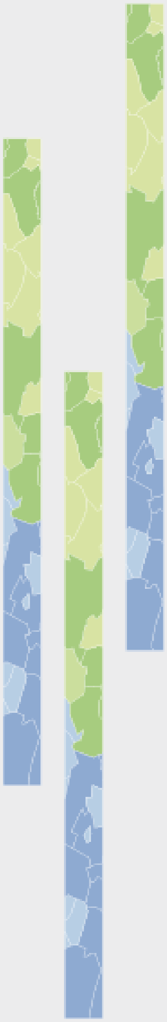
TECHNICAL REPORT



SPATIAL ANALYSIS FOR QUALITY CONTROL

Phase 1: The identification of logical input errors and statistical outliers

MAIN RESULTS

- Exceptional values can arise from logical input errors and true outlying data.
- The accurate identification of an exceptional value is important as input errors should be treated differently to true outlying data.
- Input errors can usually be identified mathematically or sometimes, statistically. Outliers are identified statistically.
- Techniques to statistically identify outliers are presented using worked examples that have been coded with R open source software.

ESPON 2013 DATABASE

MARCH 2011

EUROPEAN UNION
Part-financed by the European Regional Development Fund
INVESTING IN YOUR FUTURE

81 pages

LIST OF AUTHORS

Paul Harris, National Centre for Geocomputation (NCG), National University of Ireland (Maynooth)

Martin Charlton, National Centre for Geocomputation (NCG), National University of Ireland (Maynooth)

Contact

Paul.Harris@nuim.ie

martin.charlton@nuim.ie

tel. + 353-(0)1-7086208

TABLE OF CONTENT

Introduction

The ESPON 2013 Database should be as free from errors as possible. It follows from this that detecting errors is an important activity in both data entry and data checking. This technical report is to examine how mathematical, statistical and spatial analysis tools can be applied to the ESPON 2013 Database in order to find 'logical input errors' and 'statistical outliers'. In both cases, 'exceptional values' can arise but it is not always clear if such values relate to input errors or true values that are statistically-outlying. In this respect, reliably determining the nature of an exceptional value is important, especially as input errors should be treated differently to statistical outliers. For example, input errors are usually corrected or removed, whilst suspected outliers are usually flagged for further scrutiny.

The outcome of this report is a targeted review of existing outlier-detection tools in the field of statistics, data mining and spatial analysis, and an examination how they can assist in the detection of errors/outliers in the ESPON 2013 Database for improved quality control. This methodological review has a clear focus on spatial analysis with respect to outlier-detection; and is complemented by worked examples on an ESPON-type data set, where chosen techniques are demonstrated. Worked examples are coded using open-source software so that the applied techniques are easily transferable. The list of techniques that are applied should not be considered as exhaustive, but form a cross-section of useful techniques which are appropriate for ESPON 2013 Database.

A related aim of this report is to examine the effects of the Modifiable Areal Unit Problem (MAUP) with respect to error/outlier identification. This follows previous research by NCG for the ESPON 2006 project on this topic (ESPON 2006).

1 Exceptional values: types and identification

1.1 Logical input errors

Logical input errors can arise for a number of reasons. For example, the wrong NUTS[1] code could be specified; incorrect data values could be input; data could be repeated exactly but assigned to different variables; data could be displaced within or between columns; data could be swapped within or between columns. In general, the identification of an input error will follow some logical, mathematical approach. For example, if a land use class could only take a positive integer value from 1 to 9 say, then an input error of say, -2, 4.5 or 10 would be easily identified.

An input error may also be identified statistically. For example, if the number 27 is inadvertently entered as 72 for a region's unemployment rate, the value 72 may lie in the extreme tail of this variable's distribution and as such, is statistically-outlying. A difficulty here would be to distinguish between an input error of 72 and a true value of 72.

In this respect, when dealing with errors/outliers, most input errors can be either be corrected or removed, whilst most outliers should be flagged as: (i) suspected outliers and (ii) potential (undetected) input errors. Flagged observations would then require further scrutiny, which should ascertain whether the observation should be: (a) replaced; (b) removed; or if specifically an outlier, (c) retained or possibly down-weighted in some way (so as to provide some robust model fit or statistic of the data).

1.2 Aspatial statistical outliers: identification in univariate to multivariate data sets

A simple, graphical tool for the detection of outliers in univariate data sets is the boxplot (e.g. Frigge et al. 1989). Central to the creation of the boxplot is the inter-quartile range (Q3-Q1) around the median value Q2. Commonly, at the upper end of the distribution, the *inner fence* is defined as the value given by Q2+1.5(Q3-Q1) and the *outer fence* as the value given by Q2+3(Q3-Q1); and there are corresponding values for the lower end of the distribution. Observations whose values lie between the inner and outer fences are usually referred to as *outside* and those whose values lie beyond the outer fence are usually referred to as *far out*. In either case, such observations can be flagged as outlying, however most attention should be placed on observations that lie beyond the outer fence. In this report, we not only demonstrate the use of the standard boxplot but also an adjusted boxplot for skewed distributions (Hubert and Vandervieren 2008). For bivariate data sets, a simple extension of the boxplot, the bagplot (Rousseeuw et al. 1999) can be constructed.

[1] NUTS stands for "nomenclature of territorial units for statistics".

To detect outliers in multivariate data sets, we first demonstrate a technique where outliers are observations that have a *large* squared Mahalanobis Distance (MD^2), where the MD itself is estimated in a robust manner (Filzmoser et al. 2005). MDs are used as they take into account the covariance matrix from which the shape and size of the multivariate data set can be quantified. In this outlier detection technique, robust MD^2 values are related to some pre-determined (upper) quantile of a chi-square distribution (e.g. the 97.5th percentile), where *large* robust MD^2 values lie above this pre-determined threshold. Furthermore, to address subjectivity in choosing the threshold, the technique automatically adjusts the pre-determined threshold (downwards or upwards) via simulation reflecting specific properties of the sample data. The technique (called here RMD2-AQ-outlier) is applied incorporating useful graphical displays of suspected outliers.

We also demonstrate two further multivariate techniques that each use principal component analysis (PCA) to reduce the dimensions of the multivariate data set, where in the resultant transformed space, outliers may be more readily observable. Of the many PCA-based techniques for outlier detection that have been proposed (e.g. see Rousseeuw et al. 2006; Daszykowski et al. 2007; Filzmoser et al. 2008), we demonstrate: (a) the 'sign' approach of Locantore et al. (1999) (call this technique, PCA-outlier-1) and (b) the 'PCOut' approach of Filzmoser et al. (2008) (call this technique, PCA-outlier-2). Both techniques are computationally fast and thus suited to large, high dimensional data sets (see the comparisons given in Filzmoser et al. 2008).

1.3 Spatial statistical outliers: identification in univariate data sets

Commonly outlier detection techniques ignore any spatial element to the data. Data not observed as an outlier when an *aspatial* technique is used, may nevertheless be a *spatial* outlier. Therefore it is important to consider spatial aspects if false negatives (i.e. outliers undetected by an aspatial technique) are to be avoided. In this respect, we demonstrate a technique of Hawkins (1980) to detect spatial outliers in univariate data sets[2]. This technique has much in common with the more recent techniques of Lui et al (2001); Kou et al. (2005).

For this technique, all observations $z(\mathbf{x}_i)$ are suspected a priori as spatial outliers, where $z(\mathbf{x}_i)$ is a spatial outlier if

$$\left\{N\left(z(\mathbf{x}_i)-m_l\right)^2\right\}/\left\{(N+1)\bar{s}_l^{\,2}\right\} \geq \chi^2_{crit-1} \qquad (1)$$

Here, $i=1,\ldots,n$; $\mathbf{x}$ is spatial location; N is the number of neighbouring values of $z(\mathbf{x}_i)$; m_l is the local mean; $\bar{s}_l^{\,2}$ is the average variance for equivalently sized neighbourhoods across the sample area (i.e. the average local variance) and χ^2_{crit-1} is

[2] We only present a technique to identify spatial outliers in a univariate sense. Extensions to bivariate and multivariate spatial data sets are not considered here. However our current research in this area concerns the development of geographically weighted PCA techniques with respect to outlier identification (see Charlton et al. 2010), which should allow the identification of multivariate spatial outliers in the ESPON database.

a critical value of the chi-squared distribution for 1 degree of freedom. As there is no objective function for cross-validation, then neighbourhood definitions (for the local mean and variances) are chosen subjectively for this test statistic. In this report, the local mean and variances are found using a geographically weighted approach (see sections 2.4 and 2.5), with 95%, 99% and 99.9% critical levels chosen as appropriate cut-offs.

1.4 The use of statistical models and residual data in outlier identification

In a statistical analysis, it is common to identify outliers via large (positive or negative) prediction errors (or residuals) from some predictive model fit. Observations that are poorly predicted produce large residuals when compared with the actual data, and are therefore deemed as outlying. The key drawback to this approach is the need to specify a model in the first place, where different models may produce different outlying observations. However if several prediction models are applied, then it is reasonable to expect that the most influential outlying observations should be repeatedly identified.

In this respect, we first identify outliers (in a univariate sense) simply using the key component of expression 1, where a spatial outlier relates to a large (absolute) value of the error $z(x_i) - m_l$. Here our prediction model is simply the one chosen to find the local mean m_l, which in this case is some simple spatial predictor using geographical weights (which we shall call the local mean predictor, LM). The widely-used inverse distance weighting model would be one example of such an LM model.

Furthermore, we also identify outliers (via residual data) using univariate and multivariate regressions in both aspatial and spatial forms. In particular we apply: (a) standard multiple linear regression (MLR) models, (b) attribute-space local regression (LR) models (see Loader 2004) and (c) geographic-space local regression models (Fotheringham et al. 2002) (i.e. geographically weighted regression, GWR). Here LR accounts for nonstationarity and nonlinearity in attribute-space, whilst GWR accounts for nonstationary and nonlinearity in geographic-space. Both LR and GWR are nonparametric in design. The conventional MLR model assumes stationarity and linearity in both attribute- and geographic-space; and is parametric in design. Consequently, each of the three regression forms will identify outliers (or possibly groups of outliers, see section 2.5) according to their particular specification (or set of modelling assumptions).

The investigation of residual data plays a central role in the formulation of a robust regression model, where the influence of outlying data on the regression fit is reduced (e.g. see Faraway 2004, p98-106; Cruz Ortiz et al. 2006). MLR, LR (see Loader 2004) and GWR (Fotheringham et al. 2002, p73-82; Harris et al. 2010) all have robust forms. Commonly, a robust regression will identify outliers as observations with large standardised (or studentised) residuals via a leave-one-out approach. However, in this report we only identify outliers simply, via the raw residuals and without the benefit of a leave-one-out fit.

1.5 The identification of spatial clusters

A group of observations identified as outliers may actually be spatially clustered with a substantive reason for their 'unusualness' (i.e. false positives are to be avoided as well). In this respect, it is worthwhile applying techniques that identify local (or regional) changes in the spatial process according to some key moment or relationship[3].

Furthermore, seemingly significant clusters can be sometimes be attributable to only a few (influential and outlying) observations; so although the local techniques described below are not specifically designed to identify spatial outliers, they sometimes do so. Indeed, a corresponding robust form of the given local technique would out of necessity identify spatial outliers in order to reduce their influence.

Thus in the first instance, local summary univariate and bivariate statistics are calculated and investigated. In particular, we assess changes in the mean, standard deviation and correlation across space, where these (spatial) moments are all found in a geographically weighted form (Fotheringham et al. 2002)[4]. For the multivariate case, GWR can be applied, which complements a local correlation analysis when investigating relationship-change across space.

From a spatial autocorrelation viewpoint, a local version of Moran's I (Anselin 1995) is used. Positive spatial autocorrelation exists when neighbouring spatial units tend to have similar values of a variable; whilst negative spatial autocorrelation exists when they do not. Local Moran's I is only used to investigate univariate data, but the statistic could be adapted to investigate cross-autocorrelation in bivariate and multivariate data sets.

1.6 Summary: MAUP, temporal outliers and data imputation

We have presented a typology of techniques where variables are analysed singly or in combination; and aspatially or spatially. Underlying all of these techniques is the spatial structure of the reporting units, where results can be influenced not only by the level of spatial aggregation used but also by the spatial configuration of the reporting units (i.e. a MAUP; e.g. see Wong 1996). In this report we demonstrate the consequences of the MAUP for outlier identification via a worked example, where outlier-detection techniques are applied at different NUTS levels (NUTS level 3 through to NUTS level 0).

We have not addressed the identification of temporal (or by extension, spatio-temporal) outliers. This is not an oversight, as ESPON time series data is not expected to be of a sufficient length for an outlier detection technique to be reliably

[3] Brunsdon and Charlton (2010) assess the effectiveness of multiple hypothesis testing for detecting clusters of geographical anomalies. These tests would complement the techniques demonstrated from this section of the report.

[4] Robust forms of geographically weighted summary statistics (GWSS) can be found in Brunsdon et al. (2002) and in Harris and Brunsdon (2010).

applied. Instead it should suffice that the aspatial/spatial detection methods demonstrated here can be repeated at different time intervals. The consequences of the reporting units changing over time (i.e. another MAUP) are addressed elsewhere in ESPON 2013 database project.

As already discussed, once an input error has been identified the observation can either be corrected or removed (i.e. replaced with the missing value notation, NA[5]). On the other hand, suspected outliers (which may be an input error) can (after some additional scrutiny) be: (a) replaced; (b) removed (i.e. replaced by NA); or if indeed an outlier, (c) retained or possibly down-weighted in some way. This entails that some form of imputation or prediction of missing valued data will be required, and here the chosen regression models of section 2.4 may be of value.

1.7 Further reading

This report provides a brief overview to subject of error or outlier identification with respect to the task of identifying outliers in the ESPON 2013 Database. There is an extensive literature on outlier detection, where the following reading list may be useful.

- An evaluation of aspatial techniques to detect input errors and true outliers (here known as data editing), together with imputation techniques, for large scale survey data can be found in Charlton (2004). This and related articles arose from the EUREDIT project[6]. Related articles include: an outlier identification technique for multivariate data by Béguin and Hulliger (2004); a robust regression technique for data edits by Chambers et al. (2004); and a classification and regression tree technique for data edits by Petrakos et al. (2004).
- An aspatial Bayesian technique that both edits and imputes data in a multivariate context can be found in Ghosh-Dastidar and Schafer (2003).
- Reviews of aspatial outlier identification techniques from univariate to multivariate data sets can be found in Reimann et al. (2005); Rousseeuw et al. (2006); Daszykowski et al. (2007); Morgenthaler (2007).
- Further aspatial outlier identification techniques for multivariate data sets can found in Hoo et al. (2002); Jackson and Chen (2004), where the former article also imputes data.
- Imputation (aspatial) techniques can be found in Plaia and Bondi (2006); Vanden Branden and Verboven (2009), where the former article focuses on time series data.
- Alternative spatial outlier identification techniques can be found in D'Alimonte and Cornford (2007); Ainsworth and Dean (2008); Meiklit et al. (2009).

[5] NA is the missing data indicator used in the R statistical computing package (see section 4).

[6] See http://www.cs.york.ac.uk/euredit/. The project website was still active as of 1/12/09.

2 Data for worked examples

In the worked examples, NUTS3 level data are used. Here 1351 values (with two missing) for the variable 'evolution of gross domestic product (GDP) from the years 2000 to 2005' at NUTS2006 divisions are related to sixteen contextual variables at NUTS1999 divisions (with a maximum of 1329 values for each contextual variable). As the NUTS2006 spatial units can differ to the NUTS1999 spatial units, this combining of data results in at least 438 (1351 minus 913) missing values for each contextual variable (i.e. NUTS2006 and NUTS1999 divisions have 913 reporting units in common). Thus in summary, NUTS3 level data using the NUTS2006 divisions are the spatial units that are retained.

2.1 The full data set

The 'evolution of GDP' variable is named EVOGDP_2000_2005_2006, where the first two numbers (2000 and 2005) relate to the collection time (i.e. year) of the data and the last number (2006) relates to the NUTS division or version. Similar naming conventions were used for all other variables. EVOGDP_2000_2005_2006 is itself calculated from four stock variables which are presented in Table 1, together with the formula for calculating EVOGDP_2000_2005_2006.

The sixteen contextual variables are presented in Tables 2 to 6. These variables were selected from the basic and project indicator files posted on the ESPON website[7]. Contextual data include: two spatial typology variables, one unemployment variable, six land use variables, one natural hazards variable and six regional policy variables. In total, the full data set consists of twenty-three variables (plus the coordinates/centroids of each region).

Observe that as variables were collected over different time periods (from 1996 to 2005) this data set is purely used to demonstrate the outlier identification techniques of section 2 via the worked examples in section 4. It is essentially a fabricated data set and as such, all analytical results need to be interpreted with this in mind.

However the contextual variables were selected in expectation that if all variables were relatable (i.e. collected over the same period), then this particular set of contextual variables may help explain variation in EVOGDP_2000_2005_2006 (see sections 2.4, 4.5 and 4.6).

[7] See http://www.espon.eu/mmp/online/website/content/tools/832/850/588_EN.html and

http://www.espon.eu/mmp/online/website/content/tools/832/873/605_EN.html

Variable type	Variable name	Indicator	Year	Unit
STOCK (1)	GDP_2000_2006	Gross Domestic Product	2000	Million Euros
STOCK (2)	GDP_2005_2006	Gross Domestic Product	2005	Million Euros
STOCK (3)	POP_T_2000_2006	Total population (annual average)	2000	Thousands inhabit.
STOCK (4)	POP_T_2005_2006	Total population (annual average)	2005	Thousands inhabit.
RATIO (5)	GDP_POP_2000_2006	GDP per inhabit. = $\left[(1)/(3)\right]\times 1000$	2000	Euros
RATIO (6)	GDP_POP_2005_2006	GDP per inhabit. = $\left[(2)/(4)\right]\times 1000$	2005	Euros
RATIO	EVOGDP_2000_2005_2006	Evolution of GDP = $\left[(6)/(5)\right]\times 100$	2000-2005	Percentage

***Table 1**: Description of the EVOGDP_2000_2005_2006 variable*

Theme	**Spatial typology**	**Spatial typology**

Indicator	Typology Settlement Structure (nine basic types defined by population density and situation regarding centres) –
	1: city core region;
	2: very densely populated;
	3: densely populated;
	4: rural region;
	5: city core region;
	6: densely populated region;
	7: rural region;
	8: more densely populated region;
	9: less densely populated region
		Urban-rural typology (six basic types) –
		1: High urban influence, high human intervention;
		2: High urban influence, medium human intervention;
		3: High urban influence, low human intervention;
		4: low urban influence, high human intervention;
		5: Low urban influence, medium human intervention;
		6: Low urban influence, low human intervention

Original variable name	Settyp99N3	URTypN3
New variable name	SPAT_TYPE_1_1999_1999	SPAT_TYPE_2_1999_1999
Min. possible	1	1
Max. possible	9	6
Unit or variable type	CLASS	CLASS

***Table 2**: Descriptions of spatial typology contextual variables*

Theme	**Unemployment**	**Land use**	**Land use**	**Land use**
Indicator	Unemployment rate	Share of artificial surfaces	Artificial surfaces per 1000 inhabitants	Artificial surfaces per GDP
Original variable name	UNRT01N3	ArSu96N3	ArSc96N3	ArSg96N3
New variable name	UNEMP_R_2001_1999	LU_AS_1_1996_1999	LU_AS_2_1996_1999	LU_AS_3_1996_1999
Min. possible	0	0	0	0
Max. possible	100	100	100	100
Unit or variable type	PERCENTAGE	PERCENTAGE	PERCENTAGE	PERCENTAGE

***Table 3**: Descriptions of unemployment and three land use contextual variables*

Theme	**Land use**	**Land use**	**Land use**	**Environment - Hazards**
Indicator	Share of urban fabric	Share of arable land	Share of permanent crops	Sum of all weighted hazard values
Original variable name	UFL296N3	ALL296N3	PCL296N3	smwh04
New variable name	LU_UF_1996_1999	LU_AR_1996_1999	LU_PC_1996_1999	NAT_HAZ_2004_1999
Min. possible	0	0	0	10
Max. possible	100	100	100	INFINITY
Unit or variable type	PERCENTAGE	PERCENTAGE	PERCENTAGE	INTEGER

***Table 4**: Descriptions of three land use and an environmental hazards contextual variable*

Theme **Indicator**	**Regional policy** All Structural & Cohesion Fund expenditure	**Regional policy** Structural Fund expenditure related to Regional Development & Productive Infrastructure	**Regional policy** Structural Fund expenditure related to Social Integration & Human Resources
Original variable name	SFT99N3	SFR99N3	SFS99N3
New variable name	SF_CF_1999_1999	SF_R_1999_1999	SF_S_1999_1999
Min. possible	0	0	0
Max. possible	INFINITY	INFINITY	INFINITY
Unit or variable type	REAL NUMBER	REAL NUMBER	REAL NUMBER

***Table 5**: Descriptions of three regional policy contextual variables*

Theme **Indicator**	**Regional policy** Structural Fund expenditure related to Agriculture, Rural Development & Fishery	**Regional policy** Cohesion Fund expenditure related to Transport	**Regional policy** Cohesion Fund expenditure related to Environment
Original variable name	SFA99N3	SFCT99N3	SFCE99N3
New variable name	SF_A_1999_1999	CF_T_1999_1999	CF_E_1999_1999
Min. possible	0	0	0
Max. possible	INFINITY	INFINITY	INFINITY
Unit or variable type	REAL NUMBER	REAL NUMBER	REAL NUMBER

***Table 6**: Descriptions of three regional policy contextual variables*

2.2 Data subsets and analytical objectives

Subsets of the full data set are analysed in two basic forms: (a) in their original state and (b) in a state where some commonly encountered logical input errors are deliberately introduced. Here Tables 7 and 8 summarise how variable subsets of the full data set are used in each of six worked examples presented in section 4.

Worked example	**1**	**2**	**3**
Variables investigated	NUTS3 code, GDP_2000_2006, GDP_2005_2006, POP_T_2000_2006 & POP_T_2005_2006	EVOGDP_2000_2005_2006 (plus the coordinate data)	Some subset of EVOGDP_2000_2005_2006, its 16 contextual variables & the coordinate data
Introduced input errors?	Yes	No	No
Identification type: logical or statistical or both	Both	Statistical	Statistical
Identification type: univariate or multivariate or both	Univariate	Univariate	Multivariate
Identification type: aspatial or spatial or both	Aspatial	Both	Aspatial
Key statistical identification techniques applied	Boxplots	Boxplots; Hawkins test; residuals from LM, MLR, LR & GWR fits	Bagplots; Robust MD^2 analysis (RMD2-AQ-outlier); & PCA for outliers (PCA-outlier-1 & PCA-outlier-2)
Analysis objective	Identify logical input errors so that EVOGDP_2000_2005_2006 can be investigated for statistical outliers	Identify statistical outliers	Identify statistical outliers

***Table 7**: Data subsets and analytical objectives for worked examples 1 to 3*

Worked example	4	5	6
Variables investigated	EVOGDP_2000_2005_2006 in relation to some subset of its 16 contextual variables and the coordinate data	EVOGDP_2000_2005_2006 in relation to some subset of its 16 contextual variables and the coordinate data	EVOGDP_2000_2005_2006 (plus the coordinate data)
Introduced input errors?	No	No	No
Identification type: logical or statistical or both	Statistical	Statistical	Statistical
Identification type: univariate or multivariate or both	Multivariate	Both	Univariate
Identification type: aspatial or spatial or both	Both	Spatial	Both
Key statistical identification techniques applied	Residuals from MLR, LR & GWR fits	Data exploration with GWSS, GWR & local Moran's I	Boxplots; Hawkins test; residuals from LM, MLR, LR & GWR fits
Analysis objective	Identify statistical outliers	Identify statistical clusters	Investigate the consequences of MAUP with respect to outlier identification

***Table 8**: Data subsets and analytical objectives for worked examples 4 to 6*

It is envisaged that when identifying exceptional values in an ESPON database data set, a first pass should identify input errors using both mathematical and statistical techniques. That is the first pass screens the data. Identified input errors should then be corrected (and as such, a revised data set can be *assumed* input error-free) before a second pass is undertaken that only uses the statistical techniques to identify outlying observations. It is essential that two passes are conducted otherwise the detection of true outliers will be compromised by input errors.

Thus from Tables 7 and 8, worked example 1 relates to a first pass (for input errors) before its corresponding second pass (for outliers), which is (effectively) worked example 2. For worked examples 2, 3, 4, 5 and 6, it should be assumed that this data has already been screened for input errors. Observe that worked example 6 is the same as worked example 2, but applied at different NUTS levels (i.e. spatial scales) to investigate the effects of MAUP with respect to outlier identification.

2.3 A data subset with deliberate logical input errors

We now present a list of logical input error-types that have been introduced to: (a) the NUTS3 codes; and (b) the variables GDP_2000_2006, GDP_2005_2006, POP_T_2000_2006 and POP_T_2005_2006 (i.e. only those variables used in the calculation of EVOGDP_2000_2005_2006). This list of input errors is given with appropriate solutions (i.e. for worked example 1).

This list is not exhaustive, and should grow as different input error-types become apparent (i.e. at this stage, we are not expected to foresee all input error possibilities). The spatial location of the input errors is depicted in Fig. 1. Consequences of input errors for the correct calculation of EVOGDP_2000_2005_2006 are depicted in Fig. 2.

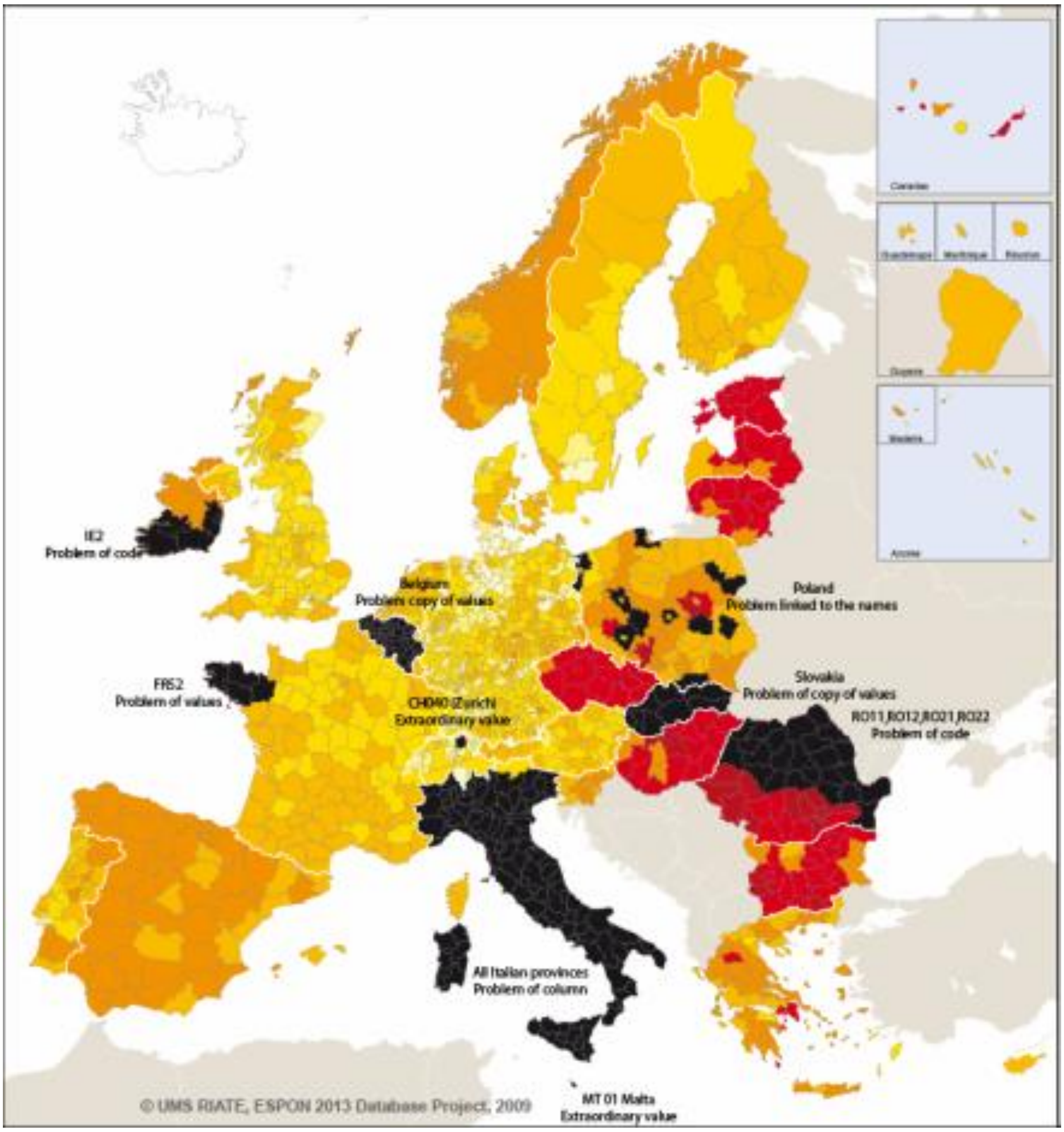


***Figure 1**: Location of input errors (in black) overlaid on the true EVOGDP_2000_2005_2006 data* *(see Fig. 2)*

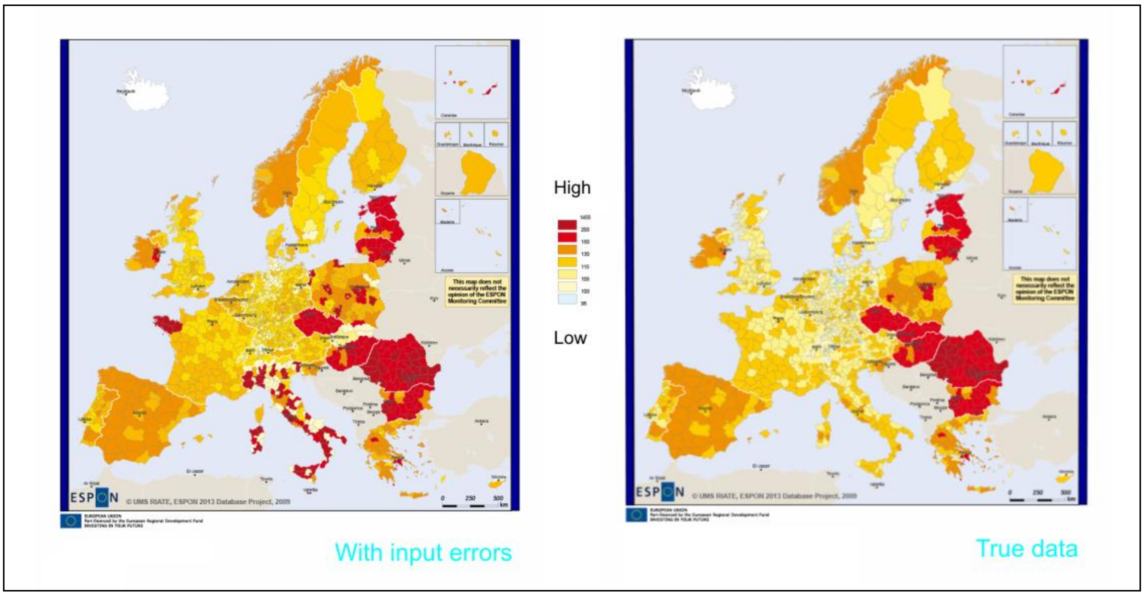

High
Low
ESPON
© UMS RIATE, ESPON 2013 Database Project, 2009
This map does not necessarily reflect the opinion of the ESPON Monitoring Committee
With input errors
True data

***Figure 2**: Maps of EVOGDP_2000_2005_2006 with and without input errors*

Problems with NUTS code (29 input errors)

Input error-type 1: For Ireland, 5 wrong codes have been input at NUTS3 level. In the NUTS hierarchy, this does not imply changes at NUTS2 level (see Fig. 3a). Solution: codes can be checked by a simple relationship to the correct NUTS name and code pairs.

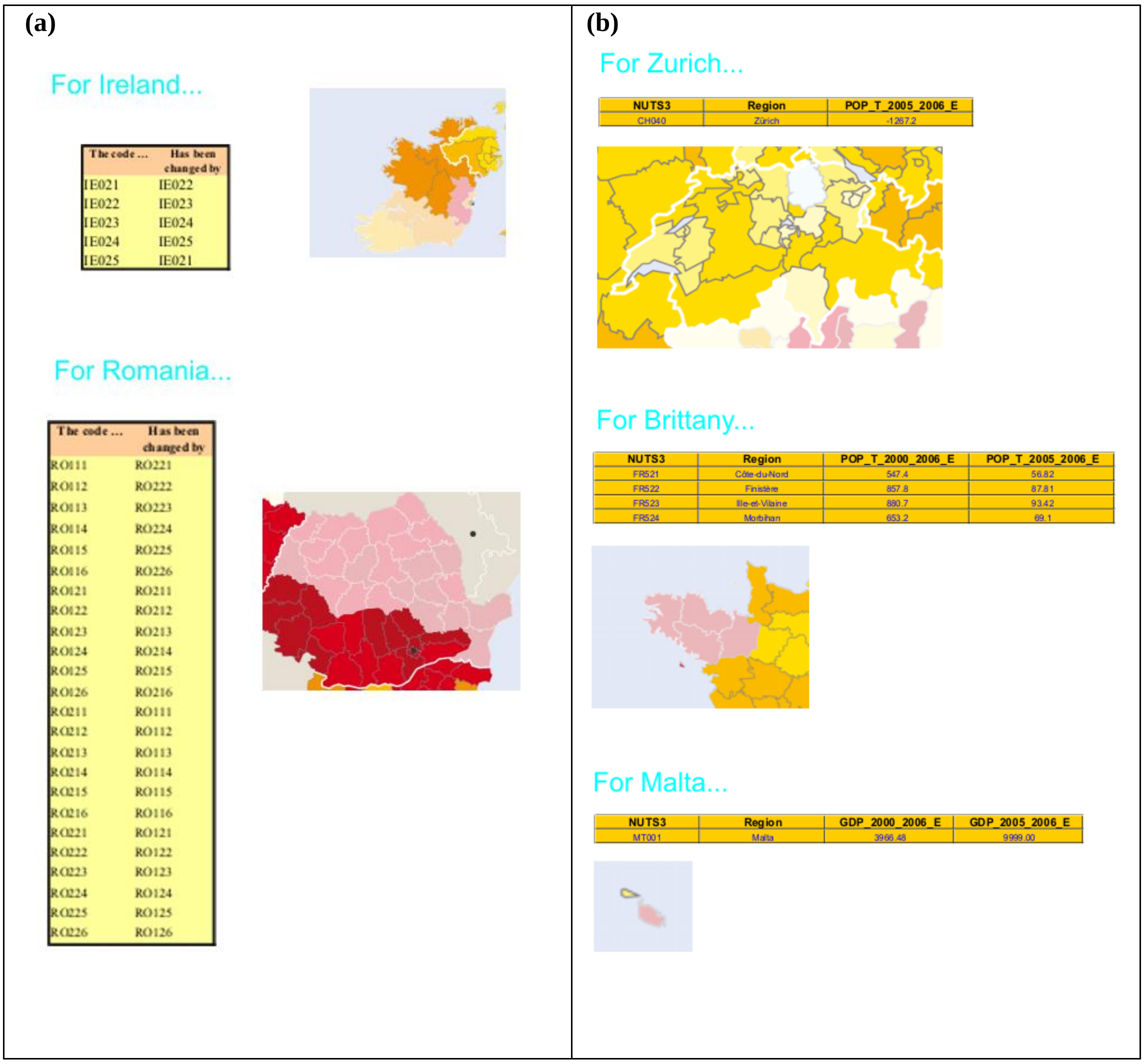
Input error-type 2: For Romania, 24 wrong codes have been input at NUTS3 level. In the hierarchy, this does imply changes at NUTS2 level (see Fig. 3a). Solution: codes can be checked by a simple relationship to the correct NUTS name and code pairs.

Problems with values (6 input errors)

Input error-type 3: For Zürich (NUTS3 - CH040), the total population in 2005 (POP_T_2005_2006) has been multiplied by -1 (see Fig. 3b). This value is impossible for this variable and as such, should be easily identified.

Input error-type 4: For Brittany (NUTS2 - FR52), the total population in 2005 (POP_T_2005_2006) has been divided by 10 at the NUTS3 level (all 4 of them, see Fig. 3b). These values are possible, but should be easily identified by a simple subtraction of both population variables (POP_T_2005_2006 minus POP_T_2000_2006) and looking for unusually large (negative) differences. Large (negative) differences could be identified as statistically outlying (which upon further scrutiny would indicate *potential* input errors).

Input error-type 5: For Malta (NUTS3 - MT001), the total GDP in 2005 (GDP_2005_2006) has been incorrectly entered as 9999. 9999 is sometimes used to denote a missing value (see Fig. 3b). This value is possible, but should be identified when all *potential* missing values (e.g. values of -99, -999, -9999, 99, 999, 9999, etc.) are identified for further scrutiny.



(a)

For Ireland...

The code ...	Has been changed by
IE021	IE022
IE022	IE023
IE023	IE024
IE024	IE025
IE025	IE021

For Romania...

The code ...	Has been changed by
RO111	RO221
RO112	RO222
RO113	RO223
RO114	RO224
RO115	RO225
RO116	RO226
RO121	RO211
RO122	RO212
RO123	RO213
RO124	RO214
RO125	RO215
RO126	RO216
RO211	RO111
RO212	RO112
RO213	RO113
RO214	RO114
RO215	RO115
RO216	RO116
RO221	RO121
RO222	RO122
RO223	RO123
RO224	RO124
RO225	RO125
RO226	RO126

(b)

For Zurich...

NUTS3	Region	POP_T_2005_2006_E
CH040	Zürich	-1267.2

For Brittany...

NUTS3	Region	POP_T_2000_2006_E	POP_T_2005_2006_E
FR521	Côte-du-Nord	547.4	56.82
FR522	Finistère	857.8	87.81
FR523	Ille-et-Vilaine	880.7	93.42
FR524	Morbihan	653.2	69.1

For Malta...

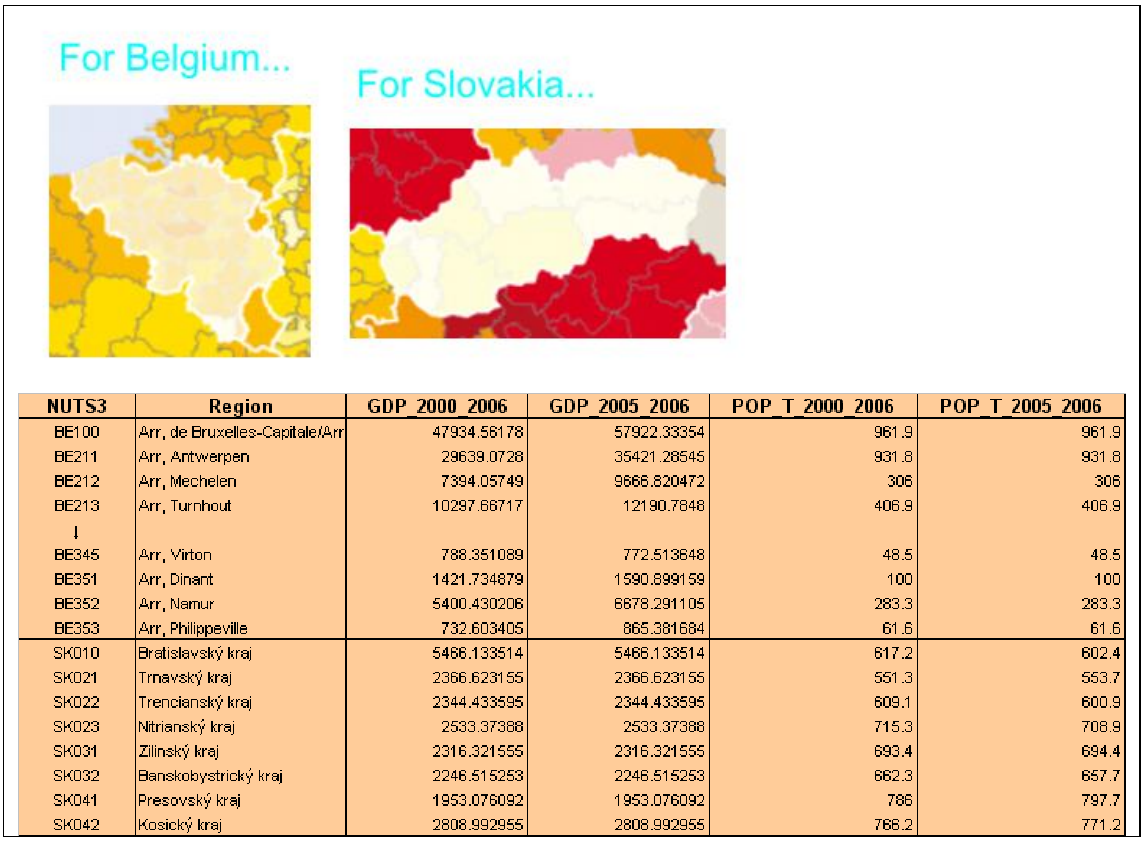
NUTS3	Region	GDP_2000_2006_E	GDP_2005_2006_E
MT001	Malta	3966.48	9999.00

Figure 3 (a): *incorrect NUTS code entries (b) incorrect value entries*

Problems with copied or repeated data (52 input errors)

Input error-type 6: For Belgium (44 entries), the total population in 2000 (POP_T_2000_2006) is repeated exactly for the total population in 2005 (POP_T_2005_2006) (all at NUTS3 level, see Fig. 4). These values are possible, but should be easily identified by a simple subtraction of the two population variables and looking for (exact) zero values. It can be assumed that equal populations for the two years are highly unlikely. Also observe that differences of zero are unlikely to be statistically outlying. A difficulty here would be to decide whether the values for POP_T_2000_2006 or the values for POP_T_2005_2006 were inputted incorrectly. This would require further scrutiny.

Input error-type 6: For Slovakia (8 entries), the total GDP in 2000 (GDP_2000_2006) is repeated exactly for the total GPD in 2005 (GDP_2005_2006) (all at NUTS3 level, see Fig. 4). These values are possible, but again should be easily identified by a simple subtraction of the two GDP variables and looking for zero values. As with the population data, it can be assumed that equal GDP data for the two years is highly unlikely. Again, it is difficult to know whether the values for GDP_2000_2006 or the values for GDP_2005_2006 were inputted incorrectly.



For Belgium...

For Slovakia...

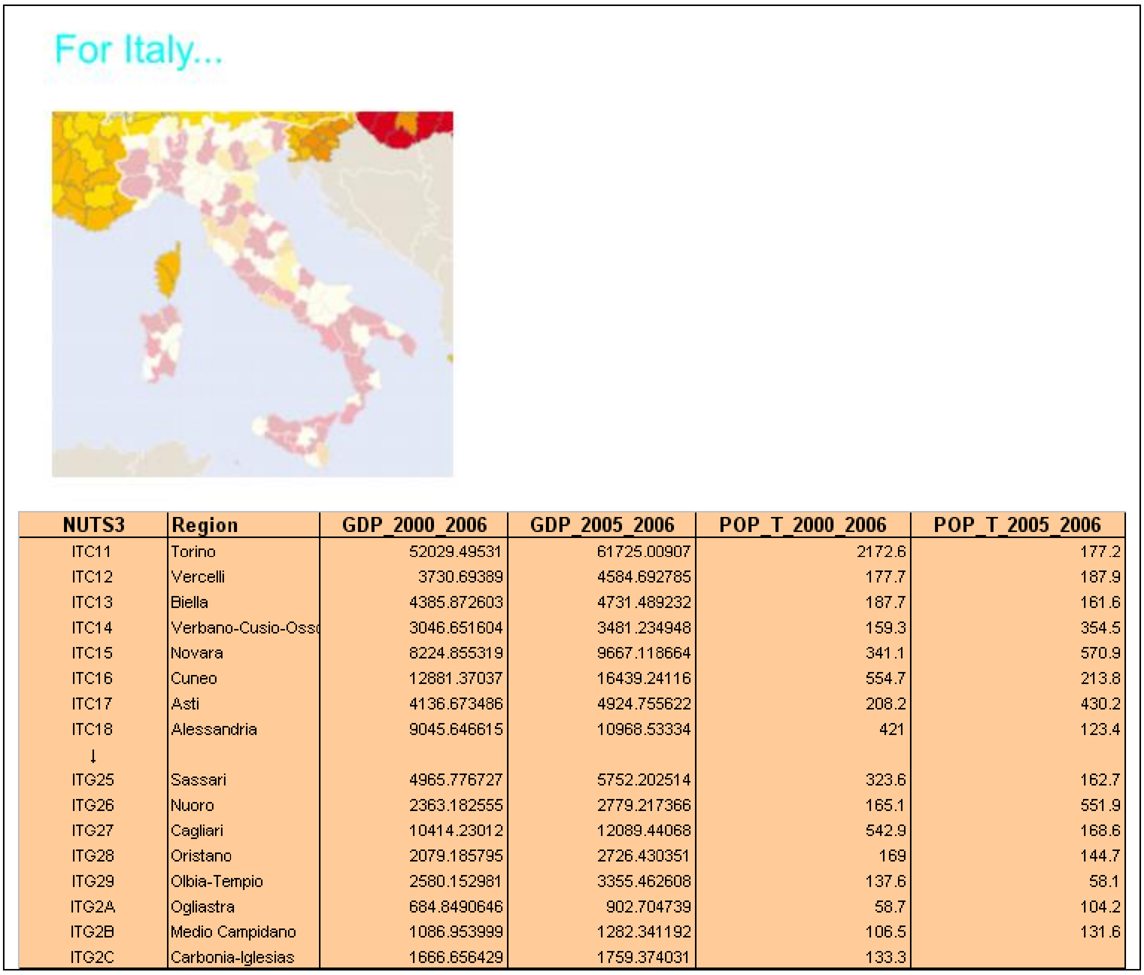
NUTS3	Region	GDP_2000_2006	GDP_2005_2006	POP_T_2000_2006	POP_T_2005_2006
BE100	Arr, de Bruxelles-Capitale/Arr	47934.56178	57922.33354	961.9	961.9
BE211	Arr, Antwerpen	29639.0728	35421.28545	931.8	931.8
BE212	Arr, Mechelen	7394.05749	9666.820472	306	306
BE213	Arr, Turnhout	10297.66717	12190.7848	406.9	406.9
↓					
BE345	Arr, Virton	788.351089	772.513648	48.5	48.5
BE351	Arr, Dinant	1421.734879	1590.899159	100	100
BE352	Arr, Namur	5400.430206	6678.291105	283.3	283.3
BE353	Arr, Philippeville	732.603405	865.381684	61.6	61.6
SK010	Bratislavský kraj	5466.133514	5466.133514	617.2	602.4
SK021	Trnavský kraj	2366.623155	2366.623155	551.3	553.7
SK022	Trenciansky kraj	2344.433595	2344.433595	609.1	600.9
SK023	Nitriansky kraj	2533.37388	2533.37388	715.3	708.9
SK031	Zilinský kraj	2316.321555	2316.321555	693.4	694.4
SK032	Banskobystrický kraj	2246.515253	2246.515253	662.3	657.7
SK041	Presovský kraj	1953.076092	1953.076092	786	797.7
SK042	Kosický kraj	2808.992955	2808.992955	766.2	771.2

Figure 4: *Problems of copied or repeated data*

Shift in data values (up one or down one line in its data column) (107 input errors)

Input error-type 7: For Italy (107 entries), the total population in 2005 (POP_T_2005_2006) has been shifted up by one line (all at NUTS3 level, see Fig. 5). These values are possible, but most values (not all) should be statistically identified as *potential* input errors. Again a subtraction of the two population variables should for most cases, produce unusually large positive or unusually large negative values which should give rise to suspicion.

Observe that this error-type has created an extra missing value (NUTS3 - ITG2C) and one value has effectively been lost (NUTS3 - ITC11).



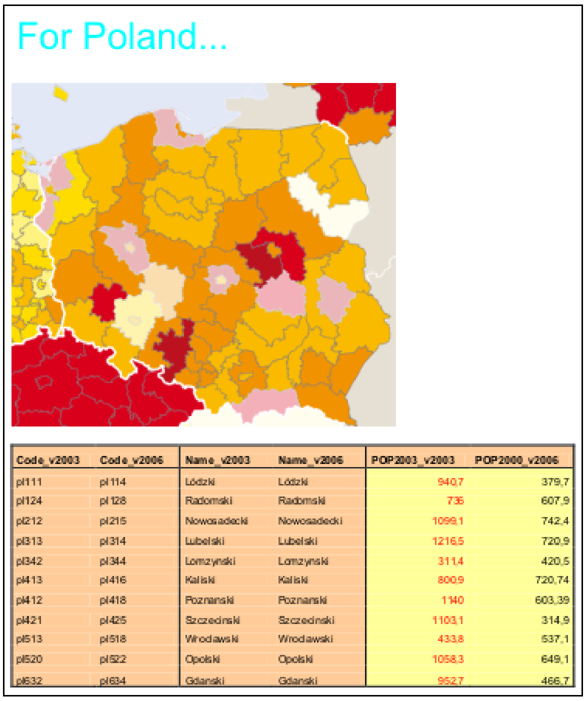
For Italy...

NUTS3	Region	GDP_2000_2006	GDP_2005_2006	POP_T_2000_2006	POP_T_2005_2006
ITC11	Torino	52029.49531	61725.00907	2172.6	177.2
ITC12	Vercelli	3730.69389	4584.692785	177.7	187.9
ITC13	Biella	4385.872603	4731.489232	187.7	161.6
ITC14	Verbano-Cusio-Oss	3046.651604	3481.234948	159.3	354.5
ITC15	Novara	8224.855319	9667.118664	341.1	570.9
ITC16	Cuneo	12881.37037	16439.24116	554.7	213.8
ITC17	Asti	4136.673486	4924.755622	208.2	430.2
ITC18	Alessandria	9045.646615	10968.53334	421	123.4
↓					
ITG25	Sassari	4965.776727	5752.202514	323.6	162.7
ITG26	Nuoro	2363.182555	2779.217366	165.1	551.9
ITG27	Cagliari	10414.23012	12089.44068	542.9	168.6
ITG28	Oristano	2079.185795	2726.430351	169	144.7
ITG29	Olbia-Tempio	2580.152981	3355.462608	137.6	58.1
ITG2A	Ogliastra	684.8490646	902.704739	58.7	104.2
ITG2B	Medio Campidano	1086.953999	1282.341192	106.5	131.6
ITG2C	Carbonia-Iglesias	1666.656429	1759.374031	133.3	

Figure 5: Problems of shifted data

Problems with NUTS codes and names (11 input errors)

Input error-type 8: For some regions of Poland (11 entries), the total population in 2000 (POP_T_2000_2006) has been estimated by the total population in 2003 using NUTS2003 divisions (i.e. POP_T_2003_2003). Such estimations are fine provided the NUTS3 codes are used to relate the regions and not the region names. In this case, the region names have been erroneously used (see Fig. 6). Here the region names have not changed but the geometries for the regions have (and thus the sensible use of different NUTS codes for such instances). The resultant (erroneous) population values are all possible, and in this case, may not be easily identified. They may be identified by a subtraction of POP_T_2005_2006 from POP_T_2000_2006 provided the subtraction happens to result in large outlying differences.



For Poland...

Code_v2003	Code_v2006	Name_v2003	Name_v2006	POP2003_v2003	POP2000_v2006
pl111	pl114	Lódzki	Lódzki	940,7	379,7
pl124	pl128	Radomski	Radomski	736	607,9
pl212	pl215	Nowosadecki	Nowosadecki	1099,1	742,4
pl313	pl314	Lubelski	Lubelski	1216,5	720,9
pl342	pl344	Lomzynski	Lomzynski	311,4	420,5
pl413	pl416	Kaliski	Kaliski	800,9	720,74
pl412	pl418	Poznanski	Poznanski	1140	603,39
pl421	pl425	Szczedinski	Szczedinski	1103,1	314,9
pl513	pl518	Wroclawski	Wroclawski	433,8	537,1
pl520	pl522	Opolski	Opolski	1058,3	649,1
pl632	pl634	Gdanski	Gdanski	952,7	466,7

***Figure 6**: Problems with NUTS codes and names.*

3 Worked examples: commented R scripts and results

3.1 The R statistical environment

All worked examples are coded in the R statistical computing environment (Ihaka and Gentleman 1996), which is open source[8]. In particular we use version 2.9.0 of the base system. For each worked example, only contributed packages are used (i.e. can be downloaded from the R website) except for a useful R mapping package, GISTools (Brunsdon pers. comm.), which is currently under development and will be made available on the R website shortly. The (unsupported) version of GISTools used here (version 0.5-4) is posted on ESPON 2013 database extranet, together with all other relevant materials that are needed to repeat each worked example. The GISTools package is not essential for outlier detection and maps could be constructed using other R packages or outside of R in a GIS.

3.2 Worked example 1: univariate & residual analyses for input errors & outliers

The R script for worked example 1 is given in Appendix 1. The results are summarised in Fig. 7, where the rates of false negatives, false positives and overall misclassification with respect to input error identification were found to be 13.2%, 2.0% and 3.7% respectively. These rates are promisingly low, where their existence, is in part, a reflection of the automated nature of the identification procedures undertaken. Rates should tend to zero upon further (manual) scrutiny of the input errors that only have the *potential* to be so. For example, it would be expected that the rate of false negatives would reduce upon a manual scrutiny of the data in Italy, where the shift in data values (input error-type 7) should be quickly identified.

Observe also that there are many instances of false positives in Spain. This may reflect (unknown) input errors that were already present in this data set (i.e. before our deliberate introduction of input errors) or may reflect true (but unusual) data values (i.e. outliers). Either way, the corresponding data should be scrutinised and checked.

[8] The R website is http://www.r-project.org/

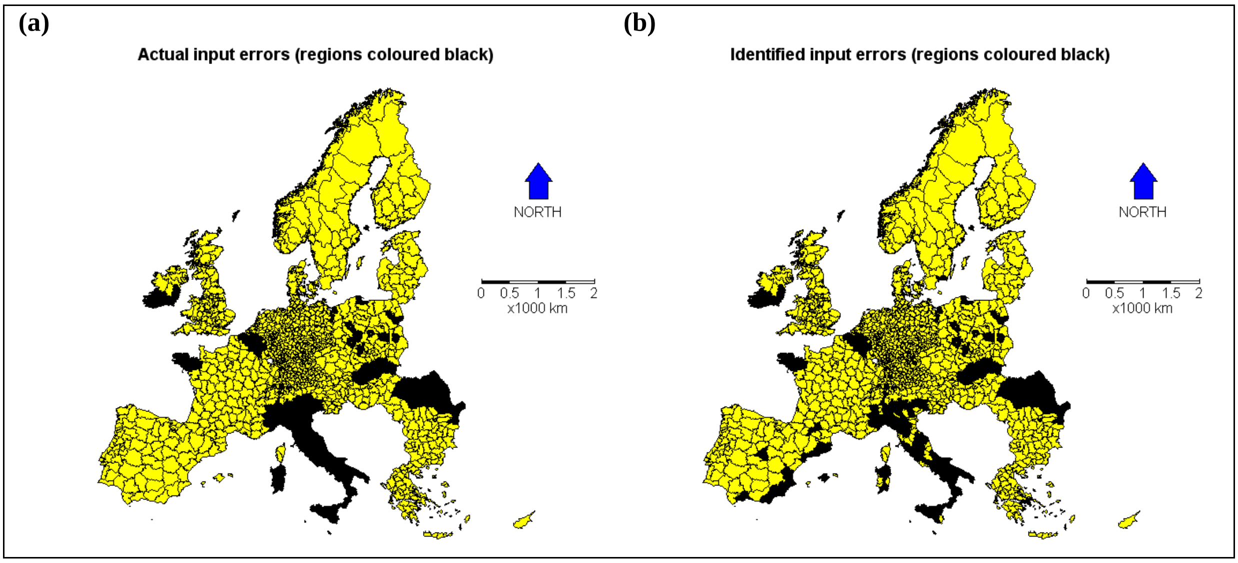

(a)
Actual input errors (regions coloured black)
NORTH
0 0.5 1 1.5 2
x1000 km
(b)
Identified input errors (regions coloured black)
NORTH
0 0.5 1 1.5 2
x1000 km

***Figure 7**: Location of (a) true (deliberate) input errors versus (b) identified input errors. Rates of false negatives, false positives and overall misclassification are 13.2%, 2.0% and 3.7% respectively*

3.3 Worked example 2: univariate & residual analyses for outliers

The R script for worked example 2 is given in Appendix 2. The results are summarised in Fig. 8, where the spatial distribution of EVOGDP_2000_2005_2006 is compared with the spatial distribution of (suspected) outliers for this variable. In total, seven indicators were used to gauge whether or not an observation is outlying: (1) standard boxplot statistics; (2) adjusted boxplot statistics; (3) Hawkins' test for spatial outliers; and (4 to 7) large (absolute and raw) residuals from LM/MLR/LR/GWR fits (each calibrated with the coordinate data). These indicators are summarised in Fig. 8b, where a strong case for an outlier relates to an observation that has positive results for all seven outlier identification analyses. It appears that the most outlying EVOGDP_2000_2005_2006 observations lie in the south-east of the ESPON region.

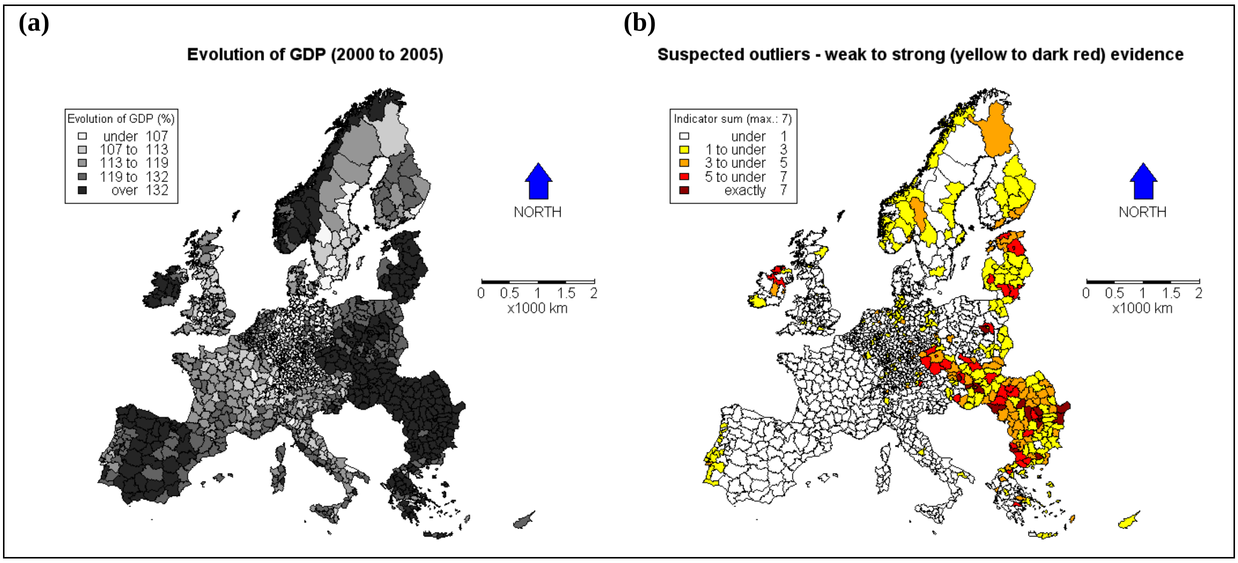


***Figure 8**: Spatial distribution of (a) EVOGDP_2000_2005_2006 and (b) suspected outliers for EVOGDP_2000_2005_2006 (seven univariate indicators)*

3.4 Worked example 3: multivariate analyses for outliers

The R script for worked example 3 is given in Appendix 3. The results are summarised in Fig. 9, where only a much reduced data set of 731 regions could be used in this set of analyses (a consequence of a considerable amount of missing data). In Fig. 9, potential outliers are found according: (a) a bagplot of EVOGDP_2000_2005_2006 with UNEMP_R_2001_1999; (b) the technique, RMD2-AQ-outlier; (c) the technique, PCA-outlier-1; and (d) the technique, PCA-outlier-2. The bagplot identifies outliers in a bivariate-sense whilst the other three techniques identify outliers in a multivariate-sense (in this case, with respect to outlying or unusual relationships amongst EVOGDP_2000_2005_2006, UNEMP_R_2001_1999 NAT_HAZ_2004_1999 and SF_CF_1999_1999). Clearly, the bivariate approach is missing key information. The three multivariate approaches give broadly similar results, but where the relatively large number of potential outliers (as many as 201 observations using RMD2-AQ-outlier) suggests multiple (statistical) populations rather than one population with many outlying observations. This would require further investigation.

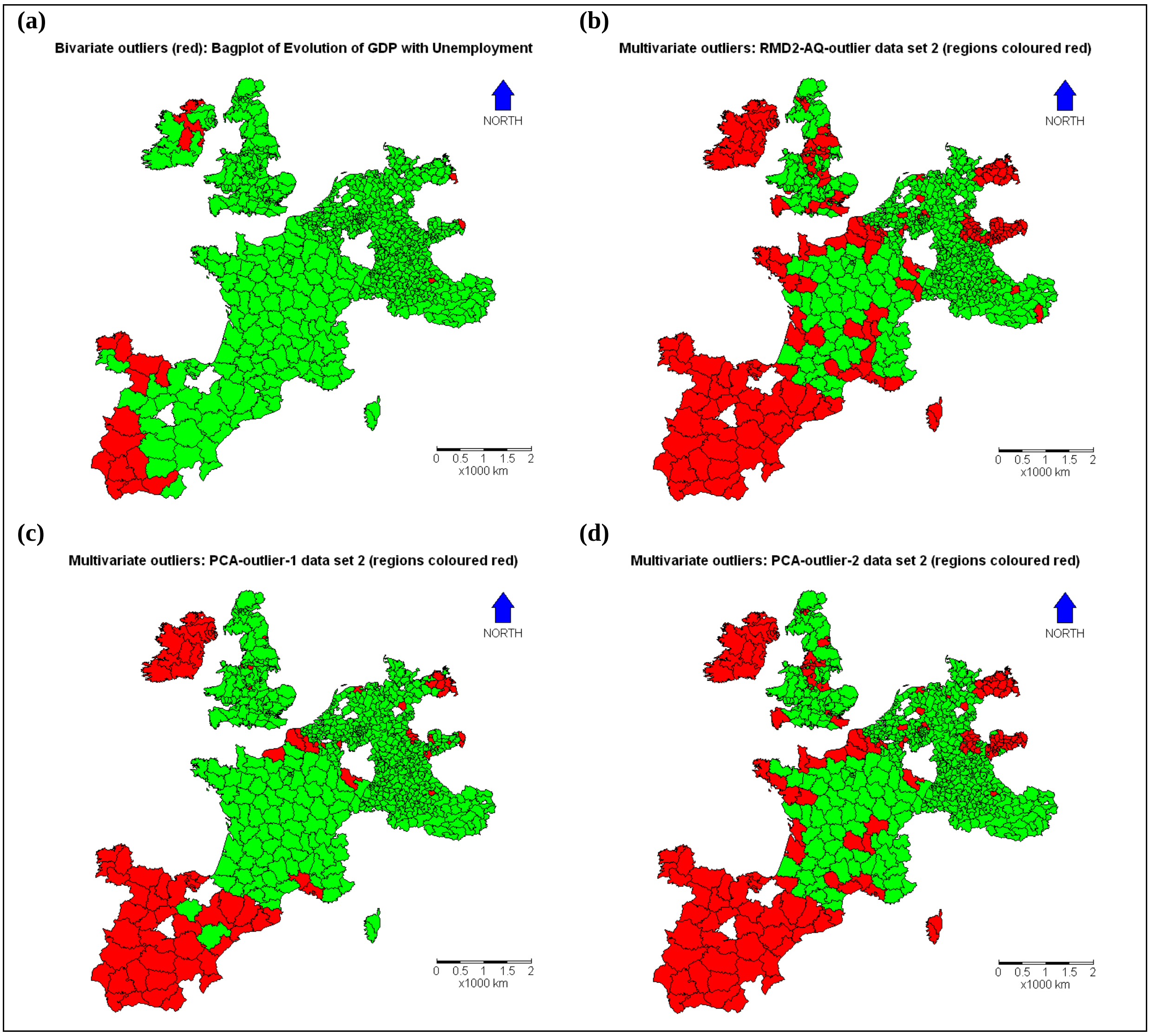

(a)
Bivariate outliers (red): Bagplot of Evolution of GDP with Unemployment
NORTH
0 0.5 1 1.5 2
x1000 km
(b)
Multivariate outliers: RMD2-AQ-outlier data set 2 (regions coloured red)
NORTH
0 0.5 1 1.5 2
x1000 km
(c)
Multivariate outliers: PCA-outlier-1 data set 2 (regions coloured red)
NORTH
0 0.5 1 1.5 2
x1000 km
(d)
Multivariate outliers: PCA-outlier-2 data set 2 (regions coloured red)
NORTH
0 0.5 1 1.5 2
x1000 km

***Figure 9**: Spatial distribution of suspected (bivariate and multivariate) outliers according to (a) a bagplot of EVOGDP_2000_2005_2006 with UNEMP_R_2001_1999; (b) the RMD2-AQ-outlier technique; (c) the PCA-outlier-1 technique; and (d) the PCA-outlier-2 technique*

3.5 Worked example 4: multivariate residual analyses for outliers

The R script for worked example 4 is given in Appendix 4. The results are summarised in Fig. 10, where the spatial distribution of EVOGDP_2000_2005_2006 is compared with the spatial distribution of (suspected) outliers for this variable. Again, only a much reduced data set of 731 regions could be used for this multivariate analysis. In total, three indicators were used to gauge whether or not an observation is outlying: (1) large (absolute and raw) residuals from an MLR fit; (2) large (absolute and raw) residuals from an LR fit; and (3) large (absolute and raw) residuals from a GWR fit. All three models were calibrated using the coordinates, SF_CF_1999_1999 and SPAT_TYPE_2_1999_1999 as independent contextual data. These indicators are summarised in Fig. 10b, where a strong case for an outlier relates to an observation that has positive results for all three outlier identification analyses.

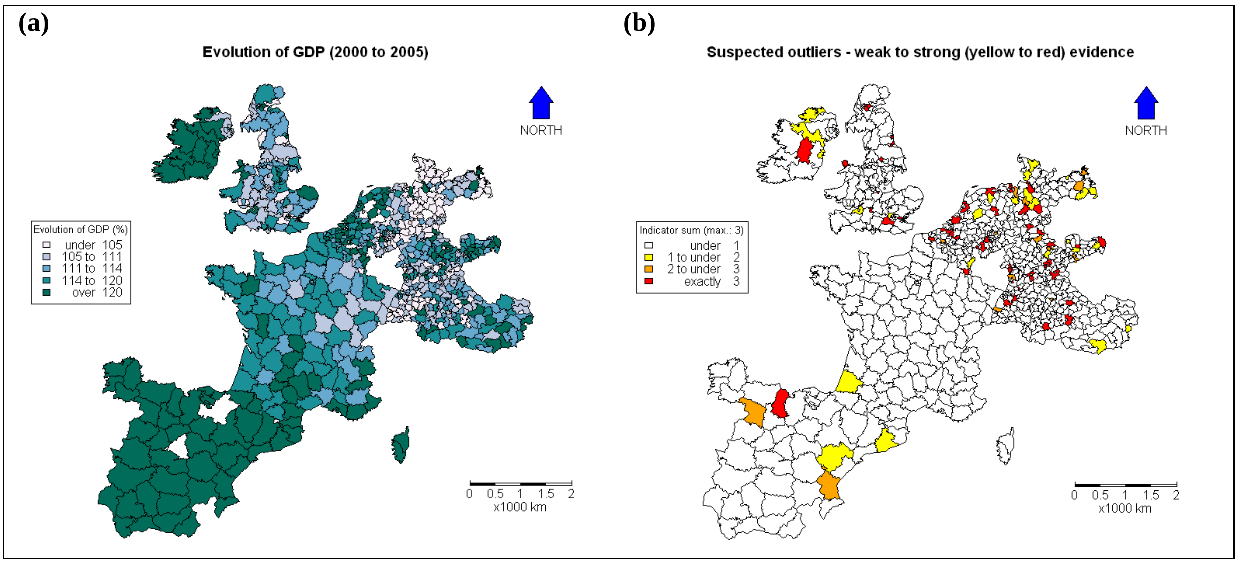
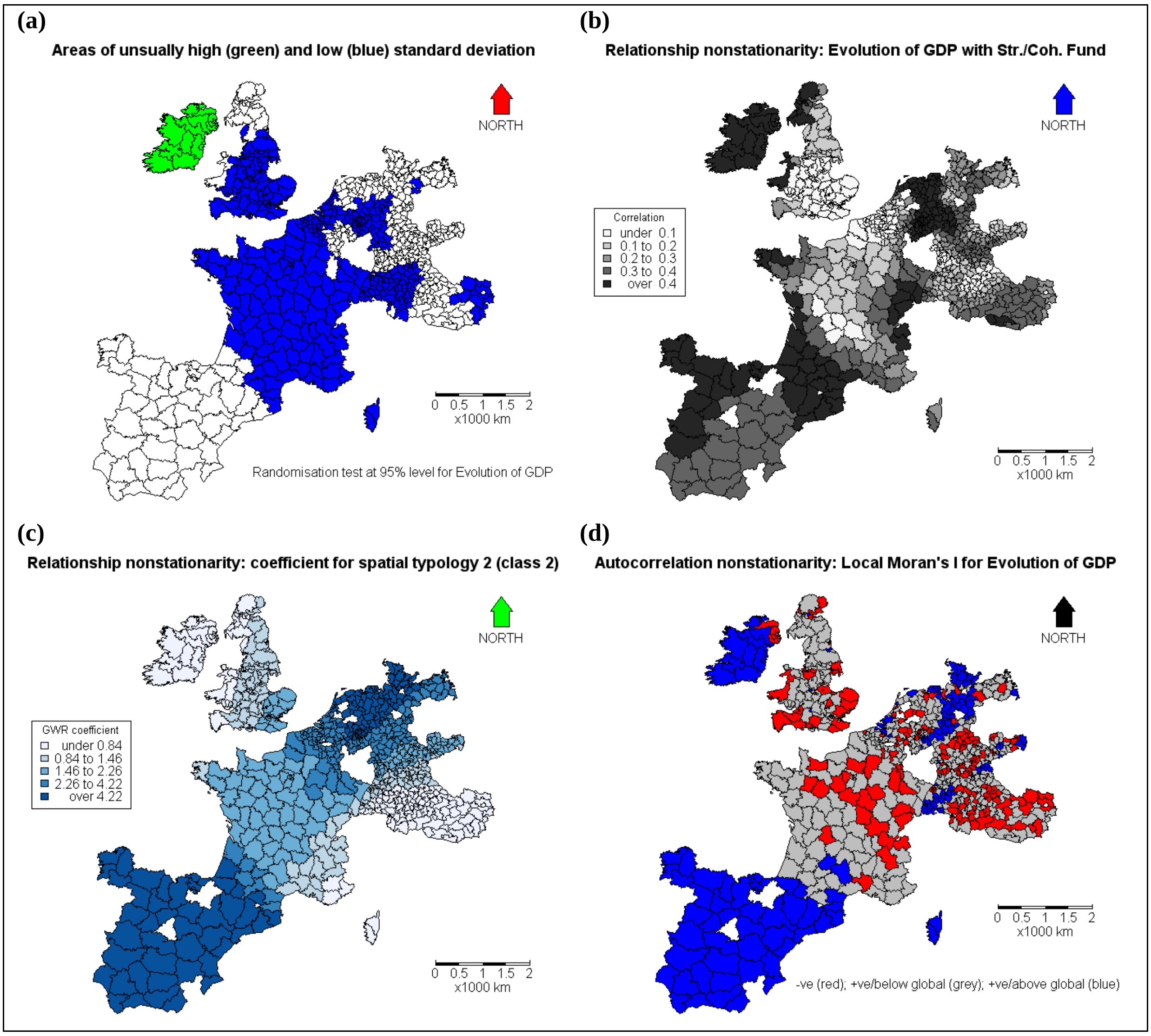


Figure 10*: Spatial distribution of (a) EVOGDP_2000_2005_2006 and (b) suspected outliers for EVOGDP_2000_2005_2006 (three multivariate indicators).*

3.6 Worked example 5: identification of spatial clusters

The R script for worked example 5 is given in Appendix 5. The results are summarised in Fig. 11, where the aim is to identify 'unusual' clusters in EVOGDP_2000_2005_2006 with respect to (a) its local variability (using GW standard deviations); (b) its local relationship to SF_CF_1999_1999 (via a GW correlation analysis); (c) its local relationship to class 2 of SPAT_TYPE_2_1999_1999 (via a GWR analysis); and (d) its local spatial autocorrelation (via local Moran's I statistic). Again, only a much reduced data set of 731 regions could be used for this combined univariate and multivariate analysis. Observe that the shown local relationships for EVOGDP_2000_2005_2006 are examples, as different local relationship can be investigated.


(a) Areas of unusually high (green) and low (blue) standard deviation

(b) Relationship nonstationarity: Evolution of GDP with Str./Coh. Fund

(c) Relationship nonstationarity: coefficient for spatial typology 2 (class 2)

(d) Autocorrelation nonstationarity: Local Moran's I for Evolution of GDP

Figure 11*: Identification of 'unusual' clusters in EVOGDP_2000_2005_2006 with respect to (a) its local variability (using GW standard deviations); (b) its local relationship to SF_CF_1999_1999 (via a GW correlation analysis); (c) its local relationship to class 2 of SPAT_TYPE_2_1999_1999 (via a GWR analysis); and (d) its local spatial autocorrelation (via local Moran's I statistic)*

Briefly and focusing on EVOGDP_2000_2005_2006 for Ireland and Northern Ireland only; Fig. 11a indicates that these regions tend to have unusually high levels of variation in EVOGDP_2000_2005_2006; Fig. 11b suggests that these regions tend to have an unusually strong relationship between EVOGDP_2000_2005_2006 and SF_CF_1999_1999; Fig. 11c suggests that these regions tend to have an unusually weak relationship between EVOGDP_2000_2005_2006 and class 2 of SPAT_TYPE_2_1999_1999; and Fig. 11d suggests that some regions of Northern Ireland can have an unusual negative spatial autocorrelation for EVOGDP_2000_2005_2006 (i.e. neighbouring values of EVOGDP_2000_2005_2006 tend to be dissimilar).

3.7 Worked example 6: some consequences of MAUP

The R script for worked example 6 is given in Appendix 6. The results are summarised in Figs. 12 and 13, where the spatial distribution of EVOGDP_2000_2005_2006 and the spatial distribution of (suspected) outliers for EVOGDP_2000_2005_2006 are shown at four different NUTS levels (i.e. 3, 2, 1 and 0), respectively. At each NUTS level, the same seven indicators are used to gauge whether or not an observation is outlying (as in worked example 2).

Scatterplots and correlations are given in Fig. 14, where the "Strongest indication of an outlier for any constituent NUTS level 3 region" is related to the "Indication of an outlier in a corresponding aggregated NUTS level 2/1/0 region". If the effects of MAUP on outlier identification are minimal, then a strong relationship (and correlation) would be expected.

From Figs. 12 to 14, we can observe that:

- A NUTS level 3 region that is an outlier does not imply that the NUTS level 2/1/0 region that contains it will also be an outlier.
- Several adjacent NUTS level 3 regions that are outliers, which belong to two or more adjacent NUTS level 2 regions, do not imply that those NUTS level 2 regions will be outliers (and so forth down the NUTS levels).
- A NUTS level 0/1/2 region that is an outlier is likely to contain one or more NUTS level 3 regions that are outliers.
- Evidence of an outlier weakens as the level of spatial aggregation increases.

In summary, it is recommended that outliers should be identified at the smallest spatial scale.

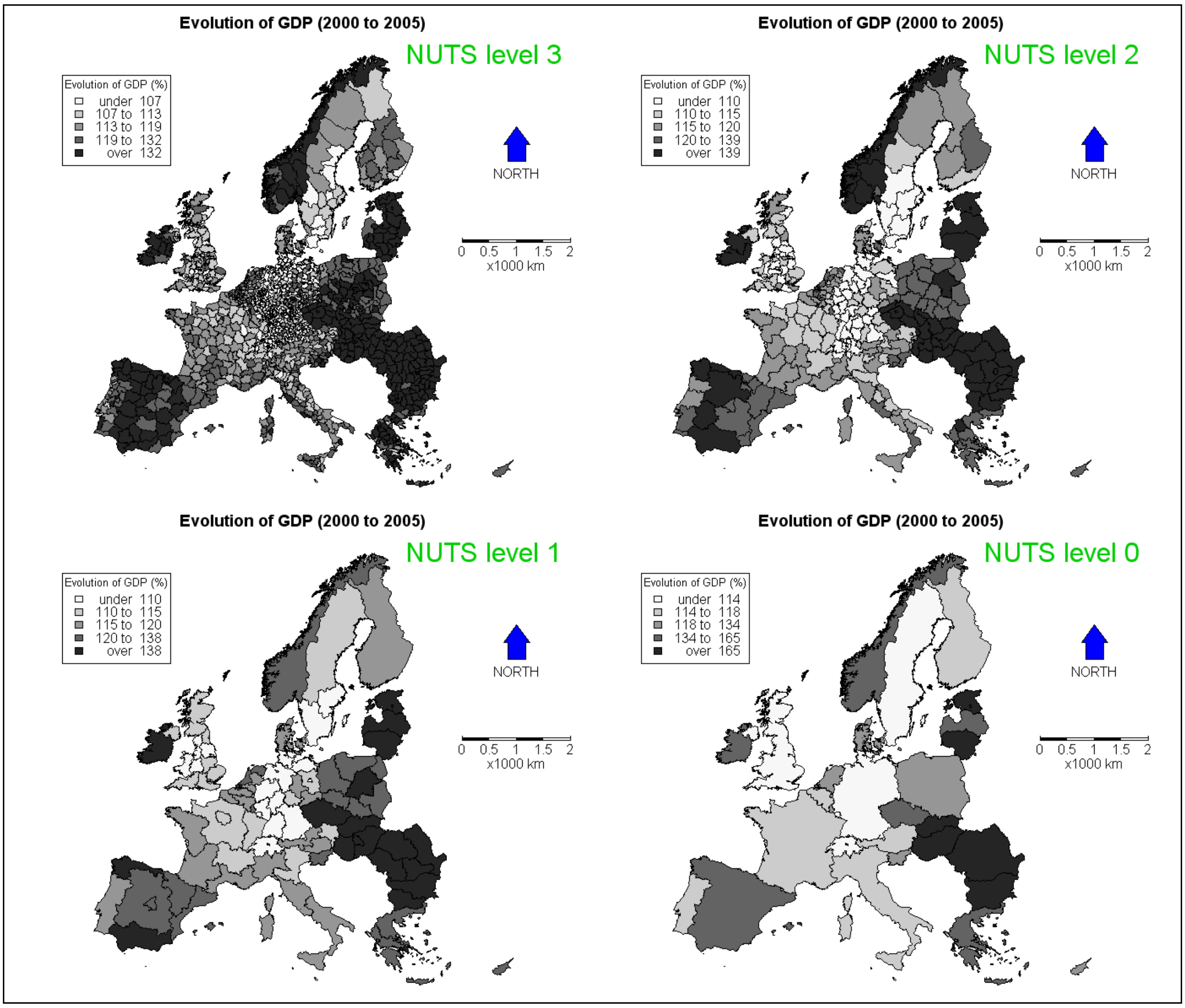


***Figure 12**: Spatial distribution of EVOGDP_2000_2005_2006 at four different NUTS levels (3, 2, 1 and 0)*

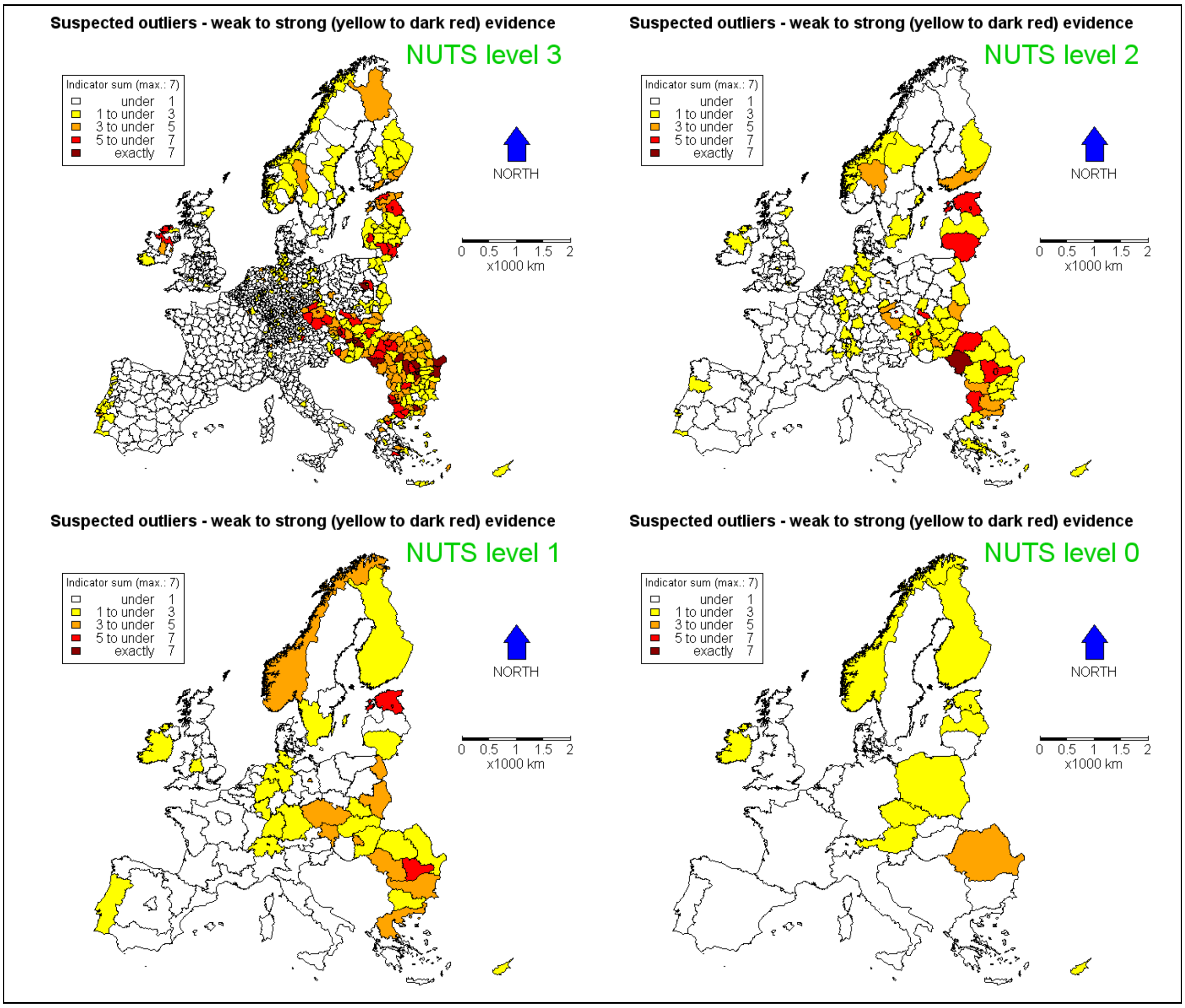
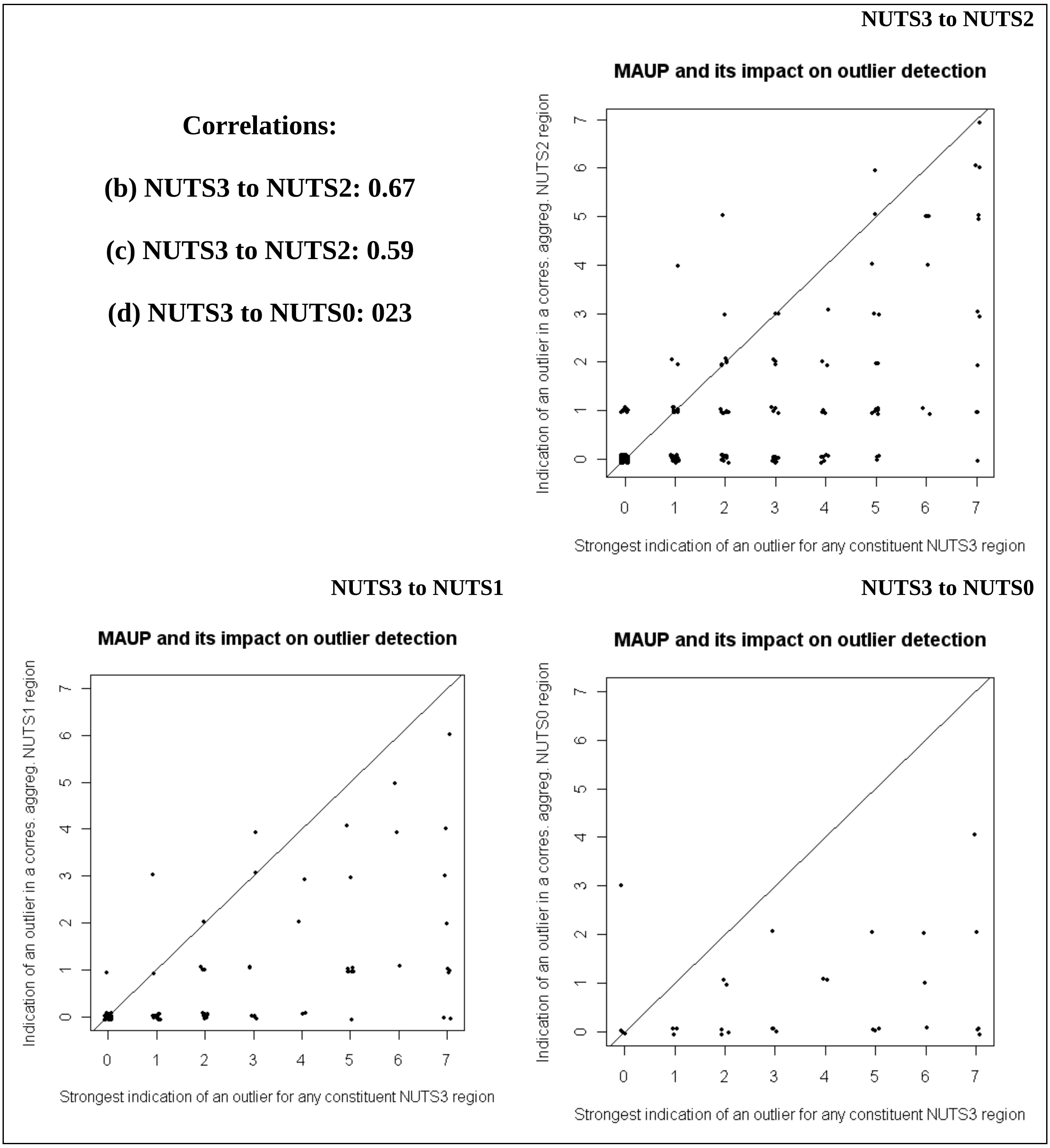

Suspected outliers - weak to strong (yellow to dark red) evidence
NUTS level 3
Suspected outliers - weak to strong (yellow to dark red) evidence
NUTS level 2
Suspected outliers - weak to strong (yellow to dark red) evidence
NUTS level 1
Suspected outliers - weak to strong (yellow to dark red) evidence
NUTS level 0
Indicator sum (max.: 7)
under 1
1 to under 3
3 to under 5
5 to under 7
exactly 7
NORTH
0 0.5 1 1.5 2
x1000 km

***Figure 13**: Spatial distribution of suspected outliers for EVOGDP_2000_2005_2006 (via seven univariate indicators), where outliers are identified at four NUTS levels (3, 2, 1 and 0)*



NUTS3 to NUTS2

Correlations:

(b) NUTS3 to NUTS2: 0.67

(c) NUTS3 to NUTS2: 0.59

(d) NUTS3 to NUTS0: 023

NUTS3 to NUTS1

NUTS3 to NUTS0

Figure 14*: Scatterplots and correlations, where the "Strongest indication of an outlier for any constituent NUTS level 3 region" is related to the "Indication of an outlier in a corresponding aggregated NUTS level 2/1/0 region". Scatterplots are jittered to aide interpretation*

4 Discussion and further developments

This technical report provides an introduction to the detection of logical input errors and statistical outliers (i.e. exceptional values) for data sets of the ESPON 2013 Database. Some important aspatial and spatial techniques have been introduced and demonstrated within the R statistical computing environment.

The field of robust statistics and outlier detection is extremely large and diverse, and as such can not be comprehensively reviewed within the terms of reference of this report. However, outlier detection techniques applicable (or designed for) *spatial* data sets are not as developed as those for *aspatial* applications.

In this respect, our current research is focused on this specific area of model development. Here robust versions of geographically weighted summary statistics (GWSS), geographically weighted regression (GWR) and geographically weighted principal component analysis (GWPCA) are to the fore, as they allow the detection of outliers in both univariate and multivariate spatial data sets.

Our expected deliveries for the final report of this phase of the ESPON project will be firmly based on the analytical techniques described and applied here. However we will now hone these procedures using a concrete, real-life data set rather than the fabricated data set used here. This new data set will no doubt present some new analytical challenges that have not been considered. This should enhance the detection methodology, which may need to include the addition of further techniques.

For the final report, we also aim to introduce a selection of the robust geographically weighted techniques that we are currently working on. An improved version of Hawkins' spatial outlier test is also under development, as is a robust version of the local Moran's I statistic (with respect to outlier identification). Here it is envisaged that our relatively advanced robust spatial methods should not be fully presented in the final report of this first phase of the ESPON project, but instead left for the next phase of the ESPON project (i.e. for the 2011 to 2013 stage), when the development of these robust spatial methods has properly matured. Work in this next phase should also include the packaging of the R code for these robust spatial methods, so that techniques are fully portable, transferable and openly documented.

Acknowledgements

Thanks are due to Ronan Ysebaert and Claude Grasland at UMS RIATE for their work on providing us with the (univariate) test data set and the deliberate introduction of a range of challenging logical input errors to this data set.

References

Ainsworth LM, Dean CB (2008), *Detection of local and global outliers in mapping studies*. Environmetrics 19, 21-37.

Anselin L. (1995) *Local indicators of spatial association*. Geographical Analysis 27, 93 -115.

Béguin C, Hulliger B (2004) *Multivariate outlier detection in incomplete survey data: the epidemic algorithm and transformed rank correlations*. Journal of the Royal Statistical Society, Series A 167(2), 275-294.

Brunsdon C, Fotheringham AS, Charlton ME (2002) *Geographically weighted summary statistics - a framework for localised exploratory data analysis*. Computers, Environment and Urban Systems 26, 501-524.

Brunsdon C, Charlton ME (2010) *An assessment of the effectiveness of multiple hypothesis testing for geographical anomaly detection*. Submitted to Environment and Planning B

Chambers R, Hentges A, Zhao X (2004) *Robust automatic methods for outlier and error detection*. Journal of the Royal Statistical Society, Series A 167(2), 323-339.

Charlton ME, Brunsdon C, Demšar U, Harris P, Fotheringham AS (2010) Principal *component analysis: from global to local*. In preparation.

Charlton S (2004) *Evaluating automatic edit and imputation methods, and the EUREDIT Project*. Journal of the Royal Statistical Society, Series A 167(2), 199-207.

Cruz Ortiz M, Sarabia LA, Herrero A (2006) *Robust regression techniques: A useful alternative for the detection of outlier data in chemical analysis*. Talanta 70, 499-512.

D'Alimonte D, Cornford D (2007) *Outlier detection with partial information: application to emergency mapping*. Stochastic Environmental Research and Risk Assessment 22, 613-620.

Daszykowski M, Kaczmarek K, Vander Heyden Y, Walczak B (2007) *Robust statistics in data analysis – a review Basic concepts*. Chemometrics and Intelligent Laboratory Systems 85, 203-219.

ESPON (2006) 3.4.3 *The modifiable areas unit problem* – Final Report http://www.espon.eu/mmp/online/website/content/projects/261/431/file_4970/

Faraway J (2004) *Linear models with R. Chapman & Hall/CRC*, Boca Raton/FL

Filzmoser P, Garrett R, Reimann C (2005) *Multivariate outlier detection in exploration geochemistry*. Computers & Geosciences 31, 579-587.

Filzmoser P, Maronna R, Werner M (2008) *Outlier identification in high dimensions*. Computational Statistics and Data Analysis 52, 1694-1711.

Fotheringham AS, Brunsdon C, Charlton ME (2002) *Geographically Weighted Regression - the analysis of spatially varying relationships*. Wiley, Chichester.
Frigge M, Hoaglin DC, Iglewicz B (1989) *Some implementations of the Boxplot*. The American Statistician 43, 50–54.

Ghosh-Dastidar B, Schafer JL (2003) *Multiple edit/multiple imputation for multivariate continuous data*. Journal of the American Statistical Association 98(464), 807-817.

Harris P, Brunsdon C (2010) *Exploring spatial variation and spatial relationships in a freshwater acidification critical load data set for Great Britain using geographically weighted summary statistics*. Computers & Geosciences 36, 54-70.

Harris P, Fotheringham AS, Juggins S (2010) *Robust Geographically Weighed Regression: A Technique for Quantifying Spatial Relationships Between Freshwater Acidification Critical Loads and Catchment Attributes*. To appear in the Annals of the Association of American Geographers.

Hawkins RM (1980) *Identification of Outliers*. Chapman & Hall, London.

Hoo KA, Tvarlapati KJ, Piovoso MJ, Hajare R (2002) *A method of robust multivariate outlier replacement*. Computers and Chemical Engineering 26, 17-39.

Hubert M, Vandervieren E (2008) *An adjusted boxplot for skewed distributions*. Computational Statistics and Data Analysis 52, 5186-5201.

Ihaka R, Gentleman R (1996) *R: A language for data analysis and graphics*. Journal of Computational and Graphical Statistics 5, 299-314.

Jackson DA, Chen Y (2004) *Robust principal component analysis and outlier detection with ecological data*. Environmetrics 15, 129-139.

Kou Y, Lu C-T, Chen D (2006) *Spatial Weighted Outlier Detection*. In proceedings of the 2006 SIAM International Conference on Data Mining No. 614 2006.

Liu H, Jezek K, O'Kelly M (2001) *Detecting outliers in irregularly distributed spatial data sets by locally adaptive and robust statistical analysis and GIS*. International Journal of Geographical Information Science 15(8), 721-741.

Loader C (2004) *Smoothing: Local Regression Techniques*. In Gentle J, Härdle W, Mori Y (eds) Handbook of Computational Statistics. Springer-Verlag, Heidelberg.

Locantore N, Marron J, Simpson D, Tripoli N, Zhang J, Cohen K (1999) *Robust principal components for functional data*. Test 8, 1–73.

Meklit T, Van Meirvenne M, Verstraete S, Bonroy J, Tack F (2009) *Combining marginal and spatial outliers identification to optimize the mapping of the regional geochemical baseline concentration of soil heavy metals*. Geoderma 148, 413-420.

Morgenthaler S (2007) *A survey of robust statistics*. Statistical Methods & Applications 15, 271-293.

Petrakos G, Conversano C, Farmakis G, Mola F, Siciliano R, Stavropoulos P (2004) *New ways of specifying data edits*. Journal of the Royal Statistical Society, Series A 167(2), 249-274.

Plaia A, Bondi A (2006) *Single imputation method of missing values in environmental pollution data sets*. Atmospheric Environment 40, 7316-7330.

Reimann C, Filzmoser P, Garrett R (2005) *Background and threshold: critical comparison of methods of determination*. Science of the Total Environment 346, 1-16.

Rousseeuw PJ, Ruts I, Tukey JW (1999) *The Bagplot: A Bivariate Boxplot.* The American Statistician 53, 382–387.

Rousseeuw PJ, Debruyne M, Engelen S, Hubert M (2006) *Robust and outlier detection in chemometrics*. Critical Reviews in Analytical Chemistry 36, 221-242.

Vanden Branden K, Verboven S (2009) *Robust data imputation*. Computational Biology and Chemistry 33, 7-13.

Wong D (1996) *Aggregation effects in geo-referenced data*. In Arlinghaus SL (ed) Practical Handbook of Spatial Statistics. CRC Press, Boca Raton, FL.

Appendices

Appendix 1 – R script for worked example 1

```
# 1. Preamble #####################################################################

# Worked example 1 - for technical report - challenge 10 - ESPON 2013 database
# NCG - P. Harris & M. Charlton
# 7/2/10

# Objective - to identify input errors in:
# "NUTS_2006" (the NUTS3 code)
# "GDP_2000_2006"
# "GDP_2005_2006"
# "POP_T_2000_2006"
# "POP_T_2005_2006"

# Methods: univariate - aspatial
# Mixture of logical & statistical methods
# Statistical methods:
# 1. Standard boxplots only

# R packages needed.....
# 1. GISTools (version 0.5-4) - depends on 2 to 11...
# 2. foreign (version 0.8-30)
# 3. gpclib (version 1.4-3)
# 4. maptools (version 0.7-16)
# 5. Matrix (version 0.999375-18)
# 6. RColorBrewer (version 1.0-2)
# 7. sp (version 0.9-28)
# 8. spam (version 0.15-2)
# 9. spdep (version 0.4-29)
# 10. spgwr (version 0.6-2)
# 11. tripack (version 1.2-11)

# Base R system version 2.9.0
# N.B. Some of the above packages may still depend on other R packages - download these from R website...

# Relevant data files (see data & ArcGIS directories):

# Excel files...
# 1. ESPON_DATA_NCG_CHALLANGE_10_original.xls
# 2. ESPON_DATA_NCG_CHALLANGE_10_subsets.xls
# Text files...
# 3. Worked example 1 true codes & new ID.txt

# ArcGIS files...
# 4. Worked_example_1a.shp - ArcGIS shapefile of the data...

# Only files 3 and 4 are needed in this worked example...



# The variables - some with deliberate input-errors...

# The following 5 variables are all suspected (i.e. in this case, known) to have input errors...
# "NUTS3_2006_E",
# "GDP_2000_2006_E",
# "GDP_2005_2006_E",
# "POP_T_2000_2006_E",
# "POP_T_2005_2006_E"

# These 3 variables are calculated from above so will be effected by an input error...
# "GDP_POP_2000_2006_E",
# "GDP_POP_2005_2006_E",
# "EVOGDP_2000_2005_2006_E"

# Remaining variables - all known to have no input errors...
# "NUTS3","NUTS23","NUTS2","NUTS1","NUTS0" - different NUTS levels
# "Error_type" - type of input error according to technical report (a number between 1 and 8)
# "New_ID" - relates to the regions name only & is purely numeric
# "Region_2006_E" - name of region (NB this does not have any errors)
# "NUTS3" - repeated (a consequence of an ArcGIS operation)
# "X","Y" - centroids of regions



# NOTE that this example dataset has been reduced to
# 1329 values from an original 1351 values (i.e. 22 values removed).
# See readme in excel files on worked example data.

# Note here that 13 of the 22 values removed relate to regions that are highly
# spatially disjoint from mainland Europe (i.e. parts of Portugal, France, and Spain
# such as the Azores, Canaries etc.). Before inclusion into the analyses, we need
# to decide on an appropriate distance metric for these regions.







# 2. Importing data as a ArcGIS shapefile & using GISTools to do some maps... ##

require(GISTools)
#help(GISTools)
# Ignore all warnings - this code is under development...

# Read in the shapefile...
data1 <- readShapePoly("Worked_example_1a.shp",
proj4string=CRS("+proj=Lambert_Azimuthal_Equal_Area+datum=D_ETRS_1989+ellps=GCS_ETRS_1989"))
colnames(data1@data)

# Renaming each variable - as they have been truncated in ArcGIS...
```

```
colnames(data1@data) <- c("NUTS3","NUTS23","NUTS2","NUTS1","NUTS0",
"Error_type","New_ID","NUTS3_2006_E","Region_2006_E",
"GDP_2000_2006_E","GDP_2005_2006_E","POP_T_2000_2006_E","POP_T_2005_2006_E",
"GDP_POP_2000_2006_E","GDP_POP_2005_2006_E","EVOGDP_2000_2005_2006_E","NUTS3","X","Y")

# Size of data set and adding an order ID...
n <- length(data1@data[,1])
Order_ID <- seq(1,n)
data1@data <- cbind(data1@data, Order_ID)
attach(data1@data)

# Creating a shading scheme and plotting a choropleth map...
shades.1 = auto.shading(GDP_2000_2006_E,5, cols=brewer.pal(5,'Greens'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,GDP_2000_2006_E,shades.1)
title("GDP_2000_2006_E: with input errors")
choro.legend(1300000,400000,shades.1,fmt="%4.0f",title='GDP',cex=0.8)
map.scale(1800000,-950000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1000000,80000, col="blue")

# Creating a shading scheme and plotting a choropleth map...
shades.2 = auto.shading(GDP_2005_2006_E,5, cols=brewer.pal(5,'Greens'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,GDP_2005_2006_E,shades.2)
title("GDP_2005_2006_E: with input errors")
choro.legend(1200000,400000,shades.2,fmt="%4.0f",title='GDP',cex=0.8)
map.scale(1800000,-950000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1000000,80000, col="blue")

# Creating a shading scheme and plotting a choropleth map...
shades.3 = auto.shading(POP_T_2000_2006_E,5, cols=brewer.pal(5,'Greens'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,POP_T_2000_2006_E,shades.3)
title("POP_T_2000_2006_E: with input errors")
choro.legend(1400000,400000,shades.3,fmt="%4.0f",title='POP.',cex=0.8)
map.scale(1800000,-950000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1000000,80000, col="blue")

# Creating a shading scheme and plotting a choropleth map...
shades.4 = auto.shading(POP_T_2005_2006_E,5, cols=brewer.pal(5,'Greens'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,POP_T_2005_2006_E,shades.4)
title("POP_T_2005_2006_E: with input errors")
choro.legend(1400000,400000,shades.4,fmt="%4.0f",title='POP.',cex=0.8)
map.scale(1800000,-950000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1000000,80000, col="blue")







# 3. Input error-types 1 and 2 - wrong NUTS code ###############################

# This is one approach to deal with these input error-types...

# Order our data by the New_ID (ie a numeric ID of the NUTS region name)...
```

```
data2 <- data1@data[order(New_ID),]
attach(data2)

# Read in a data set where NUTS codes and names (again given as new_ID) are known to be correct...
data3 <- read.table("Worked example 1 true codes & new ID.txt", header=T)
colnames(data3)
attach(data3)

# Scan for input errors in the NUTS code...
# i.e. relate the "New_ID and NUTS3_2006" variables in datasets, data2 and data3...
data4 <- cbind(data2[,7],data2[,8],data3)
#fix(data4) # data spreadsheet

# Or better still - automatically identify input errors as follows...
x <- match(data4[,2], data4[,4]) # matches the New_ID values and assigns matches by position in data set
y <- seq(1,n) # sequence of numbers from 1 to the size of data set (same as Order_ID)
z <- y-x # should be a data set of zeros if all NUTS codes are inputted correctly
sort(-1*(abs(z))) # in this case 29 NUTS codes are inputted incorrectly...

# Updating input error information in one file - using our ordered data set...
indicator.1 <-ifelse(z==0, 0, 1)
data1.update.1 <- cbind(data2, indicator.1)
data1.update.1 <- as.data.frame(data1.update.1)
attach(data1.update.1)
#fix(data1.update.1)

# Re-order our data back to its original state...
data1@data <- data1.update.1[order(data1.update.1[,20]),] # note using data1.update.1[,20] not Order_ID
attach(data1@data)

# A choropleth map of input errors ...
shades.5 = shading(c(0,1,2),c("blue","white","red")) # this actually gives: white - no errors & red - errors
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,21],shades.5) # use data1@data[,21] not indicator.1
title("Input error-types 1 & 2 (regions coloured red)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")

# Assessing the identification procedure
# Comparing "Error_types 1 & 2" with "indicator.1"
sum(indicator.1)
Assessment <- cbind(data1@data[,6], data1@data[,21])
Assessment.1 <- Assessment[order(-Assessment[,1]),]
#fix(Assessment.1)

# All 29 input errors correctly identified in Ireland & Romania...
# No false positives...

# COMMENT - THIS TYPE OF INPUT-ERROR IS PROBABLY BETTER DETECTED OUTSIDE OF R - i.e.
IN A DATABASE









# 4. Input error-type 3 - impossible values ######################################

# This is one approach to deal with this input error-type...
# Checks for impossible values (in this case, impossible values for positive continuous data,
POP_T_2005_2006_E)

# POP_T_2005_2006_E in the ordered dataset
imp.val <- data2[,13]

# Explore the data...
summary(imp.val) # summary statistics
sort(imp.val) # ordered data
X11(width=5.3,height=5.7)
boxplot(imp.val, main="Input error-type 3", pch=19, cex=0.5) # boxplot

# Define minimum and maximums
Min_pop <- 0
Max_pop <- 10000 # This upper-limit is chosen by judgement

# Identifying & updating input error information in one file - using our ordered data set...
indicator.2 <-ifelse(Min_pop < imp.val & imp.val < Max_pop, 0, 1)
data1.update.2 <- cbind(data1.update.1,indicator.2)
data1.update.2 <- as.data.frame(data1.update.2)
attach(data1.update.2)
#fix(data1.update.2)

# Again re-order our data back to its original state...
data1@data <- data1.update.2[order(data1.update.2[,20]),]
attach(data1@data)

# A choropleth map of input errors ...
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,22],shades.5)
title("Input error-type 3 (regions coloured red)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")

# Assessing the identification procedure
# Comparing "Error_type 3" with "indicator.2"
sum(indicator.2)
Assessment <- cbind(data1@data[,6], data1@data[,22])
Assessment.2 <- Assessment[order(-Assessment[,1]),]
#fix(Assessment.2)

# 1 input error correctly identified in Zurich...
# No false positives...









# 5. Input error-type 5 - potential missing value ##############################

# This is one approach to deal with this input error-type...

# Investigate all entries of -99, -999, -9999, 99, 999, 9999 as potential missing values...

# In this case do this for GDP_2005_2006_E

# GDP_2005_2006_E in the ordered dataset
miss.val <- data2[,11]
```

```
# Identifying & updating potential input error information in one file - using our ordered data set...
indicator.3 <-ifelse(miss.val!=abs(99) & miss.val!=abs(999) & miss.val!=abs(9999), 0, 1)
data1.update.3 <- cbind(data1.update.2,indicator.3)
data1.update.3 <- as.data.frame(data1.update.3)
attach(data1.update.3)
#fix(data1.update.3)

# Again re-order our data back to its original state...
data1@data <- data1.update.3[order(data1.update.3[,20]),]
attach(data1@data)

# A choropleth map of potential input errors...
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,23],shades.5)
title("Potential input error-type 5 (regions coloured red)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")

# Assessing the identification procedure
# Comparing "Error_type 5" with "indicator.3"
sum(indicator.3)
Assessment <- cbind(data1@data[,6], data1@data[,23])
Assessment.3 <- Assessment[order(-Assessment[,1]),]
#fix(Assessment.3)

# 1 input error correctly identified in Malta...
# No false positives...








# 6. Input error-type 4,6,7 and 8 - all (relatively) unexpected values #########

# This is one approach to deal with these input error-types...

# Checking POP_T_2000_2006_E with POP_T_2005_2006_E for unusual data...
# This time four input error-types (4, 6, 7 and 8) can be investigated together...
# From section 3, impossible input error-types have already been identified for POP_T_2005_2006_E
# (i.e. we do not need to account for this error-type)
# but further input error-types can be identified if we relate/compare POP_T_2000_2006_E with
POP_T_2005_2006_E

# Intuitively, these data pairs should be broadly similar (but not exactly the same, i.e. error-type 6)
# Interest lies in the data pairs that are very different (i.e. differences are statistically outlying)
# or are identical (i.e. error-type 6 - copied or repeated data)...

# Again naming the relevant variables in the ordered dataset
x1 <- data2[,12] # POP_T_2000_2006_E
y1 <- data2[,13] # POP_T_2005_2006_E

# Exploring the data with a scatterplot (data should broadly lie on the 45 degree line)...
X11(width=5.3,height=5.7)
plot(x1,y1, main="Potential input error-types 4,6,7 or 8", pch=19, cex=0.5) # scatterplot
abline(0,1) # the 45 degree line

# Difference data...
# POP_T_2005_2006_E minus POP_T_2000_2006_E
z1 <- (y1-x1) # actual differences
```

```
#z1 <- abs(y1-x1) # absolute differences

# Exploring the difference data...
summary(z1) # summary statistics
sort(z1) # ordered data
X11(width=5.3,height=5.7)
hist(z1, main="Potential input error-types 4,6,7 or 8") # histogram
X11(width=5.3,height=5.7)
boxplot(z1, main="Potential input error-types 4,6,7 or 8", pch=19, cex=0.5) # boxplot

# Boxplot statistics...
# Change 'coef' accordingly...
# Default 'coef' is 1.5...
# The higher the 'coef' value the stricter the limits/cut-offs & vice versa...
bp <- boxplot.stats(z1, coef=6)
bp$stats
bp$stats[1] # the lower limit/cut-off - i.e. differences below are deemed outlying...
bp$stats[5] # the upper limit/cut-off - i.e. differences above are deemed outlying...
bp$conf
sort(bp$out)
length(bp$out) # number of potential outliers/errors....
# help(boxplot.stats) # for boxplot details...

# Identifying & updating potential input error information in one file - using our ordered data set...
indicator.4 <-ifelse(z1!=0 & z1>bp$stats[1]& z1<bp$stats[5], 0, 1) # i.e. identical or outlying differences...
data1.update.4 <- cbind(data1.update.3,indicator.4,z1) # note - including the difference data
data1.update.4 <- as.data.frame(data1.update.4)
attach(data1.update.4)
#fix(data1.update.4)

# Again re-order our data back to its original state...
data1@data <- data1.update.4[order(data1.update.4[,20]),]
attach(data1@data)

# A choropleth map of potential input errors...
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,24],shades.5)
title("Potential input error-types 4,6,7 or 8 (regions coloured red)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")

# Observe that this methodology has also identified the impossible value for POP_T_2005_2006_E
# compare...
# indicator.2
# with
# indicators.4
# comparison.1 <- cbind(indicator.2, indicator.4) # see row 119

# We can now investigate these potential outliers more closely...

# Comparing "Error_types 4, 6, 7 & 8" with "indicator.4" and with the "difference data"
sum(indicator.4)
Assessment <- cbind(data1@data[,6], data1@data[,24], data1@data[,25])
Assessment.4 <- Assessment[order(-Assessment[,1]),]
#fix(Assessment.4)

# Results...

# Input error-type 3 - 1 out of 1 input error is correctly re-identified in Zurich...
# Input error-type 4 - 4 out of 4 input errors are correctly identified in Brittany...
# Input error-type 6 - 44 out of 44 input errors are correctly identified in Belgium...
# Input error-type 7 - 81 out of 107 input errors are correctly identified in Italy...
# i.e. 26 False negatives
# Input error-type 8 - 10 out of 11 input errors are correctly identified in Poland...
# i.e. 1 False negative

# False positives...
# For input error-types 4, 7 or 8 -
# 16 out of 1162
# i.e. 16 unusally large increases/decrceases in population are actually true...

# False positives...
# For input error-type 6 -
# 6 out of 1162
# i.e. the population remained exactly the same in 6 regions...



# 7. Input error-type 6 only - repeated or copied data ########################

# This is for GDP_2000_2006_E with GDP_2005_2006_E - but only for repeated data
# These data pairs should be exactly the same

# Again using the relevant variables in the ordered dataset
x2 <- data2[,10] #GDP_2000_2006_E
y2 <- data2[,11] #GDP_2005_2006_E

# Difference data...
z2 <- abs(y2-x2) # absolute differences
sort(z2) # ordered absolute data

# Identifying & updating potential input error information in one file - using our ordered data set...
indicator.5 <-ifelse(z2!=0, 0, 1) # i.e. identical differences...
data1.update.5 <- cbind(data1.update.4,indicator.5,z2) # note - including the difference data
data1.update.5 <- as.data.frame(data1.update.5)
attach(data1.update.5)
#fix(data1.update.5)

# Again re-order our data back to its original state...
data1@data <- data1.update.5[order(data1.update.5[,20]),]
attach(data1@data)

# A choropleth map of potential input errors...
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,26],shades.5)
title("Potential input error-type 6 (regions coloured red)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")

# Assessing the identification procedure
# Comparing "Error_type 6" with "indicator.5" and with the "difference data"
sum(indicator.5)
Assessment <- cbind(data1@data[,6], data1@data[,26], data1@data[,27])
Assessment.5 <- Assessment[order(-Assessment[,1]),]
#fix(Assessment.5)

# 8 out of 8 input errors correctly identified in Slovakia...
# No false positives...
# 8.  All input error-types together ############################################

# Put all indicator data together...
indicator.6 <- indicator.1+indicator.2+indicator.3+indicator.4+indicator.5

data1.update.6 <- cbind(data1.update.5,indicator.6)
data1.update.6 <- as.data.frame(data1.update.6)
attach(data1.update.6)
#fix(data1.update.6)

# Again re-order our data back to its original state...
data1@data <- data1.update.6[order(data1.update.6[,20]),]
attach(data1@data)

# A choropleth map of all identified input errors...
shades.6 = shading(c(0,1,3),c("blue","yellow","black")) # yellow - no errors & black - errors
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,28],shades.6)
title("Identified input errors (regions coloured black)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")

# A choropleth map of actual input errors...
shades.6 = shading(c(0,1,9),c("blue","yellow","black")) # yellow - no errors & black - errors
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,6],shades.6)
title("Actual input errors (regions coloured black)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")

# See Figure 7 in Technical report for above maps...

# Missclassification rates...

n # Data set size
tn.ie <- 205 # Total number of deliberate (known) input errors

tn.f <- 27 # Total number of false negatives
tn.p <- 22 # Total number of false positives

# Rate of false negatives
(tn.f/tn.ie)*100

# Rate of false positives
(tn.p/(n-tn.ie))*100

# Overall missclassification rate
((tn.f+tn.p)/n)*100
```

Appendix 2 – R script for worked example 2

```
# 1. Preamble #########################################################################

# Worked example 2 - for technical report - challenge 10 - ESPON 2013 database
# NCG - P. Harris & M. Charlton
# 8/2/10

# Objective - to identify statistical outliers in:
# "EVOGDP_2000_2005_2006"

# Methods: univariate - aspatial & spatial
# Only statistical methods:
# 1. Standard and Adjusted boxplots,
# 2. Hawkins' test (includes the use of GWSS -
#           geographically weighted summary statistics - GW means and variances),
# 3. LM (local mean, i.e. a GW mean)
# 4. MLR (multiple linear regression),
# 5. LR (local regression) &
# 6. GWR (geographically weighted regression)

# R packages needed.....
# 1. GISTools (version 0.5-4) - depends on 2 to 11...
# 2. foreign (version 0.8-30)
# 3. gpclib (version 1.4-3)
# 4. maptools (version 0.7-16)
# 5. Matrix (version 0.999375-18)
# 6. RColorBrewer (version 1.0-2)
# 7. sp (version 0.9-28)
# 8. spam (version 0.15-2)
# 9. spdep (version 0.4-29)
# 10. spgwr (version 0.6-2) - for GWSS & GWR
# 11. tripack (version 1.2-11)
# 12. moments (version 0.11) - for skewness
# 13. robustbase (version 0.4-5) - for adjusted boxplots
# 14. locfit (version 1.5-4)- for LR

# Base R system version 2.9.0
# N.B. Some of the above packages may still depend on other R packages - download these from R website...

# Relevant data files (see data & ArcGIS directories):

# Excel files...
# 1. ESPON_DATA_NCG_CHALLANGE_10_original.xls
# 2. ESPON_DATA_NCG_CHALLANGE_10_subsets.xls

# ArcGIS files...
# 3. Worked_example_2a.shp - ArcGIS shapefile of the data...
# The 11 variables...

# "NUTS3","NUTS23","NUTS2","NUTS1","NUTS0" - 5 different NUTS levels
# "New_ID" - relates to the regions name only & is purely numeric
# "NUTS3_2006" - the 2006 NUTS3 version
# "Region_2006" - name of 2006 NUTS3 version
# "X","Y" - centroids of regions
# "EVOGDP_2000_2005_2006" - the variable of interest



# NOTE - this example dataset has been reduced to 1329 values
# from an original 1351 values (i.e. 22 values removed)
# see readme in excel files on worked example data.

# NOTE - this dataset is NOT one corrected for input errors from worked example 1.
# It is just the corresponding dataset without the introduction of deliberate input errors.







# 2. Importing data as a ArcGIS shapefile & using GISTools to do a map... ######

require(GISTools)
#help(GISTools)
# Ignore all warnings - this code is under development...

# Read in the shapefile...
data1 <- readShapePoly("Worked_example_2a.shp",
proj4string=CRS("+proj=Lambert_Azimuthal_Equal_Area+datum=D_ETRS_1989+ellps=GCS_ETRS_1989"))
colnames(data1@data)

# renaming each variable - as they have been truncated in ArcGIS...
colnames(data1@data) <- c("NUTS3","NUTS23","NUTS2","NUTS1","NUTS0",
"New_ID","NUTS3_2006","Region_2006",
"X","Y","EVOGDP_2000_2005_2006")

# Size of data set and adding an order ID...
n <- length(data1@data[,1])
Order_ID <- seq(1,n)
data1@data <- cbind(data1@data, Order_ID)
attach(data1@data)

# Creating a shading scheme and plotting a choropleth map of EVOGDP_2000_2005_2006...
shades.1 = auto.shading(EVOGDP_2000_2005_2006,5, cols=brewer.pal(5,'Greys'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,EVOGDP_2000_2005_2006,shades.1)
title("Evolution of GDP (2000 to 2005)")
choro.legend(-2400000,2200000,shades.1,fmt="%4.0f",title='Evolution of GDP (%)',cex=0.8)
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")
#text(1500000,2400000, "NUTS level 3", cex=2, col=3)
```

```
# 3. Boxplots ######################################################################

# Let EVOGDP_2000_2005_2006 be z1...
z1 <- EVOGDP_2000_2005_2006

# Exploring this data...
summary(z1) # summary statistics
sort(z1) # ordered data

# Histogram
X11(width=5.3,height=5.7)
hist(z1, main="Histogram: Evolution of GDP (2000 to 2005)",xlab="Evolution of GDP")

# Standard boxplot with defaults
X11(width=5.3,height=5.7)
boxplot(z1, main="Std. boxplot: Evolution of GDP (2000 to 2005)", pch=19, cex=0.5)

# Standard Boxplot statistics...
# Change 'coef' accordingly...
# Default 'coef' is 1.5...
# The higher the 'coef' value the stricter the limits/cut-offs & vice versa...
bp <- boxplot.stats(z1, coef=1.5)
bp$stats
bp$stats[1] # the lower limit/cut-off - i.e. values below are deemed outlying...
bp$stats[5] # the upper limit/cut-off - i.e. values above are deemed outlying...
bp$conf
sort(bp$out)
length(bp$out) # number of potential outliers...
# help(boxplot.stats) # for details...

# Identifying & updating outlier information in one file
indicator.1 <-ifelse(z1>bp$stats[1]& z1<bp$stats[5], 0, 1) # i.e. suspected outliers...
data1@data <- cbind(data1@data, indicator.1)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of standard boxplot outliers
shades.2 = shading(c(0,1,2),c("blue","white","red")) # i.e. white - no & red - yes
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,13],shades.2)
title("Std. boxplot outliers (regions coloured red)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")

# Need moments package to assess skewness (before adjusted boxplots)
require(moments)
# Ignore warning message...
skewness(z1) # skew is fairly high and positive

# Package for adjusted boxplots...
require(robustbase)

# Adjusted boxplot with defaults
X11(width=5.3,height=5.7)
adjbox(z1, main="Adj. boxplot: Evolution of GDP (2000 to 2005)", pch=19, cex=0.5)
```

```
# Adjusted Boxplot statistics...
# Change 'coef' accordingly...
# Default 'coef' is 1.5...
# The higher the 'coef' value the stricter the limits/cut-offs & vice versa...
abp <- adjboxStats(z1, coef=1.5)
abp$stats
abp$stats[1] # the lower limit/cut-off - i.e. values below are deemed outlying...
abp$stats[5] # the upper limit/cut-off - i.e. values above are deemed outlying...
abp$conf
sort(abp$out)
length(abp$out) # number of potential outliers...
#help(adjboxStats) # for details...

# Identifying & updating outlier information in one file
indicator.2 <-ifelse(z1>abp$stats[1]& z1<abp$stats[5], 0, 1) # i.e. suspected outliers...
data1@data <- cbind(data1@data, indicator.2)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of adjusted boxplot outliers
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,14],shades.2)
title("Adj. boxplot outliers (regions coloured red)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")







# 4. GW summary statistics and Hawkins' Spatial Outlier Test ##################

# First need to find GW means (i.e. LMs) and GW variances for Hawkin's test...
# In this case, using the gw.cov function in spgwr to find the GW means/variances...

# Note 1. - we could define our own weighting scheme to use with the gw.cov function.
# For example, an IDW-type scheme.
# But in this case, the default bi-square weighting scheme is used.

# Note 2. - we could find an optimal bandwidth (i.e. the optimal number of nearby data)
# for a GW mean using 'leave-one-out' cross-validation.
# But in this case, a user-specified bandwidth is defined as the nearest 10% of data.
# It is not so easy to find an optimal bandwidth for a GW variance
# and as such, is commonly chosen subjectively.

# Note 3. - Hawkins' test should ideally use GW means/variances that have been
# calculated without the observation at each calibration/observation location.
# However, this oversight is not expected to advesely affect results.

# Future work can investigate the above issues...

# To re-cap...
colnames(data.1)

# Defining coordinates....
coordinates(data.1) <- c("X", "Y")
```

```
# GW summary statistics at observation locations (i.e. region centroids)...
# Calculated using 10% of nearby EVOGDP_2000_2005_2006 data.
bwd.1 <- 0.1
gwss <- gw.cov(data.1, vars=11, adapt=bwd.1)
#help(gw.cov) # for details...
names(gwss$SDF) # The GW summary statistics calculated...

# GW means and variances...
GW.mean <- gwss$SDF$mean.V1
GW.variance <- (gwss$SDF$sd.V1)^2

# Hawkins' Test for Spatial Outliers...
Hawk.N <- bwd.1*length(X) # number of neighbouring data
Hawk.lm <- GW.mean # the local mean at observation points
Hawk.alv <- mean(GW.variance) # the average local variance with same bandwidth

Hawk.test <- (Hawk.N*(EVOGDP_2000_2005_2006-Hawk.lm)^2)/((Hawk.N+1)*Hawk.alv)  # test statistic
summary(Hawk.test)

# Critical values of the chi-squared distribution
chi_10 <- 2.70554
chi_5 <- 3.84146
chi_2.5 <- 5.02389
chi_1 <- 6.63490
chi_0.5 <- 7.87944
chi_0.01 <- 10.828

# Updating outlier information in one file
indicator.3 <-ifelse(Hawk.test <=chi_5, 0, 1) # change critical level accordingly...
data1@data <- cbind(data1@data, Hawk.test, indicator.3)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of spatial outliers ...
shades.3 = shading(c(chi_5,chi_1,chi_0.01),c("white","yellow","orange","red"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,15],shades.3)
title("Spatial outliers: at 5/1/0.01 % (yellow/orange/red) critical levels")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")








# 5. Residual analysis with LM, MLR, LR and GWR models ########################




# LM...
# Using GW.mean from before...
GW.mean.r <- EVOGDP_2000_2005_2006-GW.mean # Actual minus prediction
summary(GW.mean.r)

# Identifying & updating outlier information in one file
cut.off.1 <- quantile(GW.mean.r, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails - alter accordingly...
```

```
indicator.4 <-ifelse(GW.mean.r>=cut.off.1[2] & GW.mean.r<=cut.off.1[20], 0, 1)
data1@data <- cbind(data1@data, GW.mean.r, indicator.4)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for LM...
shades.4 = shading(c(cut.off.1[2],cut.off.1[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,17],shades.4)
title("Raw resids. from LM: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")



# MLR...
# First- & second-order polynomial fits of the coordinate data...
mlr.1 <- lm(EVOGDP_2000_2005_2006 ~ X+Y)
mlr.2 <- lm(EVOGDP_2000_2005_2006 ~ X+Y+I(X^2)+I(Y^2)+I(X*Y))
summary(mlr.1)
summary(mlr.2)

# Choosing a second-order MLR fit...

# Using raw residuals as in LM fit...
raw.resids.mlr <- EVOGDP_2000_2005_2006-mlr.2$fitted # Actual minus prediction
summary(raw.resids.mlr)

# Identifying & updating outlier information in one file
cut.off.2 <- quantile(raw.resids.mlr, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails - alter accordingly...
indicator.5 <-ifelse(raw.resids.mlr>=cut.off.2[2] & raw.resids.mlr<=cut.off.2[20], 0, 1)
data1@data <- cbind(data1@data, raw.resids.mlr, indicator.5)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for MLR...
shades.5 = shading(c(cut.off.2[2],cut.off.2[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,19],shades.5)
title("Raw resids. from MLR: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")



# LR...
# With coordinate data as explanatory variables (i.e. first-order polynomial).

# Using locfit...
require(locfit)
# Ignore warning message...

# Finding the bandwidth for a non-robust LR (i.e. not a lowess fit)
# using generalised cross-validation (GCV) approach.
summary(gcvplot(EVOGDP_2000_2005_2006~X+Y,data=data.1, scale=F,alpha=seq(0.005,0.01,by=0.001),
deg=1,kern="tricube",lfproc=locfit.raw))
```

```
# Choosing a LR fit with bandwidth chosen from above...
bwd.2 <- 0.008
lr <- locfit(EVOGDP_2000_2005_2006~X+Y,data=data.1, scale=F, alpha=bwd.2,
deg=1,kern="tricube",lfproc=locfit.raw)

# Raw residuals...
lr.p <- fitted.locfit(lr)
raw.resids.lr <- EVOGDP_2000_2005_2006-lr.p # Actual minus prediction
summary(raw.resids.lr)

# Identifying & updating outlier information in one file
cut.off.3 <- quantile(raw.resids.lr, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails - alter accordingly...
indicator.6 <-ifelse(raw.resids.lr>=cut.off.3[2] & raw.resids.lr<=cut.off.3[20], 0, 1)
data1@data <- cbind(data1@data, raw.resids.lr, indicator.6)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for LR...
shades.6 = shading(c(cut.off.3[2],cut.off.3[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,21],shades.6)
title("Raw resids. from LR: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")



# GWR...
# With coordinate data as explanatory variables (i.e. first-order polynomial).
# Using spgwr...

# Defining the coordinates...
coords.1<-cbind(data.1[,9],data.1[,10])

# Finding the bandwidth for GWR using Akaike Information Criterion (AIC) approach.
#gwr.aic.bwd <-gwr.sel(EVOGDP_2000_2005_2006~X+Y,data=data.1,coords=coords.1,adapt=TRUE,
#gweight=gwr.bisquare, method="aic")
#gwr.aic.bwd[1] # the optimum bandwidth

# Or finding the bandwidth for GWR using cross-validation approach.
#gwr.cv.bwd <-gwr.sel(EVOGDP_2000_2005_2006~X+Y,data=data.1,coords=coords.1,adapt=TRUE,
#gweight=gwr.bisquare, method="cv")
#gwr.cv.bwd[1] # the optimum bandwidth

# Above optimisation can take a long time...
# So choosing a GWR fit with user-specified bandwidth of 0.03...
bwd.3 <- 0.03
gwr.p <-gwr(EVOGDP_2000_2005_2006~X+Y,data=data.1,coords=coords.1,adapt=bwd.3,
gweight=gwr.bisquare,predictions=T)
#gwr.p$SDF

# GWR raw residuals...
raw.resids.gwr <- EVOGDP_2000_2005_2006-gwr.p$SDF$pred
summary(raw.resids.gwr)

# Identifying & updating outlier information in one file
cut.off.4 <- quantile(raw.resids.gwr, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails - alter accordingly...
indicator.7 <-ifelse(raw.resids.gwr>=cut.off.4[2] & raw.resids.gwr<=cut.off.4[20], 0, 1)
data1@data <- cbind(data1@data, raw.resids.gwr, indicator.7)
data1@data <- as.data.frame(data1@data)
```

```
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for GWR...
shades.7 = shading(c(cut.off.4[2],cut.off.4[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,23],shades.7)
title("Raw resids. from GWR: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")



# NB - Future work should explore the identification of outliers using
# standardiased residuals & corresponding robust regression models...







# 6.  All identified outliers together ##########################################

# Put all indicator data together...
indicator.8 <- indicator.1+indicator.2+indicator.3+indicator.4+indicator.5+indicator.6+indicator.7
summary(indicator.8)
# Histogram
X11(width=5.3,height=5.7)
hist(indicator.8,br=c(0,1,2,3,4,5,6,7))

# Thus a strong case for an outlier relates to an observation
# that has a indicator.8 value of 7...

data1@data <- cbind(data1@data, indicator.8)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)
#write.table(data.1,"Outliers_NUTS_level3.txt", col.names=T,row.names=F)

# A choropleth map of suspected outliers...
shades.7 = shading(c(1,3,5,7),c("white","yellow","orange","red","dark red"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,25],shades.7)
title("Suspected outliers - weak to strong (yellow to dark red) evidence")
choro.legend(-2400000,2200000,shades.7,
over="exactly", between="to under",
fmt="%4.0f",title='Indicator sum (max.: 7)',cex=0.8)
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")
#text(1500000,2400000, "NUTS level 3", cex=2, col=3)
```

Appendix 3 – R script for worked example 3

```
# 1. Preamble #####################################################################

# Worked example 3 - for technical report - challenge 10 - ESPON 2013 database
# NCG - P. Harris & M. Charlton
# 7/2/10

# Objective - to identify statistical outliers in some subset of this data:
# "X"
# "Y"
# "EVOGDP_2000_2005_2006"
# "SPAT_TYPE_1_1999_1999"
# "SPAT_TYPE_2_1999_1999"
# "UNEMP_R_2001_1999"
# "LU_AS_1_1996_1999"
# "LU_AS_2_1996_1999"
# "LU_AS_3_1996_1999"
# "LU_UF_1996_1999"
# "LU_AR_1996_1999"
# "LU_PC_1996_1999"
# "NAT_HAZ_2004_1999"
# "SF_CF_1999_1999"
# "SF_R_1999_1999"
# "SF_S_1999_1999"
# "SF_A_1999_1999"
# "CF_T_1999_1999"
# "CF_E_1999_1999"

# Methods: multivariate - aspatial only
# Only statistical methods:
# 1. Bagplots
# 2. Robust MD-squared analysis (RMD2-AQ-outlier)
# 3. Two techniques based on PCA for outlier detection (PCA-outlier-1 & PCA-outlier-2)

# R packages needed.....
# 1. GISTools (version 0.5-4) - depends on 2 to 11...
# 2. foreign (version 0.8-30)
# 3. gpclib (version 1.4-3)
# 4. maptools (version 0.7-16)
# 5. Matrix (version 0.999375-18)
# 6. RColorBrewer (version 1.0-2)
# 7. sp (version 0.9-28)
# 8. spam (version 0.15-2)
# 9. spdep (version 0.4-29)
# 10. spgwr (version 0.6-2)
# 11. tripack (version 1.2-11)
# 12. aplpack (version 1.2-1) - for bagplots
# 13. robustbase (version 0.4-5) - required for mvoutlier package
# 14. mvoutlier (version 1.4) - for robust MD-squared analysis and PCA outlier detection
# Base R system version 2.9.0
# N.B. Some of the above packages may still depend on other R packages - download these from R website...



# Relevant data files (see data & ArcGIS directories):

# Excel files...
# 1. ESPON_DATA_NCG_CHALLANGE_10_original.xls
# 2. ESPON_DATA_NCG_CHALLANGE_10_subsets.xls

# ArcGIS files...
# 3. Worked_example_345a_reduced.shp - ArcGIS shapefile of the data...



# The 27 variables...

# "NUTS3","NUTS23","NUTS2","NUTS1","NUTS0" - 5 different NUTS levels
# "New_ID" - relates to the regions name only & is purely numeric
# "NUTS3_2006" - the 2006 NUTS3 version
# "Region_2006" - name of 2006 NUTS3 version
# "X","Y" - centroids of regions
# "EVOGDP_2000_2005_2006" - Evolution of GDP
# and 16 likely contextual variables of "EVOGDP_2000_2005_2006" ...
# "SPAT_TYPE_1_1999_1999"
# "SPAT_TYPE_2_1999_1999"
# "UNEMP_R_2001_1999"
# "LU_AS_1_1996_1999"
# "LU_AS_2_1996_1999"
# "LU_AS_3_1996_1999"
# "LU_UF_1996_1999"
# "LU_AR_1996_1999"
# "LU_PC_1996_1999"
# "NAT_HAZ_2004_1999"
# "SF_CF_1999_1999"
# "SF_R_1999_1999"
# "SF_S_1999_1999"
# "SF_A_1999_1999"
# "CF_T_1999_1999"
# "CF_E_1999_1999"



# NOTE - Methods demonstrated in this worked example do not require
# a relationship between "EVOGDP_2000_2005_2006" and its likely
# contextual data - see worked examples 4 and 5 for this.



# NOTE - This example data set has been reduced to 731 values
# from an original 1351 values
# see readme in excel files on worked example data.







# 2. Importing data as a ArcGIS shapefile & using GISTools to do some maps #####
```

```
require(GISTools)
#help(GISTools)
# Ignore all warnings - this code is under development...

# Read in the shapefile...
data1 <- readShapePoly("Worked_example_345a_reduced.shp",
proj4string=CRS("+proj=Lambert_Azimuthal_Equal_Area+datum=D_ETRS_1989+ellps=GCS_ETRS_1989"))
colnames(data1@data)

# renaming each variable - as they have been truncated in ArcGIS...
colnames(data1@data) <- c("NUTS3","NUTS23","NUTS2","NUTS1","NUTS0",
"New_ID","NUTS3_2006","Region_2006",
"X","Y","EVOGDP_2000_2005_2006",
"SPAT_TYPE_1_1999_1999","SPAT_TYPE_2_1999_1999",
"UNEMP_R_2001_1999",
"LU_AS_1_1996_1999","LU_AS_2_1996_1999","LU_AS_3_1996_1999",
"LU_UF_1996_1999","LU_AR_1996_1999","LU_PC_1996_1999",
"NAT_HAZ_2004_1999",
"SF_CF_1999_1999",
"SF_R_1999_1999","SF_S_1999_1999","SF_A_1999_1999",
"CF_T_1999_1999","CF_E_1999_1999")

# Size of data set and adding an order ID...
n <- length(data1@data[,1])
Order_ID <- seq(1,n)
data1@data <- cbind(data1@data, Order_ID)
attach(data1@data)

# Coordinate data only...
coords <- cbind(data1@data[,9],data1@data[,10])

# Example multivariate data set one...
Mult.data.1 <- cbind(EVOGDP_2000_2005_2006,SPAT_TYPE_1_1999_1999,SPAT_TYPE_2_1999_1999,
UNEMP_R_2001_1999,LU_AS_1_1996_1999,LU_AS_2_1996_1999,LU_AS_3_1996_1999,LU_UF_1996_1999,
LU_AR_1996_1999,LU_PC_1996_1999,NAT_HAZ_2004_1999,SF_CF_1999_1999,SF_R_1999_1999,
SF_S_1999_1999,SF_A_1999_1999,CF_T_1999_1999,CF_E_1999_1999)
Mult.data.1 <- as.data.frame(Mult.data.1)
attach(Mult.data.1)

# Example multivariate data set two...
Mult.data.2 <- cbind(EVOGDP_2000_2005_2006,UNEMP_R_2001_1999,
NAT_HAZ_2004_1999,SF_CF_1999_1999)
Mult.data.2 <- as.data.frame(Mult.data.2)
attach(Mult.data.2)

# Example multivariate data set three (data set two with coordinates)...
Mult.data.3 <- cbind(X,Y,EVOGDP_2000_2005_2006,UNEMP_R_2001_1999,
NAT_HAZ_2004_1999,SF_CF_1999_1999)
Mult.data.3 <- as.data.frame(Mult.data.3)
attach(Mult.data.3)

# Creating a shading scheme and plotting a choropleth map of EVOGDP_2000_2005_2006...
shades.1 = auto.shading(EVOGDP_2000_2005_2006,5, cols=brewer.pal(5,'PuBuGn'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,EVOGDP_2000_2005_2006,shades.1)
title("Evolution of GDP (2000 to 2005)")
choro.legend(-2300000,250000,shades.1,fmt="%4.0f",title='Evolution of GDP (%)',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")

# Creating a shading scheme and plotting a choropleth map of UNEMP_R_2001_1999...
```

```
shades.2 = auto.shading(UNEMP_R_2001_1999,5, cols=brewer.pal(5,'Greys'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,UNEMP_R_2001_1999,shades.2)
title("Unemployment rate")
choro.legend(-2300000,250000,shades.2,fmt="%4.1f",title='Unemployment (%)',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")

# Creating a shading scheme and plotting a choropleth map of NAT_HAZ_2004_1999...
shades.3 = auto.shading(NAT_HAZ_2004_1999,5, cols=brewer.pal(5,'Greens'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,NAT_HAZ_2004_1999,shades.3)
title("Natural hazards")
choro.legend(-2300000,250000,shades.3,fmt="%4.0f",title='Indication',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")

# Creating a shading scheme and plotting a choropleth map of SF_CF_1999_1999...
shades.4 = auto.shading(SF_CF_1999_1999,5, cols=brewer.pal(5,'Reds'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,SF_CF_1999_1999,shades.4)
title("All Structural & Cohesion Fund expenditure")
choro.legend(-2350000,250000,shades.4,fmt="%4.0f",title='Str. & Coh. Fund',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")







# 3. Bagplots ####################################################################

# Package for bagplots...
require(aplpack)

# For example, exploring these 2 variables...
z1 <- EVOGDP_2000_2005_2006
z2 <- UNEMP_R_2001_1999

# Summary statistics...
summary(z1)
summary(z2)

# Univariate boxplots...
X11(width=5.3,height=5.7)
boxplot(z1, main="Evolution of GDP", pch=19, cex=0.5)
X11(width=5.3,height=5.7)
boxplot(z2, main="Unemployment", pch=19, cex=0.5)

# The bagplot function
#help(bagplot)

# Example...
X11(width=5.3,height=5.7)
bagp.1 <- bagplot(z1,z2,xlab="Evolution of GDP",ylab="Unemployment",
main="Example bagplot: outliers in red (outside of bag)",cex=0.6)
bivariate.outliers.1 <- bagp.1$pxy.outlier
```

```
length(bivariate.outliers.1[,1])
bivariate.not.outliers.1 <-rbind(bagp.1$pxy.bag,bagp.1$pxy.outer)
length(bivariate.not.outliers.1[,1])

# Some data manipulations for mapping...
# Note can also use library(sqldf) to match datasets...
bivariate.outliers.1x <- merge(data1@data, bivariate.outliers.1,
by.x=c("EVOGDP_2000_2005_2006","UNEMP_R_2001_1999"), by.y=c("x","y"))
indicator.1 <-c(rep(1,length(bivariate.outliers.1x[,1])))
bivariate.outliers.1x <- cbind(bivariate.outliers.1x, indicator.1)

bivariate.not.outliers.1x <- merge(data1@data, bivariate.not.outliers.1,
by.x=c("EVOGDP_2000_2005_2006","UNEMP_R_2001_1999"), by.y=c("x","y"))
indicator.1 <-c(rep(0,length(bivariate.not.outliers.1x[,1])))
bivariate.not.outliers.1x <- cbind(bivariate.not.outliers.1x, indicator.1)

xx1 <- rbind(bivariate.not.outliers.1x, bivariate.outliers.1x)
data1@data <- xx1[order(xx1[,28]),] # get data in correct order with Order_ID
attach(data1@data)

# A choropleth map...
shades.5 = shading(c(0,1,2),c("white","green","red")) # i.e. green - no & red - yes
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,29],shades.5)
title("Bivariate outliers (red): Bagplot of Evolution of GDP with Unemployment")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")









# 4. Robust MD-squared analysis (RMD2-AQ-outlier) ##############################

# Following the paper of Filzmoser et al. (2005)...

# Load the necessary package...
require(mvoutlier)

# Note - multivariate data set one and similar data subsets can give rise to some technical problems
# with this technique, as it is designed for continuous multivariate normal data with outlying observations,
# whereas we have data sets that include categorical data.
# In this respect, we only explore example multivariate data sets two & three, which only have continuous
variables.
# This is not considered a problem as outliers are expected to be more hidden in continuous variables.
# Non-normality of continuous data may however still cause problems.
# Similar comments apply to the PCA methods of section 5...

# Therefore using example multivariate dataset two...

# The key function/plot for this identification technique is...
#help(aq.plot)
#X11(width=12,height=8)
#aq.plot(Mult.data.2)
#help(aq.plot)

# However, slightly adapting the aq.plot function to suit our needs...
aq.plot.1 <- function (x, delta = qchisq(0.975, df = ncol(x)), quan = 1/2,
    alpha = 0.025)
```

```
{
    if (is.vector(x) == TRUE || ncol(x) == 1) {
        stop("x must be at least two-dimensional")
    }
    covr <- covMcd(x, alpha = quan)
    dist <- mahalanobis(x, center = covr$center, cov = covr$cov)
    s <- sort(dist, index = TRUE)
    z <- x
    if (ncol(x) > 2) {
        p <- princomp(x, covmat = covr)
        z <- p$scores[, 1:2]
        sdprop <- (p$sd[1] + p$sd[2])/sum(p$sd)
        cat("Projection to the first and second robust principal components.\n")
        cat("Proportion of total variation (explained variance): ")
        cat(sdprop)
        cat("\n")
    }
    par(mfrow = c(2, 2), mai = c(0.8, 0.6, 0.2, 0.2), mgp = c(2.4,
        1, 0))
    plot(z, col = 3, type = "n",
        main="(A) Data (by ID) projected on the first two RPCs",
        xlab = "First Robust Principal Component (RPC)", ylab = "Second Robust Principal Component (RPC)")
    text(z, dimnames(as.data.frame(z))[[1]], col = 3, cex = 0.8)
    plot(s$x, (1:length(dist))/length(dist), col = 3,
        main = paste("(B) Outlier detection: above ",
        100 * (1 - alpha), "%  & adj. quantiles", sep = ""),
        xlab = "Ordered squared robust Mahalanobis distances",
        ylab = "Cumulative probability", type = "n")
    text(s$x, (1:length(dist))/length(dist), as.character(s$ix),
        col = 3, cex = 0.8)
    t <- seq(0, max(dist), by = 0.01)
    lines(t, pchisq(t, df = ncol(x)), col = 6)
    abline(v = delta, col = 5)
    xarw <- arw(x, covr$center, covr$cov, alpha = alpha)
    # note - arw() is the adaptive reweighted estimator for multivariate location and scatter...
    abline(v = xarw$cn, col = 4)
    legend(11000, 0.3, c("Chi-squared dist. func.", paste(100 * (1 - alpha), "% quantile", sep = ""),
    "Adjusted quantile"), col = c(6,5,4), lty = c(1,1,1), bty="n")
    plot(z, col = 3, type = "n", main = paste("(C) Outliers (in red) based on (user-specified) ",
        100 * (1 - alpha), "% quantile", sep = ""),
        xlab = "First RPC", ylab = "Second RPC")
    for (i in 1:nrow(x)) {
        if (dist[i] >= delta)
            text(z[i, 1], z[i, 2], dimnames(as.data.frame(x))[[1]][i],
                col = 2, cex = 0.8)
        if (dist[i] < delta)
            text(z[i, 1], z[i, 2], dimnames(as.data.frame(x))[[1]][i],
                col = 3, cex = 0.8)
    }
    plot(z, col = 3, type = "n", main = "(D) Outliers (in red) based on adjusted quantile",
        xlab = "First RPC", ylab = "Second RPC")
    for (i in 1:nrow(x)) {
        if (dist[i] >= xarw$cn)
            text(z[i, 1], z[i, 2], dimnames(as.data.frame(x))[[1]][i],
                col = 2, cex = 0.8)
        if (dist[i] < xarw$cn)
            text(z[i, 1], z[i, 2], dimnames(as.data.frame(x))[[1]][i],
                col = 3, cex = 0.8)
    }
    o <- (sqrt(dist) > min(sqrt(xarw$cn), sqrt(qchisq(0.975,
        dim(x)[2]))))
    l <- list(outliers = o)
    l
}

# Thus our take on the adjusted quantile plot...
X11(width=12,height=8)
mult.out.d2.m1 <- aq.plot.1(Mult.data.2)

# Identifying & updating outlier information in one file
indicator.2 <-ifelse(mult.out.d2.m1$outliers==F, 0, 1) # i.e. suspected outliers...
data1@data <- cbind(data1@data, indicator.2)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of the multivariate outliers...
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,30],shades.5)
title("Multivariate outliers: RMD2-AQ-outlier data set 2 (regions coloured red)")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")

# Univariate presentation of the same multivariate outliers...
# (i.e multivariate outliers in red as in the above choropleth map)
X11(width=12,height=8)
uni.plot(Mult.data.2)

# NB - see Filzmoser 2005 paper & mvoutlier reference manual for more options
# on the visualisation of multivariate outliers...




# And using example multivariate dataset three...

# The adjusted quantile plot...
X11(width=12,height=8)
mult.out.d3.m1 <- aq.plot.1(Mult.data.3)

# Identifying & updating outlier information in one file
indicator.3 <-ifelse(mult.out.d3.m1$outliers==F, 0, 1) # i.e. suspected outliers...
data1@data <- cbind(data1@data, indicator.3)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of the multivariate outliers...
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,31],shades.5)
title("Multivariate outliers: RMD2-AQ-outlier data set 3 (regions coloured red)")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")

# Univariate presentation of the same multivariate outliers...
# (i.e multivariate outliers in red as in the above choropleth map)
X11(width=16,height=8)
uni.plot(Mult.data.3)
```

```
# 5. PCA for outlier detection ####################################################

# Following the paper of Filzmoser et al. (2008)...

# Again using the mvoutlier package
# And using only Mult.data.2 data set for simplicity...

# Sign Method for Outlier Identification in High Dimensions...
# i.e. PCA-outlier-1
# Simple version (sign1) & sophisticated (sign2) versions are possible...
# Using the simple version...
mult.out.d2.m2 <- sign1(Mult.data.2)

# Identifying & updating outlier information in one file
indicator.4 <- ifelse(mult.out.d2.m2$wfinal01==1, 0, 1) # i.e. suspected outliers are the wrong way around in this case...
data1@data <- cbind(data1@data, indicator.4)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of the multivariate outliers
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,32],shades.5)
title("Multivariate outliers: PCA-outlier-1 data set 2 (regions coloured red)")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")



# PCOut Method for Outlier Identification in High Dimensions
# i.e. PCA-outlier-2
mult.out.d2.m3 <- pcout(Mult.data.2)

# Identifying & updating outlier information in one file
indicator.5 <- ifelse(mult.out.d2.m3$wfinal01==1, 0, 1) # i.e. suspected outliers are the wrong way around in this case...
data1@data <- cbind(data1@data, indicator.5)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of the multivariate outliers
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,33],shades.5)
title("Multivariate outliers: PCA-outlier-2 data set 2 (regions coloured red)")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")
```

```
# 6. Summary ##########################################################################

# Number of potential outliers

# Bagplot...
sort(-data1@data[,29])

#RMD2-AQ-outlier with multivariate data set 2
sort(-indicator.2)

#PCA-outlier-1 with multivariate data set 2
sort(-indicator.4)

#PCA-outlier-2 with multivariate data set 2
sort(-indicator.5)
```

Appendix 4 – R script for worked example 4

```
# 1. Preamble #########################################################################



# Worked example 4 - for technical report - challenge 10 - ESPON 2013 database
# NCG - P. Harris & M. Charlton
# 6/2/10



# Objective - to identify statistical outliers in "EVOGDP_2000_2005_2006"
# in relation to some subset of the following explanatory/contextual data:
# "X"
# "Y"
# "SPAT_TYPE_1_1999_1999"
# "SPAT_TYPE_2_1999_1999"
# "UNEMP_R_2001_1999"
# "LU_AS_1_1996_1999"
# "LU_AS_2_1996_1999"
# "LU_AS_3_1996_1999"
# "LU_UF_1996_1999"
# "LU_AR_1996_1999"
# "LU_PC_1996_1999"
# "NAT_HAZ_2004_1999"
# "SF_CF_1999_1999"
# "SF_R_1999_1999"
# "SF_S_1999_1999"
# "SF_A_1999_1999"
# "CF_T_1999_1999"
# "CF_E_1999_1999"



# Methods: multivariate - aspatial & spatial
# Only statistical methods:
# 1. MLR (multiple linear regression),
# 2. LR (local regression) &
# 3. GWR (geographically weighted regression)



# R packages needed.....
# 1. GISTools (version 0.5-4) - depends on 2 to 11...
# 2. foreign (version 0.8-30)
# 3. gpclib (version 1.4-3)
# 4. maptools (version 0.7-16)
# 5. Matrix (version 0.999375-18)
# 6. RColorBrewer (version 1.0-2)
# 7. sp (version 0.9-28)
# 8. spam (version 0.15-2)
# 9. spdep (version 0.4-29)
# 10. spgwr (version 0.6-2) - for GWSS & GWR
# 11. tripack (version 1.2-11)
# 12. car (version 1.2-12) - for MLR
# 13. locfit (version 1.5-4)- for LR

# Base R system version 2.9.0
# N.B. Some of the above packages may still depend on other R packages - download these from R website...



# Relevant data files (see data & ArcGIS directories):

# Excel files...
# 1. ESPON_DATA_NCG_CHALLANGE_10_original.xls
# 2. ESPON_DATA_NCG_CHALLANGE_10_subsets.xls

# ArcGIS files...
# 3. Worked_example_345a_reduced.shp - ArcGIS shapefile of the data...



# The 27 variables...

# "NUTS3","NUTS23","NUTS2","NUTS1","NUTS0" - 5 different NUTS levels
# "New_ID" - relates to the regions name only & is purely numeric
# "NUTS3_2006" - the 2006 NUTS3 version
# "Region_2006" - name of 2006 NUTS3 version
# "X","Y" - centroids of regions
# "EVOGDP_2000_2005_2006" - Evolution of GDP
# and 16 likely contextual variables of "EVOGDP_2000_2005_2006" ...
# "SPAT_TYPE_1_1999_1999"
# "SPAT_TYPE_2_1999_1999"
# "UNEMP_R_2001_1999"
# "LU_AS_1_1996_1999"
# "LU_AS_2_1996_1999"
# "LU_AS_3_1996_1999"
# "LU_UF_1996_1999"
# "LU_AR_1996_1999"
# "LU_PC_1996_1999"
# "NAT_HAZ_2004_1999"
# "SF_CF_1999_1999"
# "SF_R_1999_1999"
# "SF_S_1999_1999"
# "SF_A_1999_1999"
# "CF_T_1999_1999"
# "CF_E_1999_1999"
# NOTE - This example data set has been reduced to 731 values from an original 1351 values
# see readme in excel files on worked example data.






# 2. Importing data as a ArcGIS shapefile & using GISTools to do some maps #####

require(GISTools)
#help(GISTools)
# Ignore all warnings - this code is under development...

# Read in the shapefile...
data1 <- readShapePoly("Worked_example_345a_reduced.shp",
proj4string=CRS("+proj=Lambert_Azimuthal_Equal_Area+datum=D_ETRS_1989+ellps=GCS_ETRS_1989"))
colnames(data1@data)

# renaming each variable - as they have been truncated in ArcGIS...
colnames(data1@data) <- c("NUTS3","NUTS23","NUTS2","NUTS1","NUTS0",
"New_ID","NUTS3_2006","Region_2006",
"X","Y","EVOGDP_2000_2005_2006",
"SPAT_TYPE_1_1999_1999","SPAT_TYPE_2_1999_1999",
"UNEMP_R_2001_1999",
"LU_AS_1_1996_1999","LU_AS_2_1996_1999","LU_AS_3_1996_1999",
"LU_UF_1996_1999","LU_AR_1996_1999","LU_PC_1996_1999",
"NAT_HAZ_2004_1999",
"SF_CF_1999_1999",
"SF_R_1999_1999","SF_S_1999_1999","SF_A_1999_1999",
"CF_T_1999_1999","CF_E_1999_1999")

# Size of data set and adding an order ID...
n <- length(data1@data[,1])
Order_ID <- seq(1,n)
data1@data <- cbind(data1@data, Order_ID)
attach(data1@data)

# Coordinate data only...
coords <- cbind(data1@data[,9],data1@data[,10])

# Creating a shading scheme and plotting a choropleth map of EVOGDP_2000_2005_2006...
shades.1 = auto.shading(EVOGDP_2000_2005_2006,5, cols=brewer.pal(5,'PuBuGn'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,EVOGDP_2000_2005_2006,shades.1)
title("Evolution of GDP (2000 to 2005)")
choro.legend(-2300000,250000,shades.1,fmt="%4.0f",title='Evolution of GDP (%)',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")
```

```
# 3. Exploratory analyses for EVOGDP_2000_2005_2006 relationships ##############

data.1 <- cbind(EVOGDP_2000_2005_2006,X,Y)
data.2 <- cbind(EVOGDP_2000_2005_2006,UNEMP_R_2001_1999,NAT_HAZ_2004_1999)
data.3 <-
cbind(EVOGDP_2000_2005_2006,LU_AS_1_1996_1999,LU_AS_2_1996_1999,LU_AS_3_1996_1999)
data.4 <- cbind(EVOGDP_2000_2005_2006,LU_UF_1996_1999,LU_AR_1996_1999,LU_PC_1996_1999)
data.5 <- cbind(EVOGDP_2000_2005_2006,SF_R_1999_1999,SF_S_1999_1999,SF_A_1999_1999)
data.6 <- cbind(EVOGDP_2000_2005_2006,SF_CF_1999_1999,CF_T_1999_1999,CF_E_1999_1999)

cor(data.1,use="pairwise.complete.obs")
cor(data.2,use="pairwise.complete.obs")
cor(data.3,use="pairwise.complete.obs")
cor(data.4,use="pairwise.complete.obs")
cor(data.5,use="pairwise.complete.obs")
cor(data.6,use="pairwise.complete.obs")

X11(width=6,height=6)
pairs(data.1)
X11(width=6,height=6)
pairs(data.2)
X11(width=6,height=6)
pairs(data.3)
X11(width=6,height=6)
pairs(data.4)
X11(width=6,height=6)
pairs(data.5)
X11(width=6,height=6)
pairs(data.6)

X11(width=6,height=4)
boxplot(EVOGDP_2000_2005_2006~SPAT_TYPE_1_1999_1999,xlab="SPAT_TYPE_1_1999_1999",
ylab="EVOGDP_2000_2005_2006",cex=0.5, main="Evolution of GDP with Spatial typology 1")

X11(width=6,height=4)
boxplot(EVOGDP_2000_2005_2006~SPAT_TYPE_2_1999_1999,xlab="SPAT_TYPE_2_1999_1999",
ylab="EVOGDP_2000_2005_2006",cex=0.5, main="Evolution of GDP with Spatial typology 2")

# Exploratory investigations suggests that
# "X"
# "Y"
# "SF_CF_1999_1999"
# "SPAT_TYPE_2_1999_1999"
# have moderate relationships with EVOGDP_2000_2005_2006

# Coding for a categorical variable in a regression model using factor()...
SPAT_TYPE_2_1999_1999.f <- factor(SPAT_TYPE_2_1999_1999)

# For basic MLR analysis...
require(car)

# Full MLR model
mlr.1 <- lm(EVOGDP_2000_2005_2006 ~ X+Y+SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f)
summary(mlr.1)
vif(mlr.1) # Variance inflation factor (for collinearity)
AIC(mlr.1) # note R gives n*AIC

# AIC stepwise MLR model
mlr.2 <- step(mlr.1)
summary(mlr.2)
vif(mlr.2)
AIC(mlr.2)
```

```
# Results suggest that mlr.1 model is OK...

# We now assume (for section 4.) that the same explanatory variables
# are also important locally with LR and GWR...



# We can also investigate GW correlations using the spgwr function gw.cov

data.1 <- data1@data
coordinates(data.1) <- c("X", "Y")

# GW summary statistics at observation locations (i.e. region centroids)...
# Calculated using 10% of nearby data.
bwd.1 <- 0.1
gwss <- gw.cov(data.1, vars=c(11,22), adapt=bwd.1, cor = TRUE)
names(gwss$SDF) # The GW summary statistics calculated...

# GW correlations...
GW.corr <- gwss$SDF$cor.EVOGDP_2000_2005_2006.SF_CF_1999_1999.
summary(GW.corr) # some evidence of relationship nonstationarity...

# Updating information in one file
data1@data <- cbind(data1@data, GW.corr)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of GW correlations ...
shades.2 = auto.shading(GW.corr,5, cols=brewer.pal(5,'Greys'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,29],shades.2)
title("Relationship nonstationarity: Evolution of GDP with Str./Coh. Fund")
choro.legend(-2300000,250000,shades.2,fmt="%4.1f",title='Correlation',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")






# 4. Residual analysis with MLR, LR and GWR models ##############################


# Using raw residuals from mlr-1 fit...
raw.resids.mlr <- EVOGDP_2000_2005_2006-mlr.1$fitted # Actual minus prediction
summary(raw.resids.mlr)

# Identifying & updating outlier information in one file
cut.off.1 <- quantile(raw.resids.mlr, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails - alter accordingly...
indicator.1 <-ifelse(raw.resids.mlr>=cut.off.1[2] & raw.resids.mlr<=cut.off.1[20], 0, 1)
data1@data <- cbind(data1@data, raw.resids.mlr, indicator.1)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for MLR...
```

```
shades.3 = shading(c(cut.off.1[2],cut.off.1[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,30],shades.3)
title("Raw resids. from MLR: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")



# LR using locfit...
require(locfit)
# Ignore warning message...

# Finding the bandwidth for a non-robust LR (i.e. not a lowess fit)
# using generalised cross-validation (GCV) approach.
summary(gcvplot(EVOGDP_2000_2005_2006~X+Y+SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f,
data=data.1, scale=F,alpha=seq(0.1,1,by=0.1),
deg=1,kern="tricube",lfproc=locfit.raw))

# Choosing a LR fit with bandwidth chosen from above...
bwd.2 <- 0.7
lr <- locfit(EVOGDP_2000_2005_2006~X+Y+SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f,
data=data.1, scale=F, alpha=bwd.2,deg=1,kern="tricube",lfproc=locfit.raw)

# Raw residuals...
lr.p <- fitted.locfit(lr)
raw.resids.lr <- EVOGDP_2000_2005_2006-lr.p # Actual minus prediction
summary(raw.resids.lr)

# Identifying & updating outlier information in one file
cut.off.2 <- quantile(raw.resids.lr, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails - alter accordingly...
indicator.2 <-ifelse(raw.resids.lr>=cut.off.2[2] & raw.resids.lr<=cut.off.2[20], 0, 1)
data1@data <- cbind(data1@data, raw.resids.lr, indicator.2)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for LR...
shades.4 = shading(c(cut.off.2[2],cut.off.2[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,32],shades.4)
title("Raw resids. from LR: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")



# GWR using spgwr...
# Defining the coordinates...
coords.1<-cbind(data.1[,9],data.1[,10])

# Finding the bandwidth for GWR using Akaike Information Criterion (AIC) approach.
#gwr.aic.bwd <-gwr.sel(EVOGDP_2000_2005_2006~X+Y+SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f,
#data=data.1,coords=coords.1,adapt=TRUE,
#gweight=gwr.bisquare, method="aic")
#gwr.aic.bwd[1] # the optimum bandwidth

# Or finding the bandwidth for GWR using cross-validation (CV) approach.
gwr.cv.bwd <-gwr.sel(EVOGDP_2000_2005_2006~X+Y+SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f,
data=data.1,coords=coords.1,adapt=TRUE,
gweight=gwr.bisquare, method="cv")
gwr.cv.bwd[1] # the optimum bandwidth

# Using CV bandwidth...
bwd.3 <- gwr.cv.bwd[1]
gwr.p <-gwr(EVOGDP_2000_2005_2006~X+Y+SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f,
data=data.1,coords=coords.1,adapt=bwd.3,gweight=gwr.bisquare,predictions=T)
#gwr.p$SDF

# GWR raw residuals...
raw.resids.gwr <- EVOGDP_2000_2005_2006-gwr.p$SDF$pred
summary(raw.resids.gwr)

# Identifying & updating outlier information in one file
cut.off.3 <- quantile(raw.resids.gwr, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails - alter accordingly...
indicator.3 <-ifelse(raw.resids.gwr>=cut.off.3[2] & raw.resids.gwr<=cut.off.3[20], 0, 1)
data1@data <- cbind(data1@data, raw.resids.gwr, indicator.3)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for GWR...
shades.5 = shading(c(cut.off.3[2],cut.off.3[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,34],shades.5)
title("Raw resids. from GWR: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")



# NB - Future work should explore the identification of outliers using
# standardiased residuals & corresponding robust regression models...








# 5.  All identified outliers together ##############################################

# Put all indicator data together...
indicator.4 <- indicator.1+indicator.2+indicator.3
summary(indicator.4)
# Histogram
X11(width=5.3,height=5.7)
hist(indicator.4,br=c(0,1,2,3))

# Thus a strong case for an outlier relates to an observation
# that has a indicator.4 value of 3...

data1@data <- cbind(data1@data, indicator.4)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of suspected outliers...
shades.6 = shading(c(1,2,3),c("white","yellow","orange","red"))
```

```
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,36],shades.6)
title("Suspected outliers - weak to strong (yellow to red) evidence")
choro.legend(-2300000,250000,shades.6,over="exactly", between="to under",
fmt="%4.0f",title='Indicator sum (max.: 3)',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")
```

Appendix 5 – R script for worked example 5

```
# 1. Preamble ######################################################################

# Worked example 5 - for technical report - challenge 10 - ESPON 2013 database
# NCG - P. Harris & M. Charlton
# 7/2/10

# Objective - to identify statistical clusters in "EVOGDP_2000_2005_2006" with
# respect to key spatial moments...
# GW means/standard deviations; Relationships - GW correlations/regressions;
# Spatial autocorrelation (Moran's I)
# For relationships -
# EVOGDP_2000_2005_2006 is related to some subset of the following explanatory/contextual data:
# "X"
# "Y"
# "SPAT_TYPE_1_1999_1999"
# "SPAT_TYPE_2_1999_1999"
# "UNEMP_R_2001_1999"
# "LU_AS_1_1996_1999"
# "LU_AS_2_1996_1999"
# "LU_AS_3_1996_1999"
# "LU_UF_1996_1999"
# "LU_AR_1996_1999"
# "LU_PC_1996_1999"
# "NAT_HAZ_2004_1999"
# "SF_CF_1999_1999"
# "SF_R_1999_1999"
# "SF_S_1999_1999"
# "SF_A_1999_1999"
# "CF_T_1999_1999"
# "CF_E_1999_1999"

# Methods: univariate & multivariate - all spatial
# Only statistical methods:
# 1. GWSS (geographically weighted summary statistics)
# 2. MLR (multiple linear regression) & GWR (geographically weighted regression)
# 3. Global and local Moran's I



# R packages needed.....
# 1. GISTools (version 0.5-4) - depends on 2 to 11...
# 2. foreign (version 0.8-30)
# 3. gpclib (version 1.4-3)
# 4. maptools (version 0.7-16)
# 5. Matrix (version 0.999375-18)
# 6. RColorBrewer (version 1.0-2)
# 7. sp (version 0.9-28)
# 8. spam (version 0.15-2)
# 9. spdep (version 0.4-29) - for global and local Moran's I
# 10. spgwr (version 0.6-2) - for GWSS and GWR
# 11. tripack (version 1.2-11)
# 12. car (version 1.2-12) - for MLR

# Base R system version 2.9.0
# N.B. Some of the above packages may still depend on other R packages - download these from R website...



# Relevant data files (see data & ArcGIS directories):

# Excel files...
# 1. ESPON_DATA_NCG_CHALLANGE_10_original.xls
# 2. ESPON_DATA_NCG_CHALLANGE_10_subsets.xls

# ArcGIS files...
# 3. Worked_example_345a_reduced.shp - ArcGIS shapefile of the data...



# The 27 variables...

# "NUTS3","NUTS23","NUTS2","NUTS1","NUTS0" - 5 different NUTS levels
# "New_ID" - relates to the regions name only & is purely numeric
# "NUTS3_2006" - the 2006 NUTS3 version
# "Region_2006" - name of 2006 NUTS3 version
# "X","Y" - centroids of regions
# "EVOGDP_2000_2005_2006" - Evolution of GDP
# and 16 likely contextual variables of "EVOGDP_2000_2005_2006" ...
# "SPAT_TYPE_1_1999_1999"
# "SPAT_TYPE_2_1999_1999"
# "UNEMP_R_2001_1999"
# "LU_AS_1_1996_1999"
# "LU_AS_2_1996_1999"
# "LU_AS_3_1996_1999"
# "LU_UF_1996_1999"
# "LU_AR_1996_1999"
# "LU_PC_1996_1999"
# "NAT_HAZ_2004_1999"
# "SF_CF_1999_1999"
# "SF_R_1999_1999"
# "SF_S_1999_1999"
# "SF_A_1999_1999"
# "CF_T_1999_1999"
# "CF_E_1999_1999"



# NOTE - This example data set has been reduced to 731 values from an original 1351 values
# see readme in excel files on worked example data.


# 2. Importing data as a ArcGIS shapefile & using GISTools to do some maps #####

require(GISTools)
#help(GISTools)
# Ignore all warnings - this code is under development...

# Read in the shapefile...
data1 <- readShapePoly("Worked_example_345a_reduced.shp",
proj4string=CRS("+proj=Lambert_Azimuthal_Equal_Area+datum=D_ETRS_1989+ellps=GCS_ETRS_1989"))
colnames(data1@data)

# renaming each variable - as they have been truncated in ArcGIS...
colnames(data1@data) <- c("NUTS3","NUTS23","NUTS2","NUTS1","NUTS0",
"New_ID","NUTS3_2006","Region_2006",
"X","Y","EVOGDP_2000_2005_2006",
"SPAT_TYPE_1_1999_1999","SPAT_TYPE_2_1999_1999",
"UNEMP_R_2001_1999",
"LU_AS_1_1996_1999","LU_AS_2_1996_1999","LU_AS_3_1996_1999",
"LU_UF_1996_1999","LU_AR_1996_1999","LU_PC_1996_1999",
"NAT_HAZ_2004_1999",
"SF_CF_1999_1999",
"SF_R_1999_1999","SF_S_1999_1999","SF_A_1999_1999",
"CF_T_1999_1999","CF_E_1999_1999")

# Size of data set and adding an order ID...
n <- length(data1@data[,1])
Order_ID <- seq(1,n)
data1@data <- cbind(data1@data, Order_ID)
attach(data1@data)

# Coordinate data only...
coords <- cbind(data1@data[,9],data1@data[,10])

# Creating a shading scheme and plotting a choropleth map of EVOGDP_2000_2005_2006...
shades.1 = auto.shading(EVOGDP_2000_2005_2006,5, cols=brewer.pal(5,'YlOrBr'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,EVOGDP_2000_2005_2006,shades.1)
title("Evolution of GDP (2000 to 2005)")
choro.legend(-2300000,250000,shades.1,fmt="%4.0f",title='Evolution of GDP (%)',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")


# 3. Exploratory analyses for EVOGDP_2000_2005_2006 spatial moments ###########

# Global summary statistics...
summary(EVOGDP_2000_2005_2006)
```

```
# standard deviation
sd <- (var(EVOGDP_2000_2005_2006))^0.5
sd



# As in worked example 4 we can investigate GW summary statistics
# using the spgwr function gw.cov

data.1 <- data1@data
coordinates(data.1) <- c("X", "Y")

# GW summary statistics at observation locations (i.e. region centroids)...
# Calculated using 10% of nearby data.
bwd.1 <- 0.1
gwss <- gw.cov(data.1, vars=c(11,22), adapt=bwd.1, cor = TRUE)
names(gwss$SDF) # The GW summary statistics calculated...



# GW means...
GW.mean <- gwss$SDF$mean.EVOGDP_2000_2005_2006
summary(GW.mean) # some evidence of mean nonstationarity...

# Updating information in one file
data1@data <- cbind(data1@data, GW.mean)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of GW means ...
shades.2 = auto.shading(GW.mean,5, cols=brewer.pal(5,'YlOrBr'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,29],shades.2)
title("Mean nonstationarity: Evolution of GDP")
choro.legend(-2300000,250000,shades.2,fmt="%4.1f",title='Mean',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")



# GW standard deviations...
GW.sd <- gwss$SDF$sd.EVOGDP_2000_2005_2006
summary(GW.sd) # some evidence of SD nonstationarity...

# Updating information in one file
data1@data <- cbind(data1@data, GW.sd)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of GW SDs ...
shades.3 = auto.shading(GW.sd,5, cols=brewer.pal(5,'RdPu'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,30],shades.3)
title("Standard Deviation nonstationarity: Evolution of GDP")
choro.legend(-2300000,250000,shades.3,fmt="%4.1f",title='SD',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")
```

```
# For all GW summary statistics randomisation tests can be used to identfy
# locations of unusually different local statistics -
# see Fotheringham et al. (2002); Harris and Brunsdon (2010)

# As an example: randomisation test for the standard deviation...
n.r1 <- 99  # Number of randomisations (the more the better)
out.x <- matrix(nrow=n,ncol=n.r1)
for(i in 1:n.r1)
{
rand.dat <- sample(data.1[,11])
data.2 <- cbind(data1@data, rand.dat)
data.2 <- as.data.frame(data.2)
attach(data.2)
coordinates(data.2) <- c("X", "Y")
gwss.rand <- gw.cov(data.2, vars=c(31), adapt=bwd.1)
out.x[,i] <- gwss.rand$SDF$sd.V1
}
# combining the randomisation results with the actual result...
out.x1 <- cbind(GW.sd, out.x)
out.x2 <- t(apply(out.x1,1,rank))
Random.sd <- out.x2[,1]

# Updating information in one file
data1@data <- cbind(data1@data, Random.sd)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of randomisation test result ...
shades.4 = shading(c(2.5,97.5),c("blue","white","green")) # test at 95% level...
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,31],shades.4)
title("Areas of unsually high (green) and low (blue) standard deviation")
map.scale(100000,-750000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="red")
text(-320000,-1150000, "Randomisation test at 95% level for Evolution of GDP")








# 4. Exploratory analyses for EVOGDP_2000_2005_2006 w.r.t global relationships #

# From exploratory investigations of worked example 4:
# "SF_CF_1999_1999"
# "SPAT_TYPE_2_1999_1999"
# have moderate relationships with EVOGDP_2000_2005_2006
# i.e.

# Correlation & scatterplot...
cor(EVOGDP_2000_2005_2006,SF_CF_1999_1999,use="pairwise.complete.obs")
X11(width=6,height=6)
plot(EVOGDP_2000_2005_2006,SF_CF_1999_1999, main="Evolution of GDP with Str/Coh Fund",
pch=19, cex=0.5)
```

```
# Boxplot for categorical variable...
X11(width=6,height=4)
boxplot(EVOGDP_2000_2005_2006~SPAT_TYPE_2_1999_1999,xlab="SPAT_TYPE_2_1999_1999",
ylab="EVOGDP_2000_2005_2006",cex=0.5, main="Evolution of GDP with Spatial typology 2")

# Coding for a categorical variable in a regression model using factor()...
SPAT_TYPE_2_1999_1999.f <- factor(SPAT_TYPE_2_1999_1999)

# For useful basic MLR analysis...
require(car)

# Full MLR model
mlr.1 <- lm(EVOGDP_2000_2005_2006 ~ SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f)
summary(mlr.1)
vif(mlr.1) # Variance inflation factor (for collinearity)
AIC(mlr.1) # note R gives n*AIC

# AIC stepwise MLR model
mlr.2 <- step(mlr.1)
summary(mlr.2)
vif(mlr.2)
AIC(mlr.2)

# Results suggest that mlr.1 model is OK...

# We also assume that the same explanatory variables
# are also important locally with GWR...








# 5. Exploratory analyses for EVOGDP_2000_2005_2006 w.r.t local relationships ##

# We can also investigate GW correlations from the spgwr function gw.cov output in section 3.

# GW correlations...
GW.corr <- gwss$SDF$cor.EVOGDP_2000_2005_2006.SF_CF_1999_1999.
summary(GW.corr) # some evidence of relationship nonstationarity...

# Updating information in one file
data1@data <- cbind(data1@data, GW.corr)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of GW correlations ...
shades.5 = auto.shading(GW.corr,5, cols=brewer.pal(5,'Greys'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,32],shades.5)
title("Relationship nonstationarity: Evolution of GDP with Str./Coh. Fund")
choro.legend(-2300000,250000,shades.5,fmt="%4.1f",title='Correlation',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")



# Now using GWR to explore local relationships (using spgwr)...
# Defining the coordinates...
coords.1<-cbind(data.1[,9],data.1[,10])

# Finding the bandwidth for GWR using cross-validation (CV) approach.
gwr.cv.bwd <-gwr.sel(EVOGDP_2000_2005_2006~SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f,
data=data.1,coords=coords.1,adapt=TRUE,
gweight=gwr.bisquare, method="cv")
gwr.cv.bwd[1] # the optimum bandwidth

# GWR using CV bandwidth...
bwd.2 <- gwr.cv.bwd[1]
gwr.1 <-gwr(EVOGDP_2000_2005_2006~SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f,
data=data.1,coords=coords.1,adapt=bwd.2,gweight=gwr.bisquare)
#gwr.1$SDF

# As an example, only investigating coefficients (or parameters) for SPAT_TYPE_2_1999_1999 class 2
gwr.coeff.1 <- gwr.1$SDF$SPAT_TYPE_2_1999_1999.f2

# Updating information in one file
data1@data <- cbind(data1@data, gwr.coeff.1)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of this particular part of the GWR output ...
shades.6 = auto.shading(gwr.coeff.1, 5, cols=brewer.pal(5,'Blues'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,33],shades.6)
title("Relationship nonstationarity: coefficient for spatial typology 2 (class 2)")
choro.legend(-2300000,250000,shades.6,fmt="%1.2f",title='GWR coefficient',cex=0.8)
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="green")



# Randomisation test -
# To gauge whether or not any observed variation in the GWR coefficients is unusual...

# First get the actual variance of each local coefficient estimate...
output.x1 <- as(gwr.1$SDF, "data.frame")[,2:8]
output.1 <- as.vector(apply(output.x1,2,var))

# Now find the same variances based on 99 randomisations of the data...
output.2 <- matrix(nrow=n.r1,ncol=7) # i.e. 7 columns (intercept plus number of indep. variables)
for(i in 1:n.r1)
{
print(i) # counter
c1<-t(coords.1)
c1 <- as.data.frame(c1)
c1.s <-sample(c1,length(data.1[,1]))
c1.s <- as.data.frame(c1.s)
c1.s <- t(c1.s)
coords.2 <- as.matrix(c1.s)
gwr.2 <-gwr(EVOGDP_2000_2005_2006~SF_CF_1999_1999+SPAT_TYPE_2_1999_1999.f,
data=data.1,coords=coords.2,adapt=bwd.2,gweight=gwr.bisquare)
output.x2 <- as(gwr.2$SDF, "data.frame")[,2:8]
output.2[i,] <- as.vector(apply(output.x2,2,var))
}

# p-values for each coefficient estimate
```

```
output.3 <- rbind(output.1,output.2)
r.1 <- rank(output.3[,1])
r.11 <- ((n.r1+2)-r.1[1])/(n.r1+1)
r.2 <- rank(output.3[,2])
r.22 <- ((n.r1+2)-r.2[1])/(n.r1+1)
r.3 <- rank(output.3[,3])
r.33 <- ((n.r1+2)-r.3[1])/(n.r1+1)
r.4 <- rank(output.3[,4])
r.44 <- ((n.r1+2)-r.4[1])/(n.r1+1)
r.5 <- rank(output.3[,5])
r.55 <- ((n.r1+2)-r.5[1])/(n.r1+1)
r.6 <- rank(output.3[,6])
r.66 <- ((n.r1+2)-r.6[1])/(n.r1+1)
r.7 <- rank(output.3[,7])
r.77 <- ((n.r1+2)-r.7[1])/(n.r1+1)

# Thus in this case, all Monte Carlo tests are based on 99 randomisations of the data.
# The larger the p-value,the more support is given to the null hypothesis
# of a stationary regression coefficient estimate.
rand.test.1 <- cbind(r.11,r.22,r.33,r.44,r.55,r.66,r.77)
rand.test.1








# 6. Global and local autocorrelation ##########################################

# Global Moran's I
# Local Moran's I (a Local Indicator of Spatial Association LISA)

require(spdep) # for global and local Moran's I

# Firstly, two different examples to define spatial distances or spatial topology.
# A measure of distance is needed to calculate local and global Moran's I statistics.

data1.labs = poly.labels(data1)

# 1. Queen's case spatial topology (from chess)
data1.nb1 = poly2nb(data1)
X11(width=8,height=7)
par(mar=c(0,0,2,0))
plot(data1,col='grey')
plot(data1.nb1,coordinates(data1.labs),col='red',add=TRUE)
title("Queen's case spatial topology")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")

# 2. Alternatively, base topology on nearness of polygons rather than contiguity.
# Here we define two polygons as neighbours if they are within some distance d of one another.
# Thus letting d = 100000m, for example...
data1.nb2 = dnearneigh(poly.labels(data1),0,100000)
X11(width=8,height=7)
par(mar=c(0,0,2,0))
plot(data1,col='grey')
plot(data1.nb2,coordinates(data1.labs),col='red',add=TRUE)
title("Regions whose centroids are within 100km of each other")
map.scale(100000,-1050000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="blue")
```

```
# Tests for global Moran's I statistic using 'Queens case topology' contiguity matrix
data1.lw = nb2listw(data1.nb1,zero.policy=T)
attach(data.frame(data1))

# Global Moran's I:
moran(EVOGDP_2000_2005_2006, data1.lw, n, Szero(data1.lw), zero.policy=T)

# The assumptions underlying the following test are sensitive to the form
# of the graph of neighbour relationships and other factors.
# Results may be checked against those of moran.mc permutations.
moran.test(EVOGDP_2000_2005_2006,data1.lw, zero.policy=T)

# A permutation test for Moran’s I statistic calculated by using nsim random permutations of x for
# the given spatial weighting scheme, to establish the rank of the observed statistic in relation to the
# nsim simulated values.
moran.mc(EVOGDP_2000_2005_2006,data1.lw, zero.policy=T,nsim=10000)



# Local Moran's I also using 'Queens case topology' contiguity matrix
Local.moran <- localmoran(EVOGDP_2000_2005_2006,data1.lw, zero.policy=T)

# Summary of local Moran's I
summary(Local.moran)

# Updating information in one file
data1@data <- cbind(data1@data, Local.moran[,1])
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of local Moran's I for EVOGDP_2000_2005_2006...
shades.7 = shading(c(0,0.572),c("red","grey","blue")) # shading relates to global value...
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,34],shades.7)
title("Autocorrelation nonstationarity: Local Moran's I for Evolution of GDP")
map.scale(100000,-750000,500000,"x1000 km",4,0.5)
north.arrow(200000,750000,40000, col="black")
text(-360000,-1150000, "-ve (red); +ve/below global (grey); +ve/above global (blue)")
```

Appendix 6 – R script for worked example 6

```
# 1. Preamble ##########################################################################


# Worked example 6 - for technical report - challenge 10 - ESPON 2013 database
# NCG - P. Harris & M. Charlton
# 8/2/10



# Objective - to identify statistical outliers in:
# "EVOGDP_2000_2005_2006" at three different NUTS levels (2, 1 & 0)
# - i.e. investigate the effects of MAUP on outlier identification
# Results are compared with those of worked example 2 for NUTS3.



# Methods: univariate - aspatial & spatial (i.e. the same as that presented in worked example 2)
# Only statistical methods:
# 1. Standard and Adjusted boxplots,
# 2. Hawkins' test (includes the use of GWSS -
#            geographically weighted summary statistics - GW means and variances),
# 3. LM (local mean, i.e. a GW mean)
# 4. MLR (multiple linear regression),
# 5. LR (local regression) &
# 6. GWR (geographically weighted regression)



# R packages needed.....
# 1. GISTools (version 0.5-4) - depends on 2 to 11...
# 2. foreign (version 0.8-30)
# 3. gpclib (version 1.4-3)
# 4. maptools (version 0.7-16)
# 5. Matrix (version 0.999375-18)
# 6. RColorBrewer (version 1.0-2)
# 7. sp (version 0.9-28)
# 8. spam (version 0.15-2)
# 9. spdep (version 0.4-29)
# 10. spgwr (version 0.6-2) - for GWSS & GWR
# 11. tripack (version 1.2-11)
# 12. moments (version 0.11) - for skewness
# 13. robustbase (version 0.4-5) - for adjusted boxplots
# 14. locfit (version 1.5-4)- for LR

# Base R system version 2.9.0
# N.B. Some of the above packages may still depend on other R packages -
# download these from R website...



# Relevant data files (see data & ArcGIS directories):

# Excel files...
# 1. ESPON_DATA_NCG_CHALLANGE_10_original.xls
# 2. ESPON_DATA_NCG_CHALLANGE_10_subsets.xls

# ArcGIS files...
# 3. Worked_example_6c_Dissolve_nuts2a.shp - ArcGIS shapefile of the NUTS2 data (278 values)
# 4. Worked_example_6c_Dissolve_nuts1a.shp - ArcGIS shapefile of the NUTS1 data (95 values)
# 5. Worked_example_6c_Dissolve_nuts0a.shp - ArcGIS shapefile of the NUTS0 data (30 values)



# The 10 variables...

# "NUTSx", - the NUTS level - 2, 1 or 0
# "NUTS3_outlier_mean" - mean of outlier indicator.8 from worked example 2
#                    when going from NUTS 3 to larger scale
# "NUTS3_outlier_max" - maximum of outlier indicator.8 from worked example 2
#                    when going from NUTS 3 to larger scale
# "GDP_2000_2006" - mean of NUTS3 data when going from NUTS 3 to larger scale
# "GDP_2005_2006" - mean of NUTS3 data when going from NUTS 3 to larger scale
# "POP_T_2000_2006" - mean of NUTS3 data when going from NUTS 3 to larger scale
# "POP_T_2005_2006" - mean of NUTS3 data when going from NUTS 3 to larger scale
# "X","Y" - centroids of NUTS regions
# "EVOGDP_2000_2005_2006" - the variable of interest re-calculated from
#                      the relevant variables above




# Change the following script in six places to go through each NUTS level...








# 2. Importing data as a ArcGIS shapefile & using GISTools to do a map... ######

require(GISTools)
#help(GISTools)
# Ignore all warnings - this code is under development...

# Read in the shapefile...
#data1 <- readShapePoly("Worked_example_6c_Dissolve_nuts2a.shp",
data1 <- readShapePoly("Worked_example_6c_Dissolve_nuts1a.shp",
#data1 <- readShapePoly("Worked_example_6c_Dissolve_nuts0a.shp",
proj4string=CRS("+proj=Lambert_Azimuthal_Equal_Area+datum=D_ETRS_1989+ellps=GCS_ETRS_1989"))
colnames(data1@data)

# Renaming each variable - as they have been altered by ArcGIS commands...
colnames(data1@data) <- c("NUTS2", "NUTS3_outlier_mean", "NUTS3_outlier_max",
"GDP_2000_2006", "GDP_2005_2006", "POP_T_2000_2006", "POP_T_2005_2006",
"X","Y")

# Size of data set and adding an order ID...
n <- length(data1@data[,1])
Order_ID <- seq(1,n)
data1@data <- cbind(data1@data, Order_ID)
attach(data1@data)

# Calculating the new EVOGDP_2000_2005_2006 values...
EVOGDP_2000_2005_2006 <-
((data1@data[,5]/data1@data[,7])*1000)/((data1@data[,4]/data1@data[,6])*1000)*100
data1@data <- cbind(data1@data, EVOGDP_2000_2005_2006)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Creating a shading scheme and plotting a choropleth map of EVOGDP_2000_2005_2006...
shades.1 = auto.shading(EVOGDP_2000_2005_2006,5, cols=brewer.pal(5,'Greys'))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,EVOGDP_2000_2005_2006,shades.1)
title("Evolution of GDP (2000 to 2005)")
choro.legend(-2400000,2200000,shades.1,fmt="%4.0f",title='Evolution of GDP (%)',cex=0.8)
```

```
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")
#text(1500000,2400000, "NUTS level 2", cex=2, col=3)
text(1500000,2400000, "NUTS level 1", cex=2, col=3)
#text(1500000,2400000, "NUTS level 0", cex=2, col=3)









# 3. Boxplots ################################################################

# Let EVOGDP_2000_2005_2006 be z1...
z1 <- EVOGDP_2000_2005_2006

# Exploring this data...
summary(z1) # summary statistics
sort(z1) # ordered data

# Histogram
X11(width=5.3,height=5.7)
hist(z1, main="Histogram: Evolution of GDP (2000 to 2005)",xlab="Evolution of GDP")

# Standard boxplot with defaults
X11(width=5.3,height=5.7)
boxplot(z1, main="Std. boxplot: Evolution of GDP (2000 to 2005)", pch=19, cex=0.5)

# Standard Boxplot statistics...
# Change 'coef' accordingly...
# Default 'coef' is 1.5...
# The higher the 'coef' value the stricter the limits/cut-offs & vice versa...
bp <- boxplot.stats(z1, coef=1.5)
bp$stats
bp$stats[1] # the lower limit/cut-off - i.e. values below are deemed outlying...
bp$stats[5] # the upper limit/cut-off - i.e. values above are deemed outlying...
bp$conf
sort(bp$out)
length(bp$out) # number of potential outliers...
# help(boxplot.stats) # for details...

# Identifying & updating outlier information in one file
indicator.1 <-ifelse(z1>bp$stats[1]& z1<bp$stats[5], 0, 1) # i.e. suspected outliers...
data1@data <- cbind(data1@data, indicator.1)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of standard boxplot outliers
shades.2 = shading(c(0,1,2),c("blue","white","red")) # i.e. white - no & red - yes
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,12],shades.2)
title("Std. boxplot outliers (regions coloured red)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")

# Need moments package to assess skewness (before adjusted boxplots)
require(moments)
# Ignore warning message...
```

```
skewness(z1)

# Package for adjusted boxplots...
require(robustbase)

# Adjusted boxplot with defaults
X11(width=5.3,height=5.7)
adjbox(z1, main="Adj. boxplot: Evolution of GDP (2000 to 2005)", pch=19, cex=0.5)

# Adjusted Boxplot statistics...
# Change 'coef' accordingly...
# Default 'coef' is 1.5...
# The higher the 'coef' value the stricter the limits/cut-offs & vice versa...
abp <- adjboxStats(z1, coef=1.5)
abp$stats
abp$stats[1] # the lower limit/cut-off - i.e. values below are deemed outlying...
abp$stats[5] # the upper limit/cut-off - i.e. values above are deemed outlying...
abp$conf
sort(abp$out)
length(abp$out) # number of potential outliers...
#help(adjboxStats) # for details...

# Identifying & updating outlier information in one file
indicator.2 <-ifelse(z1>abp$stats[1]& z1<abp$stats[5], 0, 1) # i.e. suspected outliers...
data1@data <- cbind(data1@data, indicator.2)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of adjusted boxplot outliers
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,13],shades.2)
title("Adj. boxplot outliers (regions coloured red)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")







# 4. GW summary statistics and Hawkins' Spatial Outlier Test ##################

# First need to find GW means (i.e. LMs) and GW variances for Hawkin's test...
# In this case, using the gw.cov function in spgwr to find the GW means/variances...

# To re-cap...
colnames(data.1)

# Defining coordinates....
coordinates(data.1) <- c("X", "Y")

# GW summary statistics at observation locations (i.e. region centroids)...
# Calculated using 10% of nearby EVOGDP_2000_2005_2006 data.
bwd.1 <- 0.1
gwss <- gw.cov(data.1, vars=11, adapt=bwd.1)
#help(gw.cov) # for details...
names(gwss$SDF) # The GW summary statistics calculated...
```

```
# GW means and variances...
GW.mean <- gwss$SDF$mean.V1
GW.variance <- (gwss$SDF$sd.V1)^2

# Hawkins' Test for Spatial Outliers...
Hawk.N <- bwd.1*length(X) # number of neighbouring data
Hawk.lm <- GW.mean # the local mean at observation points
Hawk.alv <- mean(GW.variance) # the average local variance with same bandwidth

Hawk.test <- (Hawk.N*(EVOGDP_2000_2005_2006-Hawk.lm)^2)/((Hawk.N+1)*Hawk.alv)  # test statistic
summary(Hawk.test)

# Critical values of the chi-squared distribution
chi_10 <- 2.70554
chi_5 <- 3.84146
chi_2.5 <- 5.02389
chi_1 <- 6.63490
chi_0.5 <- 7.87944
chi_0.01 <- 10.828

# Updating outlier information in one file
indicator.3 <-ifelse(Hawk.test <=chi_5, 0, 1) # change critical level accordingly...
data1@data <- cbind(data1@data, Hawk.test, indicator.3)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of spatial outliers ...
shades.3 = shading(c(chi_5,chi_1,chi_0.01),c("white","yellow","orange","red"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,14],shades.3)
title("Spatial outliers: at 5/1/0.01 % (yellow/orange/red) critical levels")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")



# 5. Residual analysis with LM, MLR, LR and GWR models ########################



# LM...
# Using GW.mean from before...
GW.mean.r <- EVOGDP_2000_2005_2006-GW.mean # Actual minus prediction
summary(GW.mean.r)

# Identifying & updating outlier information in one file
cut.off.1 <- quantile(GW.mean.r, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails
indicator.4 <-ifelse(GW.mean.r>=cut.off.1[2] & GW.mean.r<=cut.off.1[20], 0, 1)
data1@data <- cbind(data1@data, GW.mean.r, indicator.4)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for LM...
```

```
shades.4 = shading(c(cut.off.1[2],cut.off.1[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,16],shades.4)
title("Raw resids. from LM: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")



# MLR...
# First- & second-order polynomial fits of the coordinate data...
mlr.1 <- lm(EVOGDP_2000_2005_2006 ~ X+Y)
mlr.2 <- lm(EVOGDP_2000_2005_2006 ~ X+Y+I(X^2)+I(Y^2)+I(X*Y))
summary(mlr.1)
summary(mlr.2)

# Choosing a second-order MLR fit...

# Using raw residuals as in LM fit...
raw.resids.mlr <- EVOGDP_2000_2005_2006-mlr.2$fitted # Actual minus prediction
summary(raw.resids.mlr)

# Identifying & updating outlier information in one file
cut.off.2 <- quantile(raw.resids.mlr, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails
indicator.5 <-ifelse(raw.resids.mlr>=cut.off.2[2] & raw.resids.mlr<=cut.off.2[20], 0, 1)
data1@data <- cbind(data1@data, raw.resids.mlr, indicator.5)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for MLR...
shades.5 = shading(c(cut.off.2[2],cut.off.2[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,18],shades.5)
title("Raw resids. from MLR: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")



# LR...
# With coordinate data as explanatory variables (i.e. first-order polynomial).

# Using locfit...
require(locfit)
# Ignore warning message...

# Finding the bandwidth for a non-robust LR (i.e. not a lowess fit)
# using generalised cross-validation (GCV) approach.
summary(gcvplot(EVOGDP_2000_2005_2006~X+Y,data=data.1, scale=F,
#alpha=seq(0.02,0.1,by=0.01), # for NUTS2
alpha=seq(0.1,0.2,by=0.01), # for NUTS1
#alpha=seq(0.2,1,by=0.1), # for NUTS0
deg=1,kern="tricube",lfproc=locfit.raw))

# Choosing a LR fit with bandwidth chosen from above...
#bwd.2 <- 0.03 # for NUTS2
bwd.2 <- 0.2 # for NUTS1
#bwd.2 <- 0.6 # for NUTS0
lr <- locfit(EVOGDP_2000_2005_2006~X+Y,data=data.1, scale=F, alpha=bwd.2,
deg=1,kern="tricube",lfproc=locfit.raw)

# Raw residuals...
lr.p <- fitted.locfit(lr)
raw.resids.lr <- EVOGDP_2000_2005_2006-lr.p # Actual minus prediction
summary(raw.resids.lr)

# Identifying & updating outlier information in one file
cut.off.3 <- quantile(raw.resids.lr, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails
indicator.6 <-ifelse(raw.resids.lr>=cut.off.3[2] & raw.resids.lr<=cut.off.3[20], 0, 1)
data1@data <- cbind(data1@data, raw.resids.lr, indicator.6)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for LR...
shades.6 = shading(c(cut.off.3[2],cut.off.3[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,20],shades.6)
title("Raw resids. from LR: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")



# GWR...
# With coordinate data as explanatory variables (i.e. first-order polynomial).
# Using spgwr...

# Defining the coordinates...
coords.1<-cbind(data.1[,8],data.1[,9])

# Finding the bandwidth for GWR using Akaike Information Criterion (AIC) approach.
#gwr.aic.bwd <-gwr.sel(EVOGDP_2000_2005_2006~X+Y,data=data.1,coords=coords.1,adapt=TRUE,
#gweight=gwr.bisquare, method="aic")
#gwr.aic.bwd[1] # the optimum bandwidth

# Or finding the bandwidth for GWR using cross-validation approach.
gwr.cv.bwd <-gwr.sel(EVOGDP_2000_2005_2006~X+Y,data=data.1,coords=coords.1,adapt=TRUE,
gweight=gwr.bisquare, method="cv")
gwr.cv.bwd[1] # the optimum bandwidth

bwd.3 <- gwr.cv.bwd[1]
gwr.p <-gwr(EVOGDP_2000_2005_2006~X+Y,data=data.1,coords=coords.1,adapt=bwd.3,
gweight=gwr.bisquare,predictions=T)
#gwr.p$SDF

# GWR raw residuals...
raw.resids.gwr <- EVOGDP_2000_2005_2006-gwr.p$SDF$pred
summary(raw.resids.gwr)

# Identifying & updating outlier information in one file
cut.off.4 <- quantile(raw.resids.gwr, probs = seq(0, 1, 0.05), na.rm=T) # for 5% tails
indicator.7 <-ifelse(raw.resids.gwr>=cut.off.4[2] & raw.resids.gwr<=cut.off.4[20], 0, 1)
data1@data <- cbind(data1@data, raw.resids.gwr, indicator.7)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# Raw residual map for GWR...
```

```
shades.7 = shading(c(cut.off.4[2],cut.off.4[20]),c("red","white","black"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,22],shades.7)
title("Raw resids. from GWR: High/-ve(red) & High/+ve(black) (5% tails)")
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")







# 6.  All identified outliers together ###########################################

# Put all indicator data together...
indicator.8 <- indicator.1+indicator.2+indicator.3+indicator.4+indicator.5+indicator.6+indicator.7
summary(indicator.8)
# Histogram
X11(width=5.3,height=5.7)
hist(indicator.8,br=c(0,1,2,3,4,5,6,7))

# Thus a strong case for an outlier relates to an observation
# that has a indicator.8 value of 7...

data1@data <- cbind(data1@data, indicator.8)
data1@data <- as.data.frame(data1@data)
attach(data1@data)
data.1 <- data1@data
#fix(data.1)

# A choropleth map of suspected outliers...
shades.7 = shading(c(1,3,5,7),c("white","yellow","orange","red","dark red"))
X11(width=8,height=7)
par(mar=c(0,0,2,0))
choropleth(data1,data1@data[,24],shades.7)
title("Suspected outliers - weak to strong (yellow to dark red) evidence")
choro.legend(-2400000,2200000,shades.7,
over="exactly", between="to under",
fmt="%4.0f",title='Indicator sum (max.: 7)',cex=0.8)
map.scale(1800000,650000,1000000,"x1000 km",4,0.5)
north.arrow(1800000,1400000,80000, col="blue")
#text(1500000,2400000, "NUTS level 2", cex=2, col=3)
text(1500000,2400000, "NUTS level 1", cex=2, col=3)
#text(1500000,2400000, "NUTS level 0", cex=2, col=3)







# 7. The effects of MAUP ########################################################

# This relationship would be expected to weaken from NUTS2 to NUTS0
X11(width=5.3,height=5.7)
plot(jitter(NUTS3_outlier_max,factor=0.4), jitter(indicator.8,factor=0.4),
main="MAUP and its impact on outlier detection",
xlab="Strongest indication of an outlier for any constituent NUTS3 region",
#ylab="Indication of an outlier in a corres. aggreg. NUTS2 region", ylim=c(0,7),
ylab="Indication of an outlier in a corres. aggreg. NUTS1 region", ylim=c(0,7),
#ylab="Indication of an outlier in a corres. aggreg. NUTS0 region", ylim=c(0,7),
pch=19, cex=0.5)
abline(0,1)
cor(NUTS3_outlier_max, indicator.8)
```