REAT: A Regional Economic Analysis Toolbox for R

Thomas Wieland

1 Karlsruhe Institute of Technology, Karlsruhe, Germany

Received: 7 June 2019/Accepted: 4 November 2019

Abstract. Methods of regional economic analysis are widely used in regional and urban economics as well as in economic geography. This paper introduces the REAT (Regional Economic Analysis Toolbox) package for the programming environment R, which provides a collection of mathematical regional analysis methods in a user-friendly way. The focus is on the identification of regional inequality, beta and sigma convergence, measurement of agglomerations, point-based measures of clustering and accessibility, as well as regional growth. The theoretical basics of the applications are briefly introduced, while the usage of the most important functions is presented and explained using real data.

1 Introduction

Methods of regional economic analysis (or regional analysis) are used frequently in theory-based, empirical studies from regional and urban economics as well as (quantitative) economic geography. These methods aim at analyzing some of the most important issues in the mentioned research fields, including (but not limited to) the existence and evolution of agglomerations, regional economic growth and regional disparities (Capello, Nijkamp, 2009; Dinc, 2015; Farhauer, Kröll, 2014; Schätzl, 2000). In any of the mentioned fields, a growing amount of quantitative data has to be processed when using traditional or novel methods and models of regional analysis. This paper introduces the package (add-on) REAT (Regional Economic Analysis Toolbox) (Wieland, 2019) for the programming environment R (R Core Team, 2018a). The package provides a collection of mathematical regional analysis applications, designed in a relatively user-friendly way.

The main topics in the regional analysis context can be summarized as follows, showing also the structure of the present paper with respect to the presented approaches and their application in REAT:

1. Identifying regional inequality (or regional disparities) using indicators of concentration and/or dispersion (Section 2)
2. Regional disparities over time leading to the concept of beta and sigma convergence (Section 3)
3. Measuring agglomerations, which means the specialization of regions and the spatial concentration of industries as well as more complex cluster indices (Section 4)
4. Point-based measures of clustering and accessibility (Section 5)
5. Regional growth, especially shift-share analysis (Section 6)
Note that, in its original form, the open source software R is a command-line environment including a lot of mathematical and statistical features. For the installation of R and its packages as well as the basics of navigation and implemented statistical functions, see the R documentations (R Core Team, 2018b). A good supplement for working with R is RStudio (RStudio Team, 2016). The REAT package deals with several R data types: The most functions require and calculate numeric vectors, but, in some cases, also objects of type matrix, data frame and list, depending on the complexity of calculation. For a quick introduction to the data types in R and their properties, see e.g. Kabacoff (2017).

2 Concentration, dispersion and regional disparities

2.1 Indicators of concentration and dispersion

Regional disparities are a frequent topic in economic geography and regional economics. The spatial inequality with respect to e.g. regional output, income or employment is an essential element of polarization theory (Myrdal, 1957) and ”New Economic Geography” (Krugman, 1991; Fujita et al., 2001). Assessing regional disparities is possible using concentration and dispersion indicators, which belong to the univariate and descriptive analysis in statistics. Apart from regional economics, these measures are used in several contexts, such as competition economics (market concentration of firms) or welfare economics (income inequality). For a review of the most common indicators with respect to regional inequality, see Portnov, Felsenstein (2010), for studies comparing different indicators in the regional economic context using empirical data, see e.g. Gluschenko (2018); Habánik et al. (2013); Huang, Leung (2009); Palan (2017); Petrakos, Psycharls (2016).

Concentration is operationalized as the discrepancy between an empirical distribution of a variable \( x \) (e.g. annual turnover, income, gross domestic product [GDP]) with \( n \) observations or objects (e.g. competing firms, households, regions) and a (theoretical) equal distribution or a reference distribution (e.g. population distribution). Dispersion indicators aim at the deviation from the arithmetic mean of \( \bar{x} \). In this context, Portnov, Felsenstein (2005, 2010) distinguish between measures of deprivation and variation.

Typical measures of regional disparities are the Gini coefficient, the Herfindahl-Hirschman index and the coefficient of variation (Lessmann, 2005). The most popular measure of concentration is the Gini coefficient (Gini, 1912) in combination with the Lorenz curve (Lorenz, 1905). There are several calculation approaches for the Gini coefficient, all producing the same result. The Lorenz curve is a graphical indicator, showing the deviation of the empirical shares of the regarded variable \( x \) from a (theoretical) equal distribution. Another well-known indicator is the Herfindahl-Hirschman index, which was developed independently by Hirschman (1945) and Herfindahl (1950), both in the context of competition economics. Several other concentration indicators are also applied in the fields of regional economics with respect to regional disparities, such as the Hoover coefficient (Hoover, 1936) and the Theil coefficient (Theil, 1967).

Except for the standard deviation, whose unit is equal to the unit of \( x \), all common indicators are dimensionless. Most of them (except for standard deviation and coefficient of variation) have a fixed value range, normally between zero (indicating complete equality/dispersion) and one (indicating complete inequality/concentration).

Most of the common indicators are mathematically formulated in an unweighted and in a weighted form, while, in the context of regional disparities, the latter is mostly done using the regions’ proportion of the total (e.g. national) population (Doran, Jordan 2013; Lessmann 2014; Mussini 2017; Petrakos, Psycharls 2016; for a critical discussion of weighting these coefficients, see Gluschenko 2018). In the literature, there are different formulations where the weighted coefficients also include a weighted arithmetic mean. Note that, in the case of the population-weighted Gini coefficient, a weighted arithmetic mean is mandatory to keep the indicators’ value range.

Especially when dealing with GDP per capita as an indicator of regional economic output, several recent studies use dispersion measures rather than concentration measures, especially the (weighted) coefficient of variation (e.g. Lessmann 2005, 2014, 2016;
Lessmann, Seidel 2017; Petrakos, Psycharis 2016). This dispersion indicator is a dimensionless normalization of the standard deviation. Weighting the coefficient of variation with population shares was introduced by Williamson (1965), which has led to calling this coefficient the Williamson index. As regional incomes or outputs are not normally distributed in most cases, resulting in biased arithmetic means used in the calculation of dispersion measures, the regarded variable may be log-transformed, which means replacing \( x_i \) with \( \log(x_i) \) in the calculations.

Table 1 shows the common indicators, including their (population-)weighted and their normalized form (if there exist any) and the corresponding value ranges. The formulae are shown in a way that includes several ways of application. The regarded variable is always named \( x_i \), while the (population) weighting is called \( w_i \). Some indicators, such as the Hoover or the Coulter coefficient, require a variable representing a reference distribution the shares of \( x_i \) are compared to. This reference is not a weighting. However, in many studies, the regional population is also used for the reference distribution. In these cases, reference and weighting are the same data. The reference distribution may also be equal to \( 1/n \).

Several indicators are also used for the analysis of regional specialization or the spatial concentration of industries, such as the Hoover coefficient or the Herfindahl-Hirschman index or its inverse \( 1/\text{HHI} \); also known as the “equivalent number” in the competition context). Other coefficients of concentration and specialization are discussed in Section 4. The last coefficient in Table 1, the mean square successive difference (von Neumann et al., 1941) is a measure for time variability not originating from but also transferable to regional economics.

2.2 Application in REAT

2.2.1 REAT functions for concentration and dispersion indicators

Table 2 shows the functions for concentration and dispersion measures implemented in the REAT package. All functions require at least one argument, a numeric vector with a length equal to \( n \), containing the regarded variable \( x \) (e.g. income) with \( i \) observations (e.g. regions), where \( i = 1, \ldots, n \). This data may be a single vector or a column of a data frame or matrix.

An optional weighting of the vector \( x \) can be done using the function argument \texttt{weighting} which is also a numeric vector of length \( n \). By default, the functions remove missing \( \text{(NA)} \) values. The \texttt{hoover()} function always needs a reference distribution (see the Hoover coefficient formula in Table 1), which is stated via the \texttt{ref} argument, also requiring a numeric vector of length \( n \). If no reference variable is stated (\texttt{ref = NULL}), the reference is set to \( 1/n \).

All functions (except for \texttt{disp()} ) return the single value of the computed coefficient. In the relevant cases (\texttt{gini()}, \texttt{gini2()}, \texttt{herf()} and \texttt{cv()}), a normalization of the coefficient is possible using the function argument \texttt{coefnorm = TRUE}, returning the normalized coefficient instead of the raw coefficient. The function \texttt{disp()} is a wrapper for all mentioned functions, calculating all coefficients (except for the \texttt{MSSD}) at once for one vector \( x \) or a set of variables/columns from a data frame or matrix.

Note that there are two functions for the Gini coefficient, \texttt{gini()} and \texttt{gini2()}, both producing the same result in the unweighted case. The former function is designed for income inequality, where the \texttt{weighting} option is designed for the calculation of the Gini coefficient for groups (e.g. income classes), where the weighting represents the group mean. The function \texttt{gini2()} is designed for the population-weighted analysis of regional inequality.

2.2.2 Application example: Small-scale regional disparities in health care provision

Regional inequality with respect to health care providers is a topic of high societal significance. In Germany, the health care planning system (\textit{Kassenärztliche Bedarfsplanung}) attempts to flatten the disparities of local health care provision (\textit{Kassenärztliche Bundesvereinigung, 2013}). Here, we analyze small-scale regional disparities in health care
Table 1: Indicators of concentration and dispersion for analyzing regional disparities

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Unweighted</th>
<th>Weighted</th>
<th>Normalized</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gini</td>
<td>(G = \frac{1}{2n^2} \sum_{i=1}^{n} \sum_{j=1}^{n}</td>
<td>x_i - x_j</td>
<td>)</td>
</tr>
<tr>
<td>HHI</td>
<td>(HHI = \frac{\sum_{i=1}^{n} (\frac{x_i}{\sum_{i=1}^{n} x_i})^2}{\frac{1}{n^2} \sum_{i=1}^{n} x_i^2})</td>
<td>(HHI^* = \frac{HHI - \frac{1}{n}}{1 - \frac{1}{n}})</td>
<td>(0 \leq HHI^* \leq 1)</td>
</tr>
<tr>
<td>Hoover</td>
<td>(HC = \frac{1}{\sum_{i=1}^{n} \sum_{j=1}^{n} (\frac{x_i}{\sum_{i=1}^{n} x_i} - \frac{x_j}{\sum_{j=1}^{n} x_j})^2}{\frac{1}{n^2} (\sum_{i=1}^{n} x_i)^2})</td>
<td>(HC^w = \frac{1}{\sum_{i=1}^{n} \sum_{j=1}^{n} (\frac{w_i x_i}{\sum_{i=1}^{n} w_i x_i} - \frac{w_j x_j}{\sum_{j=1}^{n} w_j x_j})^2}{\frac{1}{n^2} (\sum_{i=1}^{n} w_i x_i)^2})</td>
<td>(0 \leq HC \leq 1)</td>
</tr>
<tr>
<td>Theil</td>
<td>(TC = \frac{1}{\sum_{i=1}^{n} \ln(\frac{\bar{x}_i}{\bar{x}})})</td>
<td>(TC^w = \frac{1}{\sum_{i=1}^{n} w_i \ln(\frac{\bar{x}_i}{\bar{x}})})</td>
<td>(0 \leq TC \leq 1)</td>
</tr>
<tr>
<td>Coulter</td>
<td>(CC = \sqrt{\frac{1}{\sum_{i=1}^{n} w_i (\frac{x_i}{\sum_{i=1}^{n} x_i} - \frac{\bar{x}}{n})^2}})</td>
<td>(0 \leq CC \leq 1)</td>
<td></td>
</tr>
<tr>
<td>Atkinson</td>
<td>(AF = 1 - \left(\frac{1}{\sum_{i=1}^{n} x_i^1} - \frac{1}{\sum_{i=1}^{n} x_i} \right))</td>
<td></td>
<td>(0 \leq AF \leq 1)</td>
</tr>
<tr>
<td>Dalton</td>
<td>(d = \frac{\log\left(\frac{1}{\sum_{i=1}^{n} x_i}\right) - \log\left(\frac{1}{n} \sum_{i=1}^{n} x_i\right)}{\log(n)\sqrt{\sum_{i=1}^{n} x_i^2 - \bar{x}^2}})</td>
<td></td>
<td>(0 \leq d \leq \infty)</td>
</tr>
<tr>
<td>SD</td>
<td>(s = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2})</td>
<td>(s^w = \sqrt{\frac{1}{n} \sum_{i=1}^{n} w_i (x_i - \bar{x})^2})</td>
<td>see CV</td>
</tr>
<tr>
<td>CV</td>
<td>(v = \frac{1}{\sum_{i=1}^{n} w_i (x_i - \bar{x})^2})</td>
<td></td>
<td>see Williamson</td>
</tr>
<tr>
<td>Williamson</td>
<td>(WI = \frac{1}{\sum_{i=1}^{n} w_i (x_i - \bar{x})^2})</td>
<td></td>
<td>(0 \leq v \leq \infty)</td>
</tr>
<tr>
<td>MSSD</td>
<td>(MSSD = n \sum_{i=1}^{n} \frac{(x_{i+1} - x_i)^2}{1 - \epsilon})</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 2: REAT functions for concentration and dispersion indicators

<table>
<thead>
<tr>
<th>Indicator</th>
<th>REAT function</th>
<th>Mandatory arguments</th>
<th>Optional arguments</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gini/Lorenz</td>
<td>gini()</td>
<td>vector x</td>
<td>weighting vector, remove NAs, Lorenz curve, normalization</td>
<td>value: G or G* or G* or G_w, optional: plot (LC)</td>
</tr>
<tr>
<td></td>
<td>gini2()</td>
<td>vector x</td>
<td>weighting vector P_1, remove NAs, normalization</td>
<td>value: G or G* or G* or G_w,</td>
</tr>
<tr>
<td></td>
<td>lorenz()</td>
<td>vector x</td>
<td>weighting vector, remove NAs,</td>
<td>plot LC, value: G or G* or G* or G*</td>
</tr>
<tr>
<td>HHI</td>
<td>herf()</td>
<td>vector x</td>
<td>remove NAs, normalization</td>
<td>value: HHI or HHI* or N_HHI</td>
</tr>
<tr>
<td>Hoover</td>
<td>hoover()</td>
<td>vector x</td>
<td>weighting vector P_1, remove NAs</td>
<td>value: HC or HC* or HC_w</td>
</tr>
<tr>
<td>Theil</td>
<td>theil()</td>
<td>vector x</td>
<td>weighting vector P_1, remove NAs</td>
<td>value: TC or TC* or TC_w</td>
</tr>
<tr>
<td>Coulter</td>
<td>coulter()</td>
<td>vector x</td>
<td>weighting vector P_1, remove NAs</td>
<td>value: CC</td>
</tr>
<tr>
<td>Atkinson</td>
<td>atkinson()</td>
<td>vector x</td>
<td>remove NAs, epsilon</td>
<td>value: AI</td>
</tr>
<tr>
<td>Dalton</td>
<td>dalton()</td>
<td>vector x</td>
<td>remove NAs</td>
<td>value: δ</td>
</tr>
<tr>
<td>SD</td>
<td>sd2()</td>
<td>vector x</td>
<td>weighting vector, remove NAs, treating as sample</td>
<td>value: s or s_W</td>
</tr>
<tr>
<td>CV</td>
<td>cv()</td>
<td>vector x</td>
<td>weighting vector, remove NAs, normalization, treating as sample</td>
<td>value: v or v_W or v^*</td>
</tr>
<tr>
<td>Williamson</td>
<td>williamson()</td>
<td>vector x, weighting vector P_1</td>
<td>remove NAs</td>
<td>value: WI</td>
</tr>
<tr>
<td>MSSD</td>
<td>mssd()</td>
<td>vector x</td>
<td>remove NAs</td>
<td>value: MSSD</td>
</tr>
<tr>
<td>All indicators</td>
<td>disp()</td>
<td>vector x or vectors x_1, x_2,... from dataframe</td>
<td>weighting vector P_1, remove NAs</td>
<td>matrix with 13 (no weighting) or 19 indicators (incl. weighted)</td>
</tr>
</tbody>
</table>

Source: own compilation.

(column district) and geographic coordinates (columns lat and lon, respectively) and the number of general practitioners, psychotherapists and pharmacies located there (columns phys_gen, psych and pharm, respectively) as well as the local population (column pop). First, we load the dataset:

data(GoettingenHealth2)

Now, we investigate how the health care providers are dispersed over the whole area. In the first step, we calculate the Gini coefficient for the concentration of general practitioners using the REAT function gini():

gini(GoettingenHealth2$phys_gen)
[1] 0.8386269

The empirical Gini coefficient is equal to 0.839, indicating a relatively strong concentration. If we want to calculate the normalized (unbiased) indicator instead, we use the same function with the optional argument coefnorm = TRUE:
In the same way, we calculate e.g. the Herfindahl-Hirschman index, non-normalized and normalized:

```
gerf(GoettingenHealth2$phys_gen)
```

```
gerf(GoettingenHealth2$phys_gen, coefnorm = TRUE)
```

Remember that the minimum of $HHI$ is $1/n$ (here: $1/420 \approx 0.00238$) and the minimum of $HHI^*$ is equal to zero.

If we want to inspect the concentration graphically, we could use the Lorenz curve, which can be plotted using either the functions `gini()` or `lorenz()`. Here, we use `gini()`, tell the function to plot the curve (`lc = TRUE`), and include several graphical parameters (such as `lc.col` for the color of the Lorenz curve or `lcx` and `lcy` for the x/y axes labels). As we want to compare the population distribution to the location distribution, we start by plotting the Lorenz curve for the local population:

```
gini(GoettingenHealth2$pop, lc = TRUE, lsize = 1, le.col = "black",
    lc.col = "orange", lc = "Shares of districts", lcy = "Shares of
    providers", lctitle = "Spatial concentration of health care
    providers", lc = TRUE, lcg = TRUE, lcg.caption =
        "Population 2016:"#, lcg.lab.x = 0, lcg.lab.y = 1)
# Gini coefficient and Lorenz curve for the no. of inhabitants
```

```
0.5840336
```

Now, we overlay the Lorenz curves of general practitioners and psychotherapists, which means adding two more curves (function argument `add.lc = TRUE`):

```
gini(GoettingenHealth2$phys_gen, lc = TRUE, lsize = 1, add.lc = TRUE,
    lc.col = "red", lc = TRUE, lcg = TRUE, lcg.caption =
        "Physicians 2016:"#, lcg.lab.x = 0, lcg.lab.y = 0.85)
# Adding Gini coefficient and Lorenz curve for the general practitioners
```

```
0.8386269
```

```
gini(GoettingenHealth2$psych, lsize = 1, lc = TRUE, add.lc = TRUE,
    lc.col = "blue", lcg = TRUE, lcg = TRUE, lcg.caption =
        "Psychotherapists 2016:"#, lcg.lab.x = 0, lcg.lab.y = 0.7)
# Adding Gini coefficient and Lorenz curve for psychotherapists
```

```
0.9329298
```

Our commands result in the output of Figure 1, showing three Lorenz curves (population, general practitioners and psychotherapists) and the line of equality (diagonal). All three empirical distributions differ from an equal distribution. In about 72% of the regions, representing about 23% of the whole population (orange curve; $G \approx 0.584$), no general practitioner is located (red curve; $G \approx 0.839$). But the psychotherapists are more concentrated, as they are located only in about 13% of all districts (blue curve; $G \approx 0.933$). As we can see, the physicians are more concentrated than the inhabitants but the psychotherapists are more concentrated than the physicians.

Now, we calculate all mentionended concentration and dispersion coefficients at once for all three types of providers using the function `disp()`, including a population weighting:

```
disp(GoettingenHealth2[c(5,6,7)], weighting = GoettingenHealth2$pop)
# column 5 = general practitioners, column 6 = psychotherapists,
# column 7 = pharmacies, column "pop" = local population
```

```
0.8406284
```

```
0.01528053
```

```
0.01293036
```
Figure 1: Lorenz curves for the spatial concentration of health care providers

Our output is:

**Concentration and dispersion measures**

Note: w = weighted, n = normalized, eq = equivalent number

<table>
<thead>
<tr>
<th>REGION</th>
<th>Volume 6, Number 3, 2019</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th></th>
<th>phys_gen</th>
<th>psych</th>
<th>pharm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gini</td>
<td>0.8386</td>
<td>0.9329</td>
<td>0.8915</td>
</tr>
<tr>
<td>Gini n</td>
<td>0.8406</td>
<td>0.9351</td>
<td>0.8937</td>
</tr>
<tr>
<td>Gini w</td>
<td>0.6294</td>
<td>0.7709</td>
<td>0.7056</td>
</tr>
<tr>
<td>Gini w n</td>
<td>0.6309</td>
<td>0.7727</td>
<td>0.7073</td>
</tr>
<tr>
<td>HHI</td>
<td>0.0153</td>
<td>0.0385</td>
<td>0.0241</td>
</tr>
<tr>
<td>HHI n</td>
<td>0.0129</td>
<td>0.0362</td>
<td>0.0218</td>
</tr>
<tr>
<td>HHI eq</td>
<td>65.44</td>
<td>25.98</td>
<td>41.38</td>
</tr>
<tr>
<td>Hoover</td>
<td>0.7214</td>
<td>0.0833</td>
<td>0.0218</td>
</tr>
<tr>
<td>Hoover w</td>
<td>0.0018</td>
<td>0.0031</td>
<td>0.0034</td>
</tr>
<tr>
<td>Theil</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Theil w</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Coulter</td>
<td>0.0498</td>
<td>0.1233</td>
<td>0.0656</td>
</tr>
<tr>
<td>Atkinson</td>
<td>0.7611</td>
<td>0.9007</td>
<td>0.8542</td>
</tr>
<tr>
<td>Dalton</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>SD</td>
<td>1.7145</td>
<td>1.0954</td>
<td>0.8653</td>
</tr>
<tr>
<td>SD w</td>
<td>4.0102</td>
<td>1.8477</td>
<td>2.4014</td>
</tr>
<tr>
<td>CV</td>
<td>2.3303</td>
<td>3.8992</td>
<td>3.0285</td>
</tr>
<tr>
<td>CV n</td>
<td>0.1138</td>
<td>0.1904</td>
<td>0.1479</td>
</tr>
<tr>
<td>Williamson</td>
<td>1.4294</td>
<td>1.9654</td>
<td>1.7092</td>
</tr>
</tbody>
</table>

We conclude that any concentration/dispersion measure is the highest for psychotherapists and the lowest for the general practitioners, while the values for pharmacies lie between them. The regional disparities with respect to pharmacies are higher than those with respect to general practitioners, while the most unequal distribution is that of psychotherapists. In other words: The pharmacies are more spatially concentrated than the general practitioners and the psychotherapists are the most concentrated health locations here.

In most cases, population weighting reduces the coefficient values. That is, because districts with a large (small) population have a high (low) impact on the resulting coefficient and the districts without health service providers are also small districts. Furthermore, as the regarded variables contain zero values (which means no health service locations), the Theil coefficient (including the term \( \ln(\bar{x}/x_i) \)) and the Dalton coefficient (including the \( n \)-th root) cannot be computed, resulting in an output of NA.
The visible output of any function presented above can be saved in a new R object:

```r
gini_phys <- gini(GoettingenHealth2$phys_gen)
# save as gini_phys (numeric vector of length = 1)
```

We can simply access our result:

```r
gini_phys
[1] 0.8386269
```

The function `disp()` returns a matrix with 13 rows (when only unweighted coefficients are computed) or 19 rows (in the case of additional weighted coefficients) and one column for each regarded variable:

```r
disp_Goettingen <- disp(GoettingenHealth2[c(5,6,7)],
weighting = GoettingenHealth2$pop)
# save as disp_Goettingen (matrix)
```

We call our results:

```r
disp_Goettingen
phys_gen psych pharm
Gini 0.83862691 0.93292978 0.89154762
Gini n 0.84062840 0.93515634 0.89367542
...
```

3 Regional convergence

3.1 The concept of beta and sigma convergence

Regional convergence is derived from (regional) growth theory (for an extensive survey, see Barro, Sala-i Martin 2004) and means the decline of regional disparities over time. The neoclassical growth model states that a region’s economic output (e.g. GDP per capita) depends on its stock of factors of production, capital and labor (aggregate production function), on condition of constant returns to scale and diminishing marginal product of the factor inputs. As a consequence, regions with a high (low) initial level of factor input grow slower (faster) than “poor” (“rich”) regions, what is called beta convergence. It is assumed that all regions converge to the same regional output level (steady-state). Sigma convergence means the decline of regional inequality with respect to regional output over time itself (Allington, McCombie, 2007; Capello, Nijkamp, 2009).

Both types of convergence can be tested empirically, as presented in Table 3. When testing for beta convergence, the natural logarithms of output growth over $T$ time periods in $i$ regions is regressed against the natural logarithms of the initial output values at time $t$. The original convergence formula was presented by Barro, Sala-i Martin (2004) using a nonlinear least squares (NLS) estimation approach. But in many cases, a linear transformation is used which allows for ordinary least squares (OLS) estimation (Allington, McCombie, 2007; Dapena et al., 2016; Schmidt, 1997; Young et al., 2008). The outcome variable of the convergence equation can be the regional growth between two years (e.g. Young et al. 2008) or the average growth rate per year (e.g. Goecke, Hüther 2016; Puente 2017; Weddige-Haaf, Kool 2017). Significance tests are carried out with $t$-tests for the regression coefficients and, in the OLS case, the $F$-test for the significance of $R^2$.

The estimated parameter of interest is the slope of the model, here denoted $\beta$ (that is why the modeled process is called beta convergence): If $\beta < 0$ and statistically significant, there is absolute beta convergence. If additional variables (conditional variables) are included into the convergence equation, we have a test for conditional beta convergence. A further interpretation of the $\beta$ coefficient is possible using the speed of convergence, $\lambda$, and $H$, the so-called half-life, which means the time (measured in the regarded time periods) to reduce the regional disparities by one half (Allington, McCombie, 2007; Schmidt, 1997).

Sigma convergence (which is named after the Greek letter for the standard deviation, $\sigma$) can be tested in two ways depending on the number of time periods: The regional
The analysis of beta convergence is provided by the functions \( \text{betaconv.ols()} \) and \( \text{betaconv.nls()} \) for OLS and NLS estimation, respectively. Speed of convergence and inequality between all regions at time \( t \) is measured using the standard deviation, \( \sigma_1 \), or the coefficient of variation, \( cv_1 \), for the GDP per capita in its original or natural-logged form. If only two years are regarded, the quotient of both parameters is computed. If e.g. \( \sigma_{11} > \sigma_{12} \), the regional inequality has declined from \( 11 \) to \( 12 \). A significance test can be applied with a simple ANOVA (analysis of variance), where the test statistic is the quotient of the underlying variances (\( \sigma^2 \)) (Furceri, 2005; Schmidt, 1997; Young et al., 2008). Within a time series, the dispersion parameter is regressed (and plotted) against time. If the slope coefficient of time is negative, there is sigma convergence (Gorecke, Hüther, 2016; Huang, Leung, 2009; Schmidt, 1997).

### 3.2 Application in REAT

#### 3.2.1 REAT functions for beta and sigma convergence

Table 4 shows the functions for beta and sigma convergence as implemented in REAT. The analysis of beta convergence is provided by the functions \( \text{betaconv.ols()} \) and \( \text{betaconv.nls()} \) for OLS and NLS estimation, respectively. Speed of convergence and

<table>
<thead>
<tr>
<th>Type of convergence</th>
<th>Two time periods</th>
<th>More than two time periods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beta convergence</td>
<td>NLS</td>
<td>NLS</td>
</tr>
<tr>
<td></td>
<td>[ \frac{1}{T} \ln\left( \frac{Y_{i,t+1}}{Y_{i,t}} \right) ]</td>
<td>[ \frac{1}{T} \sum_{t=1}^{T} \ln\left( \frac{Y_{i,t+1}}{Y_{i,t}} \right) ]</td>
</tr>
<tr>
<td></td>
<td>[ \alpha - \frac{1}{T} \ln(Y_{i,t}) + \epsilon ]</td>
<td>[ \alpha - \frac{1}{T} \sum_{t=1}^{T} \ln(Y_{i,t}) + \epsilon ]</td>
</tr>
<tr>
<td></td>
<td>OLS</td>
<td>OLS</td>
</tr>
<tr>
<td></td>
<td>[ \frac{1}{T} \ln\left( \frac{Y_{i,t+1}}{Y_{i,t}} \right) ]</td>
<td>[ \frac{1}{T} \sum_{t=1}^{T} \ln\left( \frac{Y_{i,t+1}}{Y_{i,t}} \right) ]</td>
</tr>
<tr>
<td></td>
<td>[ \alpha + \beta \ln(Y_{i,t+1}) + \epsilon ]</td>
<td>[ \alpha + \beta \ln(Y_{i,t+1}) + \epsilon ]</td>
</tr>
<tr>
<td></td>
<td>conditional</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NLS</td>
<td>NLS</td>
</tr>
<tr>
<td></td>
<td>[ \frac{1}{T} \ln\left( \frac{Y_{i,t+1}}{Y_{i,t}} \right) ]</td>
<td>[ \frac{1}{T} \sum_{t=1}^{T} \ln\left( \frac{Y_{i,t+1}}{Y_{i,t}} \right) ]</td>
</tr>
<tr>
<td></td>
<td>[ \alpha - \frac{1}{T} \ln(Y_{i,t}) + \theta X_i + \epsilon ]</td>
<td>[ \alpha - \frac{1}{T} \ln(Y_{i,t+1}) + \theta X_i + \epsilon ]</td>
</tr>
<tr>
<td></td>
<td>OLS</td>
<td>OLS</td>
</tr>
<tr>
<td></td>
<td>[ \frac{1}{T} \ln\left( \frac{Y_{i,t+1}}{Y_{i,t}} \right) ]</td>
<td>[ \frac{1}{T} \sum_{t=1}^{T} \ln\left( \frac{Y_{i,t+1}}{Y_{i,t}} \right) ]</td>
</tr>
<tr>
<td></td>
<td>[ \alpha + \beta \ln(Y_{i,t+1}) + \theta X_i + \epsilon ]</td>
<td>[ \alpha + \beta \ln(Y_{i,t+1}) + \theta X_i + \epsilon ]</td>
</tr>
</tbody>
</table>

Convergence speed: \( \lambda = -\frac{\ln(1+\beta)}{\beta} \)

Half-life: \( H = \frac{\ln(2)}{\beta} \)

Sigma convergence

\[ \sigma_t = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (Y_{i,t} - \bar{Y}_t)^2} \] or

\[ cv_t = \frac{\sigma_t}{\bar{Y}_t} \]

\[ \frac{cv_{t1}}{cv_{t2}} > 1 \text{ or } \frac{cv_{t1}}{cv_{t2}} > 1 \]

\[ \sigma = a + b \ln + \epsilon \text{ or } cv = a + b \ln + \epsilon \]

Test statistic: \[ \frac{\sigma_{t1}^2}{\sigma_{t2}^2} \]

Notes: \( Y_{i,t} \) is the regional output (e.g. GDP per capita) of region \( i \) at time \( t \), \( \bar{Y}_t \) is the arithmetic mean of \( Y_{i,t} \) for all regions at time \( t \), \( T \) is the number of regarded time periods (e.g. years), \( X_i \) is a set of other variables (conditions), \( \sigma_t \) is the standard deviation of the regional output of all regions, \( cv_t \) is the corresponding coefficient of variation, \( \alpha, \beta, \theta, a \) and \( b \) are estimated coefficients, \( \epsilon \) is an error term and \( n \) is the number of regions.

Compiled from: Allington, McCombie (2007); Barro, Sala-i Martin (2004); Furceri (2005); Schmidt (1997)
Table 4: REAT functions for beta and sigma convergence

<table>
<thead>
<tr>
<th>Convergence</th>
<th>REAT function</th>
<th>Mandatory arguments</th>
<th>Optional arguments</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beta convergence</td>
<td>betaconv.ols()</td>
<td>vectors $Y_{i,t1}$ and $Y_{i,t2}, ..., Y_{i,T}$, $t_1$ and $t_T$</td>
<td>Conditions, scatterplot</td>
<td>visible: model estimates, invisible: list with model estimates and regression data, optional: plot</td>
</tr>
<tr>
<td></td>
<td>betaconv.nls()</td>
<td>vectors $Y_{i,t1}$ and $Y_{i,t2}, ..., Y_{i,T}$, $t_1$ and $t_T$</td>
<td>Conditions, scatterplot</td>
<td>visible: model estimates, invisible: list with model estimates and regression data, optional: plot</td>
</tr>
<tr>
<td></td>
<td>betaconv.speed()</td>
<td>values $\beta$ and $T$</td>
<td></td>
<td>matrix with $A$ and $H$</td>
</tr>
</tbody>
</table>

Sigma convergence

- `sigmaconv()` (when $T = 2$): vectors $Y_{i,t1}$ and $Y_{i,t2}, t_1$ and $t_2$
  - Sigma measure, log, weighting, normalization
  - visible: estimates, invisible: matrix with estimates
- `sigmaconv.t()` (when $T > 2$): vectors $Y_{i,t1}$ and $Y_{i,t2}, ..., Y_{i,T}$, $t_1$ and $t_T$
  - Sigma measure, log, weighting, normalization, line plot
  - visible: model estimates, invisible: matrix with model estimates, optional: plot

All at once:

- `rca()`
  - vectors $Y_{i,t1}$ and $Y_{i,t2}, ..., Y_{i,T}$, $t_1$ and $t_T$
  - Beta estimation, conditions, scatterplot, sigma measure, log, weighting, line plot
  - visible: model estimates, invisible: list with model estimates and regression data, optional: plot

Source: own compilation.

Half-life can be computed with the function `betaconv.speed()`. The ratio test of sigma convergence for two time periods can be done using the function `sigmaconv()`, while a trend regression over time is implemented into the function `sigmaconv.t()`. Both convergence types can be analyzed at once with the function `rca()`, which is a wrapper for all functions mentioned above.

The functions require (at least) two numeric vectors, containing the regarded variable $Y$ (e.g. GDP per capita) for at least two different time periods, e.g. from the same data frame. Also the start and end time periods ($t_1$ and $t_T$) have to be stated. Optionally, a graphical output can be generated (scatterplot for beta convergence, line plot for sigma convergence with respect to longitudinal data). Furthermore, when analyzing sigma convergence, the user can choose whether $Y$ should be log-transformed or not and/or which sigma measure is computed (variance, standard deviation or coefficient of variation; weighted or non-weighted).

Note that, unlike the functions for regional inequality indicators (Section 2), the REAT functions for regional convergence distinguish between a visible and an invisible output. The latter can be saved as a new R object. While the visible output shows the main results, the invisible output goes beyond that: `betaconv.ols()`, `betaconv.nls()` and `rca()` return a list, which is the most flexible data type in R, because it consists of a non-predetermined number of different data objects. Apart from the model results, e.g. the (transformed) regression data is returned in this invisible output.

3.2.2 Application example: Beta and sigma convergence in Germany on the county level

In this example, we look at regional convergence in Germany. The REAT package includes the example dataset `G.counties.gdp` with the GDP (gross domestic product), the population and the GDP per capita for the 402 counties (“Kreise”) in Germany 1992 to 2014.
(complete data only for 2000-2014). First, we load the dataset:

```r
data(G.counties.gdp)
```

In our case, we prevent scientific notation of numbers in R and set a limit of 4 digits:

```r
options(scipen = 100, digits = 4)
```

We need the columns named `gdppcxxxx`, containing the GDP per capita for each year, e.g. `G.counties.gdp$gdppc2010` contains the GDP per capita for 2010. In the first step, we test absolute beta convergence comparing the years 2010 and 2014 with OLS estimation using the function `betaconv.ols()`:

```r
```

# Two years, no conditions (Absolute beta convergence)

The output is:

**Absolute Beta Convergence**

Model coefficients (Estimation method: OLS)

|            | Estimate | Std. Error | t value | Pr(>|t|) |
|------------|----------|------------|---------|----------|
| Alpha      | 0.104159 | 0.018934   | 5.501   | 0.00000006743 |
| Beta       | -0.007373 | 0.001848 | -3.990 | 0.00007867475 |
| Lambda     | 0.001850 | NA         | NA      | NA       |
| Halflife   | 374.640507 | NA | NA | NA   |

Model summary

|            | Estimate | F value | df 1 | df 2 | Pr(>|F|) |
|------------|----------|---------|------|------|----------|
| R-Squared  | 0.03827  | 15.92   | 1    | 400  | 0.00007867 |

We see that both regression coefficients, α and β, are statistically significant (t ≈ 5.50 and −3.99, respectively, both p < 0.001) and the linear regression model is significant as a whole (F ≈ 15.92, p < 0.001). The negative sign of β shows that, on average, the higher the initial GDP per capita, the lower its growth, which indicates absolute beta convergence. However, the convergence process is very slow: The speed of convergence, represented by λ, shows a harmonization by 0.185% per year. This implies that the output gap will be reduced by 50% in approximately 375 years.

Now we check sigma convergence for the same time using the function `sigmaconv()`.

We choose the coefficient of variation as measure, while using the GDP per capita values in their original form:

```r
```

# Using the coefficient of variation

The output is:

**Sigma convergence for two periods (ANOVA)**

|            | Estimate | F value | df1 | df2 | Pr(>|F|) |
|------------|----------|---------|-----|-----|----------|
| CV 2010    | 0.03416  | NA      | NA  | NA  | NA       |
| CV 2014    | 0.03316  | NA      | NA  | NA  | NA       |
| Quotient   | 1.03004  | 1.038   | 401 | 401 | 0.7117 |

The coefficient of variation is a little smaller in 2014, which means the spatial inequality declined between 2010 and 2014. The quotient of the variances is slightly above one (√F = σ^2_{2010}/σ^2_{2014} ≈ 1.04), but not statistically significant (p ≈ 0.71).

When analyzing regional convergence with `REAT`, it is preferable (and more convenient) to use the wrapper function `rca()`.

Instead of repeating the results above, we test for (absolute) beta and sigma convergence between 2000 and 2014. The analysis of sigma convergence uses trend regression (function argument `sigma.type = "trend"`) for the coefficient of variation (`sigma.measure = "cv"`). We also want plots for both convergence types (`beta.plot = TRUE` and `sigma.plot = TRUE`, respectively) with specific axis labels (e.g. `beta.plotX = "Ln (initial GDP p.c.)"`). Our code is:
rca (G.counties.gdp$gdppc2000, 2000, G.counties.gdp[55:68], 2014, conditions = NULL, sigma.type = "trend", sigma.measure = "cv", beta.plot = TRUE, beta.plotLine = TRUE, beta.plotX = "Ln (initial GDP p.c.)", beta.plotY = "Ln (av. growth GDP p.c.)", sigma.plot = TRUE, sigma.plotY = "cv of ln (GDP p.c.)", sigma.plotTitle = "Sigma convergence of German counties 2000-2014")

# 14 years: 2000 (column 55) to 2014 (column 68)
# no conditions (Absolute beta convergence)
# with plots for both beta and sigma convergence

This results in the following output:

Regional Beta and Sigma Convergence

Absolute Beta Convergence

Model coefficients (Estimation method: OLS)

| Estimate | Std. Error | t value | Pr(>|t|) |
|----------|------------|---------|---------|
| Alpha    | 0.0954564  | 0.0099087 | 9.634 | 0.00000000000000000006845 |
| Beta     | -0.0071323 | 0.0009885 | -7.215 | 0.00000000000271925822550 |
| Lambda   | 0.0005113  | NA       | NA     | NA |
| Halflife | 1355.7282963 | NA | NA     | NA |

Model summary

| Estimate | F value | df 1 | df 2 | Pr(>|F|) |
|----------|---------|------|------|---------|
| R-Squared| 0.1152  | 52.06 | 400  | 0.000000000002719 |

Sigma convergence (Trend regression)

| Estimate | Std. Error | t value | Pr(>|t|) |
|----------|------------|---------|---------|
| Intercept| 0.5523659  | 0.0308485 | 17.91  | 0.00000000001526 |
| Time     | -0.0002579 | 0.00001537 | -16.78 | 0.00000000034446 |

Model summary

| Estimate | F value | df 1 | df 2 | Pr(>|F|) |
|----------|---------|------|------|---------|
| R-Squared| 0.9558  | 281.4 | 13   | 0.000000003446 |

This function also produces the plots in Figures 2a and 2b, both showing a declining curve, which is a first indication of both beta and sigma convergence. The beta convergence model is statistically significant ($F \approx 52.06, p < 0.001$), as well as the coefficients $\alpha$ ($t \approx 9.63, p < 0.001$) and $\beta$ ($t \approx -7.21, p < 0.001$). Again, we find evidence for absolute beta convergence because of a negative slope ($\beta \approx -0.007$). The trend regression model for sigma convergence is significant ($F \approx 281.4, p < 0.001$). The slope is significant and negative ($b \approx -0.00026, t \approx 17.91, p < 0.001$), which indicates sigma convergence. However, both types of convergence can be regarded as very slow processes: The half-life value shows that, resulting from the beta convergence model, the regional disparities in GDP per capita will be halved in approximately 1,356 years. When looking at the trend regression, we see that the coefficient of variation declines only by 0.000026 per year. Another aspect is that we only regarded absolute beta convergence, ignoring other spatial effects or the impact of regional policy. The latter is also not considered in neoclassical regional growth theory.

Remembering German reunification, we want to test if there are average growth differences between West Germany and East Germany (former German Democratic Republic), which leads to conditional beta convergence. The dataset G.regions.emp contains the column regional, where the counties are attributed either to West or East Germany, expressed as character string ("West" or "East"). We need to include our condition into the convergence equation. Thus, we use the REAT function to.dummy() to create dummy variables (1/0) out of (nominal scaled) variables, and add the indicator for West Germany (1, otherwise 0) to our data:

```
regionaldummies <- to.dummy(G.counties.gdp$regional)
# Creating dummy variables for West/East
# regionaldummies[,1] = East (1/0), regionaldummies[,2] = West (1/0)
G.counties.gdp$West <- regionaldummies[,2]
# Adding the dummy variable for West
```
Now, we test for conditional beta and sigma convergence, including the condition “West”, again using the rca() function, but without plots and using the standard deviation (default setting) instead of the cv for sigma convergence. This time, we save the results in an object:

```r
converg_results <- rca(G.counties.gdp$gdppc2000, 2000,
                       G.counties.gdp[55:68], 2014, conditions = G.counties.gdp[c(70)],
                       sigma.type = "trend")
```

The output is:

**Regional Beta and Sigma Convergence**

**Absolute Beta Convergence**

Model coefficients (Estimation method: OLS)

|            | Estimate | Std. Error | t value | Pr(>|t|) |
|------------|----------|------------|---------|----------|
| Alpha      | 0.0954564| 0.0099087  | 9.634   | 0.00000000000000000006845 |
| Beta       | -0.0071323| 0.0009885  | -7.215  | 0.00000000071925822550 |
| Lambda     | 0.0005113 | NA         | NA      | NA       |
| Halflife   | 1355.7282963 | NA         | NA      | NA       |

Model summary

<table>
<thead>
<tr>
<th>Estimate</th>
<th>F value</th>
<th>df 1</th>
<th>df 2</th>
<th>Pr (&gt;F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-Squared</td>
<td>0.1152</td>
<td>52.06</td>
<td>1 400</td>
<td>0.0000000000000002719</td>
</tr>
</tbody>
</table>

**Conditional Beta Convergence**

Model coefficients (Estimation method: OLS)

|            | Estimate | Std. Error | t value | Pr(>|t|) |
|------------|----------|------------|---------|----------|
| Alpha      | 0.0754412| 0.0102354  | 7.371   | 0.0000000009872 |
| Beta       | -0.0047020| 0.0010517  | -4.471  | 0.0000101294 |
| West       | -0.0053559| 0.0009745  | -5.496  | 0.0000100790 |
| Lambda     | 0.0003366 | NA         | NA      | NA       |
| Halflife   | 2058.955949 | NA        | NA      | NA       |

Model summary

<table>
<thead>
<tr>
<th>Estimate</th>
<th>F value</th>
<th>df 1</th>
<th>df 2</th>
<th>Pr (&gt;F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-Squared</td>
<td>0.1774</td>
<td>43.04</td>
<td>2 399</td>
<td>0.000000000000001192</td>
</tr>
</tbody>
</table>

**Sigma convergence (Trend regression)**

|            | Estimate | Std. Error | t value | Pr(>|t|) |
|------------|----------|------------|---------|----------|
| Intercept  | 3.895236 | 0.3267817  | 11.92   | 0.0000002264 |
| Time       | -0.001764 | 0.0001628  | -10.84  | 0.0000007041 |

Model summary

<table>
<thead>
<tr>
<th>Estimate</th>
<th>F value</th>
<th>df 1</th>
<th>df 2</th>
<th>Pr (&gt;F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-Squared</td>
<td>0.9003</td>
<td>117.4</td>
<td>1 13</td>
<td>0.000000007041</td>
</tr>
</tbody>
</table>
In the \texttt{rca()} output, we can compare the results of absolute and conditional beta convergence. In the conditional model, the explained variance increases from $R^2 \approx 0.12$ to $R^2 \approx 0.18$, which indicates an increased explanatory power of the model due to the added condition variable. Both models are statistically significant, also the $\beta$ values are negative and significant ($p < 0.001$ in both cases). The condition “West” is significant ($t \approx -5.50$, $p < 0.001$) and negative, which means that, on average, the GDP per capita in West German counties grew slower than in East Germany. These results seem to support the convergence hypothesis from growth theory, but one should not forget that e.g. political aspects (such as the German and/or EU regional policy) are not considered in this simple analysis.

As we have saved the invisible function output, we can access specific parts of our analysis, such as the regression data for the absolute convergence model:

\begin{verbatim}
  converg_results$betaconv$regdata
  # All results in list converg_results
  # converg_results contains list betaconv (beta convergence results)
  # betaconv contains data frame regdata (regression data)

  ln_initial ln_growth
  1      11.002 0.01997436
  2      10.552 0.02980133
  3      10.283 0.01794207
  4      10.090 0.01763444
  5      10.287 0.02361006
  ...
\end{verbatim}

If we want to look at the single sigma values, we can address them via:

\begin{verbatim}
  converg_results$sigmaconv$sigma.trend
  # All results in list converg_results
  # converg_results contains list sigmaconv (sigma convergence results)
  # sigmaconv contains data frame sigma.trend (sigma values)

  years sigma.years
  gdp1  2000 0.3646
  gdppc2001 2001 0.3662
  gdppc2002 2002 0.3618
  gdppc2003 2003 0.3606
  gdppc2004 2004 0.3592
  ...
\end{verbatim}

4 Specialization of regions and spatial concentration of industries

4.1 Indicators of regional specialization and industry concentration

Specialization of regions or countries and the spatial concentration of industries or firms are phenomena linked to several research fields in regional economics and economic geography: Specialization is a key point in traditional theories of international trade with respect to comparative advantages (Ricardo, 1821) as well as in the generation of the “New Trade Theory” (introduced by Krugman 1979). Spatial clustering of firms or industries due to agglomeration economies is a perennial issue in all spatial economic fields. It especially reemerged in the context of the “New Economic Geography” (e.g. Krugman 1991; Fujita et al. 2001) as well as through the work of Porter (1990) regarding clusters. The common indicators are broadly discussed in Farhauer, Kröll (2014) or Nakamura, Morrison Paul (2009). For studies comparing some different indicators, see e.g. Goschin et al. (2009); Moga, Constantin (2011); Palan (2017).

When looking at the family of indicators of regional specialization and industry concentration, we have to distinguish between indicators for aggregate data, such as regional employment data, and those requiring individual firm data. The first group, compiled in Table 5, can be differentiated into indicators of specialization and indicators of spatial concentration. As both types of agglomeration are closely linked to each other, so are the
Table 5: Coefficients of regional specialization and industry concentration

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Specialization of region ( j )</th>
<th>Spatial concentration of industry ( i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hoover/Balassa</td>
<td>( LQ_{ij} = \frac{e_{ij}/e_j}{e_{ij}/e_i} \equiv MRCA_{ij} = \frac{e_{ij}/e_j}{e_{ij}/e_i} )</td>
<td>( LR_{ij} = \frac{1}{2} \sum_{l=1}^I LR_{ij,l} )</td>
</tr>
<tr>
<td>( TQ_{ij} = \frac{1}{2} \sum_{l=1}^I LQ_{ij,l} )</td>
<td>( LR_i = \frac{1}{2} \sum_{j=1}^J LQ_{ij} )</td>
<td></td>
</tr>
<tr>
<td>Extensions:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>O’Donoghue-Gleave</td>
<td>( SLQ_{ij} = \frac{LQ_{ij} - TQ_{ij}}{\sum_{l=1}^I TQ_{ij,l}} )</td>
<td>( \frac{LQ_{ij}}{\sum_{l=1}^I TQ_{ij,l}} )</td>
</tr>
<tr>
<td>Tian</td>
<td>( SLLQ_{ij} = \frac{\log(LQ_{ij}) - \log(TQ_{ij})}{\sum_{l=1}^I \log(TQ_{ij,l})} )</td>
<td></td>
</tr>
<tr>
<td>Hoen-Oosterhaven</td>
<td>( ARCA_{ij} = \frac{e_{ij}}{e_j} - \frac{e_{ij}}{e_i} )</td>
<td>( H_i = \frac{1}{2} \left( \sum_{j=1}^J \frac{</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( 0 \leq H_i \leq 1 )</td>
</tr>
<tr>
<td>Gini</td>
<td>( G_j = \frac{2}{T \cdot R} \sum_{i=1}^I \lambda_i (R_i - \bar{R}) )</td>
<td>( G_i = \frac{2}{T \cdot R} \sum_{j=1}^J \lambda_j (C_j - \bar{C}) )</td>
</tr>
<tr>
<td>( 0 \leq G_j \leq 1 )</td>
<td>( 0 \leq G_i \leq 1 )</td>
<td></td>
</tr>
<tr>
<td>where: ( R_i = \frac{e_{i}/e_j}{e_{ij}/e_i} ), ( \bar{C} = \frac{\sum_{j=1}^J C_j}{J} )</td>
<td>where: ( C_j = \frac{e_{j}/e_i}{e_{ij}/e_i} ), ( \bar{C} = \frac{\sum_{j=1}^J C_j}{J} )</td>
<td></td>
</tr>
<tr>
<td>( \lambda_i = 1, ..., I ) (( \lambda_i &lt; \lambda_{i+1} ))</td>
<td>( \lambda_j = 1, ..., J ) (( \lambda_j &lt; \lambda_{j+1} ))</td>
<td></td>
</tr>
<tr>
<td>Krugman</td>
<td>( K_{jl} = \sum_{i=1}^I</td>
<td>s_{ij}^l - s_{ij}^l</td>
</tr>
<tr>
<td>( J = 2, I = 2 )</td>
<td>( 0 \leq K_{jl} \leq 2 )</td>
<td>( 0 \leq K_{iu} \leq 2 )</td>
</tr>
<tr>
<td>where: ( s_{ij}^l = \frac{e_{ij}}{e_j} ) and ( s_{ij}^u = \frac{e_{ij}}{e_i} )</td>
<td>where: ( s_{ij}^l = \frac{e_{ij}}{e_j} ) and ( s_{ij}^u = \frac{e_{ij}}{e_i} )</td>
<td></td>
</tr>
<tr>
<td>Extensions:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Midelfart et al., Vogiatzoglou</td>
<td>( K_j = \sum_{i=1}^I</td>
<td>s_{ij}^l - s_{ij}^l</td>
</tr>
<tr>
<td>( J &gt; 2, I &gt; 2 )</td>
<td>( 0 \leq K_j \leq 2 )</td>
<td>( 0 \leq K_i \leq 2 )</td>
</tr>
<tr>
<td>where: ( s_{ij}^l = \frac{e_{ij}}{e_j} ) and ( s_{ij}^u = \frac{e_{ij}}{e_i} )</td>
<td>where: ( s_{ij}^l = \frac{e_{ij}}{e_j} ) and ( s_{ij}^u = \frac{e_{ij}}{e_i} )</td>
<td></td>
</tr>
<tr>
<td>( s_{ij}^l = \frac{1}{I} \sum_{l \neq j} s_{ij}^l )</td>
<td>( s_{ij}^u = \frac{1}{J} \sum_{u \neq i} s_{ij}^u )</td>
<td></td>
</tr>
<tr>
<td>Duranton-Puga</td>
<td>( RDI_j = \frac{1}{\sum_{l=1}^I</td>
<td>s_{ij}^l - s_{ij}^l</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Litzenberger-Sternberg</td>
<td>( CI_{ij} = \frac{IS_{ij}</td>
<td>DI_{ij}</td>
</tr>
<tr>
<td>where ( IS_{ij} = \frac{e_{ij}/e_j}{e_{ij}/e_i}, ID_{ij} = \frac{e_{ij}/p_j}{e_{ij}/p_i} )</td>
<td>and ( PS_{ij} = \frac{e_{ij}/b_j}{e_{ij}/b_i} )</td>
<td></td>
</tr>
</tbody>
</table>

Notes: \( e_{ij} \) and \( e_{il} \) equal the employment of industry \( i \) in regions \( j \) and \( l \), respectively, \( e_i \) is the total employment in industry \( i \), \( e_{ij} \) ist the employment of industry \( u \) in region \( j \), \( e_j \) is the total employment in region \( j \), \( e \) is the total employment in the whole economy, \( I \) is the number of industries, \( J \) is the number of regions, \( a_j \) is the area of region \( j \), \( a \) is the total area in the whole economy, \( p_j \) is the population in region \( j \), \( p \) is the total population, \( b_i \) is the number of firms in industry \( i \) in region \( j \) and \( b_i \) is the number of firms in industry \( i \).

Compiled from: Farhauer, Kröll (2014); Hoen, Oosterhaven (2006); Hoffmann et al. (2017); Nakamura, Morrison Paul (2009); O’Donoghue, Gleave (2004); Tian (2013); Schätzl (2000); Störmann (2009)

The empirical basis of all those measures is the employment \( e \) in industry \( i \) in region \( j \), \( e_{ij} \). This employment stock is compared to some reference, mostly including the total employment in region \( j \), \( e_j \), and/or the total employment in industry \( i \), \( e_i \), as well as the all-over employment \( e \). The individual firm level indicators in Table 6 can be segmented into indicators for agglomeration of one industry due to local economic conditions and indicators for the coagglomeration of different industries due to urbanization economies.
Table 6: Coefficients of agglomeration and coagglomeration using individual firm data

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Agglomeration</th>
<th>Coagglomeration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ellison-Glaeser</td>
<td>$\gamma_i = \frac{G_i - (1 - \sum_{j=1}^{J} s_{ij}^2) \cdot HHI_i}{(1 - \sum_{j=1}^{J} s_{ij}^2) \cdot (1 - HHI_i)}$</td>
<td>$\gamma^c = \frac{G^c(1 - \sum_{j=1}^{J} s_{ij}^2) \cdot HHI_i - \sum_{k=1}^{K} \gamma_k s_k^2 (1 - HHI_i)}{1 - \sum_{j=1}^{J} s_{ij}^2}$</td>
</tr>
</tbody>
</table>

where: $G_i = \sum_{j=1}^{J} (s_{ij} - s_j)^2, s_{ij} = \frac{s_{ij}}{e}$ and $HHI_i = \frac{\sum_{k=1}^{K} (s_{ik}^2 e^k)}{\sum_{k=1}^{K} e^k}$

$z$-standardization:

$$z_i = \frac{G_i - (1 - \sum_{j=1}^{J} s_{ij}^2) \cdot HHI_i}{\sqrt{\text{var}(G_i)}}$$

where: $\text{var}(G_i) = 2 \left\{ HHI_i^2 \left( \sum_{j=1}^{J} s_j^2 \right) - 2 \sum_{j=1}^{J} s_j^3 + \left( \sum_{j=1}^{J} s_j^2 \right)^2 \right\} - \sum_{k=1}^{K} s_k^4 \left( \sum_{j=1}^{J} s_j^3 - 4 \sum_{j=1}^{J} s_j^2 \right) + 3 \left( \sum_{j=1}^{J} s_j^2 s_k^2 \right)^2 \right\}$

Howard et al. |
| $CL_{ab} = \sum_{k=1}^{K_a} \sum_{b=1}^{K_b} C_{kl} \frac{U_{ab}}{K_a K_b}$ | $XCL_{ab} = CL_{ab} - CL_{RN}^{ab}$ |

where: $C_{kl} = 1$ if firms $k$ and $l$ are located in the same region and $C_{kl} = 0$ otherwise.

Notes: $e_{ij}$ is the employment of industry $i$ in region $j$, $e_i$ is the total employment in industry $i$, $e_j$ is the total employment in region $j$, $e$ is the total employment in the whole economy, $e_{ik}$ is the employment of firm $k$ from industry $i$, $k$ and $l$ are indices for single firms, $I$ is the number of industries, $J$ is the number of regions, $U$ is a subset of all $I$ industries ($U \subseteq I$), $K$ is the number of firms and $K_a$ and $K_b$ are the numbers of firms in industry $a$ and $b$.

Compiled from: Farhauer, Kröll (2014); Howard et al. (2016); Nakamura, Morrison Paul (2009)

The most popular indicator is the Location Quotient (LQ), which is attributed to Hoover (1936) and mathematically equivalent to the Revealed Comparative Advantage (RCA) index, developed by Balassa (1965) in the context of international trade. The LQ is utilized in many studies (e.g. Bai et al. 2008; Kim 1995) as well as in the OECD Territorial Reviews (OECD, 2019). Following O’Donoghue, Gleave (2004) and Tian (2013), the original formulation can be extended: As the location quotient is not normalized, there is no cut-off value for defining a cluster, which leads to a standardization of the computed values via $z$-transformation. Hoen, Oosterhaven (2006) developed an additive alternative to the RCA index. The original LQ provides the main mathematical basis for several indicators developed later, such as the spatial Gini coefficients described below.

Some indicators which are known from the context of regional inequality (see Section 2) are also used for the analysis of agglomeration: A modification of the Gini coefficient is used for the spatial concentration of industries as well as regional specialization (e.g. Ceapraz 2008; Wieland, Fuchs 2018). As we can see in the calculation of $R_k$ and $C_{ij}$, respectively, the spatial Gini coefficient is based on the LQ. Another popular option for analyzing agglomeration is the Hoover coefficient, comparing the structure of an industry/a region to a reference structure of all industries/regions (e.g. Dixon, Freebairn 2009; Jiang et al. 2007). Both indicator types range between zero (no specialization/concentration) and one (total specialization/concentration). Also the Herfindahl-Hirschman index and its derivates are used to measure concentration, specialization and diversification (e.g. Duranton, Puga 2000; Goschin et al. 2009; Lehocký, Rusnák 2016).

Another type of specialization/concentration indicator was introduced by Krugman (1991), originally designed for comparing the specialization of two regions. An extension of this indicator was established by Midelfart-Knarvik et al. (2000) for the comparison of regional specialization/industry concentration with respect to the sum or mean of all
regions/industries (furthermore used e.g. by Haas, Südekum 2005; Vogiatzoglou 2006). Unlike the Gini- or Hoover-type measures, the Krugman coefficients range between zero (no specialization/concentration) and two (total specialization/concentration).

The cluster index developed by Litzenberger, Sternberg (2006) goes beyond employment data and includes additional information about the industry-specific firm size, population density and region size. It is composed of three parts: the relative industrial stock with respect to industry \( i \) and region \( j \), \( IS_{ij} \), the relative industrial density, \( ID_{ij} \), and the relative firm size, \( PS_{ij} \). All three components are modified location quotients. This is done to control for small and monostructural regions, which are identified as clusters otherwise (which is a problem in the original \( LQ \)). The cluster index \( CI_{ij} \) has a potential range from zero to infinity. This extended indicator is used e.g. by Hoffmann et al. (2017) for the German food processing industry.

The cluster indicators by Ellison, Glaeser (1997) compare the empirical distribution of firms to an arbitrary location pattern where agglomeration economies are absent (often referred to as a dartboard approach). Ellison, Glaeser (1997) differentiate between the clustering of firms from one industry (agglomeration) due to localization economies and the clustering of multiple industries (coagglomeration) due to urbanization economies. Their indices also take into account the industry-specific structure of the firms by including the Herfindahl-Hirschman index, \( HHI_i \), for the employment concentration in industry \( i \). This is the reason why individual firm-level data is required for the computation. The Herfindahl-Hirschman indicator is included to control the raw measures of spatial concentration, \( G_i \) and \( G \), for firm employment concentration, which occurs especially when there are just a few firms with many employees. The Ellison-Glaeser (\( EG \)) index for agglomeration, \( \gamma_i \), is designed for identifying the clustering of industry \( i \), while the coagglomeration index, \( \gamma_c \), aims at the clustering of a set of \( U \) industries, where \( U \leq I \). Values of \( \gamma \) equal to zero imply the absence of agglomeration economies, while values above zero indicate positive effects due to spatial clustering. When \( \gamma \) is negative, firm locations are less spatially concentrated than expected on condition of the dartboard approach, which indicates negative agglomeration economies. The \( EG \) index is used in several current regional economic studies (e.g. Dauth et al. 2015, 2018; Yamamura, Goto 2018).

In contrast, Howard et al. (2016) argue that agglomeration economies should not be analyzed regarding employment but the firms itself. Their colocation index, \( CL_{ab} \), sums the colocation of \( K_i \) and \( K_q \) firms from two industries, \( i \) and \( q \), controlling for all possible combinations. This colocation measure is compared to a counterfactual location structure constructed via bootstrapping: specifically the arithmetic mean of a number of (e.g. 50) random assignments of the regarded firms to the locations. The value of the resulting excess colocation index, \( XCL_{ab} \), ranges between -1 and 1.

4.2 Application in REAT

4.2.1 REAT functions for regional specialization and industry concentration

Table 7 shows the REAT functions for agglomeration measures based on aggregate (employment) data. All functions require at least information about the employment in one or more regions \( j \) in one or more industries \( i \), \( e_{ij} \). The Herfindahl-Hirschman index (function \( herf() \)) for measuring regional diversity is not displayed as it is used exactly in the same way as described in Section 2, replacing \( x_i \) with \( e_{ij} \).

Location quotients for one region and one or more industries are computed by the function \( locq() \), including the option for an additive indicator instead of the multiplicative. When calculating the LQ for a set of \( J \) regions and \( I \) industries, one can use function \( locq2() \), which is a kind of batch processing extension of \( locq() \). As the dimension of the Litzenberger-Sternberg cluster index is the same as in the \( LQ \) (a single value for each combination of region \( j \) and industry \( i \)), the related functions \( litzenberger() \) and \( litzenberger2() \) work in the same way. When using \( locq2() \) or \( litzenberger2() \), the user may choose the type of function output: either a matrix with \( I \) columns and \( J \) rows or a data frame with \( I * J \) rows.

The Hoover-, Gini- and Krugman-type indicators require the same kind of input data. The \( hoover() \) function was already explained in Section 2, as it can be also
Table 7: REAT functions for regional specialization and industry concentration

<table>
<thead>
<tr>
<th>Indicator</th>
<th>REAT function</th>
<th>Mandatory arguments</th>
<th>Optional arguments</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hoover LQ/ Balassa RCA incl. extensions</td>
<td>locq()</td>
<td>vectors or single values of $e_{ij}$ and $e_i$, single values of $e_j$ and $e$</td>
<td>LQ method, plot</td>
<td>Single value or matrix with $LQ_{ij}$</td>
</tr>
<tr>
<td></td>
<td>locq2()</td>
<td>vectors of $e_{ij}$, industry ID $i$ and region ID $j$</td>
<td>normalization, output type, remove NAs</td>
<td>matrix or data frame with $I * J$ values of $LQ_{ij}$</td>
</tr>
<tr>
<td>Hoover specialization/ concentration</td>
<td>hoover()</td>
<td>vectors of $e_{ij}$ and $e_j$ and $e_i$</td>
<td>remove NAs</td>
<td>value: $H_j$ or $H_i$</td>
</tr>
<tr>
<td>Gini specialization</td>
<td>gini.spec()</td>
<td>vectors $e_{ij}$ and $e_i$</td>
<td>plot LC</td>
<td>value: $G_j$, optional: LC plot</td>
</tr>
<tr>
<td>concentration</td>
<td>gini.conc()</td>
<td>vectors $e_{ij}$ and $e_j$</td>
<td>plot LC</td>
<td>value: $G_i$, optional: LC plot</td>
</tr>
<tr>
<td>Krugman specialization</td>
<td>krugman.spec()</td>
<td>vectors $e_{ij}$ (regions $j$ and $l$) and $e_{ij}$</td>
<td>value: $K_{jl}$</td>
<td></td>
</tr>
<tr>
<td>concentration</td>
<td>krugman.conc2()</td>
<td>vector $e_{ij}$ and matrix or data frame $e_{il}$</td>
<td>value: $K_{ij}$</td>
<td></td>
</tr>
<tr>
<td>All at once: specialization</td>
<td>spec()</td>
<td>vectors of $e_{ij}$, industry ID $i$ and region ID $j$</td>
<td>remove NAs</td>
<td>matrix with $H_j$, $G_j$ and $K_{ij}$ (columns) for $J$ regions (rows)</td>
</tr>
<tr>
<td>concentration</td>
<td>conc()</td>
<td>vectors of $e_{ij}$, industry ID $i$ and region ID $j$</td>
<td>remove NAs</td>
<td>matrix with $H_i$, $G_i$ and $K_i$ (columns) for $I$ industries (rows)</td>
</tr>
<tr>
<td>Duranton-Puga</td>
<td>durpug()</td>
<td>vectors $e_{ij}$ and $e_i$</td>
<td>value: $RDI_j$</td>
<td></td>
</tr>
<tr>
<td>Litzenberger-Sternberg</td>
<td>litzenberger()</td>
<td>single values of $e_{ij}$, $e_i$, $a_j$, $a_i$, $p_j$, $p_i$, $b_{ij}$ and $b_i$</td>
<td>value: $CI_{ij}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>litzenberger2()</td>
<td>vectors of $e_{ij}$, industry ID $i$, region ID $j$, $a_j$, $p_j$ and $b_{ij}$</td>
<td>output type, remove NAs</td>
<td>matrix or data frame with $I * J$ values of $CI_{ij}$</td>
</tr>
</tbody>
</table>

Source: own compilation.

used for measuring spatial concentration of industries or the specialization of regions with all-over employment vectors, $e_i$ and $e_j$, respectively, as reference distributions. The spatial Gini coefficients are available through functions `gini.spec()` for regional specialization and `gini.conc()` for spatial concentration. The Krugman coefficients are divided into functions for the comparison of two regions/industries (`krugman.spec()` and `krugman.conc()`, respectively) and for applying all regions/industries as reference (`krugman.spec2()` and `krugman.conc2()`, respectively). The functions `spec()` and `conc()` are wrapper functions providing a convenient way to compute Hoover, Gini and Krugman coefficients of a given set of $J$ regions and $I$ industries at once, e.g. originating from official statistics on regional employment.

Table 8 shows the functions operating on the level of individual firm data. The Ellison-Glaeser (EG) indices are available through the functions `ellison.a()` (agglomeration index for industry $i$) and `ellison.a2()` (agglomeration indices for $I$ industries) as well as `ellison.c()` (coagglomeration index for $U$ industries) and `ellison.c2()` (co-agglomeration indices for $I * I - I$ industry combinations). All functions require the firm size (e.g. no. of employees) for the $k$-th firm from industry $i$ (`numeric vector`) and the
Table 8: REAT functions for agglomeration and coagglomeration using firm data

<table>
<thead>
<tr>
<th>Indicator</th>
<th>REAT function</th>
<th>Mandatory arguments</th>
<th>Optional arguments</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ellison-Glaeser agglomeration</td>
<td>ellison.a()</td>
<td>vectors of $e_{ik}$, $e_j$ and region ID $j$</td>
<td>visible: value $\gamma_i$; invisible: matrix with $\gamma_i$, $G_i$, $z_i$, $K_i$ and $HHI_i$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ellison.a2()</td>
<td>vectors $e_{ik}$, industry ID $i$ and region ID $j$</td>
<td>visible: values $\gamma_i$; invisible: matrix with $\gamma_i$, $G_i$, $z_i$, $K_i$ and $HHI_i$, for $I$ industries (rows)</td>
<td></td>
</tr>
<tr>
<td>Coagglomeration</td>
<td>ellison.c()</td>
<td>vectors $e_{ik}$, industry ID $i$ and region ID $j$</td>
<td>vectors $e_j$ and $U$ industries</td>
<td>value: $\gamma^c$</td>
</tr>
<tr>
<td></td>
<td>ellison.c2()</td>
<td>vectors $e_{ik}$, industry ID $i$ and region ID $j$</td>
<td>vector $e_j$</td>
<td>matrix with $\gamma^c$ for $I \ast I - I$ industry combinations (rows)</td>
</tr>
<tr>
<td>Howard et al. colocation</td>
<td>howard.cl()</td>
<td>firm ID $k$, industry ID $i$, and region ID $j$, industries $a$ and $b$</td>
<td></td>
<td>value: $CL_{ab}$</td>
</tr>
<tr>
<td>Excess colocation</td>
<td>howard.xcl()</td>
<td>firm ID $k$, industry ID $i$, and region ID $j$, industries $a$ and $b$, no. of samples</td>
<td></td>
<td>value: $XCL_{ab}$</td>
</tr>
<tr>
<td></td>
<td>howard.xcl2()</td>
<td>firm ID $k$, industry ID $i$ and region ID $j$</td>
<td></td>
<td>matrix with $XCL_{ab}$ for $I \ast I - I$ industry combinations (rows)</td>
</tr>
</tbody>
</table>

Source: own compilation.

region $j$ the firm is located in. The functions incorporating more than one industry (all except for ellison.a()) require a vector containing the industry $i$. The data could e.g. be stored in a data frame with at least three columns (firm size, region, industry). Like some of the convergence functions (see Section 3), the EG agglomeration index functions in REAT also distinguish between a visible and an invisible output: ellison.a() and ellison.a2() show the value(s) auf $\gamma_i$ but return an invisible matrix including the raw measure of concentration ($G_i$), the $z$-standardized results ($z_i$) and the related Herfindahl-Hirschman index for industry-specific firm concentration ($HHI_i$) as well as the number of firms in industry $i$ ($K_i$).

The Howard-Newman-Tarp coagglomeration measure is distributed over the functions howard.cl() (calculation of the colocation index for one pair of industries $a$ and $b$), howard.xcl() (calculation of the excess colocation index for industries $a$ and $b$) and howard.xcl2() (calculation of the excess colocation index for $I \ast I - I$ combinations of $I$ industries). As this cluster index works with firms instead of employment, we only need a vector containing the IDs of the firms $k$, the corresponding industry $i$ and the region $j$ where the firm is located. When calculating this measure for one pair of industries, the user must state the IDs of industries $a$ and $b$. Note that calculation time for this index increases heavily with the number of firms and/or industries.

4.2.2 Application example 1: Regional specialization of Göttingen

We use the German classification of economic activities (WZ2008) on the level of 21 sections (A-U) for the classification of industries in the following examples (see Table 9).

Starting with a simple example, we analyze the regional specialization of Göttingen, a city with a population of about 134,000 in Niedersachsen, Germany. The example dataset Goettingen, which is included in REAT, contains the dependent employees in Göttingen and Germany for 2008 to 2017 in industries A to R (rows 2 to 16; row 1 contains the all-over employment). First, we load the data:
Table 9: Classification of economic activities in Germany, edition 2008 (WZ 2008)

<table>
<thead>
<tr>
<th>WZ2008 Code</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Agriculture, forestry and fishing</td>
</tr>
<tr>
<td>B</td>
<td>Mining and quarrying</td>
</tr>
<tr>
<td>C</td>
<td>Manufacturing</td>
</tr>
<tr>
<td>D</td>
<td>Electricity, gas, steam and air conditioning supply</td>
</tr>
<tr>
<td>E</td>
<td>Water supply; sewerage, waste management and remediation activities</td>
</tr>
<tr>
<td>F</td>
<td>Construction</td>
</tr>
<tr>
<td>G</td>
<td>Wholesale and retail trade; repair of motor vehicles and motorcycles</td>
</tr>
<tr>
<td>H</td>
<td>Transportation and storage</td>
</tr>
<tr>
<td>I</td>
<td>Accommodation and food service activities</td>
</tr>
<tr>
<td>J</td>
<td>Information and communication</td>
</tr>
<tr>
<td>K</td>
<td>Financial and insurance activities</td>
</tr>
<tr>
<td>L</td>
<td>Real estate activities</td>
</tr>
<tr>
<td>M</td>
<td>Professional, scientific and technical activities</td>
</tr>
<tr>
<td>N</td>
<td>Administrative and support service activities</td>
</tr>
<tr>
<td>O</td>
<td>Public administration and defence; compulsory social security</td>
</tr>
<tr>
<td>P</td>
<td>Education</td>
</tr>
<tr>
<td>Q</td>
<td>Human health and social work activities</td>
</tr>
<tr>
<td>R</td>
<td>Arts, entertainment and recreation</td>
</tr>
<tr>
<td>S</td>
<td>Other service activities</td>
</tr>
<tr>
<td>T</td>
<td>Activities of households as employers; undifferentiated goods-and services-producing activities of households for own use</td>
</tr>
<tr>
<td>U</td>
<td>Activities of extraterritorial organisations and bodies</td>
</tr>
</tbody>
</table>

Source: own compilation based on Statistisches Bundesamt (2008).

data(Goettingen)

Using the REAT function `locq()`, we calculate a location quotient for Göttingen with respect to the manufacturing industry ("Verarbeitendes Gewerbe"), which is represented by letter C:

```r
locq(Goettingen$Goettingen2017[4], Goettingen$Goettingen2017[1],
     Goettingen$BRD2017[4], Goettingen$BRD2017[1])
# Industry: manufacturing (letter C) in row 4
# row 1 = all-over employment

[1] 0.5369
```

The output is simply the LQ value ($LQ_{ij}$, where $i$ is manufacturing and $j$ is Göttingen). We see that the LQ is very low, indicating that manufacturing is underrepresented in Göttingen as compared to Germany. Now, we calculate LQ values for all industries (A-R), including a simple plot (function argument `plot.results = TRUE`):

```r
locq(Goettingen$Goettingen2017[2:16], Goettingen$Goettingen2017[1],
     Goettingen$BRD2017[2:16], Goettingen$BRD2017[1],
     industry.names = Goettingen$WZ2008_Code[2:16], plot.results = TRUE,
     plot.title = "Location quotients for Göttingen 2017")
# all industries (rows 2-16 in the dataset)
```

The output is a matrix with one row for each industry:

```
Location quotients
I = 15 industries

LQ
A  0.08407652
BDE 0.40085663
C  0.53687366
F  0.34366928
G  0.74603541
H  0.67117311
```

`REGION`: Volume 6, Number 3, 2019
Figure 3: Location quotients for Göttingen

I 0.98141916
J 0.91654277
K 0.82650178
M 1.53027645
N 0.95843423
O 1.03509027
Q 2.77790858
R 0.35317012

The result is plotted in Figure 3. The function plots a vertical line at $LQ_{ij} = 1$ automatically. This is the (only) reference value for the LQ. It indicates a stock of the related industry equal to the whole economy. The highest LQ values can be found for the industries with letters P (education) and Q (health). This is because Göttingen is mainly characterized by a large university (about 30,000 students) with a university hospital with about 7,000 employees.

Now, we want to measure the specialization of Göttingen with a single indicator. First, we simply use the Herfindahl-Hirschman coefficient for both Göttingen and Germany using the function `herf()`:

\[
\text{herf(Goettingen$Goettingen2017[2:16])} \\
\text{herf(Goettingen$BRD2017[2:16])}
\]

\[0.127314 \quad 0.1104567\]

The HHI for Göttingen is slightly larger than for Germany, which indicates a higher specialization (or lower economic diversity) of the region. To combine this information in one indicator, we calculate the Hoover coefficient of specialization using the function `hoover()` where the reference distribution is the German industry structure:

\[
\text{hoover(Goettingen$Goettingen2017[2:16], ref = Goettingen$BRD2017[2:16])}
\]

\[0.2254234\]

We finish our analysis of Göttingen’s regional specialization by calculating both the Gini and the Krugman coefficient of regional specialization with the same data, using the REAT functions `gini.spec()` and `krugman.spec()`, respectively. Note that, here,
we use the Krugman coefficient to compare the industry structure of Göttingen to the structure of whole Germany (instead of another region within the country, for which this coefficient was originally formulated):

\[
\text{gini.spec}(\text{Goettingen$Goettingen2017}[2:16], \text{Goettingen$BRD2017}[2:16])
\]

\[
\begin{bmatrix}
0.359852
\end{bmatrix}
\]

\[
\text{krugman.spec}(\text{Goettingen$Goettingen2017}[2:16], \text{Goettingen$BRD2017}[2:16])
\]

\[
\begin{bmatrix}
0.4508469
\end{bmatrix}
\]

There seems to be some specialization in Göttingen, but, unfortunately, we do not have any real reference value to interpret the results.

4.2.3 Application example 2: Identifying clusters in Germany using aggregate data

In this example, we will compute indicators of regional specialization and industry concentration for a set of \(J\) regions and \(I\) industries at once. We load the included test dataset `G.regions.industries` containing employment and firms on the level of \(I = 17\) industries (WZ2008 codes B-S) and \(J = 16\) regions (“Bundesländer”) in Germany:

\[
data(G \text{.regions.industries})
\]

The number of employees in the column `emp_all` includes dependent employees and self-employed persons. The classification code of industries (see Table 9) can be found in column `ind_code`, while the region code (abbreviation of the region’s official name) is in column `region_code`. First, we want to detect the spatial concentration of the 17 industries in Germany by calculating Hoover, Gini and Krugman coefficients for all industries at once, applying the `REAT` function `conc()` which is a wrapper function for the mentioned indicators. We save our output in the matrix object `conc_i`:

\[
\text{conc}_i \leftarrow \text{conc} (\text{e}_{ij} = G \text{.regions.industries$emp_all, ind}_{\text{industry.id}} = G \text{.regions.industries$ind_code, region}_{\text{region.id}} = G \text{.regions.industries$region_code})
\]

The output is:

**Spatial concentration of industries**

\(I = 17\) industries, \(J = 16\) regions

\[
\begin{array}{ccc}
\text{H}_i & \text{G}_i & \text{K}_i \\
\text{WZ08-B} & 0.22969050 & 0.42334831 & 0.45675385 \\
\text{WZ08-C} & 0.09933363 & 0.17047620 & 0.26813759 \\
\text{WZ08-D} & 0.07754576 & 0.12509360 & 0.16260016 \\
\text{WZ08-E} & 0.11972072 & 0.16742909 & 0.20369011 \\
\text{WZ08-F} & 0.07676634 & 0.15357575 & 0.16996098 \\
\text{WZ08-G} & 0.03034962 & 0.05471323 & 0.07977056 \\
\text{WZ08-H} & 0.06006957 & 0.11921850 & 0.10076748 \\
\text{WZ08-I} & 0.05177262 & 0.09939075 & 0.11450791 \\
\text{WZ08-J} & 0.10230712 & 0.22605802 & 0.24450967 \\
\text{WZ08-K} & 0.08998271 & 0.17610712 & 0.20565974 \\
\text{WZ08-L} & 0.09798632 & 0.16784764 & 0.17472656 \\
\text{WZ08-M} & 0.06490185 & 0.14760918 & 0.14931991 \\
\text{WZ08-N} & 0.06714816 & 0.08575299 & 0.09053327 \\
\text{WZ08-P} & 0.03019678 & 0.05053848 & 0.07043586 \\
\text{WZ08-Q} & 0.04679962 & 0.06170335 & 0.06406058 \\
\text{WZ08-R} & 0.09424708 & 0.16748405 & 0.17023603 \\
\text{WZ08-S} & 0.04507988 & 0.07246697 & 0.06441360 \\
\end{array}
\]

The function returns a matrix with 17 rows (one for each industry) and three columns: \(H_i\) is the Hoover coefficient, \(G_i\) is the Gini coefficient and \(K_i\) is the Krugman coefficient for industry \(i\). We cannot interpret or compare all of these results, but we may pick out some findings: The strongest spatial concentration is found with respect to mining
and quarrying (WZ08-B), no matter which indicator is regarded, which may be interpreted with “natural advantages” due to the spatial distribution of mineral resources in Germany. Services (such as retailing) as well as education and health are least concentrated, as these industries are bound to regional demand and/or their locations are regulated by policy and planning authorities.

At a first glance, the three indicators seem to produce similar results. Now, we want to test the similarity between Hoover, Gini and Krugman coefficients of concentration. As we saved our result matrix, we now calculate Pearson correlation coefficients \( r \) for each pair of indicators using the basic \( R \) function \( \text{cor()} \), which is implemented in the \( \text{stats} \) package (included automatically in any \( R \) release). The function is applied to the three columns of \( \text{conc}_i \), producing a 3 \( \times \) 3 correlation matrix:

\[
\text{cor(conc\_i[,1:3])}
\]

\[
\begin{array}{ccc}
H & G & K \\
H & 1.0000000 & 0.9676518 & 0.9527747 \\
G & 0.9676518 & 1.0000000 & 0.9681770 \\
K & 0.9527747 & 0.9681770 & 1.0000000
\end{array}
\]

As we can see, each combination of the three indicators shows a strong positive correlation \( (H_i \text{ vs. } G_i: r \approx 0.97, H_i \text{ vs. } K_i: r \approx 0.95, G_i \text{ vs. } K_i: r \approx 0.97) \). At least in this context, we may conclude that these indicators are interchangeable. However, we have to recognize that the analysis presented here is on a large-scale regional level (German “Bundesländer”) and all of the mentioned indicators are affected by the \textit{modifiable areal unit problem}, which means that the results depend on the aggregation unit in the analysis (see e.g. Dapena et al. 2016 for a discussion of this effect).

Now, we do exactly the same with respect to regional specialization of the 16 regions, using the same data. Analogously, we use the wrapper function \( \text{spec()} \) for calculating Hoover, Gini and Krugman coefficients of regional specialization, also saving the resulting matrix:

\[
\text{spec\_j <- spec (e\_ij = G\_regions\_industries\$emp\_all,}
\text{industry\_id = G\_regions\_industries\$ind\_code,}
\text{region\_id = G\_regions\_industries\$region\_code)}
\]

The output is:

Specialization of regions
\( I = 17 \) industries, \( J = 16 \) regions

\[
\begin{array}{ccc}
H & G & K \\
BB & 0.11530353 & 0.20632682 & 0.18555259 \\
BE & 0.17891265 & 0.29040841 & 0.3452331 \\
BW & 0.08024011 & 0.10300695 & 0.22675612 \\
BY & 0.05008135 & 0.07659148 & 0.16019603 \\
HB & 0.09502615 & 0.18563500 & 0.17467291 \\
HE & 0.05494422 & 0.12160142 & 0.11282696 \\
HH & 0.16413456 & 0.22616814 & 0.33190321 \\
MV & 0.13270849 & 0.18974606 & 0.22058688 \\
NI & 0.03772799 & 0.08237225 & 0.07972852 \\
NW & 0.02940091 & 0.05997505 & 0.07181569 \\
RP & 0.04793147 & 0.07432361 & 0.12036513 \\
SH & 0.08901907 & 0.11384295 & 0.15994524 \\
SL & 0.05726933 & 0.11921727 & 0.15071159 \\
SN & 0.05400855 & 0.10643512 & 0.10341280 \\
ST & 0.08821395 & 0.21120287 & 0.15280711 \\
TH & 0.08234046 & 0.13902924 & 0.17720208
\end{array}
\]

The strongest specialization can be found in the city states Berlin (BE) and Hamburg (HH), while Niedersachsen (NI) and Nordrhein-Westfalen (NW) show the lowest values in all three indicators. As already mentioned in the concentration example, we have to remember the large-scale aggregation unit. If we used smaller scale units (e.g. counties like in Section 3.2.2), our results would surely be more differentiated. Again, we check the correlation between the indicators:
Again, we find a strong positive correlation between the Hoover coefficient and both Gini and Krugman coefficient ($H_j$ vs. $G_j$: $r \approx 0.92$, $H_j$ vs. $K_j$: $r \approx 0.93$), while the third Pearson correlation coefficient is a little lower, but still showing the same direction ($G_j$ vs. $K_j$: $r \approx 0.79$).

Now we check for clusters in a combination of a specific industry and a specific region. First, we calculate location quotients for the dataset `G.regions.industries` using the `REAT` function `locq2()`. Here, the optional function argument `LQ.norm` could be used for computing $z$-standardized location quotients according to O’Donoghue, Gleave (2004) ($LQ.norm = "OG"$) or $z$-standardized values of the natural-logged LQs according to Tian (2013) ($LQ.norm = "T"$). However, we produce the original LQs, since we need exactly the same columns as in the examples above:

```
locq2(e_ij = G.regions.industries$emp_all, G.regions.industries$ind_code, G.regions.industries$region_code)
```

The output is a matrix with $J$ rows and $I$ columns:

<table>
<thead>
<tr>
<th>REGION</th>
<th>BB</th>
<th>BE</th>
<th>BW</th>
<th>BY</th>
<th>HB</th>
<th>HE</th>
</tr>
</thead>
<tbody>
<tr>
<td>WZ08-B</td>
<td>2.5314363</td>
<td>0.04030901</td>
<td>0.6607950</td>
<td>0.8078054</td>
<td>0.0000000</td>
<td>0.3735773</td>
</tr>
<tr>
<td>WZ08-C</td>
<td>0.6857231</td>
<td>0.37224900</td>
<td>1.3968652</td>
<td>1.1902785</td>
<td>0.7863570</td>
<td>0.8580352</td>
</tr>
<tr>
<td>WZ08-D</td>
<td>1.1736475</td>
<td>0.46721079</td>
<td>1.0861988</td>
<td>0.8343784</td>
<td>0.9179718</td>
<td>0.9627955</td>
</tr>
<tr>
<td>WZ08-E</td>
<td>1.7945685</td>
<td>1.30128835</td>
<td>0.5896526</td>
<td>0.7137388</td>
<td>1.2393228</td>
<td>0.8532203</td>
</tr>
<tr>
<td>WZ08-F</td>
<td>1.5997778</td>
<td>0.77160172</td>
<td>1.3968652</td>
<td>1.1902785</td>
<td>0.7863570</td>
<td>0.8580352</td>
</tr>
<tr>
<td>WZ08-G</td>
<td>0.9550127</td>
<td>0.83133221</td>
<td>0.9492523</td>
<td>0.9879826</td>
<td>0.9013193</td>
<td>1.0006321</td>
</tr>
<tr>
<td>WZ08-H</td>
<td>1.3212794</td>
<td>0.87982228</td>
<td>0.819666</td>
<td>0.8664163</td>
<td>1.7692815</td>
<td>1.2208728</td>
</tr>
<tr>
<td>WZ08-I</td>
<td>1.0379426</td>
<td>1.35561299</td>
<td>0.9132949</td>
<td>1.0390886</td>
<td>0.9904308</td>
<td>0.9571339</td>
</tr>
<tr>
<td>WZ08-J</td>
<td>0.5625876</td>
<td>0.62357960</td>
<td>1.4987086</td>
<td>1.4595767</td>
<td>1.0371236</td>
<td>0.5078145</td>
</tr>
<tr>
<td>WZ08-K</td>
<td>0.4781934</td>
<td>0.62357960</td>
<td>1.4987086</td>
<td>1.4595767</td>
<td>1.0371236</td>
<td>0.5078145</td>
</tr>
<tr>
<td>WZ08-L</td>
<td>0.4332870</td>
<td>0.1188846</td>
<td>0.7310014</td>
<td>0.8633894</td>
<td>1.3245723</td>
<td>1.1132939</td>
</tr>
<tr>
<td>WZ08-M</td>
<td>0.7366238</td>
<td>0.36720820</td>
<td>0.0337193</td>
<td>0.1265993</td>
<td>1.1202532</td>
<td>1.1455770</td>
</tr>
<tr>
<td>WZ08-N</td>
<td>1.2571301</td>
<td>1.2461162</td>
<td>0.7977054</td>
<td>0.8486971</td>
<td>1.2938161</td>
<td>1.052912</td>
</tr>
<tr>
<td>WZ08-P</td>
<td>0.9052976</td>
<td>1.3842157</td>
<td>0.9649298</td>
<td>0.9252245</td>
<td>1.0631692</td>
<td>1.0085207</td>
</tr>
<tr>
<td>WZ08-Q</td>
<td>1.1540423</td>
<td>0.2939920</td>
<td>0.869241</td>
<td>0.9079679</td>
<td>0.9544891</td>
<td>0.8980680</td>
</tr>
<tr>
<td>WZ08-R</td>
<td>1.0656945</td>
<td>2.5559102</td>
<td>0.8151892</td>
<td>0.8220540</td>
<td>1.3196613</td>
<td>0.8651451</td>
</tr>
<tr>
<td>WZ08-S</td>
<td>1.1409373</td>
<td>1.32596177</td>
<td>0.8626829</td>
<td>0.9092125</td>
<td>1.1396616</td>
<td>1.0528184</td>
</tr>
</tbody>
</table>

```
\begin{verbatim}
cor(spec_j[,1:3])
    H j G j K j
H j 1.0000000 0.9179127 0.9322604
G j 0.9179127 1.0000000 0.7907841
K j 0.9322604 0.7907841 1.0000000
\end{verbatim}
```
These $I \times J = 17 \times 16 = 272$ coefficients are too much information. Thus, we calculate them again using the optional argument \texttt{LQ.output = "df"}, which produces a data frame with \(I \times J\) rows and three columns (\texttt{j_region}: ID of region \(j\), \texttt{i_industry}: ID of industry \(i\) and \texttt{LQ}: location quotient \(LQ_{ij}\)). We save the results in the object \texttt{lqs}:

\[
lqs <- \text{locq2(e_ij = G.regions.industries$emp_all,}
G.regions.industries$ind_code, G.regions.industries$region_code,
\text{LQ.output = "df")}
\]

As we forego an inspection of these single values, the results are not displayed here. Instead, we only deal with the five highest LQs in our results (the “top five”). We sort the resulting data frame decreasing and take a look at the first five rows:

\[
lqs\_sort <- lqs[\text{order(lqs$LQ, decreasing = TRUE),}]
\# Sort decreasing by size of LQ
\]

\[
lqs\_sort[1:5,]
\]

<table>
<thead>
<tr>
<th>\texttt{j_region}</th>
<th>\texttt{i_industry}</th>
<th>\texttt{LQ}</th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>BE</td>
<td>WZ08-R</td>
</tr>
<tr>
<td>1</td>
<td>BB</td>
<td>WZ08-B</td>
</tr>
<tr>
<td>239</td>
<td>ST</td>
<td>WZ08-B</td>
</tr>
<tr>
<td>28</td>
<td>BE</td>
<td>WZ08-L</td>
</tr>
<tr>
<td>111</td>
<td>HH</td>
<td>WZ08-J</td>
</tr>
</tbody>
</table>

The highest LQ is found for the arts, entertainment, and recreation sector (WZ08-R) in the German capital Berlin. Note that this result is congruent with several studies about the “creative class”, showing a large stock of “creative” employment in Berlin (e.g. Martin 2015). We also find a strong concentration of mining and quarrying in two Eastern regions, Brandenburg and Sachsen-Anhalt. Note that the \texttt{LQ} is a relative measure with respect to the total regional employment as well as the total industry-specific employment and the employment in the whole economy, not considering other aspects of industry or spatial structure.

These deficiencies should be overcome with the Litzenberger-Sternberg cluster index, also taking into account area, population and firm size. This additional data is also included in our current dataset (columns \texttt{area_sqkm}, \texttt{pop} and \texttt{firms}). The functions \texttt{litzenberger()} and \texttt{litzenberger2()} work equivalently to \texttt{locq()} and \texttt{locq2()}. To compute cluster indices for all \(I \times J\) combinations, we use the function \texttt{litzenberger2()}:

\[
litzenberger2(G.regions.industries$emp_all,
G.regions.industries$ind_code, G.regions.industries$region_code,
G.regions.industries$area_sqkm, G.regions.industries$pop,
G.regions.industries$firms)
\]
Like in `locq2()`, the default output is a matrix with $I$ rows and $J$ columns:

**Litzenberger-Sternberg cluster indices**  
$I = 17$ industries, $J = 16$ regions

<table>
<thead>
<tr>
<th>Region</th>
<th>Volume 6, Number 3, 2019</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Region</th>
<th>BB</th>
<th>BE</th>
<th>BW</th>
<th>BY</th>
<th>HB</th>
<th>HE</th>
</tr>
</thead>
<tbody>
<tr>
<td>WZ08-B</td>
<td>0.5736692</td>
<td>0.05611505</td>
<td>0.8041813</td>
<td>1.1073446</td>
<td>NaN</td>
<td>0.4745084</td>
</tr>
<tr>
<td>WZ08-C</td>
<td>0.1610679</td>
<td>3.24717820</td>
<td>2.6258085</td>
<td>1.2043415</td>
<td>5.119669</td>
<td>0.0603087</td>
</tr>
<tr>
<td>WZ08-D</td>
<td>0.2213625</td>
<td>1.37720778</td>
<td>1.7043505</td>
<td>1.4178208</td>
<td>4.172162</td>
<td>0.8541359</td>
</tr>
<tr>
<td>WZ08-E</td>
<td>0.8810260</td>
<td>10.14891585</td>
<td>0.8235517</td>
<td>0.6890213</td>
<td>6.705744</td>
<td>1.1427285</td>
</tr>
<tr>
<td>WZ08-F</td>
<td>0.7512888</td>
<td>11.36434353</td>
<td>1.2372108</td>
<td>0.9442221</td>
<td>3.498921</td>
<td>1.9225087</td>
</tr>
<tr>
<td>WZ08-G</td>
<td>0.2707787</td>
<td>12.10626625</td>
<td>1.3903532</td>
<td>0.9406777</td>
<td>7.136205</td>
<td>1.3361060</td>
</tr>
<tr>
<td>WZ08-H</td>
<td>0.4368768</td>
<td>13.26265081</td>
<td>1.0955747</td>
<td>0.7962074</td>
<td>22.566824</td>
<td>0.8352872</td>
</tr>
<tr>
<td>WZ08-J</td>
<td>0.2672336</td>
<td>26.60020727</td>
<td>1.4098657</td>
<td>0.9633029</td>
<td>8.481338</td>
<td>1.2880210</td>
</tr>
<tr>
<td>WZ08-K</td>
<td>0.1524825</td>
<td>10.28664194</td>
<td>1.4774980</td>
<td>1.1686261</td>
<td>6.650213</td>
<td>2.4739044</td>
</tr>
<tr>
<td>WZ08-L</td>
<td>0.2564814</td>
<td>56.65943460</td>
<td>0.9594093</td>
<td>0.8695636</td>
<td>11.839900</td>
<td>1.6099929</td>
</tr>
<tr>
<td>WZ08-M</td>
<td>0.1685895</td>
<td>39.30149403</td>
<td>1.5306799</td>
<td>1.0110472</td>
<td>9.410498</td>
<td>1.7302434</td>
</tr>
<tr>
<td>WZ08-N</td>
<td>0.4232166</td>
<td>26.91532975</td>
<td>1.0228326</td>
<td>0.7471872</td>
<td>10.796027</td>
<td>1.5265846</td>
</tr>
<tr>
<td>WZ08-P</td>
<td>0.2043023</td>
<td>25.30839556</td>
<td>1.3656409</td>
<td>0.9509028</td>
<td>7.322709</td>
<td>1.4627871</td>
</tr>
<tr>
<td>WZ08-Q</td>
<td>0.3445630</td>
<td>21.86956483</td>
<td>1.1770297</td>
<td>0.7837877</td>
<td>7.850886</td>
<td>1.2163624</td>
</tr>
<tr>
<td>WZ08-R</td>
<td>0.2450932</td>
<td>104.73565741</td>
<td>1.0839767</td>
<td>0.7821779</td>
<td>10.555369</td>
<td>1.0489672</td>
</tr>
<tr>
<td>WZ08-S</td>
<td>0.2891132</td>
<td>24.71310833</td>
<td>1.3435576</td>
<td>0.954882</td>
<td>9.893688</td>
<td>1.5421903</td>
</tr>
</tbody>
</table>

Note that there is a value equal to NaN, which means “not a number”, due to a division by zero; this is because there is no mining and quarrying (WZ08-B) in Bremen (HB). However, we take a look at the “top five” again:
lss <- litzenberger2(G.regions.industries$emp_all,
G.regions.industries$ind_code, G.regions.industries$region_code,
G.regions.industries$area_sqkm, G.regions.industries$pop,
G.regions.industries$firms, CI.output = "df")

lss_sort <- lss[order(lss$CI, decreasing = TRUE),]
lss_sort[1:5,]

<table>
<thead>
<tr>
<th>j_region</th>
<th>i_industry</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>BE</td>
<td>WZ08-R</td>
</tr>
<tr>
<td>111</td>
<td>HH</td>
<td>WZ08-J</td>
</tr>
<tr>
<td>26</td>
<td>BE</td>
<td>WZ08-J</td>
</tr>
<tr>
<td>28</td>
<td>BE</td>
<td>WZ08-L</td>
</tr>
<tr>
<td>114</td>
<td>HH</td>
<td>WZ08-M</td>
</tr>
</tbody>
</table>

Again, we find the largest cluster value for the arts and entertainment sector in Berlin. Also the other four highest indicators are discovered in the largest city states Berlin and Hamburg, especially with respect to the information and communication industry (WZ08-J) and other knowledge-intensive services. Obviously, the results of the Litzenberger-Sternberg index differ in a noticeable way from those of the LQ, which can be attributed to the consideration of other spatial aspects, especially controlling for the size of the regions.

4.2.4 Application example 3: Identifying clusters using micro-data

In our last example about agglomerations, we use the Ellison-Glaeser indices and the Howard-Newman-Tarp colocation index, which both require individual firm data. As this kind of micro-data is sensitive and, of course, not available in official statistics, we have to use fictional data from the textbook by Farhauer, Kröll (2014).

At first, we compute the Ellison-Glaeser agglomeration index for one industry $i$, $\gamma_i$. We use the REAT function ellison.a(), which is designed for this purpose and requires three vectors: the size (employment) of firm $k$, $e_{ik}$, the IDs of the regions $j$ each firm is located in, and the total regional employment, $e_j$. The numerical example in Farhauer, Kröll (2014), Table 14.11, contains ten firms in three regions (Wien, Linz, and Graz). We simply compile the data from the original table into separate vectors:

# regions (Austrian cities)
emp_firm <- c(200, 650, 12000, 100, 50, 16000, 13000, 1500, 1500, 25000)
# employment of the ten firms
emp_region <- c(500000, 400000, 100000)
# employment of the three regions

Now, we apply ellison.a() to this data:

ellison.a (emp_firm, emp_region, region)

[1] 0.05990628

The $EG$ agglomeration index of $\gamma_i \approx 0.06$, which is, by the way, the same result as in the textbook, indicates a stronger clustering than expected from a dartboard approach. Since this data is fictional, we refrain from interpreting this result.

The REAT package contains the dataset FK2014_EGC, which is compiled from the numerical example in Farhauer, Kröll (2014), Tables 14.14 to 14.17. There are $k = 42$ firms from $I = 4$ industries (clothing trade, forestry, textiles dyeing and textiles trade) in $J = 3$ regions (1, 2 and 3). We load this example data:

data(FK2014_EGC)

We compute $\gamma_i$ for all industries in the dataset. This can be done with the function ellison.a2(), which requires vectors containing the size of firm $k$, the corresponding industry $i$, and region $j$. We save the results in the object ega:
ega <- ellison.a2 (FK2014_EGC$emp_firm, FK2014_EGC$industry, FK2014_EGC$region)

Here, we see the output of the function:

Ellison–Glaeser Agglomeration Index
K = 42 firms, I = 4 industries, J = 3 regions

\[
\begin{align*}
\text{Gamma } & \text{ } i \\
\text{Clothing trade} & \quad -0.09379384 \\
\text{Forestry} & \quad 0.16838003 \\
\text{Textiles dyeing} & \quad -0.08012539 \\
\text{Textiles trade} & \quad -0.13040134
\end{align*}
\]

We see a strong clustering of the forestry industry, which is attributed to localization economies, but spatial avoidance in the three other industries. The visible output of \texttt{ellison.a2()} contains the \( \gamma_i \) values only, but the invisible \texttt{matrix} output also includes the other information referring to the \( EG \) agglomeration index:

\[
\begin{align*}
\text{ega} \\
\text{Gamma } & \text{ } i \\
\text{Clothing trade} & \quad -0.09379384 \quad 0.017909653 \quad -0.5025978 \quad 11 \quad 0.13124350 \\
\text{Forestry} & \quad 0.16838003 \quad 0.088262934 \quad 1.3660878 \quad 13 \quad 0.09240553 \\
\text{Textiles dyeing} & \quad -0.08012539 \quad 0.027764811 \quad -0.3801644 \quad 9 \quad 0.14559983 \\
\text{Textiles trade} & \quad -0.13040134 \quad 0.002734966 \quad -0.7663541 \quad 9 \quad 0.12208059
\end{align*}
\]

When looking at the forestry industry, we also see a high standardized value \( (z_i \approx 1.37) \) and a relatively low firm concentration \( (HHI_i \approx 0.09) \).

In the next step, we compute the \( EG \) coagglomeration index, \( \gamma^c \), for the same data using the function \texttt{ellison.c()} . This function requires the same information as \texttt{ellison.a2()} plus the total employment in the regarded regions (column \texttt{emp_region}):

\[
\text{ellison.c (FK2014_EGC$emp_firm, FK2014_EGC$industry, FK2014_EGC$region, FK2014_EGC$emp_region)}
\]

\[
[1] \ 12.0729
\]

Congruent with the calculation in Farhauer, Kröll (2014), the function returns \( \gamma^c \approx 12.07 \). This value is very large, which indicates urbanization economies in this fictional example.

If we want to analyze the coagglomeration of industry pairs instead, we may use the function \texttt{ellison.c2()} , which requires the same data:

\[
\text{ellison.c2 (FK2014_EGC$emp_firm, FK2014_EGC$industry, FK2014_EGC$region, FK2014_EGC$emp_region)}
\]

The output is a \texttt{matrix} with \( I \times I - I \) rows (one for each industry pair, omitting the combination of the same industry \( i \)):

Ellison–Glaeser Co-Agglomeration Index
K = 42 firms, I = 4 industries, J = 3 regions

\[
\begin{align*}
\text{Gamma } & \text{ } c \\
\text{Forestry-Clothing trade} & \quad 1.382257 \\
\text{Textiles dyeing-Clothing trade} & \quad 2.465609 \\
\text{Textiles trade-Clothing trade} & \quad 2.067766 \\
\text{Clothing trade-Forestry} & \quad 1.382257 \\
\text{Textiles dyeing-Forestry} & \quad 1.570292 \\
\text{Textiles trade-Forestry} & \quad 1.336020 \\
\text{Clothing trade-Textiles dyeing} & \quad 2.465609 \\
\text{Forestry-Textiles dyeing} & \quad 1.570292 \\
\text{Textiles trade-Textiles dyeing} & \quad 2.294259 \\
\text{Clothing trade-Textiles trade} & \quad 2.067766 \\
\text{Forestry-Textiles trade} & \quad 1.336020 \\
\text{Textiles dyeing-Textiles trade} & \quad 2.294259
\end{align*}
\]
If we want to focus on firm numbers instead of employment size, we may compute the Howard-Newman-Tarp excess colocation index, which is included in REAT through the functions `howard.cl()` for one colocation index for one pair of industries, `howard.xcl()` for the corresponding excess colocation index and `howard.xcl2()` for all combinations of $I \times I$ industries. Subsequent to the numerical example above, we calculate $XCL_{ab}$ for all industry pairs in the dataset FK2014_EGC, where the firm ID of $k$ is stored in the column firm:

```r
howard.xcl2 (FK2014_EGC$firm, FK2014_EGC$industry,
FK2014_EGC$region)
# this takes some seconds
```

The output has the same structure as the output from `ellison.c2()`:

```
Howard-Newman-Tarp Excess Colocation Index
K = 42 firms, I = 4 industries, J = 3 regions

  XCL
Forestry-Clothing trade  0.01902098
Textiles dyeing-Clothing trade  0.02909091
Textiles trade-Clothing trade  0.02020202
Clothing trade-Forestry  0.02377622
Textiles dyeing-Forestry  0.03282051
Textiles trade-Forestry  0.03589744
Clothing trade-Textiles dyeing  0.02707071
Forestry-Textiles dyeing  0.02666667
Textiles trade-Textiles dyeing  0.02814815
Clothing trade-Textiles trade  0.01743590
Forestry-Textiles trade  0.03012346
```

We see that the index by Howard et al. (2016) is structured differently than the indicators presented above: Although they are based on exactly the same data, the value for forestry and clothing trade ($XCL \approx 0.019$) is not equal to the value for clothing trade and forestry ($XCL \approx 0.024$). Why? The $XCL_{ab}$ is the difference between the colocation index, $CL_{ab}$, and the mean of a set of bootstrap samples, $CL_{ab}^{RND}$ (see Table 6). These random samples are drawn again each time a $XCL$ value is computed, consequently, also the $XCL$ value changes.

5 Proximity and accessibility

5.1 Distance-based measures of accessibility and proximity using individual point-level data

In this chapter, we mix two different concepts of indicators, accessibility and spatial proximity (see Table 10), both frequently used especially in the context of GIS (geographic information systems). Both concepts are discussed together because they have two aspects in common: 1) they are based on the geographical distance between point locations, in particular, the distance between an origin point $i$ or several origin points ($i = 1, ..., n$) and one or more destination points $j$ ($j = 1, ..., m$), and 2) for the calculation, they require geocoded (with geographical coordinates) individual point data.

One popular indicator of accessibility is the Hansen accessibility, developed by Hansen (1959) in the context of land use theory. The basic idea is that “accessibility” equals the sum of opportunities outgoing from a specific origin $i$. These opportunities are spread over a set of $m$ locations ($j = 1, ..., m$). The summation is weighted with the distance between $i$ and the $j$-th location. This distance, no matter how measured (e.g. street distance, Euclidean distance, driving time) is assumed to be perceived in a nonlinear way, which is operationalized by a nonlinear distance decay function (a.k.a. distance impedance function or response function), e.g. power, exponential or logistic. A similar concept was introduced by Harris (1954) attempting to model the market potential of
locations. If we replace the inverse distance weighting in the Harris indicator with another type of distance weighting, we see that both concepts are mathematically equivalent. The only difference is that the Harris indicator is conceptualized from the supplier’s perspective (e.g., market potential of a retail store) and the Hansen accessibility takes the demand location $i$ as a starting point (Pooler, 1987; Reggiani et al., 2011). As these indicators are dimensionless and range from zero to infinity, a normalization with a range from zero to one can be computed by weighting the results with the opportunities without distance correction.

This accessibility/potential concept can be used in the regional economic context e.g. to quantify the over-regional job potential (e.g. Wieland, Fuchs 2018) or the clustering of point locations of a specific type, such as retail stores (e.g. Larsson, Öner 2014). The most common application of these indicators may be the context of transport economics and transport geography (e.g. Albacete et al. 2017).

In the GIS context, spatial proximity can be measured using concentric zones within a radius of $t$ (buffers) around point $i$, where the number of the $j$ points within this radius is counted (Longley et al., 2005). A systematic analysis of spatial proximity or cluster patterns is possible using Ripley’s $K$ function (Ripley, 1976). It compares empirical point counts with expected values from a random spatial point process based on a Poisson distribution. Ripley’s $K$ computes empirical values for each distance band with a maximum distance of $t$, which can be compared to the expected value. A more comprehensible (and linear) interpretation is provided when normalizing the $K$ function in the form of the $L$ or $H$ function. Also, confidence intervals for the expected values can be calculated by bootstrapping (Kiskowski et al., 2009; Smith, 2016). All of these measures are based on a simple indicator function, $I(d_{ij} \leq t)$, which takes the value of $I = 1$ if point $j$ is within a distance of $t$ from point $i$ or not ($I = 0$). Originating from natural sciences, especially Ripley’s $K$ is frequently used when analyzing location patterns in spatial economic contexts, such as the clustering of retail stores (e.g. Krider, 2013).

Table 10: Accessibility and proximity indicators using point-level data

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Non-normalized</th>
<th>Normalized</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessibility/Market potential</td>
<td>$M_j = \sum_{i=1}^{n} O_i d_{ij}^{-1}$, $0 \leq M_j \leq \infty$</td>
<td>$A^<em><em>i = \sum</em>{j=1}^{m} O_j f(d_{ij}) / \sum_{j=1}^{m} O_j$, $0 \leq A^</em>_i \leq 1$</td>
</tr>
<tr>
<td>Harris</td>
<td>$A_i = \sum_{j=1}^{m} O_j f(d_{ij})$ if $i \neq j$, $0 \leq A_i \leq \infty$</td>
<td>where: $f(d_{ij}) = d_{ij}^{-\lambda}$ or $f(d_{ij}) = e^{-\lambda d_{ij}}$</td>
</tr>
<tr>
<td>Hansen</td>
<td>$A_i = \sum_{j=1}^{m} O_j f(d_{ij})$ if $i \neq j$, $0 \leq A_i \leq \infty$</td>
<td>$f(d_{ij}) = 1$ if $d_{ij} \leq t$, $0$ otherwise.</td>
</tr>
</tbody>
</table>

Notes: $d_{ij}$ is the distance from origin location $i$ ($i = 1, \ldots, n$) to destination location $j$ ($j = 1, \ldots, m$), $O_j$ is a variable quantifying the size of destination $j$, $t$ is a maximum search radius and $I(d_{ij} \leq t)$ is the indicator function taking the value of $I = 1$ if $d_{ij} \leq t$, and $I = 0$ otherwise. Compiled from: Kiskowski et al. (2009); Krider, Butler (2013); Peña Carrera (2002); Pooler (1987); Reggiani et al. (2011); Smith (2016)
Table 11: REAT functions for accessibility and proximity on the point level

<table>
<thead>
<tr>
<th>Indicator</th>
<th>REAT function</th>
<th>Mandatory arguments</th>
<th>Optional arguments</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance matrix</td>
<td>dist.mat()</td>
<td>data frame(s) with start points i (ID, lat, lon) and end points j (ID, lat, lon), distance unit</td>
<td>i ≠ j</td>
<td>data frame with from, to, from-to and distance d_{ij} (distance matrix)</td>
</tr>
<tr>
<td>Buffer</td>
<td>dist.buf()</td>
<td>data frame(s) with start points (ID, lat, lon) and end points (ID, lat, lon), max. distance t, distance unit</td>
<td>i ≠ j, sum O_j at endpoints</td>
<td>list with distance matrix (data frame) and count table (data frame)</td>
</tr>
<tr>
<td>Hansen/ Harris</td>
<td>hansen()</td>
<td>distance matrix (data frame with start points i and end points j as well as distance d_{ij} and O_j), weighting functions, parameters λ and γ</td>
<td>distance constant, max. distance t, i ≠ j</td>
<td>data frame with origins i and accessibility A_i</td>
</tr>
<tr>
<td>Ripley</td>
<td>ripley()</td>
<td>data frame with points (ID, lat, lon), total area A, max. distance t, number of distance intervals</td>
<td>local K values, confidence intervals no. of samples, significance level, plot (K, L or H)</td>
<td>visible: matrix with t, K_t, E(K_t), K_t − E(K_t), L_t and H_t for each distance interval, invisible: matrix (as described above) and optional: matrices with local K values and confidence intervals</td>
</tr>
</tbody>
</table>

Source: own compilation.

Putler 2013) or other types of firms assumed to be connected in a network (e.g. Espa et al. 2010).

5.2 Application in REAT

5.2.1 REAT functions for accessibility and proximity on the point level

Table 11 shows the REAT functions for the accessibility and proximity methods described above. A simple Euclidean distance matrix for georeferenced points (data frame with latitude and longitude) can be calculated using the function dist.mat(). The function dist.buf() computes a “count points within buffer”, where also a weighting, O_j, can be summarized (e.g., if the destination points are cities of a given population, one could count the number of cities within 50 kilometers and their corresponding population). The latter function uses dist.mat(), thus, it is not necessary to create a distance matrix before.

The same is the case for the function ripley(), which calculates Ripley’s K function for georeferenced data (data frame with lat/lon) and a given number of distance intervals up to a maximum distance of t. The differences between the empirical values, K_t, and the expected values, E(K_t), as well as the normalizations (L_t and H_t) are calculated and returned automatically. Optionally, local K values for each distance interval and corresponding confidence intervals are computed. These confidence intervals are based on bootstrapping with a given number of samples (default: 100) on a given significance level (the default value is α = 0.05, which leads to confidence intervals of a range from α/2 = 2.5% to 1 − α/2 = 97.5%). Note that the plot of the K function (or, when desired, L or H function) provides a graphical and more intuitive interpretation of the analyzed point pattern, especially when including confidence intervals.

When calculating the Hansen accessibility (or the Harris market potential) with hansen(), a distance matrix including the opportunities, O_j, is required. This can be, of course, done with dist.mat() (if straight-line distances are sufficient), but also with any other software creating distance matrices (and any type of transport costs indicator). In hansen(), the user may choose between a power, exponential or logistic distance decay function. Optionally, the normalized Hansen accessibility is returned additionally.
5.2.2 Application example 1: Location analysis of medical practices

In the example in Section 2.2.2, we dealt with small-scale regional inequality in health care in South Lower Saxony, Germany. We have seen that e.g. psychotherapists are more spatially clustered than general practitioners (GPs). Returning to this topic, we want to use proximity and accessibility measures for determining the market potential (in the sense of the Harris model) of these health care locations. Obviously, there are different location patterns of general practitioners and psychotherapists. In the related study, there was evidence that psychotherapists are not just clustered but clustered within some districts of larger cities (Wieland, Dittrich, 2016). In the German health care planning system, the market potential of medical practices is the main determinant of the official authorization to be included into the allocation system of health insurance, while psychotherapists are assumed to need quite larger market areas than GPs (Kassenärztliche Bundesvereinigung, 2013). Consequently, our research hypothesis is that the population potential of psychotherapists is larger than that of general practitioners.

We use the same test data as in the mentioned example, containing the health locations (GoettingenHealth1) and the corresponding settlements (GoettingenHealth2). We load both R datasets:

```r
data(GoettingenHealth1)
data(GoettingenHealth2)
```

Table GoettingenHealth1 contains 617 locations, whose ID is stored in the column `location`. Columns `lat` and `lon` contain the latitude and longitude, respectively, while the corresponding location type can be found in column `type` (phys_gen: general practitioners, psych: psychotherapists, pharm: pharmacies). As the following applications may be time-consuming, we extract the general practitioners from GoettingenHealth1 and draw a random sample of ten doctor’s practices:

```r
physgen <- GoettingenHealth1[GoettingenHealth1$type == "phys_gen",]
# general practitioners: column "type" is equal to "phys_gen"
physgen_sample <- physgen[sample(nrow(physgen),10),]
# random sampling of ten general practitioners
```

Now, we want to summarize the population potential of these health locations in a 1,000 meters buffer. We apply the function `dist.buf()` to the sample data `physgen_sample` and sum up the local population of the districts within this distance (column `pop` in GoettingenHealth2):

```r
physgen_pot <- dist.buf(physgen_sample, "location", "lat", "lon",
GoettingenHealth2, "district", "lat", "lon", bufdist = 1000,
ep_sum = "pop")
# counting all districts within a radius of 1000 meters
# and summing the corresponding population
```

We calculate the arithmetic mean of all ten potentials:

```r
mean2(physgen_pot$count_table$sum_pop)
[1] 8027.7
```

On average, the ten GP practices have a population potential of about 8,028 inhabitants. One problem related to the buffer technique is the lack of distance weighting: All origin points up to a given distance are included completely, while all points above 1,000 meters are ignored. Thus, we repeat estimating the population potential using the Hansen accessibility. At first, we need an origin-destination matrix (distance matrix) from the origin points to the sampled GP locations. We use the function `dist.mat()` and merge the returned distance matrix with the population values from GoettingenHealth2:

```r
...
physgen_od <- dist.mat(GoettingenHealth2, "district", "lat", "lon",
physgen_sample, "location", "lat", "lon")
# creating OD matrix from all districts to the
# sampled general practitioners

physgen_od <- merge (physgen_od, GoettingenHealth2,
by.x = "from", by.y = "district")
# merging with GoettingenHealth2 to include the
# population values of the districts

Then, we use the function `hansen()` to calculate the Hansen accessibility (used in
the sense of the Harris market potential model) for each GP location in `physgen_od`.

The required columns in this dataset are the IDs of the GP locations (`to`), the IDs of
the districts (`from`) and the population of the districts (`pop`) as well as the distances
calculated above (`distance`). Finally, we have to set a distance weighting (which has an
important influence in all types of spatial interaction models like this). For this purpose,
we fall back on the results of a study by Fülöp et al. (2011): Based on empirical patient’s
choice of doctor, they estimated distance decay functions in spatial interaction models
(Huff model) for several types of physicians. For GPs, an exponential distance decay
function with $\lambda = -0.28$ was found to fit the empirical data best. To set a distance
decay function type and the related weighting(s), the function arguments `dtype` and
`lambda` must be used. We save the results under the name `physgen_hansen`:

```
physgen_hansen <- hansen(physgen_od, "to", "from", "pop",
"distance", dtype = "exp", lambda = -0.28)
# calculating Hansen accessibility for the ten
# sampled general practitioners
```

The output of the `hansen()` function is:

Hansen Accessibility

J = 420 locations with mean attractivity = 1138.486
I = 10 origins with mean transport costs = 28.07581
Attractivity weighting (pow) with Gamma = 1
Distance weighting (exp) with Lambda = -0.28

to accessibility
1 1103 24267.054
2 1171 17629.564
3 1206 9581.732
4 1220 9213.407
5 197 10023.854
6 301 6489.571
7 600 69676.232
8 755 66921.123
9 966 13154.921
10 974 3666.171

Again, we calculate the arithmetic mean of the distance-weighted market potentials:

```
mean2(physgen_hansen$accessibility)
[1] 23062.36
```

The average population potential of the ten GPs is equal to 23,063 inhabitants.

As we want to compare the market potential of GPs and psychotherapists, we repeat
the same analysis for them, now in the “fast mode”, leaving out most comments, as the
functions and commands are exactly the same as above, only applied to psychotherapists.
psychgen <- GoettingenHealth1[GoettingenHealth1$type == "psych",]

psych_sample <- psychgen[sample(nrow(psychgen),10),]

psych_pot <- dist.buf (psych_sample, "location", "lat", "lon", GoettingenHealth2, "district", "lat", "lon", bufdist = 1000, ep_sum = "pop")

mean2(psych_pot$count_table$sum_pop)
[1] 12245.88

The calculation of Hansen accessibility is different from the one for GPs with respect to the assumed distance reaction of the (potential) clients: For psychotherapists, Fülöp et al. (2011) found a distance impedance which is considerably smaller than for GPs (and any other type of doctor), resulting in a weighting parameter of $\lambda = -0.11$ in the exponential decay function:

psych_od <- dist.mat(GoettingenHealth2, "district", "lat", "lon", psych_sample, "location", "lat", "lon")

psych_od <- merge (psych_od, GoettingenHealth2, by.x = "from", by.y = "district")

psych_hansen <- hansen (psych_od, "to", "from", "pop", "distance", dtype = "exp", lambda = -0.11)

Hansen Accessibility

J = 420 locations with mean attractivity = 1138.486
I = 10 origins with mean transport costs = 25.56756
Attractivity weighting (pow) with Gamma = 1
Distance weighting (exp) with Lambda = -0.11

to accessibility
1 1031 43415.63
2 1213 39226.26
3 179 33491.41
4 313 51228.41
5 506 147887.43
6 786 147969.39
7 791 147971.80
8 811 148021.51
9 872 147475.57
10 922 42424.51

mean2(psych_hansen$accessibility)
[1] 94911.19

We see that the average population potential of the sampled psychotherapists on the 1,000 meters buffer level is equal to 12,246 inhabitants, which is about one third more than for GPs. The Hansen/Harris market potential of psychotherapists of about 94,911 persons is a fourfold increase compared to the GPs. We have to remember that the last result is not only a matter of location but also due to a lower assumed distance decay. However, the population potential of the sampled psychotherapists is obviously higher than the potential of the GPs, which can be attributed to a different location pattern, where psychotherapists are more clustered within larger city districts.

5.2.3 Application example 2: Clustering of health service providers

We stick to the example of health care locations. As we have found different degrees of regional inequality with respect to suppliers (Section 2.2.2) and of market potentials
(Section 5.2.2), we now analyze the clustering patterns of health service providers. In South Lower Saxony there is nearly the same number of psychotherapists (118) and pharmacies (120), but we should not expect their location patterns to be similar or even equal. Following the results above, we hypothesize that psychotherapists are more spatially clustered than pharmacies (as we already know about clustering with respect to districts in the former case and we can expect an avoidance tendency in the latter case due to a high degree of substitutability).

For this analysis, we compute Ripley’s $K$ with the REAT function `ripley()`. Before going on, we have to prepare two things: First, we load the required dataset. Then, we must calculate the total area of the study area manually (here: in square meters).

```r
data(GoettingenHealth1)

area_goe <- 1753000000
# area of Landkreis Goettingen (sqm)
area_nom <- 1267000000
# area of Landkreis Northeim (sqm)
area_gn <- area_goe+area_nom
```

Now, we compute Ripley’s $K$ for the pharmacies only, which means processing only those locations in `GoettingenHealth1` which are pharmacies (`type == "pharm"`). We set our maximum search radius equal to $t = 30000$ (function argument `t.max`), divided into 300 distance intervals (`t.sep`), resulting in distance steps of 100 meters. As we want to check for a significant deviation from a random spatial pattern, we instruct the function to construct confidence intervals (`ci.boot = TRUE`) using the default settings ($\alpha = 0.05$, 100 bootstrapping samples). We also plot the results (default function argument: `K.plot = TRUE`) to inspect our results graphically. Here, we plot $K$, which is also the default setting (if the user wants to plot $L_t$ or $H_t$ instead, the function argument `Kplot.func` has to be changed to “$L$” or “$H$”, respectively):

```r
ripley(GoettingenHealth1[GoettingenHealth1$type == "pharm",],
      "location", "lat", "lon", area = area_gn, t.max = 30000, t.sep = 300,
      K.local = TRUE, ci.boot = TRUE, ci.alpha = 0.05, ciboot.samples = 100,
      plot.title = "Ripley’s K: Clustering of pharmacies")
```

The output is a matrix with six columns and one row for each distance interval. Thus, we skip the full output here:

Ripley’s $K$

$n = 120$ points

<table>
<thead>
<tr>
<th>$t$</th>
<th>$K_t$</th>
<th>$K_t \exp$</th>
<th>$K_t - K_t \exp$</th>
<th>$L_t$</th>
<th>$H_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100</td>
<td>31415.93</td>
<td>3355556</td>
<td>1033.492</td>
<td>933.49238</td>
</tr>
<tr>
<td>2</td>
<td>200</td>
<td>125663.71</td>
<td>25583333</td>
<td>32297222</td>
<td>31794567</td>
</tr>
<tr>
<td>3</td>
<td>300</td>
<td>282743.34</td>
<td>25586111</td>
<td>25303368</td>
<td>2853.824</td>
</tr>
<tr>
<td>4</td>
<td>400</td>
<td>502654.82</td>
<td>32297222</td>
<td>3206.326</td>
<td>2806.3258</td>
</tr>
<tr>
<td>5</td>
<td>500</td>
<td>785398.16</td>
<td>39008333</td>
<td>38222935</td>
<td>3523.739</td>
</tr>
</tbody>
</table>

We repeat the computation of Ripley’s $K$ for the psychotherapists:

```r
ripley(GoettingenHealth1[GoettingenHealth1$type == "psych",],
      "location", "lat", "lon", area = area_gn, t.max = 30000, t.sep = 300,
      K.local = TRUE, ci.boot = TRUE, ci.alpha = 0.05, ciboot.samples = 100,
      plot.title = "Ripley’s K: Clustering of psychotherapists")
```
The output is (also truncated):

\[
\text{Ripley's K} \\
\text{n = 118 points}
\]

<table>
<thead>
<tr>
<th>t</th>
<th>K t exp</th>
<th>K t</th>
<th>Rt-Kt exp</th>
<th>L t</th>
<th>H t</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>31415.93</td>
<td>30798621</td>
<td>30767205.2</td>
<td>3131.055</td>
<td>3031.055025</td>
</tr>
<tr>
<td>2</td>
<td>125663.71</td>
<td>48583740</td>
<td>48458076.6</td>
<td>3932.516</td>
<td>3732.516350</td>
</tr>
<tr>
<td>3</td>
<td>282743.34</td>
<td>80249928</td>
<td>79967184.8</td>
<td>5054.141</td>
<td>4754.141421</td>
</tr>
<tr>
<td>4</td>
<td>502654.82</td>
<td>132737719</td>
<td>132235064.2</td>
<td>6500.133</td>
<td>6100.132940</td>
</tr>
<tr>
<td>5</td>
<td>785398.16</td>
<td>202143062</td>
<td>201357664.2</td>
<td>8021.480</td>
<td>7521.479612</td>
</tr>
</tbody>
</table>

... 

The graphical output is shown in Figures 4a (pharmacies) and 4b (psychotherapists), respectively. The expected value of $K_t$ is plotted as blue line, while the empirical $K_t$ values are red and the corresponding confidence intervals are colored in green (These colors are the default values in `ripley()` and can be changed by the function arguments `lcol.exp` and `lcol.emp`, respectively). As we have nearly the same number of points in both cases within the same field area, a direct comparison seems reasonable. Obviously, both types of locations show a significant spatial clustering: Also the pharmacies are more clustered than expected on condition of complete spatial randomness up to a distance of about 15,000 meters. We have to remember that also the population is already clustered (see Section 2.2.2) and the spatial distribution of pharmacies may follow this pattern. However, the clustering of psychotherapists exceeds this level enormously, especially within smaller distances up to about 8,000 meters. In conclusion, the psychotherapists are more spatially clustered than pharmacies.

6 Analysis and prognosis of regional growth

6.1 Tools and models concerning regional growth

6.1.1 Analyzing regional growth: shift-share analysis and portfolio matrix

Aspects of regional growth have already been discussed in the context of regional convergence in Section 3. The identification of clusters was the topic of Section 4. Combining some aspects of both, this section presents a collection of tools and models concerning regional growth with respect to industries. Like the indicators in Section 4, these techniques are of high significance especially in the context of local economic policy and municipal business promotion activities, aiming at e.g. strengthening a city’s or region’s competitiveness, defining its profile or increasing the number of jobs (Dinc, 2015; Nischwitz et al., 2017). Inspired by Farhauer, Kröll (2014) and congruent with the mathematical formulations in Section 4, we calculate on the basis of local/regional employment, $e_{ij}$,
which is the number of employees of industry $i$ in region $j$. Its growth from time $t$ to time $t+y$ can be operationalized as an absolute value ($\Delta e_{ij} = e_{ij,t+y} - e_{ij,t}$) or as a relative growth ($\Delta e_{ij}^r = e_{ij,t+y}/e_{ij,t}$) or as a (percentage) growth rate ($\Delta e_{ij}^{gr} = e_{ij,t+y}/e_{ij,t} - 1$).

The first technique described is the regional economic portfolio matrix, originating from the portfolio matrix in marketing, developed by the Boston Consulting Group (BCG) for the identification of growing and declining business fields of firms (Henderson, 1973). However, this technique can also be applied to several regional economic contexts (Baker et al., 2002; Howard, 2007). Here, we present a portfolio matrix which compares the growth in one region with the growth in a superordinate reference region (e.g. whole economy). When using the matrix in this way, it is a plot of the growth rate with respect to industry $i$ in the region ($\Delta e_{ij}^{gr}$) on the $x$ axis and the corresponding growth in the reference region ($\Delta e_{ij}^{gr}$) on the $y$ axis (see Figure 5a). The size of the points for each industry may be the total size of employment in the region ($e_{ij}$) to reflect the absolute relevance of the $i$-th industry. The plot is segmented into four quadrants, differentiated with respect to positive or negative growth rates. As implied by the colors of the quadrants, they can be interpreted as follows: Quadrant I (top right) contains the industries growing in both the region and the whole economy (or any other reference region). Quadrant II (top left) shows all industries growing in the whole economy but shrinking in the regarded region, which may indicate significant locational handicaps. Quadrant III (bottom left) includes all industries shrinking in the region as well as in the whole economy. Quadrant IV (bottom right) shows the special case of “star” industries, indicating that these industries grow in the regarded region while shrinking in the whole economy. Note that this segmentation (and the corresponding interpretation) differs from the original BCG matrix.

Another variant of the portfolio matrix, which was developed in the context of designing the REAT package, is shown in Figure 5b. Combining the aspects of regional specialization (see Section 4) and regional growth, we can plot the location quotient as an indicator of local specialization on the $x$ axis, while plotting an industry-specific growth indicator on the $y$ axis. For identifying “growing” industries, there are at least three options of operationalization: We can plot the industry-specific regional growth rate ($\Delta e_{ij}^{gr}$) on the $y$ axis (which is on the $x$ axis in the portfolio matrix in Figure 5a) or the industry-specific national rate ($\Delta e_{ij}^{gr}$) or, if we want to show regional growth in relation to national growth, the quotient of industry-specific regional and national growth rates ($\Delta e_{ij}^{gr}/\Delta e_{ij}^{gr}$). In quadrant I, we see now all industries overrepresented in the region (in terms of the location quotient) as well as growing on the regional/national level. Quadrant II shows all industries underrepresented in the region but growing as well. In quadrants III and IV, we can identify all industries with negative growth rates, which are underrepresented or overrepresented, respectively.
### Table 12: Shift-share analysis: Dunn and Gerfin type

<table>
<thead>
<tr>
<th>Component</th>
<th>Dunn-type (absolute)</th>
<th>Gerfin-type (index)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Net total shift</td>
<td>$\Delta e_j = e_{jt+y} - e_{jt} = \frac{m_{jt,t+y} + c_{jt,t+y}}{n_{jt,t+y} + m_{jt,t+y} + c_{jt,t+y}}$</td>
<td>$t_{t+y} = m_{jt,t+y} c_{jt,t+y} = \frac{e_{jt+y}}{e_{jt}}$</td>
</tr>
<tr>
<td>National share</td>
<td>$n_{jt,t+y} = e_{jt} e_{yt+y} - e_{jt}$</td>
<td>$n_{jt,t+y} = 1$ (omitted)</td>
</tr>
<tr>
<td>Industrial mix</td>
<td>$m_{jt,t+y} = \sum_{i=1}^{I} e_{ijt} \frac{e_{iy+y}}{e_{yt}} - e_{jt} \frac{e_{yt+y}}{e_{yt}}$</td>
<td>$m_{jt,t+y} = \sum_{i=1}^{I} e_{ijt} \frac{e_{iy+y}}{e_{yt}}$</td>
</tr>
<tr>
<td>Regional share</td>
<td>$c_{jt,t+y} = \sum_{i=1}^{I} e_{ijt} \left( \frac{e_{iy+y}}{e_{yt} e_{yt}} - \frac{e_{yt+y}}{e_{yt}} \right)$</td>
<td>$c_{jt,t+y} = \frac{e_{iy+y}}{e_{yt}}$</td>
</tr>
</tbody>
</table>

**Dynamic (T time periods, while $T > 2$)**

| National share | $n_{jt,T} = \sum_{t=1}^{T} e_{ijt} \frac{e_{yt+y}}{e_{yt}} - e_{jt}$ |
| Industrial mix | $m_{jt,T} = \sum_{t=1}^{T} \sum_{i=1}^{I} e_{ijt} \frac{e_{iy+y}}{e_{yt}} - e_{jt} \frac{e_{yt+y}}{e_{yt}}$ |
| Regional share | $c_{jt,T} = \sum_{t=1}^{T} \sum_{i=1}^{I} e_{ijt} \left( \frac{e_{iy+y}}{e_{yt} e_{yt}} - \frac{e_{yt+y}}{e_{yt}} \right)$ |

**Industry-specific**

| National share | $n_{ij,t+y} = e_{ijt} e_{yt+y} - e_{ijt}$ |
| Regional share | $c_{ij,t+y} = e_{ijt} \left( \frac{e_{iy+y}}{e_{yt} e_{yt}} - \frac{e_{yt+y}}{e_{yt}} \right)$ |

**Prognosis for time period $z$**

| Employment | $\Delta e_{ij,t+z} = e_{ij,t+y} \left( \frac{e_{iy+y}}{e_{yt+y}} \right) c_{jt,t+y}$ |

Notes: $e_{jt}$ is the employment in region $j$ at time $t$, $e_{ijt}$ is the employment of industry $i$ in region $j$ at time $t$, $e_{yt}$ is the total employment in the whole economy at time $t$, $e_{yt}$ is the total employment in industry $i$, $y$ and $z$ are numbers of time periods added to $t$ ($z > y$), $T$ is the number of regarded time periods and $I$ is the number of industries.

Compiled from: Farhauer, Kröll (2014); Haynes, Parajuli (2014); Schätzel (2000); Schönebeck (1996); Spielermann, Wegener (2008); Barff, Knight (1988)

A well-established model of regional growth is the shift-share analysis, which is, although developed independently from the portfolio matrix, closely linked to the concept presented above. The original shift-share analysis was introduced by Dunn Jr. (1960) and given a theoretical foundation by Casler (1989). Parallelly and independently, Gerfin (1964) developed a variant of shift-share analysis, which is more popular in the German-speaking regional economic science. Both concepts have been extended in several ways. Table 12 shows the basics of shift-share analysis with respect to “Dunn” and “Gerfin” type. As there are several ways of formulating the shift-share formulae and calling the particular elements of the shift-share analysis, the description here is based on the mathematical formulations in Farhauer, Kröll (2014) and the terms used in Haynes, Parajuli (2014).

The basic idea of shift-share analysis is the decomposition of regional growth into components, recognizing that single economic regions are embedded into and influenced by a larger regional system, normally the whole economy, just called “the nation” hereinafter: The (employment or e.g. gross value added) growth of industry $i$ in region $j$ from time $t$ to time $t + y$ can be attributed to 1) a national trend, which means the economic climate in the whole system of regions, 2) the all-over growth or decline of the regarded industries and 3) the industry-specific performance of the region, which is linked to locational advantages or disadvantages. The first component is called national share and reflects the growth in region $j$ that would have occurred if region $j$ would have developed exactly as the nation. The second component is the industrial mix, represent-
ing the aggregated industry-specific growth in region $j$ if the regarded industries would have developed like in the whole economy, adjusted by the national effect. The third component is the regional share, which is the “residuum” of the first two components: this share of growth is attributed to locational advantages (or disadvantages), showing the regional growth adjusted by national and industry effects (Farhauer, Kröll, 2014; Haynes, Parajuli, 2014).

The Dunn-type models deal with absolute growth ($\Delta e_{ij}$ or $\Delta e_j$), which is the sum of all shift-share components, and a net total shift, which is the sum of the industrial mix and the regional share (as these components are region-specific). Thus, this technique is also called the “difference method”. The Gerfin-type approaches express growth in terms of indices, while the net total shift for region $j$ is the result of a multiplication of the industrial mix index and the regional share index, resulting in the alternative denomination “index method” (Schätzl, 2000).

Several extensions have been developed for the Dunn-type shift-share analysis (Haynes, Parajuli, 2014). One regular application calculates a shift-share analysis for each industry $i$ in region $j$ (instead of computing components for the whole region), while skipping the industrial mix effect. A main contribution was the dynamic shift-share analysis by Barff, Knight (1988). It extended the Dunn model by dealing with growth within a longitudinal cut of $T$ years. Other extensions of the Dunn-type technique provide a deeper differentiation of the three components, which are regarded as correlated (e.g. Arcelus 1984; Esteban-Marquillas 1972).

6.1.2 Commercial area prognosis

Also developed independently in the context of German urban planning, a commercial area prognosis deals with an absolute (assumed) employment growth ($\Delta e_{ij}$) over $T$ years, which is used to forecast the required commercial area within a city or region $j$ up to time $T$. Note that “commercial area” represents the type of urban area which is used by specific economic activities, especially industrial plants, and/or designated for this purpose in municipal land-use plans. This technique is a demand-side approach, as it derives the required commercial area from the (expected) demand for it (Bonny, Kahnert, 2005). See Table 13 for the calculation of two types of commercial area prognosis based on employment growth.

The basic model called GIFPRO (German abbreviation for “Gewerbe- und Industrieflächenbedarfsprognose”, roughly translated: prognosis of future demand of commercial area) was developed by Stark et al. (1981). The usual procedure is to estimate – starting from the current employment – the future industry-specific employment in region $j$. This number of employees is weighted by the industry-specific shares of workers usually located in commercial areas and multiplied by a resettlement rate ($sq_{ij}$ percent of employees from industry $i$ are resettled in one time period) and a relocation rate ($rq_{ij}$ percent of employees from industry $i$ are relocated in one time period) as well as a reutilization rate ($ru_{ij}$ percent of employees from industry $i$ will be located at reused commercial area). This “commercial area-relevant” employment is weighted with an areal index, $a_{ij}$ (commercial area per employee), to compute the commercial area for industry $i$ in region $j$ for one time period $t$. The expected commercial area is summed over all $I$ industries ($A_{j}t$) and, finally, over all $T$ years and $I$ industries ($A_{jt}$) (Bonny, Kahnert, 2005; Planungsgruppe MWM, 2009).

A significant extension was developed in the context of establishing a land-use plan for Dresden: The TBS-GIFPRO (German abbreviation for “Trendbasierte und standortspezifische Gewerbe- und Industrieflächenbedarfsprognose”, roughly translated: trend-based and location-specific prognosis of future demand of commercial area) technique (Deutsches Institut für Urbanistik GmbH, Spath + Nagel (GbR), 2010). It includes a stochastic approach for forecasting employment as well as other region-specific data. The employment prognosis is done using a trend regression model (employment against time) based on past empirical employment data for region $j$ (mostly from official employment statistics) which are used for forecasting future employment. For each $i$ industry, a single regression model is estimated, where the function type is not pre-defined but chosen e.g. based on the explained variance ($R^2$) and/or plausibility considerations.
Table 13: Commercial area prognosis

<table>
<thead>
<tr>
<th>Prognosis</th>
<th>GIFPRO</th>
<th>TBS-GIFPRO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employment</td>
<td>$e_{ij}^1 = \left( e_{ij0} a_{i} \frac{sq_{ij}}{100} + e_{ij0} a_{i} \frac{ru_{ij}}{100} \right) - \left( e_{ij0} \frac{ru_{ij}}{100} \right)$</td>
<td>$e_{ij}^A = \left( e_{ij1} a_{i} \frac{rq_{ij}}{100} + e_{ij1} a_{i} \frac{ru_{ij}}{100} \right) - \left( e_{ij1} \frac{ru_{ij}}{100} \right)$ where: $e_{ij} = f(t) = a + bt$ or $f(t) = abt$ or $f(t) = e_{MAX} \frac{e^{(t-1)}}{1+e^{(t-1)}}$</td>
</tr>
</tbody>
</table>

Areal index
pre-defined: $ai_{ij}$
empirical estimation: $ai_{ij} = \frac{A_{ij}}{e_{ij}}$

Commercial area

$A_{ij} = e_{ij1} ai_{ij}$

$A_{jt} = \sum_{i=1}^{I} A_{ij}$

$A_{jt} = \sum_{i=1}^{I} \sum_{t=1}^{T} A_{ij}$

Notes: $e_{ij1}$ is the (expected) number of employees of industry $i$ in region $j$ which is located in commercial areas at time $t$, $e_{ij0}$ is the employment of industry $i$ in region $j$ at start time $t0$ (empirical value), $e_{ij}$ is the (expected) employment of industry $i$ in region $j$ at time $t$, $a_{i}$ is the share of employees in industry $i$ which is located in commercial areas, $sq_{ij}$ is the resettlement rate with respect to industry $i$ in region $j$ in one time period, $rq_{ij}$ is the relocation rate with respect to industry $i$ in region $j$ in one time period, $ru_{ij}$ is the reutilization rate with respect to industry $i$ in region $j$ in one time period, $ai_{ij}$ is the areal index with respect to industry $i$ in region $j$ (commercial area per employee), $A_{ij}$ is the (expected) commercial area for industry $i$ in region $j$ at time $t$, $A_{jt}$ is the (expected) commercial area in region $j$ at time $t$ and $A_{jt}$ is the sum of the (expected) commercial area in region $j$ over all $T$ time periods.

Compiled from: Bonny, Kahnert (2005); CIMA Projekt + Entwicklung GmbH et al. (2011); Deutsches Institut für Urbanistik GmbH, Spath + Nagel (GbR) (2010); Planungsgruppe MWM (2009); Mulligan (2006); Vallée et al. (2012)

The function may be linear (which seems unrealistic) or not: Deutsches Institut für Urbanistik GmbH, Spath + Nagel (GbR) (2010) use linear and exponential functions. However, from the growth perspective, also a logistic function may be applied (see Mulligan 2006 for a discussion of logistic growth with respect to population). If possible, the areal index and, maybe, other parameters are also estimated empirically for the specific region $j$ (e.g. via firm-level surveys and/or official statistical data).

6.2 Application in REAT

6.2.1 REAT functions for analyzing and forecasting regional growth

Table 14 shows the functions for the analysis of regional growth as implemented in REAT. Table 15 presents the functions related to commercial area prognosis. All of these functions require at least current employment data for each industry in the regarded region $j$, $e_{ij}$, which may be a single numeric vector or the column of a data frame or matrix. Another similarity of all mentioned functions is the optional argument of the industry names (or codes). If no industry names are stated by the user (default function argument: industry.names = NULL), the industries are numbered consecutively. With respect to the function output, all regional growth functions distinguish between a visible and an invisible output (see e.g. Section 3), where the main results are returned automatically and the details are included in the invisible output (mostly a list with several entries of type matrix).

The portfolio matrix (growth portfolio and growth-specialization portfolio, respectively) can be plotted using the functions portfolio() and locq.growth(), respectively. The different techniques of shift-share analysis are distributed over five functions (shift(), shiftd(), shifti(), shiftid() and shiftp()). The usage of portfolio and shift-share functions is similar: In any case, the user needs industry-specific employment data for the regarded region and the reference region (e.g. whole economy) for at least two time periods (e.g. years).
### Table 14: REAT functions for analyzing regional growth

<table>
<thead>
<tr>
<th>Model</th>
<th>REAT function</th>
<th>Mandatory arguments</th>
<th>Optional arguments</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Growth portfolio matrix</strong></td>
<td>portfolio()</td>
<td>vectors of (e_{ijt}) and vectors of (e_{ijt+y}) or matrix/data frame with (e_{ijt}) and (e_{ijt}) for (T) years, point size (e.g. (e_{ijt+y}))</td>
<td>point size factor, industry names</td>
<td>visible: plot, invisible: growth rates (matrix)</td>
</tr>
<tr>
<td><strong>Growth and specialization portfolio matrix</strong></td>
<td>locq.growth()</td>
<td>vectors of (e_{ijt}) and vectors of (e_{ijt+y}) or matrix/data frame with (e_{ijt}) and (e_{ijt}) for (T) years, point size (e.g. (e_{ijt+y}))</td>
<td>point size factor, industry names</td>
<td>visible: plot, invisible: list with portfolio data (matrix), (LQ_{ijt}) (matrix) and growth rates (matrix)</td>
</tr>
<tr>
<td><strong>Shift-share analysis</strong></td>
<td>shift()</td>
<td>vectors of (e_{ijt}) and (e_{ijt+y}), vectors of (e_{ijt}) and (e_{ijt+y})</td>
<td>shift-share method (default: Dunn), industry names, plot components, plot portfolio</td>
<td>visible: matrix with components, invisible: list with components (matrix), growth (matrix) and shift method (char), optional: plots (matrix)</td>
</tr>
<tr>
<td><strong>dynamic</strong></td>
<td>shift()</td>
<td>vectors of (e_{ijt0}) and (e_{ijt+y}), matrix/data frame with (e_{ijt}) and (e_{ijt}) for (T) years</td>
<td>shift-share method (default: Dunn), industry names, plot components, plot portfolio</td>
<td>visible: matrix with annual components, invisible: list with components (matrix), annual components (matrix), growth (matrix) and shift method (char), optional: plots (matrix)</td>
</tr>
<tr>
<td><strong>industry-specific</strong></td>
<td>shift()</td>
<td>vectors of (e_{ijt}) and (e_{ijt+y}), vectors of (e_{ijt}) and (e_{ijt+y})</td>
<td>shift-share method (default: Dunn), industry names, plot components, plot portfolio</td>
<td>visible: matrix with industry components, invisible: list with components (matrix), industry components (matrix), growth (matrix) and shift method (char), optional: plots (matrix)</td>
</tr>
<tr>
<td><strong>industry-specific and dynamic</strong></td>
<td>shiftid()</td>
<td>vectors of (e_{ijt0}) and (e_{ijt+y}), matrix/data frame with (e_{ijt}) and (e_{ijt}) for (T) years</td>
<td>shift-share method (default: Dunn), industry names, plot components, plot portfolio</td>
<td>visible: matrix with industry components, invisible: list with components (matrix), industry components (matrix), growth (matrix) and shift method (char), optional: plots (matrix)</td>
</tr>
<tr>
<td><strong>prognosis</strong></td>
<td>shiftp()</td>
<td>vectors of (e_{ijt}) and (e_{ijt+y}), vectors of (e_{ijt}) and (e_{ijt+y}), vector of (e_{ijt+y})</td>
<td>industry names, plot</td>
<td>visible: matrix with industry components, invisible: list with industry employment prognosis (matrix), components (matrix), industry components (matrix), growth (matrix) and shift method (char), optional: plots</td>
</tr>
</tbody>
</table>

Source: own compilation.
### Table 15: REAT functions for commercial area prognosis

<table>
<thead>
<tr>
<th>Model</th>
<th>REAT function</th>
<th>Mandatory arguments</th>
<th>Optional arguments</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>GIFPRO</td>
<td>gifpro()</td>
<td>vectors of $e_{ij}$, $a_i$, $sq_{ij}$, $rq_{ij}$ and $ai_{ij}$, time interval, time base</td>
<td>vector of $ru_{ij}$, industry names, type of output</td>
<td>visible: total commercial area and (optional) annual values, invisible: list with components (matrices), annual and all-over results (list with two matrices)</td>
</tr>
<tr>
<td>TBS-GIFPRO</td>
<td>gifpro.tbs()</td>
<td>vectors of $e_{ij}$ for $T$ years, $a_i$, $sq_{ij}$, $rq_{ij}$ and $ai_{ij}$, time interval, time base, trend function types</td>
<td>vector of $ru_{ij}$, industry names, type of output, employment forecast only</td>
<td>visible: total commercial area and (optional) annual values, invisible: list with components (matrices), annual and all-over results (list with two matrices), industry-specific forecast model results (list with $I$ matrices)</td>
</tr>
</tbody>
</table>

Source: own compilation.

All functions for shift-share analysis (except for shift-share prognosis with `shiftp()` provide three variants of calculation of the components: The classical Dunn method (default function argument `shift.method="Dunn"`), the Dunn extension by Esteban-Marquillas (1972) (`shift.method="Esteban"`) producing four components instead of three, and the Gerfin method (`shift.method="Gerfin"`). When calculating a dynamic shift-share analysis, the user must choose the function `shiftd()`. Industry-specific components are returned by the function `shifti()`. With `shiftd()` one can combine both approaches. Here, it is important to recognize that the function structure allows a combination of e.g. industry-specific and dynamic components while calculating the components from the Esteban-Marquillas extension of shift-share analysis. Additionally, the shift-share functions may plot a portfolio matrix (function argument `plot.portfolio = TRUE`), allowing portfolio and shift-share analysis at once.

Both functions for commercial area prognosis (`gifpro()` and `gifpro.tbs()`) require vectors of employment data as well as the coefficients for resettlement etc. When forecasting commercial area using the trend-specific technique with `gifpro.tbs()`, the user needs time series data of previous industry-specific employment and has to specify a trend function type (linear, power, exponential or logistic) for each industry. The “best” function type may be examined visually by regarding the employment forecasting output (optional function argument `prog.plot = TRUE`) and the related $R^2$ values which is part of the invisible function output. Note that this function uses the REAT function `curvefit()`, which is a simple tool for bivariate regression, similar to the curve fitting functions in other spreadsheet or statistics software.

### 6.2.2 Application example 1: Analysis of regional growth in Göttingen

Referring to the example in Section 4.2.2, we perform a regional growth analysis for the German city Göttingen. We use the same dataset `Goettingen` as before, that contains industry-specific employment data for Göttingen and Germany from 2008 to 2017. We load our example data:

```r
data(Goettingen)
```

In the first step, we want to examine the industry-specific growth in Göttingen visually. Using the function `portfolio()`, we plot a regional growth matrix with respect to the 15 industries (rows 2 to 16). We also set a plot title (argument `pctitle`) and axis labels (arguments `pmx` and `pmy`, respectively) as well as industry-specific colors (argument `pcol`):
Similarly, we plot a growth-specialization portfolio matrix using `locq.growth()` with the same options (colors etc.). On the $y$ axis, we put the industry-specific regional growth which is stated by the function argument `y.axis = "r"` (if we would like to see the national growth instead, we had to set `y.axis = "n"`; for the quotient of regional and national growth, use `y.axis = "rn"`):

```r
locq.growth (Goettingen$Goettingen2008[2:16],
Goettingen$Goettingen2017[2:16],
Goettingen$BRD2008[2:16], Goettingen$BRD2017[2:16],
psize = Goettingen$Goettingen2017[2:16], psize.factor = 15,
ptitle = "Growth and specialization in Göttingen",
industry.names = Goettingen$WZ2008_Code[2:16],
pmx = "Growth Göttingen 2008-2017 [%]",
pmy = "Growth Germany 2008-2017 [%]",
pcol.border = "grey",
pcol = c("darkgreen", "powderblue", "chocolate", "darkred",
"orange", "cadetblue1", "chartreuse1", "red", "coral",
"coral4", "cyan", "darkcyan", "yellow", "green", "deeppink"),
leg = TRUE, leg.x = 0.1)
```

The resulting growth portfolio matrix is shown in Figure 6a, the growth-specialization portfolio in Figure 6b. The size of the points (or bubbles) is equal to the current industry-specific employment ($e_{ij}$) for 2017 (rows 2 to 16 of column `Goettingen2017` in the example data), normalized with respect to a maximum point size of 15 (argument `psize.factor = 15`). As we can see, the health sector (industry code Q, green bubble) has the highest absolute relevance, which can be attributed to the local university hospital (see Section 4.2.2). The axes in the growth portfolio are segmented at $x = 0$ and $y = 0$, respectively, which means a differentiation between positive and negative growth.
As we can see, most industries have grown from 2008 to 2017 in both the region and the whole economy (see quadrant I) with similar growth rates. There is one outlier: Industry R (arts, entertainment, and recreation) shows a regional growth of more than 75 percent, while the national growth is about 10 percent. Note that we see percentage growth rates from 2008 to 2017 here (if average growth rates are desired, use the function argument `time.periods`).

Looking at the growth-specialization portfolio, we can identify absolute relevance and growth rate as well as regional specialization of the industries (The colors and bubble sizes are equal to those in Figure 6a). In quadrant I, we find the industries which are overrepresented in Göttingen (specialization) and growing at this regional level. As expected in this university city and related to our results in Section 4.2.2, the “stars” in Göttingen are education (code P), health (code Q) and professional, scientific and technical services (code M).

While the portfolio matrix analysis tells us about the industry-specific growth, the shift-share analysis decomposes this growth into the national, industrial and regional components. In the first step, we perform a static shift-share analysis in the sense of Dunn Jr. (1960) for the same data as in the portfolio analysis by applying the function `shift()`:

```r
shift(Goettingen$Goettingen2008[2:16], Goettingen$Goettingen2017[2:16],
Goettingen$BRD2008[2:16], Goettingen$BRD2017[2:16])
```

This is our (visible) output:

**Shift-Share Analysis**

**Method:** Dunn

**Shift-share components**

<table>
<thead>
<tr>
<th>Components</th>
<th>Growth (t1-t)</th>
<th>National share</th>
<th>Industrial mix</th>
<th>Regional share</th>
<th>Net total shift</th>
</tr>
</thead>
<tbody>
<tr>
<td>Growth (t1-t)</td>
<td>10411.0000</td>
<td>9178.1916</td>
<td>2204.8202</td>
<td>-972.0118</td>
<td>1232.8084</td>
</tr>
<tr>
<td>National share</td>
<td>9178.1916</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Industrial mix</td>
<td>2204.8202</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Regional share</td>
<td>-972.0118</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Net total shift</td>
<td>1232.8084</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Calculation for 15 industries**

- **Regional employment at time t:** 56872, at time t+1:
  - 67283 (10411 / 18.30602 %)
- **National employment at time t:** 27695398, at time t+1:
  - 32164973 (4469575 / 16.13833 %)

In this cross-sectional analysis, we see that the overall employment in Göttingen increased by 10,411 persons from 2008 to 2017. However, a large share of this growth is due to the growth in the national economy \((n_{j,t+1} \approx 9,178 \text{ employees})\), which is only a bit lower than Göttingen. The industrial mix component \((m_{j,t+1})\) shows that approximately 2,205 additional employees must be attributed to an overrepresentation of growing industries in Göttingen. The regional share is negative \((c_{j,t+1} \approx -972)\), which indicates locational disadvantages. When interpreting the industrial mix also as a regional aspect (which seems plausible), we can look at the sum of the industrial mix and the regional share: The net total shift \((t_{t+1})\) is equal to 1,233 employees, representing the growth difference between the region and the whole economy.

We confirm our results using the Gerfin technique. We request it by setting the argument `shift.method` of the `shift()` function equal to "Gerfin":

```r
shift(Goettingen$Goettingen2008[2:16], Goettingen$Goettingen2017[2:16],
Goettingen$BRD2008[2:16], Goettingen$BRD2017[2:16],
shift.method = "Gerfin")
```
The output is:

Shift-Share Analysis
Method: Gerfin

Shift-share components

Components
Industrial mix 1.0333810
Regional share 0.9857591
Net total shift 1.0186647

Calculation for 15 industries
Regional employment at time t: 56872, at time t+1: 67283 (10411 / 18.30602 %)
National employment at time t: 27695398, at time t+1: 32164973 (4469575 / 16.13833 %)

In the index method, there is no national share component (implicitly, it is equal to one), thus, we only take a look at the industrial mix and the regional share as well as the net total shift. The industrial mix component is above one \( \left(r_{jt,ty} \approx 1.03\right) \), showing a more advantageous sector structure in Göttingen compared to Germany. While the regional share in the Dunn-type shift-share analysis was negative, this component in the Gerfin analysis is slightly below one \( \left(c_{jt,ty} \approx 0.99\right) \), indicating locational disadvantages.

These traditional techniques only regard the overall growth with respect to cross-sectional data. To gain a deeper insight and take into account also seasonal effects, we perform a dynamic shift-share analysis in the sense of Barff, Knight (1988) which distinguishes between the 15 industries simultaneously. This can be done via the REAT function `shiftid()`, requiring data for the initial time period and at least for two following periods. In the Goettingen dataset, the rows 2 to 16 represent the industries and the columns represent the years (2008 to 2017). Data for the regarded region and the whole economy is arranged successively. We also use the industry codes in column WZ2008_Code. In this function, we have to define the start and end periods explicitly:

```r
```

# columns 3-12: employment in Göttingen 2009-2017
# columns 13-22: employment in Germany 2009-2017

The result is:

Dynamic Shift-Share Analysis
Method: Dunn

Shift-share components

\[
\begin{array}{cccccc}
A & B & C & D & E & F \\
\text{Growth (t1-t)} & -3.000000 & 29.000000 & -1117.0000 & -255.0000 & -51.0000 \\
\text{National share} & 6.103502 & -9.46377 & 254.5217 & 160.0638 & 561.7436 \\
\text{Regional share} & -9.103502 & 38.46377 & -1371.5217 & -415.0638 & -612.7436 \\
\text{Net total shift} & -9.103502 & 38.46377 & -1371.5217 & -415.0638 & -612.7436 \\
\end{array}
\]

\[
\begin{array}{cccccc}
G & H & I & J & K & M \\
\text{Growth (t1-t)} & 524.0000 & 470.00000 & 274.0000 & -465.0000 & 2229.000 \\
\text{National share} & 368.2053 & 515.03493 & 286.3238 & 6.356612 & 1821.392 \\
\text{Regional share} & 155.7947 & -45.03493 & -12.3238 & 654.308 & 315.49647 \\
\text{Net total shift} & 155.7947 & -45.03493 & -12.3238 & 654.308 & 315.49647 \\
\end{array}
\]

\[
\begin{array}{cccccc}
L & M & N & O & P & Q \\
\text{Growth (t1-t)} & 1178.0000 & 268.00000 & 1272.0000 & 4211.0000 & 363.00000 \\
\text{National share} & 977.9869 & 167.9118 & 1138.5383 & 3556.692 & 47.50353 \\
\text{Regional share} & 200.0131 & 100.0882 & 133.4617 & 654.308 & 315.49647 \\
\text{Net total shift} & 200.0131 & 100.0882 & 133.4617 & 654.308 & 315.49647 \\
\end{array}
\]
Calculation for 15 industries
Regional employment at time t: 56872, at time t+1: 67283 (10411 / 18.30602 %)
National employment at time t: 27695398, at time t+1: 32164973 (4469575 / 16.13833 %)

The visible output is a matrix containing one row for each component (the number of components depends on the selected shift-share method, here: Dunn) and I columns (one for each industry). As we calculate industry-specific components, there is no industrial mix effect, which means that the calculations are on the level of single industries. Again, we detect large absolute growth for industries P (education) and Q (health) (see Table 9). Interestingly, this growth can be mainly attributed to effects in the whole economy. The corresponding regional shares are small but positive, showing locational advantages with respect to these industries in Göttingen.

The logic of shift-share analysis can also be regarded in two other examples: If industry C (manufacturing) had developed as in the national trend, the absolute growth in Göttingen would be equal to 255 employees. In fact, there was a decline of 1,117 employees, resulting in a negative regional share of -1,372 employees, indicating locational disadvantages with respect to the manufacturing sector. The opposite is true for the industries with code BDE (including electricity, gas, water supply, etc.): The absolute growth of 29 employees would not have occurred if this sector had developed as in the whole economy (negative national share equal to -9 employees). The residuum (regional share) is equal to 38 employees, indicating a trend contrary to the national.

6.2.3 Application example 2: Commercial area prognosis for Göttingen

Using the same data, we now perform a commercial area prognosis for Göttingen. We load our data:

```r
data(Goettingen)
```

When using the GIFPRO-based commercial area prognosis techniques, several parameters have to be defined (employment shares in commercial areas $a_i$, resettlement rate $sq_{ij}$, relocation rate $rq_{ij}$ and areal index $ai_{ij}$; a reutilization rate $ru_{ij}$ is optional, thus, we ignore the reutilization of commercial area in this example). These parameters have to be defined for each industry. In our example, we use the employment shares as well as the resettlement and relocation rates from Deutsches Institut für Urbanistik GmbH, Spath + Nagel (GbR) (2010). Note that some sectors are, per definition, not located within commercial areas (e.g. agriculture), resulting in an employment share of $a_i = 0$.

As we want to reuse the sets of parameters, we save them as single numeric vectors:

```r
c_a_share <- c(0, 0, 100, 90, 70, 100, 10, 20, 20, 20, 20, 0, 0, 0, 0)
# industry-specific shares of employees in commercial areas
sq_quote <- c(0.77, 0.77, 0.15, 0.15, 0.77, 0.15, 0.77, 0.77, 0.77, 0.77, 0.77, 0.77, 0.77, 0.77, 0.77)
# industry-specific resettlement quote
rq_quote <- rep(0.7, 15)
# industry-specific relocation quote (0.7 for each of the 15 industries)
area_index <- c(0, 0, 200, 75, 250, 250, 50, 100, 100, 100, 100, 50, 50, 50, 50)
# industry-specific area index (sqm commercial area per employee)
```

Now, we compute the traditional commercial area prognosis using the `gifpro()` function and the `Goettingen` data as well as the parameters defined above. We forecast the commercial area for five years ($tinterval = 5$). Our base is 2017 ($time.base = 2017$), as this is the last year empirical data is available for. We save the (invisible) output in the list object `gifpro_goettingen`:

```r
gifpro_goettingen <- gifpro (e_ij = Goettingen$Goettingen2017[2:16],
a_i = c_a_share, sq_ij = sq_quote, rq_ij = rq_quote, tinterval = 5,
a_i_j = area_index, time.base = 2017,
industry.names = Goettingen$WZ2008_Code[2:16], output = "full")
```
As we have set `output = "full"`, the visible function output contains overall as well as annual values:

GIFPRO
Method: GIFPRO

Employment and commercial area changes (allover)

<table>
<thead>
<tr>
<th>Employment</th>
<th>Commercial Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum</td>
<td>1113.8785</td>
</tr>
<tr>
<td>Average</td>
<td>222.7757</td>
</tr>
</tbody>
</table>

Employment and commercial area changes (per time unit)

<table>
<thead>
<tr>
<th>Employment</th>
<th>Commercial Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018</td>
<td>222.7757</td>
</tr>
<tr>
<td>2019</td>
<td>222.7757</td>
</tr>
<tr>
<td>2020</td>
<td>222.7757</td>
</tr>
<tr>
<td>2021</td>
<td>222.7757</td>
</tr>
<tr>
<td>2022</td>
<td>222.7757</td>
</tr>
</tbody>
</table>

Calculation for 15 industries

In all 15 industries, 1,114 new employees are predicted for the year 2022, resulting in 212,928 square meters required for new commercial area. As the employment prognosis is not based on (nonlinear) trend regression but on constant growth, the absolute employment growth and the required commercial area are equal in each year (223 employees and 42,596 sqm, respectively).

The object `gifpro_goettingen` contains a list called `components` containing the single components of prognosis as well as the results already shown in the visible output (`results`). To understand the GIFPRO technique and the related `REAT` function, we take a look at the single components:

`gifpro_goettingen$components`

$resettlement

<table>
<thead>
<tr>
<th>A</th>
<th>0.00000</th>
<th>0.00000</th>
<th>0.00000</th>
<th>0.00000</th>
<th>0.00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>BDE</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>C</td>
<td>11.81100</td>
<td>11.81100</td>
<td>11.81100</td>
<td>11.81100</td>
<td>11.81100</td>
</tr>
<tr>
<td>F</td>
<td>1.80090</td>
<td>1.80090</td>
<td>1.80090</td>
<td>1.80090</td>
<td>1.80090</td>
</tr>
<tr>
<td>G</td>
<td>38.00489</td>
<td>38.00489</td>
<td>38.00489</td>
<td>38.00489</td>
<td>38.00489</td>
</tr>
<tr>
<td>H</td>
<td>3.72150</td>
<td>3.72150</td>
<td>3.72150</td>
<td>3.72150</td>
<td>3.72150</td>
</tr>
<tr>
<td>I</td>
<td>1.73327</td>
<td>1.73327</td>
<td>1.73327</td>
<td>1.73327</td>
<td>1.73327</td>
</tr>
<tr>
<td>J</td>
<td>3.12928</td>
<td>3.12928</td>
<td>3.12928</td>
<td>3.12928</td>
<td>3.12928</td>
</tr>
<tr>
<td>K</td>
<td>2.67806</td>
<td>2.67806</td>
<td>2.67806</td>
<td>2.67806</td>
<td>2.67806</td>
</tr>
<tr>
<td>N</td>
<td>7.50750</td>
<td>7.50750</td>
<td>7.50750</td>
<td>7.50750</td>
<td>7.50750</td>
</tr>
<tr>
<td>O</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>P</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>Q</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>R</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
</tbody>
</table>

$relocation

<table>
<thead>
<tr>
<th>A</th>
<th>0.00000</th>
<th>0.00000</th>
<th>0.00000</th>
<th>0.00000</th>
<th>0.00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>BDE</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
<td>0.00000</td>
</tr>
<tr>
<td>C</td>
<td>55.11800</td>
<td>55.11800</td>
<td>55.11800</td>
<td>55.11800</td>
<td>55.11800</td>
</tr>
<tr>
<td>F</td>
<td>8.40420</td>
<td>8.40420</td>
<td>8.40420</td>
<td>8.40420</td>
<td>8.40420</td>
</tr>
<tr>
<td>G</td>
<td>34.54999</td>
<td>34.54999</td>
<td>34.54999</td>
<td>34.54999</td>
<td>34.54999</td>
</tr>
<tr>
<td>H</td>
<td>17.36700</td>
<td>17.36700</td>
<td>17.36700</td>
<td>17.36700</td>
<td>17.36700</td>
</tr>
<tr>
<td>I</td>
<td>1.57570</td>
<td>1.57570</td>
<td>1.57570</td>
<td>1.57570</td>
<td>1.57570</td>
</tr>
<tr>
<td>J</td>
<td>2.84480</td>
<td>2.84480</td>
<td>2.84480</td>
<td>2.84480</td>
<td>2.84480</td>
</tr>
<tr>
<td>K</td>
<td>2.43460</td>
<td>2.43460</td>
<td>2.43460</td>
<td>2.43460</td>
<td>2.43460</td>
</tr>
</tbody>
</table>
As we defined some industries as not relevant for commercial areas \( (a_i = 0) \), they do not contribute any employees neither resettled nor relocated (such as A - agriculture, B - mining and quarrying or R - arts, entertainment, and recreation). We see that e.g. in the manufacturing sector (code C), there is an annual increase of about 12 employees attributed to resettlement and 55 employees related to relocation each year (see row 3 in resettlement and relocation, respectively). As we ignored the reutilization of commercial area, the matrix containing the commercial area-relevant employment related to reutilization (reuse) contains only zeros. The sum of all three components is stored in the fourth matrix, employment. There is an annual increase of nearly 67 employees in the manufacturing sector. The contents of the results list is the same as shown in the visible output.

In the next step, we apply the trend-based commercial area prognosis (TBS-GIFPRO) to the Goettingen data. In the gifpro.tbs() function, we use the employment data from 2008 to 2017 (columns 3 to 12), and assume an exponential function for employment prognosis (function argument prog.func, repeating the argument "exp" for each industry). The employment prognosis is plotted \( (\text{prog.plot} = \text{TRUE}) \), showing all 15 plots in one \( (\text{plot.single} = \text{FALSE}) \):
Figure 7: Employment prognosis for 15 industries in Goettingen (TBS-GIFPRO)

```r
gifpro.tbs (e_ij = Goettingen[2:16,3:12],
a_i = ca_share, sq_ij = sq_quote, rq_ij = rq_quote, tinterval = 5,
prog.func = rep("exp", nrow(Goettingen[2:16,3:12])),
ai_ij = area_index, time.base = 2008,
industry.names = Goettingen$WZ2008_Code[2:16],
prog.plot = TRUE, plot.single = FALSE, output = "full")
```

The visible function output is similar to the output above:

**GIFPRO**

**Method:** TBS-GIFPRO

Employment and commercial area changes (allover)

<table>
<thead>
<tr>
<th>Employment</th>
<th>Commercial Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum</td>
<td>1139.6592</td>
</tr>
<tr>
<td>Average</td>
<td>227.9318</td>
</tr>
</tbody>
</table>

Employment and commercial area changes (per time unit)

<table>
<thead>
<tr>
<th></th>
<th>Employment</th>
<th>Commercial Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018</td>
<td>224.9565</td>
<td>42945.53</td>
</tr>
<tr>
<td>2019</td>
<td>226.2904</td>
<td>43054.76</td>
</tr>
<tr>
<td>2020</td>
<td>227.7755</td>
<td>43182.97</td>
</tr>
<tr>
<td>2021</td>
<td>229.4169</td>
<td>43330.70</td>
</tr>
<tr>
<td>2022</td>
<td>231.2199</td>
<td>43498.50</td>
</tr>
</tbody>
</table>

Calculation for 10 industries

The resulting plot containing the employment forecasting functions is shown in Figure 7. The black vertical lines divide the plots into the estimation segment (2008 to 2017) and the prognosis segment (2018 to 2022). Four function types are supplied: linear (blue), power (green), exponential (yellow) and logistic (red). Note that a linear trend seems unrealistic as it implies continuous growth and may result in negative employment if the slope is negative. At this point, we should normally discuss and find the “best”
forecasting model for each industry and rerun our analysis a few times. In our example, we skip this step and just take a look at the prognosis functions: In most cases, an exponential growth (or decline) seems to be an appropriate approximation. The power functions (green lines) are nearly invisible as their data fit is nearly the same as that of the exponential functions. Thus, we could choose them instead. In our case, the exponential function seems sufficient.

As expected, a nonlinear industry growth results in a nonlinear overall employment growth and, consequently, the commercial area-relevant employment also grows in a nonlinear way. As we can see from the `gifpro.tbs()` output, employment increases by about 228 employees per year on average and by about 1,140 employees over the five years regarded (2018 to 2022). The annual commercial area required ranges from 42,946 sqm (2018) to 43,499 sqm (2022), all in all 216,012 sqm up to 2022. In our case, the estimated commercial area exceeds the prognosis derived from the simple GIFPRO analysis, which can be attributed to the positive differences between the exponential prognosis and a linear prognosis (see Figure 7). We skip the inspections of the components, which could be addressed by saving the results in an object (`list`), as we did in the first GIFPRO example.

7 Final remarks

This paper has shown how R and specifically the package `REAT` can be used for regional economic analysis. It should be noted that this package aims at width with respect to the treated analysis subjects rather than depth. The subsections provide the basic analysis methods regarded as most important from the package developer’s point of view (with respect to usage in current papers and discussion in current textbooks as well as application in own research projects), while there are several other approaches as well as extensions of the basic methods. A more detailed survey of the common methods can be found in the cited literature, especially in review articles (e.g. Nakamura, Morrison Paul 2009; Portnov, Felsenstein 2010) and textbooks (e.g. Farhauer, Kröll 2014).

Finally, we have to keep in mind that this package (like nearly any other free software) was developed in a non-commercial context (and published under the GNU General Public License). All functions have been tested several times using various real data and single functions have already been used in a few research projects. However, there is no warranty that all functions always work perfectly. Like nearly any other R package, `REAT` is continuously refined, which means extending functions as well as correcting errors. This requires attentive usage and, of course, constructive feedback from the package users. It can be easily transmitted using the contact information on the CRAN package website.
References


*REGION*: Volume 6, Number 3, 2019


Herfindahl OC (1950) Concentration in the U.S. Steel Industry. Colombia University Press


Myrdal G (1957) *Economic theory and under-developed regions*. G. Duckworth


R Core Team (2018b) The R Manuals. Manual. [https://cran.r-project.org/manuals.html](https://cran.r-project.org/manuals.html)


Schönebeck C (1996) Wirtschaftsstruktur und Regionalentwicklung: theoretische und empirische Befunde für die Bundesrepublik Deutschland, Volume 75 of Dortmunder Beiträge zur Raumplanung Blaue Reihe. IRPUD


*REGION*: Volume 6, Number 3, 2019


