of the indicator scale does not get distorted. The last Figure 6.1(c) shows another possibility of visualization, by using a triangle for the confidence intervals.

6.3.1 Gini coefficients for Austria

In Table 6.5 the Gini coefficients for Austria as well as for the three NUTS1 levels of Austria are displayed. In the region AT1 (Eastern Austria, consisting of Vienna, Lower Austria and Burgenland) the Gini coefficient has the highest value. The Gini coefficient measures the inequality of the income distribution (the higher the inequality, the higher the Gini coefficient), and the high value for Eastern Austria is explained by the difference of income between Vienna, Lower Austria and Burgenland. The gross regional product per inhabitant of Vienna was 43.300 in the year 2007, while it was just 21.600 for Burgenland, so around the half. Compared to the others, region AT2 (South Austria) has a low Gini coefficient. South Austria consists of Styria and Carinthia, both states have nearly the same gross regional product per inhabitant, 27.800 for Carinthia and 28.200 for Styria[1].

Someone could now interpret these numbers as 'South Austria is the richest region in Austria because it has the lowest Gini coefficient'. This would be a misinterpretation of the Gini coefficient, because for a region in which everyone has a low income the Gini coefficient would be very low.

The confidence intervals in Table 6.5 have been calculated in **R** with the function *bootVar* from the package *laeken* (Alfons et al., 2010). It computes the confidence interval estimation based on a bootstrap resampling method, to be more precise it is a bootstrap percentile interval. This works as follows (Efron and Tibshirani, 1986):

Given the data X, n samples of the data are taken with replacement. For every sample X_{boot_i} the unknown parameter $\hat{\theta}$ of interest is estimated, in this case the Gini coefficient. That way n indicator values $\hat{G}_1, \ldots \hat{G}_n$ are calculated. The bootstrap percentile confidence interval is determined by percentiles of the bootstrap distribution. That means leaving off $\alpha/2 \cdot 100\%$ of each tail of the distribution. α is the given significance level which is 5% in this case.

The percentile method was used because it cannot be assumed that the income is normal distributed. This way the confidence interval is asymmetric around the point estimator of the Gini coefficient. The smaller confidence intervals for AT compared to the NUTS1 regions AT1-AT3 are explained by the larger sample which is available for Austria.

[1] source: http://www.statistik.at/web_de/services/wirtschaftsatlas_oesterreich/oesterreich_und_seine_bundeslaender/021513.html

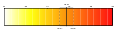
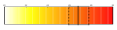
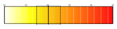
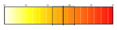
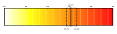
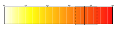
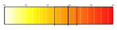
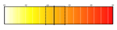
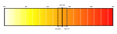
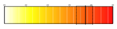
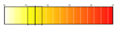
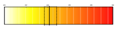
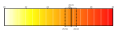
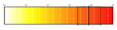
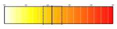
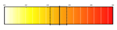
	region	year	ind	lower	upper	sparkline
1	AT	2004	25.77	25.14	26.35	
2	AT1	2004	26.79	25.94	27.79	
3	AT2	2004	24.05	22.97	25.16	
4	AT3	2004	25.42	24.43	26.45	
5	AT	2005	26.13	25.74	26.68	
6	AT1	2005	27.38	26.54	28.56	
7	AT2	2005	25.87	24.61	26.69	
8	AT3	2005	24.58	23.8	25.58	
9	AT	2006	25.33	24.93	25.77	
10	AT1	2006	27.47	26.63	28.14	
11	AT2	2006	22.83	22.15	23.47	
12	AT3	2006	24.14	23.68	24.82	
13	AT	2007	26.15	25.59	26.62	
14	AT1	2007	27.78	26.75	28.89	
15	AT2	2007	24.38	23.57	25.30	
16	AT3	2007	25.08	24.21	25.75	

Table 6.5: Table of the Austrian Gini coefficient and the Austrian NUTS1 level, including the confidence intervals for the Gini.

One further application of sparktables is shown in Table 6.6. Statistical institutions often presents their tables as displayed in Table 6.6 either in their print-outs or by using web dissemination systems.

However, the information given in Table 6.6 is not easy readable for humans.

On the other hand, the graphical table given in Table 6.7 clearly shows the main figures of Table 6.6 by sparklines, much easier readable by humans.

Bibliography

Alfons, A., **Filzmoser, P.**, **Hulliger, B.**, **Meindl, B.**, **Schoch, T.** and **Templ, M.** (**2009**): *State-of-the-art in visualization of indicators in survey statistics.* Technical report CS-2009-4, Vienna University of Technology.

Alfons, A., **Holzer, J.** and **Templ, M.** (**2010**): laeken: Laeken indicators for measuring social cohesion. R package version 0.1.1.

Cleveland, W. S. (**1993**): Visualizing Data. New Jersey: AT&T Bell Labs.

Efron, B. and **Tibshirani, R.** (**1986**): *Bootstrap Methods for Standard Errors, Confi-*

Table 6.6: Usual representation of aggregated data.

	2002	2003	2004	2005	2006	2007	2008	2009
EU-14 until 1995	110861	115090	122394	131839	143473	154033	167401	181383
Germany	75262	78227	83592	91194	100439	109193	119807	130684
EU-10 / 2004	57407	59730	67791	75273	80840	87059	94256	
EU-2 / 2007	22440	24817	26339	27598	28422	28301	35282	41356
Switzerland, EWR	7311	7202	7355	7567	7737	7957	8145	8390
former Yugoslawia	306922	310827	305549	302332	300525	295005	290506	292730
Turkey	127147	127156	123043	116544	113068	108189	109179	110678
Other States in Europe	7819	10176	14496	21740	25623	27400	28967	31226
Africa	15127	16749	17574	19577	20366	20007	20656	21460
North America	7350	7233	7326	7527	7779	8043	8422	8755
Latin America	5350	6056	6693	7104	7611	7667	8179	8716
Asia	36889	41668	45392	48726	50987	52606	56252	59538
Ociania	1076	1108	1148	1139	1178	1219	1278	1377
stateless	26917	21264	17177	14917	14624	13512	13856	10839

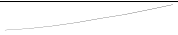
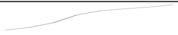
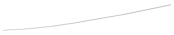
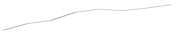
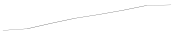
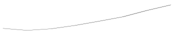
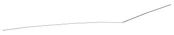
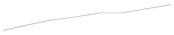
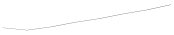
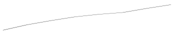
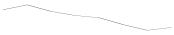
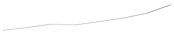
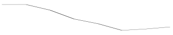
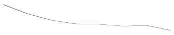
Domain	2002–2009	%	min	max	**Domain**	2002–2009	%	min	max
EU-14 until 1995		64	110	181	Other Eur.		299	8	31
Germany		74	75	130	Africa		42	15	21
EU-10		65	57	95	North Am.		19	7	9
EU-2		84	22	41	Latin Am.		63	5	9
Switzerl., EWR		15	7	8	Asia		61	37	60
former Yugos.		-5	290	311	Oceania		28	1	1.3
Turkey		-13	108	127	stateless		-60	11	27

Table 6.7: Graphical table with sparklines showing the relative change from 2002 to 2009 (in %) as well as the minimum and the maximum values.

dence Intervals, and Other Measures of Statistical Accuracy. Statistical Science, 1, pp. 54–75.

Kowarik, A., **Meindl, B.** and **Zechner, S.** (**2010**): sparkTable: Sparklines and graphical tables for tex and html. R package version 0.1.1.
URL http://CRAN.R-project.org/package=sparkTable

Tufte, E. (**2006**): Beautiful Evidence. Cheshire: Graphics Press.

Chapter 7

Visualisation of Simulation Results

7.1 Visualization of simulation results in simFrame

For the simulations within the AMELI project, a framework has been implemented in the R package **simFrame** (Alfons et al., 2010; Alfons, 2011). Visualization of simulation results with various plots is based on **lattice** graphics (Sarkar, 2008, 2011). The use of the available plot functions is demonstrated in the following examples.

However, it is not the aim of this section to discuss the presented methodology, nor will there be a thorough description of how to use the framework for carrying out simulation studies. A detailed description of the package contents, along with step-by-step instructions on using **simFrame** for different simulation designs are given in Alfons et al. (2010). This section is focused only on the presentation of the visualization methods. Nevertheless, the implementation of the framework offers two features that are important for visualization of the simulation results:

- Proportions of the data may be contaminated or set as missing.
- The simulations may be split into different domains.

Using **lattice** graphics, the results from different domains are displayed in separate panels of the plot. The following plot functions are currently implemented:

`simBwplot()` produces a boxplot of the simulation results.

`simDensityplot()` produces a kernel density plot of the simulation results.

`simXyplot()` plots the average results against the contamination levels or missing value rates.

`plot()` selects a suitable visualization method of the results depending on their structure.

The following two examples focus on visualization of simulation results in the case that contamination is used. However, the same principles apply if missing values are inserted in the simulations, and examples can be found in Alfons et al. (2010). Results from simulations in which no contamination or nonresponse is added are visualized in exactly the same way as the second example (with only one contamination level, see Section 7.1.2) illustrates.

7.1.1 Example 1

In the first example, the standard estimation of the *quintile share ratio* (QSR; Eurostat, 2004) is compared to two semi-parametric approaches that fit a *Pareto* distribution (e.g. Kleiber and Kotz, 2003) to the upper tail of the data: the classical *Hill* estimator (Hill, 1975) and the robust *partial density component* estimator (PDC; Vandewalle et al., 2007). Note that the semiparametric approaches flag values larger than the theoretical 99% quantile of the fitted distribution as nonrepresentative outliers. Since these are considered to be unique to the population, the sample weights of the flagged observations are then set to 1, and the weights of the remaining variables are calibrated accordingly. Details on the semiparametric methods can be found in Alfons et al. (2011a). All the aforementioned methods are implemented in the R package **laeken** (Alfons et al., 2011b).

With the following code, 100 samples are drawn from synthetic EU-SILC population data. Note that different contamination levels are used (no contamination, 0.5% and 1%), and that the simulations are carried out separately for each gender.

```
R> library(simFrame)
R> library(laeken)
R> data(eusilcP)
R> set.seed(1234)
R> set <- setup(eusilcP, design = "region", grouping = "hid",
+     size = c(75, 250, 250, 125, 200, 225, 125, 150, 100),
+     k = 100)
R> cc <- DCARContControl(target = "eqIncome", epsilon = c(0,
+     0.005, 0.01), dots = list(mean = 5e+05, sd = 20000))
R> sim <- function(x, k) {
+     q <- qsr(x$eqIncome, x$.weight)$value
+     fitHill <- paretoTail(x$eqIncome, k = k, method = "thetaHill",
+         w = x$.weight)
+     wHill <- reweightOut(fitHill, calibVars(x$region))
+     qHill <- qsr(x$eqIncome, wHill)$value
+     fitPDC <- paretoTail(x$eqIncome, k = k, method = "thetaPDC",
+         w = x$.weight)
+     wPDC <- reweightOut(fitPDC, calibVars(x$region))
+     qPDC <- qsr(x$eqIncome, wPDC)$value
+     c(standard = q, Hill = qHill, PDC = qPDC)
+ }
R> results <- runSimulation(eusilcP, set, contControl = cc,
+     design = "gender", fun = sim, k = 125)
```

In order to add reference lines to the plot, the true values of the QSR are computed from the population.

```
R> tv <- simSapply(eusilcP, "gender", function(x) qsr(x$eqIncome)$value)
```

The following commands create the plots shown in Figure 7.1. Since multiple contamination levels are used in this example, the default plot obtained with the `plot()` method draws the average results against the contamination levels (at the top of the figure). With the second command, results for 1% contamination are extracted and a density plot is produced (at the bottom).

```
R> plot(results, true = tv, ylab = "Quintile share ratio")
R> simDensityplot(results, true = tv, epsilon = 0.01,
+     xlab = "Quintile share ratio")
```

Figure 7.1 shows that the standard estimation of the QSR is highly influenced even by small proportions of outliers. Fitting a Pareto distribution to the upper tail of the data with the Hill estimator does not solve the problem. Using the PDC estimator, on the other hand, gives excellent results.

7.1.2 Example 2

Simulations in the AMELI project involve repeated sampling from very large data sets and are thus time consuming. Therefore, many different estimators will typically be computed in one simulation experiment. This example is similar to the previous one, but only contamination level 0.5% is used. Moreover, both the quintile share ratio (QSR) and the Gini coefficient are computed using the standard estimation and the semi-parametric approach with the PDC estimator. Even though some of the objects have already been defined in the previous section, the code for the whole simulation process is given below for a complete picture.

```
R> library(simFrame)
R> library(laeken)
R> data(eusilcP)
R> set.seed(1234)
R> set <- setup(eusilcP, design = "region", grouping = "hid",
```

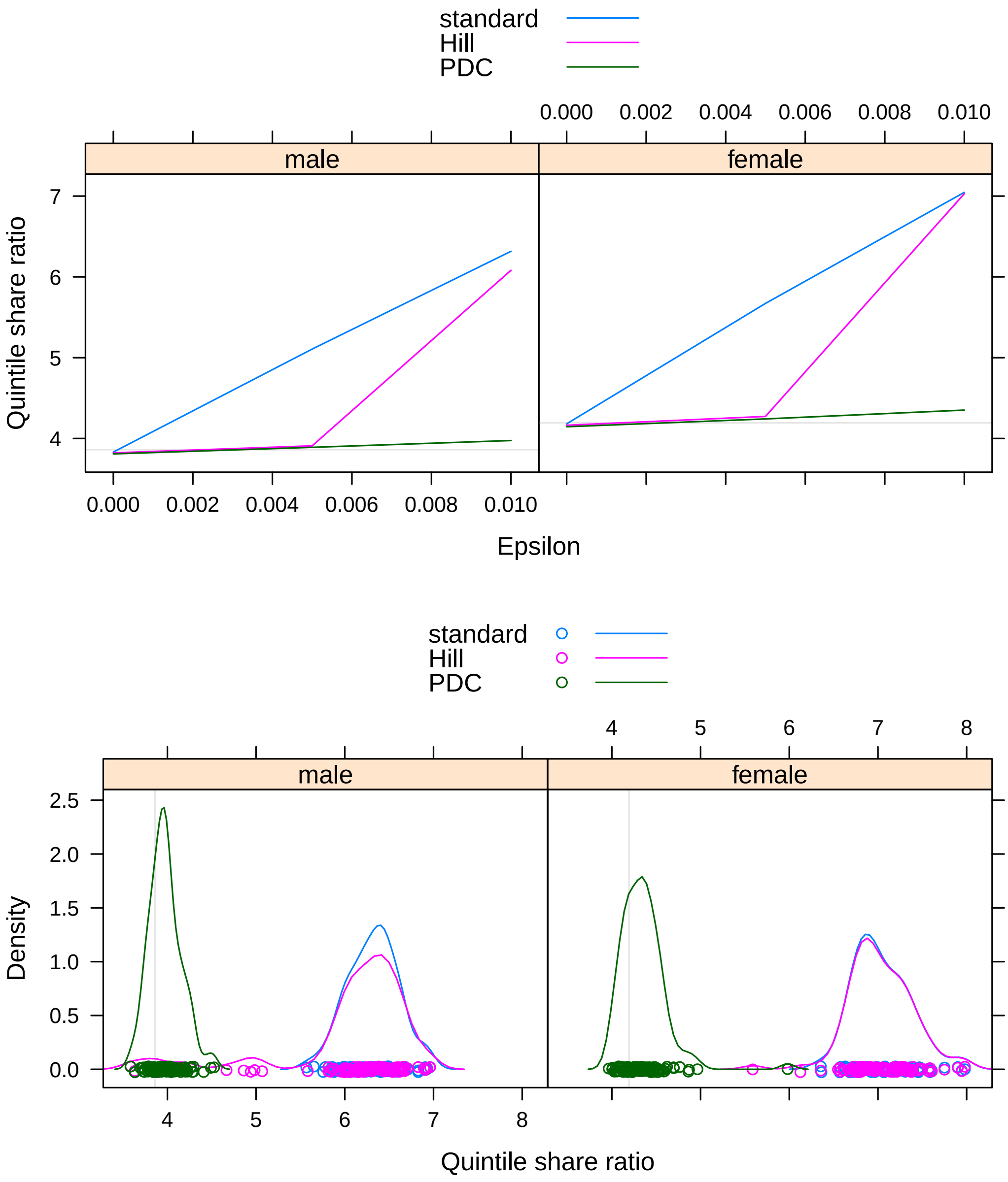


Figure 7.1: *Top*: Default plot of results from a simulation study with multiple contamination levels. *Bottom*: Kernel density plot of the simulation results for a specified contamination level (1%).

```
+     size = c(75, 250, 250, 125, 200, 225, 125, 150, 100),
+     k = 100)
R> cc <- DCARContControl(target = "eqIncome", epsilon = 0.005,
+     dots = list(mean = 5e+05, sd = 20000))

R> sim <- function(x, k) {
+     q <- qsr(x$eqIncome, x$.weight)$value
+     g <- gini(x$eqIncome, x$.weight)$value
+     fitPDC <- paretoTail(x$eqIncome, k = k, method = "thetaPDC",
+         w = x$.weight)
+     wPDC <- reweightOut(fitPDC, calibVars(x$region))
+     qPDC <- qsr(x$eqIncome, wPDC)$value
+     gPDC <- gini(x$eqIncome, wPDC)$value
+     c(QSR = q, qPDC = qPDC, Gini = g, gPDC = gPDC)
+ }
R> results <- runSimulation(eusilcP, set, contControl = cc,
+     design = "gender", fun = sim, k = 125)
```

The true values of the QSR and the Gini coefficient are computed to add reference lines as well.

```
R> tv <- simSapply(eusilcP, "gender", function(x) {
+     c(qsr(x$eqIncome)$value, gini(x$eqIncome)$value)
+ })
```

All plot functions in **simFrame** have a `select` argument, which provides a convenient way to specify to variables to be plotted.

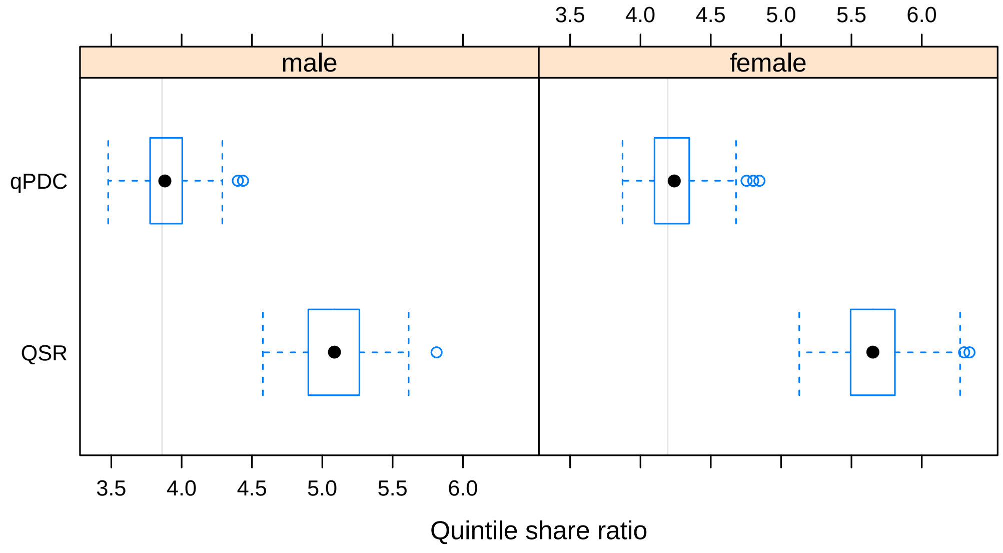
The following two commands visualize the simulation results for the QSR (see Figure 7.2). In the case of only one contamination level (or no contamination, for that matter), the `plot()` method produces a box-and-whisker plot (top of the figure). A density plot is produced with the second command (bottom of the figure).

```
R> plot(results, true = tv[1, ], select = c("QSR", "qPDC"),
+     xlab = "Quintile share ratio")
R> simDensityplot(results, true = tv[1, ], select = c("QSR",
+     "qPDC"), xlab = "Quintile share ratio")
```

In Figure 7.3, the simulation results for the Gini coefficient are displayed, which is obtained with the following code. Again, the top of the figure shows a box-and-whisker plot obtained with the `plot()` method, whereas the bottom of the figure shows a density plot.

```
R> plot(results, true = tv[2, ], select = c("Gini", "gPDC"),
+     xlab = "Gini coefficient")
R> simDensityplot(results, true = tv[2, ], select = c("Gini",
+     "gPDC"), xlab = "Gini coefficient")
```

Figure 7.2 and Figure 7.3 illustrate that the standard estimation of the QSR and the Gini coefficient,respectively, is already highly biased when only a small proportion of



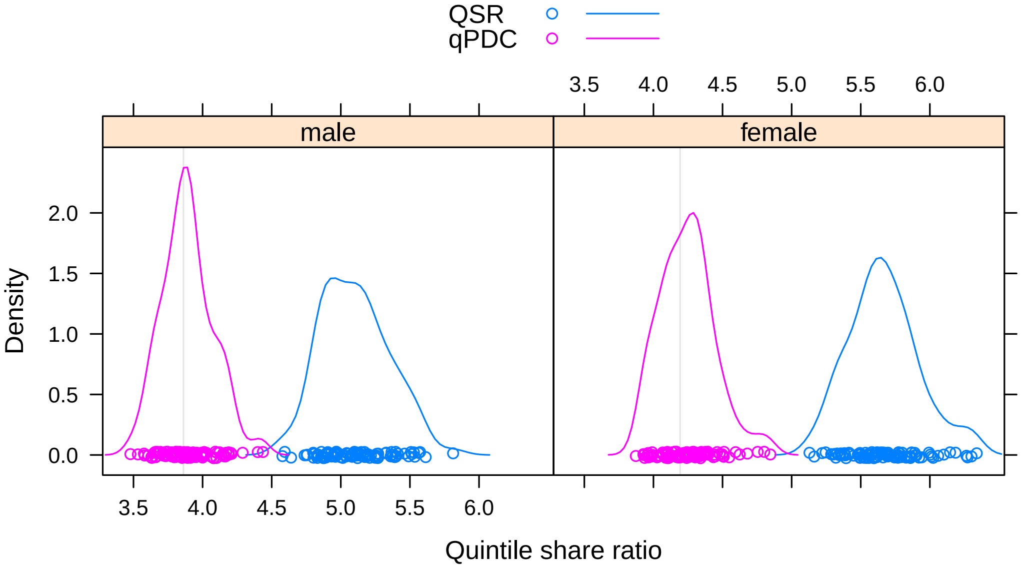


Figure 7.2: Simulation results for the QSR. *Top*: Default plot of results from a simulation study with one contamination level. *Bottom*: Kernel density plot of the simulation results.

contamination is present. Nevertheless, the semi-parametric approach with the PDC estimator performs well in this simple example. The similar results for QSR and Gini are not surprising, though, since these two indicators are closely related.

Bibliography

Alfons, A. (**2011**): **simFrame**: Simulation framework. R package version 0.4.1. URL http://CRAN.R-project.org/package=simFrame

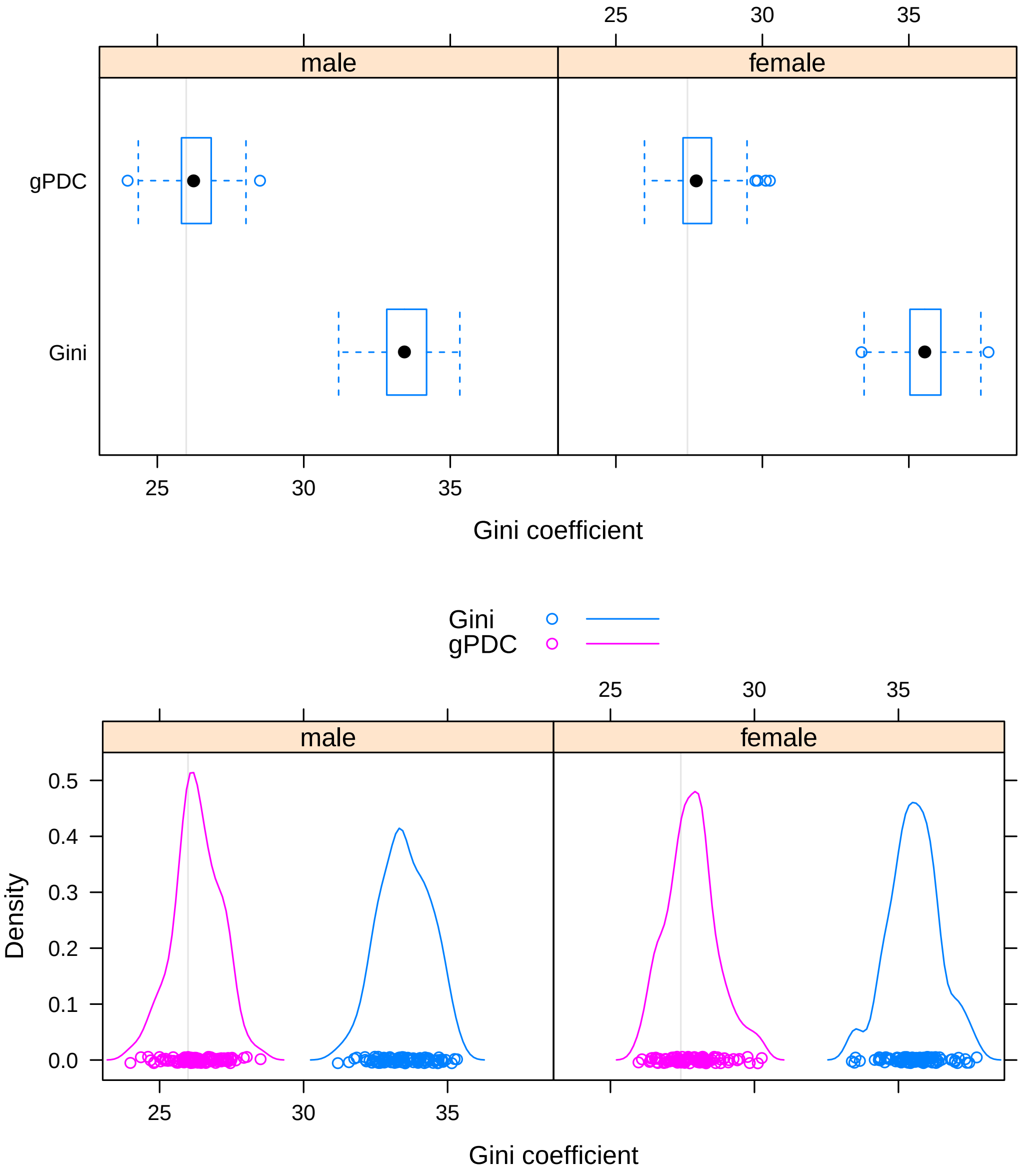


Figure 7.3: Simulation results for the Gini coefficient. *Top*: Default plot of results from a simulation study with one contamination level. *Bottom*: Kernel density plot of the simulation results.

Alfons, A., **Filzmoser, P.**, **Hulliger, B.**, **Meraner, A.**, **Schoch, T.** and **Templ, M.** (**2011**a): *Robust methods for Laeken indicators.* Deliverable 4.2, AMELI project.
URL http://ameli.surveystatistics.net

Alfons, A., **Holzer, J.** and **Templ, M.** (**2011**b): **laeken**: Laeken indicators for measuring social cohesion. R package version 0.2.1.
URL http://CRAN.R-project.org/package=laeken

Alfons, A., **Templ, M.** and **Filzmoser, P.** (**2010**): *An object-oriented framework for statistical simulation: The R package simFrame.* Journal of Statistical Software, 37 (3),

pp. 1–36.
URL http://www.jstatsoft.org/v37/i03/

Eurostat (**2004**): *Common cross-sectional EU indicators based on EU-SILC; the gender pay gap*. EU-SILC 131-rev/04, Unit D-2: Living conditions and social protection, Directorate D: Single Market, Employment and Social statistics, Eurostat, Luxembourg.

Hill, B. (**1975**): *A simple general approach to inference about the tail of a distribution.* The Annals of Statistics, 3 (5), pp. 1163–1174.

Kleiber, C. and **Kotz, S.** (**2003**): Statistical Size Distributions in Economics and Actuarial Sciences. Hoboken: John Wiley & Sons, ISBN 0-471-15064-9.

Sarkar, D. (**2008**): Lattice: Multivariate Data Visualization with R. New York: Springer, ISBN 978-0-387-75968-5.

Sarkar, D. (**2011**): **lattice**: Lattice graphics. R package version 0.19-17.
URL http://CRAN.R-project.org/package=lattice

Vandewalle, B., **Beirlant, J.**, **Christmann, A.** and **Hubert, M.** (**2007**): *A robust estimator for the tail index of Pareto-type distributions.* Computational Statistics & Data Analysis, 51 (12), pp. 6252–6268.

Chapter 8

Diagnostic Tools for Missing Values

Abstract: Package **VIM** allows to explore and to analyze the structure of missing values in data, as well as to produce high-quality graphics for publications. This paper illustrates an application of **VIM** to a highly complex data set – the European Statistics on Income and Living Conditions (EU-SILC).

8.1 Missing Values

Visualization of missing values is an absolutely necessary step before imputation. If rows including missing values are not omitted, missing values are always imputed, either as a data preparation step or within a statistical method.

Whenever missing values are imputed, one must be aware of the missing values mechanism(s). It can be shown that visualization tools provide deeper insights into the distribution of the missing values. Afterwards, a suitable imputation model could be chosen.

In the following the corresponding R functions and in package VIM and their application to EU-SILC are presented.

8.2 The graphical user interface of VIM

The graphical user interface (GUI) has been developed using the R package **tcltk** (R Development Core Team, 2009) and allows easy handling of the functions included in package **VIM**. Figure 8.1 shows the GUI, which pops up automatically after loading the package.

```
> library(VIM)
```

If the GUI has been closed, it can be reopened with the following command. All selections and settings from the last session are thereby recovered.

```
> vmGUImenu()
```

For visualization, the most important menus are the *Data*, the *Visualization* and the *Options* menus.

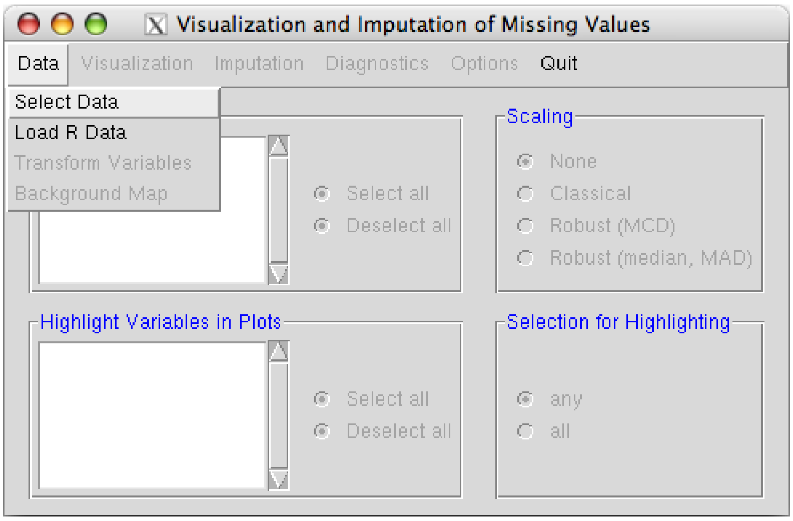


Figure 8.1: The **VIM** GUI and the *Data* menu.

8.2.1 Handling data

The *Data* menu allows to select a data frame from the R workspace (see Figure 8.2). In addition, a data set in `.RData` format can be imported from the file system into the R workspace, which is then loaded into the GUI directly.

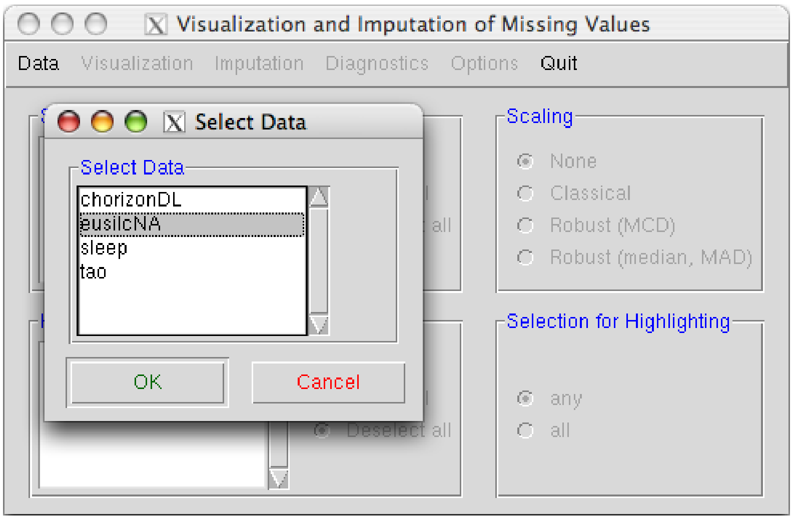
Transformations of variables are available via `Data` → `Transform Variables`. The transformed variables are thereby appended to the data set in use. Commonly used transformations in official statistics are available, e.g., the Box-Cox transformation (Box and Cox, 1964) and the log-transformation as an important special case of the Box-Cox transformation. In addition, several other transformations that are frequently used for compositional data (Aitchison, 1986) are implemented. Background maps and coordinates for spatial data can be selected in the *Data* menu as well.

Functionality to scale variables, on the other hand, is offered in the upper right frame of the GUI. Note that scaling is performed on-the-fly, i.e., the scaled variables are simply passed to the underlying plot functions, they are not permanently stored.

8.2.2 Selecting variables

After a data set has been chosen, variables can be selected in the main dialog (see Figure 8.3). An important feature is that the variables will be used in the same order as they were selected, which is especially useful for parallel coordinate plots.

Variables for highlighting are distinguished from the plot variables and can be selected separately (see the lower left frame in Figure 8.3). If more than one variable chosen for highlighting, it is possible to select whether observations with missing values in any or in all of these variables should be highlighted (see in the lower right frame in Figure 8.3).



Select Data

- chorizonDL
- eusilcNA
- sleep
- tao

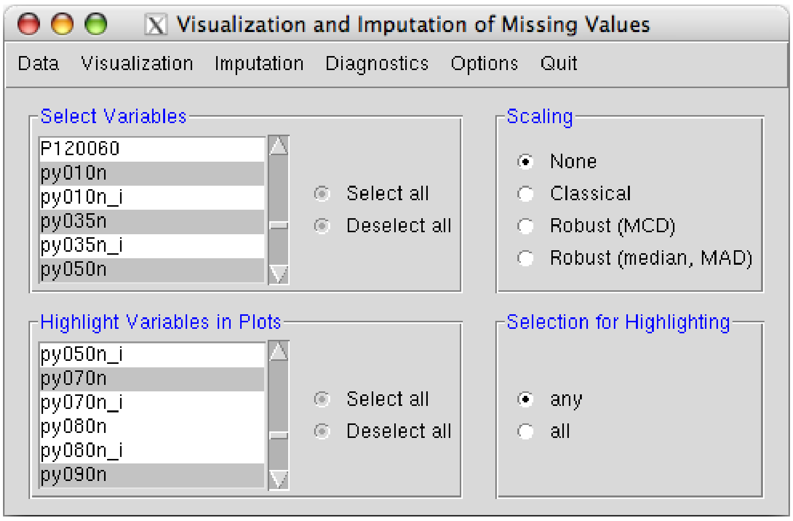
Scaling

- ◉ None
- ○ Classical
- ○ Robust (MCD)
- ○ Robust (median, MAD)

Selection for Highlighting

- ◉ any
- ○ all

Figure 8.2: The dialog for data selection.



Select Variables

- P120060
- py010n
- py010n_i
- py035n
- py035n_i
- py050n

○ Select all
○ Deselect all

Scaling

- ● None
- ○ Classical
- ○ Robust (MCD)
- ○ Robust (median, MAD)

Highlight Variables in Plots

- py050n_i
- py070n
- py070n_i
- py080n
- py080n_i
- py090n

○ Select all
○ Deselect all

Selection for Highlighting

- ● any
- ○ all

Figure 8.3: Variable selection with the **VIM** GUI.

8.2.3 Selecting plots

A plot method can be selected from the *Visualization* menu. Note that plots that are not applicable with the selected variables are disabled, e.g., if only one plot variable is selected, multivariate plots are not available.

8.3 An application to EU-SILC data

In this section, some of the visualization tools are illustrated on the public use sample of the Austrian EU-SILC data from 2004 (Statistics Austria, 2007), which can be obtained from Statistics Austria (see Table 8.1 for an explanation of the variables used here). This well-known and complex data set is mainly used for measuring risk-of-poverty and social cohesion in Europe, and for monitoring the Lisbon 2010 strategy of the European Union. The raw data set contains a high amount of missing values, which are imputed with model-based and donor-based imputation methods before public release (Statistics Austria, 2006). Since a high amount of missing values are not MCAR, the variables to be included for imputation need to be selected carefully. This problem can be solved with our proposed visualization tools.

Table 8.1: Explanation of the used variables from the EU-SILC data set.

name	meaning
age	Age
R007000	Occupation
P033000	Years of employment
py010n	Employee cash or near cash income
py035n	Contributions to individual private pension plans
py050n	Cash benefits or losses from self-employment
py070n	Values of goods produced by own-consumption
py080n	Pension from individual private plans
py090n	Unemployment benefits
py100n	Old-age benefits
py110n	Survivors' benefits
py120n	Sickness benefits
py130n	Disability benefits
py140n	Education-related allowances

```
> incvars <- c(paste("py", c("010", "035", "050", "070", "080",
+     "090", "100", "110", "120", "130", "140"), "n", sep=""))
> eusilcNA[, incvars] <- log10(eusilcNA[, incvars] + 1)
```

First of all, it may be of interest how many missing values are contained in each variable. Even more interesting, missing values may frequently occur in certain combinations of variables. This can easily investigated by selecting variables of interest (see Figure 8.3) and by clicking on `Visualization` → `Aggregate Missings`. If one prefers the command line language of R, the the plot in Figure 8.4 can be created by invoking:

```
> aggr(eusilcNA[, incvars], numbers=TRUE, prop = c(TRUE, FALSE))
```

Here `eusilcNA` denotes the data frame in use (see also Figure 8.2). The barplot on the left hand side shows the proportion of missing values in each of the selected variables. On the right hand side, all existing combinations of missing and non-missing values in the observations are visualized. A red rectangle indicates missingness in the corresponding variable, a blue rectangle represents available data. In addition, the frequencies of the different combinations are represented by a small bar plot and by numbers. Variables may be sorted by the number of missing values and combinations by the frequency of occurrence to give more power to finding the structure of missing values. For example, the top row in Figure 8.4 (right) represents the combination with missing values in variables *py010n* (employee cash or near cash income), *py035n* (contributions to individual private pension plans) and *py090n* (unemployment benefits), and observed values in the remaining variables, which appears only once in the data.

The plot reveals an exceptionally high number of missing values in variable *py010n*. The combination with variable *py035n* still contains 32 missing values. Note that it is possible to display proportions of missing values and combinations rather than absolute numbers.

8.3.1 Univariate plots

When only one variable is selected, only plots emphasized in Figure 8.5 can be applied. Standard univariate plots, such as barplots and spine plots for categorical variables and histograms, spinograms and different types of boxplots for continuous variables, have been adapted to display information about missing values.

For example, it may be of interest to display the distribution of years of employment, with missing values in *py010n* (employee cash or near cash income) highlighted. A spinogram (Hofmann and Theus, 2005) can easily be generated by clicking `Visualization` → `Spinogram with Missings`. Alternatively, the output shown in Figure 8.6 can be produced with the following command:

```
> spineMiss(eusilcNA[, c("P033000", "py010n")])
```

Figure 8.6 indicates that the probability of missingness in *py010n* depends on the years of employment.

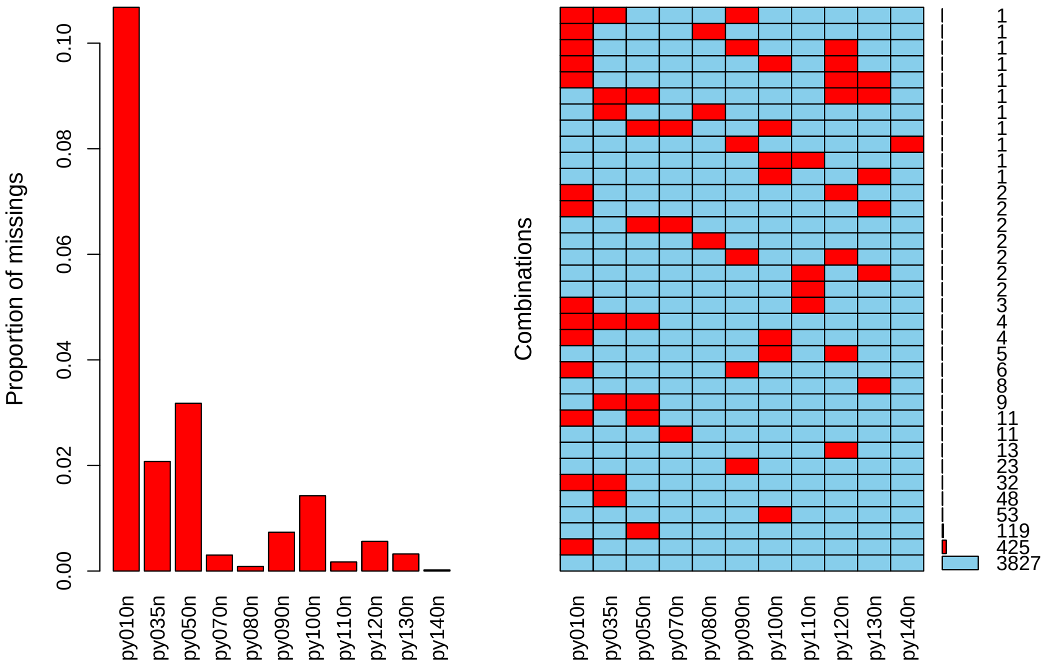


Figure 8.4: Aggregation plot of the income components in the public use sample of the Austrian EU-SILC data from 2004. *Left*: barplot of the proportions of missing values in each of the income components. *Right*: all existing combinations of missing (red) and non-missing (blue) values in the observations. The frequencies of the combinations are visualized by small horizontal bars.

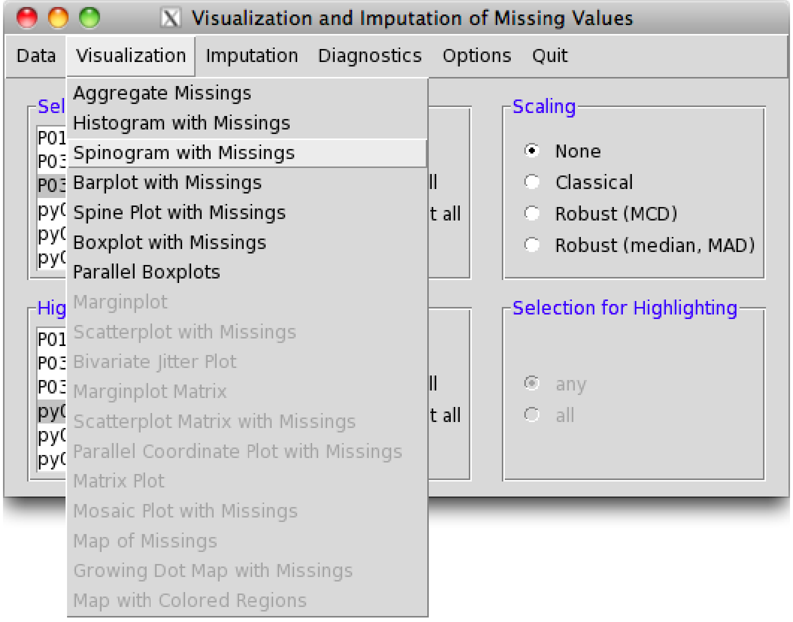


Figure 8.5: Univariate Plots supported by the **VIM** GUI.

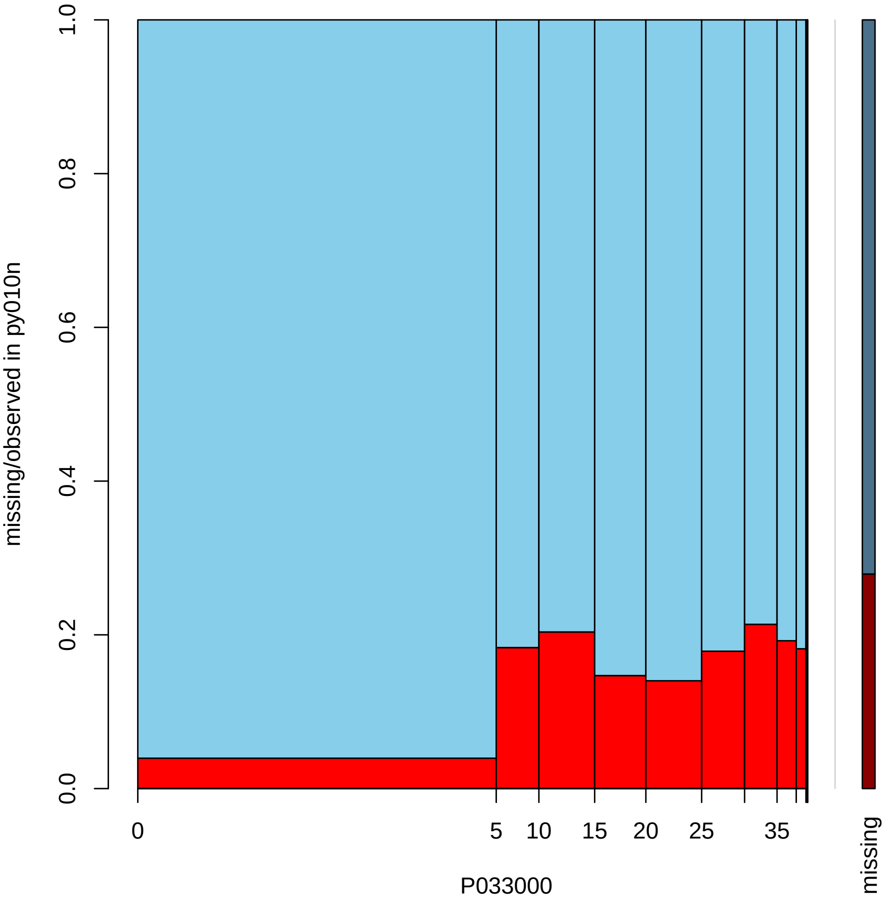
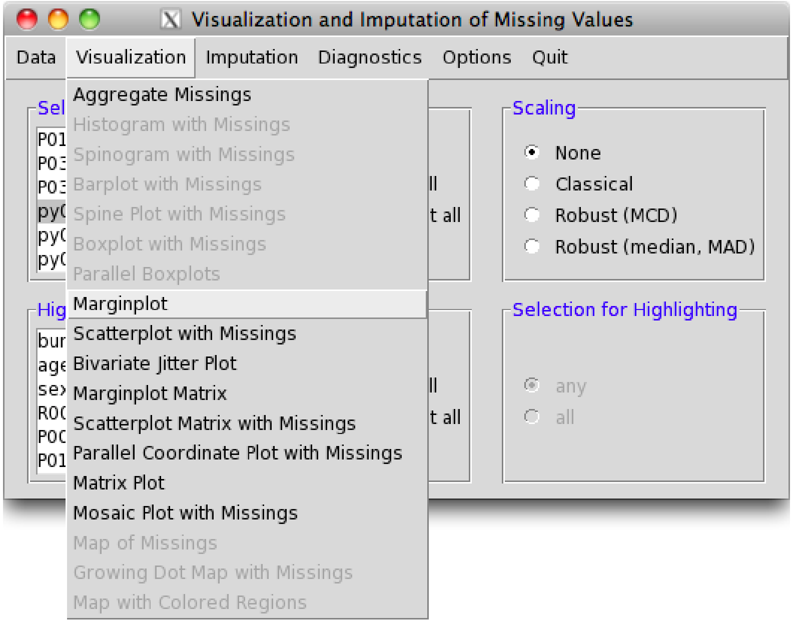


Figure 8.6: Spinogram of *P033000* (years of employment) with color coding for missing (red) and available (blue) data in *py010n* (employee cash or near cash income).



Scaling

● None
○ Classical
○ Robust (MCD)
○ Robust (median, MAD)

Selection for Highlighting

● any
○ all

Figure 8.7: Bivariate plots (as well as multivariate plots) available in the **VIM** GUI.

8.3.2 Bivariate plots

For bivariate data, different kinds of scatterplots are implemented. Figure 8.7 lists the plots applicable when two variables are selected. Multivariate plots are also highlighted because they can be used in the bivariate case, too.

Figure 8.8 shows a scatterplot with information about the univariate distributions and missingness of the variables in the plot margins (`Visualization` → `Marginplot`). The boxplots in red indicate observations with missing values in the other variable. It is clearly visible that the amount of missingness in *py010n* (employee cash or near cash income) is less for older people. Note that semi-transparent colors are used to prevent overplotting. The figure can also be produced with the command line interface of R, using the following command:

```
> marginplot(eusilcNA[, c("age", "py010n")], alpha = 0.6)
```

8.3.3 Multivariate plots

Parallel coordinate plots (WEGMAN, 1990) are very powerful for displaying multivariate relationships in data. A natural way of displaying information about missing data is to highlight observations according to missingness in a certain variable or a combination of variables. However, plotting variables with missing values results in disconnected lines, making it impossible to trace the respective observations across the graph. As a remedy, missing values may be represented by a point above the corresponding coordinate axis,

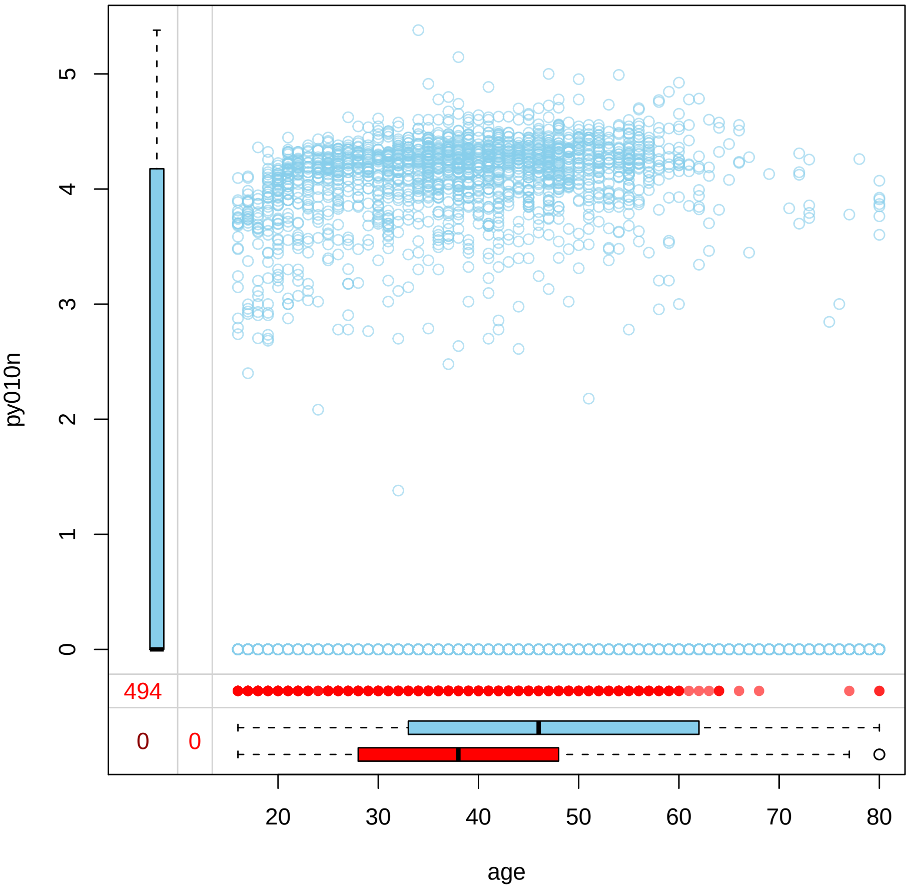


Figure 8.8: Scatterplot of *age* and transformed *py010n* (employee cash or near cash income) with information about missing values in the plot margins.

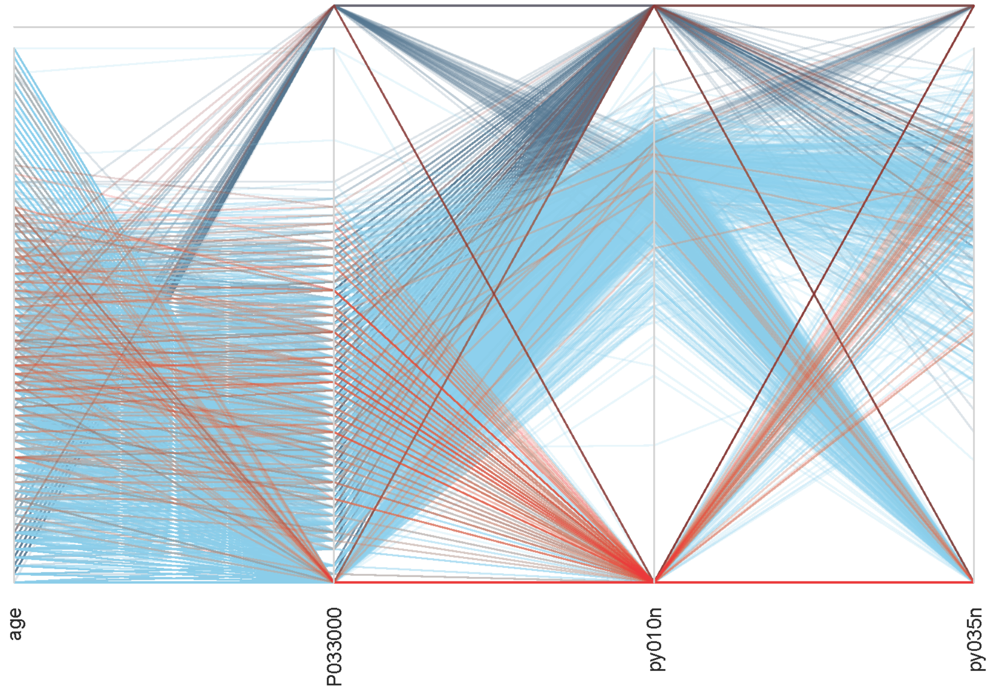


Figure 8.9: Parallel coordinate plot of *age*, *P033000* (years of employment), transformed *py010n* (employee cash or near cash income) and transformed *py035n* (contributions to individual private pension plans), with color coding for missing (red) and available (blue) data in variable *py050n* (cash benefits or losses from self-employment).

which is separated from the main plot by a small gap and a horizontal line (see Figure 8.9). Connected lines can then be drawn for all observations.

Such parallel coordinate plots can be genereated by clicking `Visualization` → `Parallel Coordinate Plot with Missings` in the GUI or by using the function `parcoordMiss()` on the command line. The example in Figure 8.9 can be produced with:

```
> parcoordMiss(eusilcNA[, c("age", "P033000", "py010n", "py035n",
+      "py050n")], plotvars = 1:4, highlight = 5, alpha = 0.2)
```

A data frame containing all variables of interest needs to supplied as the first argument, the variables to be plotted are given by argument `plotvars`, and the variables to be used for highlighting are specified by argument `highlight`.

Due to the large number of lines, a very low alpha value (i.e., very high transparancy) is used in Figure 8.9 to prevent overplotting. Missing values in *py050n* occur mainly for middle-aged people. Moreover, observations with missing values in *py050n* behave in an entire different way for the variables *py010n* (employee cash or near cash income) and *py035n* (contributions to individual private pension plans) than the main part of the data.

The *matrix plot* is an even more powerful multivariate plot. It visualizes all cells of the data matrix by (small) rectangles. In the example in Figure 8.10, red rectangles are drawn for missing values, and a greyscale is used for the available data. To determine the grey level, the variables are first scaled to the interval $[0, 1]$. Small values are assigned a light grey and high values a dark grey (0 corresponds to white, 1 to black). In addition, the observations can be sorted by the magnitude of a selected variable, which can also be done interactively by clicking in the corresponding column of the plot. Using the GUI, a matrix plot can be produced by clicking `Visualization` → `Matrix Plot`. The example in Figure 8.10 can also be created on the command line by invoking the following command:

```
> matrixplot(eusilcNA[, c("age", "R007000", incvars)],
+      sortby = "R007000")
```

Figure 8.10 shows a matrix plot of *age*, *R007000* (occupation) and the transformed income components, sorted by variable *R007000* (occupation). It is clearly visible that missing values in most income components depend on the occupation of the corresponding person. Thus the missing data mechanism was found to be MAR for these variables, which should be considered when applying imputation methods.

8.3.4 Other plots

Various other plots are availabe in the package and can also be created with the GUI (see Figures 8.5 and 8.7). For spatial data, mapping is supported if a background map is provided by the user, e.g., as a shape file, data frame or list of coordinates.

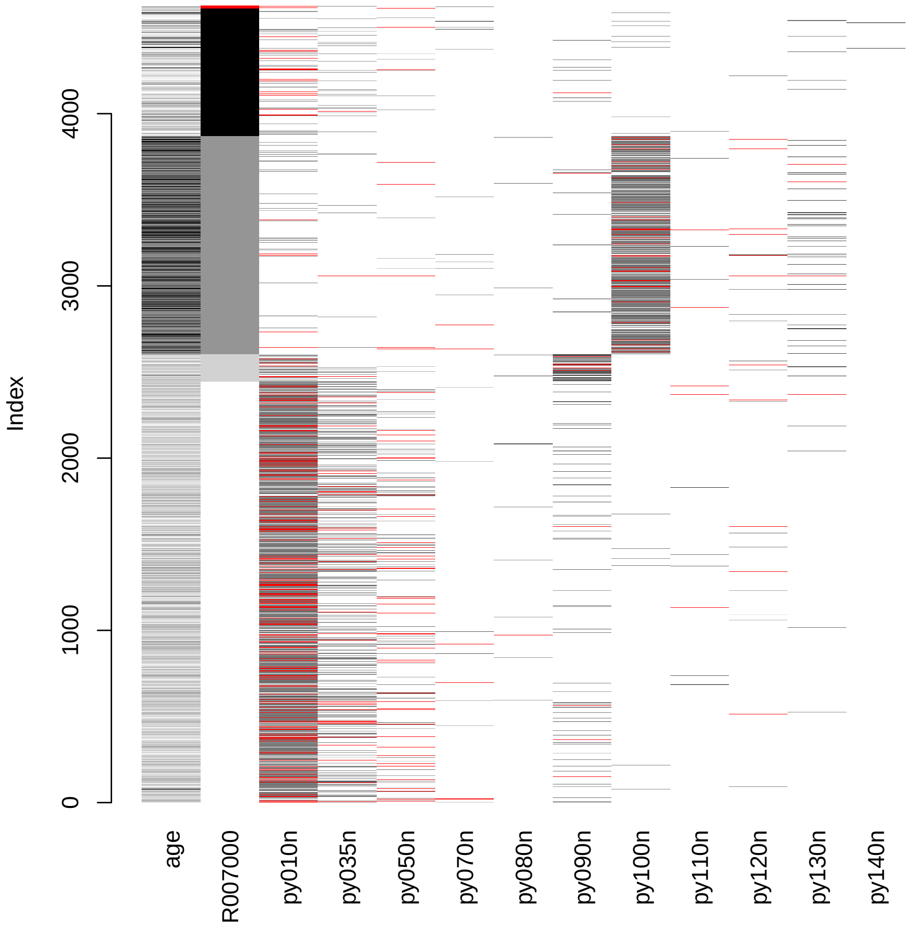
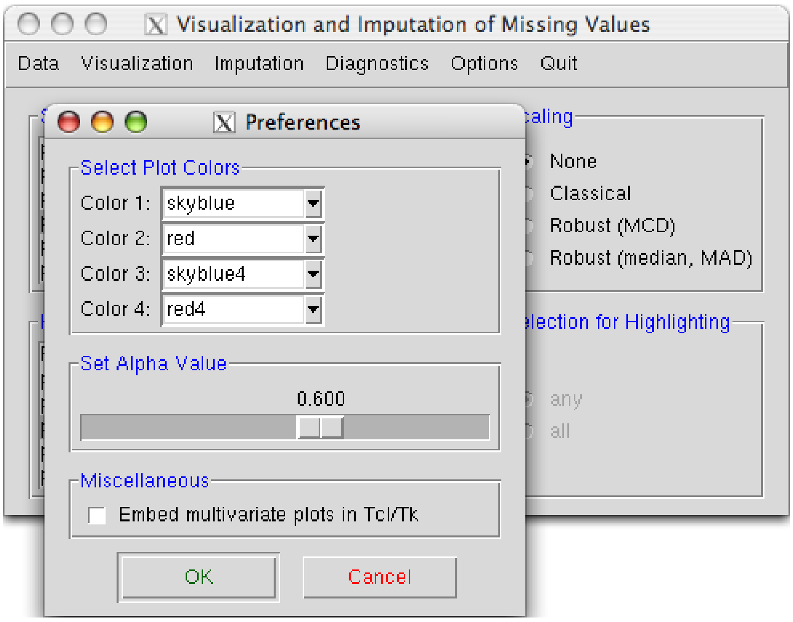


Figure 8.10: Matrixplot of *age*, *R007000* (occupation) and the transformed income components, sorted by variable *R007000* (occupation).



Select Plot Colors

Color 1: skyblue

Color 2: red

Color 3: skyblue4

Color 4: red4

Set Alpha Value

0.600

Miscellaneous

☐ Embed multivariate plots in Tcl/Tk

Figure 8.11: The *Preferences* dialog of the **VIM** GUI.

8.4 Fine tuning

In the *Preferences* dialog from the *Options* menu (click `Options` → `Preferences`), which is displayed in Figure 8.11, the colors and alpha channel to be used in the plots can be set. In addition, it contains an option to embed multivariate plots in Tcl/Tk windows. This is useful if the number of observations and/or variables is large, because scrollbars allow to move from one part of the plot to another.

8.5 Interactive features

Many interactive features are implemented in the plot functions in order to allow easy modification of the plots.

When variables are selected for highlighting in univariate plots such as histograms, barplots, spine plots or spinograms, it is possible to switch between the variables. Clicking in the right plot margin of a histogram, for example, corresponds with creating a histogram (or barplot) for the next variable, and clicking in the left margin switches to the previous variable. This interactive feature is particularly usedful for parallel boxplots, as it allows to view all possible $p(p-1)$ combinations with $p-1$ clicks, where p denotes the number of variables.

For multivariate plots (scatterplot matrix and parallel coordinate plot), variables for highlighting can be selected and deselected interactively, by clicking in a diagonal panel of the scatterplot matrix or on a coordinate axis in the parallel coordinate plot. Information about the current selection is printed on the R-console.

The matrixplot is particulary powerful if the observations are sorted by a specific variable (see Figure 8.10). This can be done by clicking on the corresponding column.

8.6 Highlighting of Imputed Values

Package **VIM** (TEMPL and FILZMOSER, 2008; TEMPL and ALFONS, 2009) can be easily enhanced to visualize imputed values. This is future work, however, we show two simple applications.

Visualization of imputed values is necessary to explore how well the imputations fits to the observed data part. With visualization it should be easy to see, how well an imputation method works, for example, it could be easily seen how worst column-wise mean imputation performs (the imputed values lay on “crosses”).

Here we do not assume knowledge about the observed values. The goal is to visualize the imputed values in an appropriate way.

The first diagnostic plot is a multiple scatterplot where the imputed values are highlighted. Figure 8.12 shows the result using four variables of the EU-SILC data, and we can see that the imputed values are placed certain hyperplanes, because the imputation was made by column-wise mean imputation. It is also visible that the former missing values of the imputated variables *py010n* and *py035n* (colored in red) do have a quite different behaviour than the observed data (colored in skyblue). This clearly indicates that mean imputation, although often used in practice, destroys the multivariate structure of the EU-SILC data. Of course, this example should demonstrate the plot methods, and in the AMELI-project much more sophisticated imputation methods will be developed for data like EU-SILC.

A parallel coordinate plot (WEGMAN, 1990) is shown in Figure 8.13. Again, the imputed values in certain variables are highlighted. One can select variables interactively, and imputed values in any of the selected variables will be highlighted. It is easy to see, that the imputation carried out imputes to low values.

The third diagnostic plot (see Figure 8.14), a ternary diagram (AITCHISON, 1986), assumes that the data are compositions or subcompositions (see AITCHISON (1986)). The 3-part compositions (here: *py010n*, *py035n*, *py050n*) are presented by red and skyblue colored pointsFigure 8.14 reports that the imputation made was really worst. The red points (the imputed observations) lay on very different places than the observed ones.

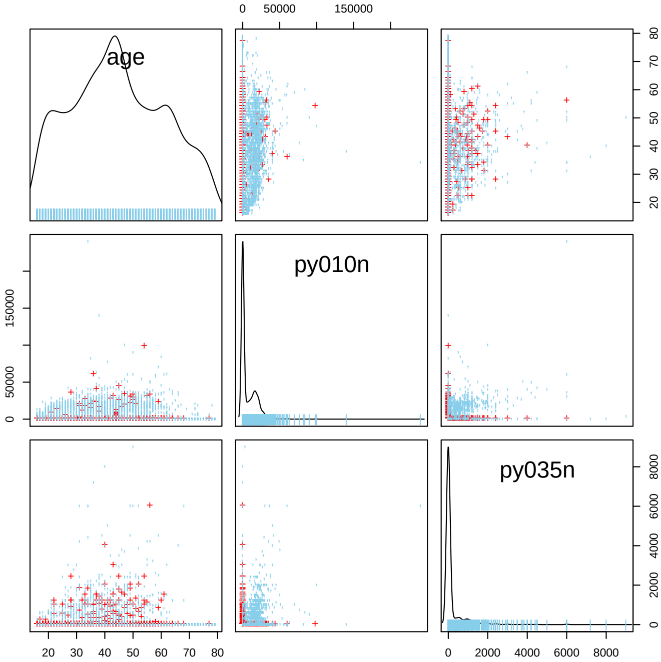


Figure 8.12: Multiple Scatterplot to highlight imputed observations.

8.7 Summary

We showed that the visualization of missing values is extremely simple with package **VIM**, either by using the GUI or by typing code on the R command line. With the visualization techniques in **VIM**, it is possible to gain insight into the data and to understand the structure of missing values. The latter is absolutely necessary when dealing with missing values, e.g., before imputation is performed.

8.8 Acknowledgments

This work was partly funded by the European Union (represented by the European Commission) within the 7th framework programme for research (Theme 8, Socio-Economic Sciences and Humanities, Project AMELI (Advanced Methodology for European Laeken Indicators), Grant Agreement No. 217322). Visit http://ameli.surveystatistics.net for more information on the project.

Bibliography

Aitchison, J. (**1986**): The Statistical Analysis of Compositional Data. Wiley, New York.

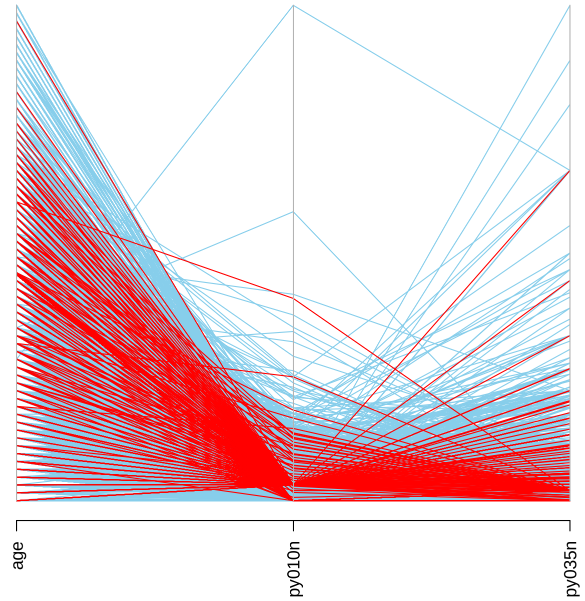


Figure 8.13: Interactive parallel coordinate plot in which observations with imputed values are highlighted in red.

Box, G. and **Cox, D.** (**1964**): *An Analysis of Transformations*. Journal of the Royal Statistical Society, Series B, 26, pp. 211–252.

Hofmann, H. and **Theus, M.** (**2005**): *Interactive graphics for visualizing conditional distributions*, unpublished manuscript.

R Development Core Team (**2009**): R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0.
URL http://www.R-project.org

Statistics Austria (**2006**): *Einkommen, Armut und Lebensbedingungen 2004, Ergebnisse aus EU-SILC 2004*. In German. ISBN 3-902479-59-0.

Statistics Austria (**2007**): *EU-SILC 2004. Erläuterungen: Mikrodaten-Subsample für externe Nutzer*. In German.

Templ, M. and **Alfons, A.** (**2009**): VIM: Visualization and Imputation of Missing Values. R package version 1.3.
URL http://cran.r-project.org/package=VIM

Templ, M. and **Filzmoser, P.** (**2008**): *Visualization of missing values using the R-package VIM*. Research report CS-2008-1, Department of Statistics and Probability Theory, Vienna University of Technology.
URL http://www.statistik.tuwien.ac.at/forschung/CS/CS-2008-1complete.pdf

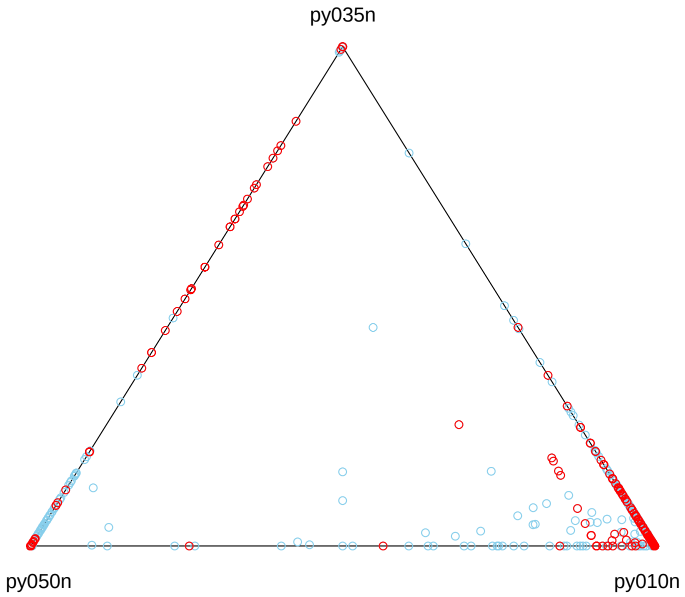


Figure 8.14: Ternary diagram with special plotting symbols for highlighting imputed parts of the compositional data.

Wegman, E. (**1990**): *Hyperdimensional data analysis using parallel coordinates*. Journal of the American Statistical Association, 85 (411), pp. 664–675.