# lczexplore: an R package to explore Local Climate Zone classifications

Matthieu Gousseff [1*], **Erwan Bocher** [1*], **Jérémy Bernard** [1,2,4¶], **and Elisabeth Le Saux Wiederhold** [3¶]

1 CNRS, Lab-STICC, UMR 6285, Vannes, France **2** University of Savoie Mont-Blanc, LOCIE, UMR 5271, France **3** Université Bretagne Sud, Lab-STICC, UMR 6285, Vannes, France **4** University of Gothenburg, Department of Earth Sciences, Sweden ¶ Corresponding author * These authors contributed equally.

Summary

Climate change is a growing concern for city planners as Urban Heat Islands have an impact on mortality Clarke (1972), health in general (Lowe, 2016) and consumption of energy for building cooling (Malys et al., 2012) among other effects. A first step towards large scale study of urban climate is to define classes based on logical division of the landscape, such as Local Climate Zones (LCZ) defined by (Stewart & Oke, 2012).

The lczexplore package aims at comparing different LCZ classifications, but can be used to compare any pair of classifications on geographical units.

A spatial comparison is performed by producing agreement maps between classifications, agreement statistics and a confusion matrix to help qualify and quantify the misclassifications.

This software is available as a free and opensource R package.

Statement of need

Comparing maps

As stated in (Visser & De Nijs, 2006) comparing maps is an important issue in environmental research. The four main reasons to compare categorical variables on geographical units are: - to assess the differences between maps generated by models under different scenarios and assumptions, - to detect temporal changes, - to calibrate or validate models, - to perform uncertainty and sensitivity analysis.

Comparing specifically LCZ maps

Climate change is a growing concern for city planners with a special focus on Urban Heat Island phenomenons. The terms *rural* and *urban* lack of a clear definition and different classifications of urban and rural landscapes were proposed. Stewart & Oke (2012) defined an approach that complies with criteria of geographical classification defined in (Grigg, 1965). Their Local Climate Zones (LCZ) are based on a logical segmentation of land-use and urban canopy parameters and define 10 urban types and 7 land cover types.

LCZ have gained popularity in the past decade as they sum up relevant information and can help, for instance, apprehend the intensity of the Urban Heat Island (Kotharkar & Bagade, 2018).

Several methods aim to classify a territory into LCZ, but only few workflows allow an automatic classification for any given area.

Quan & Bansal ([2021](#)) distinguishes two main streams of production of these LCZ: - the raster stream processes remotely sensed information, and applies machine learning algorithms trained using local experts' knowledge. In this way, the WUDAPT community (Ching et al., 2018) produced thousands of city-based LCZ maps (accessible via the LCZ Generator ((Demuzere et al., 2021))) but also large-scale maps for Europe, the continental United States and the whole world ((Demuzere et al., 2019), (Demuzere et al., 2020), (Demuzere et al., 2022)). - the vector stream uses Geographic Information System (GIS) layers that represent the main topographic features, defines spatial units, computes urban canopy parameters and uses them to classify spatial units into LCZ. For instance, the GeoClimate geospatial toolbox produces LCZ classifications from OpenStreetMap or french BDTopo data (Bocher et al., 2021).

The existence of several methods to produce LCZ classifications, or the use of a method with different input data, raises the need for a tool to quickly get:

1. Statistics measuring the general agreement between two classifications
2. A spatial representation to allow a fast visual assessment of the zones where the classifications agree or disagree.
3. A confusion matrix, to explore how the levels of one classification break up in levels of the other classification, shown in a graphical way to help visualize the main differences between the classifications.

State of the field and feature comparison

State of the field on spatial classification comparison

The question of going beyond the simple visual observation of differences between maps is not a new one. A first approach would be, on raster maps, to simply compute the agreement between two maps as the proportion of pixels for which the two maps have the same value of the variable of interest. This approach is often sufficient to help specialists compare pairs of maps, but it has two main drawbacks:

- two totally random maps won't have a value of agreement of zero, as some pixel values may agree by chance,
- only raster maps where pixels match perfectly (same size, not translated) can be treated, or some pre-treatment are needed (like nearest neighbour interpolation for instance).

To prevent the first drawback, Monserud & Leemans ([1992](#)) proposed the use of Cohen's kappa coefficient of agreement for nominal scales (Cohen, 1960). Cohen's Kappa allow to assess how two raters agree or differ in the task of rating individual. In our case, each pixel is seen as an individual, each map is seen as a rater and the value of the raster at each pixel is seen as the rate.

The comprehensive Map Comparison kit, which was released in 2001 by the Netherlands Environmental Assessment Agency (Visser & De Nijs, 2006), is an example of a tool that provides multiple methods to compare raster maps. It includes a fuzzy algorithm which allows to tackle small shifts of one map from another. It only works on raster maps, only on Windows OS and doesn't allow automation for several pairs of maps.

To tackle the second drawback, one can rasterize vector files, for instance by applying a regular grid to a vector layer and assign to each pixel the value of the vector geometry most present in the pixel (in terms of area of the intersection). This can produce severe effect of smoothing one may want to prevent.

Other approaches also rely on comparing the distribution of some variables of interest. For instance (Hammerberg et al., 2018), rather than comparing the maps, compare some model output variables using the data of the map.

State of the field on Local Climate Zone maps comparison.

As Local Climate Zones emerged as a new standard for characterizing urban landscapes (Demuzere et al., 2021), specific tools are needed to allow easy automation of LCZ maps. As a matter of fact, "ground truth" is hard to define in matters of LCZs, and one may want to compare LCZs produced by experts and by different algorithms.

The very definition of LCZs by Stewart & Oke (2012) relies on ranges of values for several indicators, and sometimes several LCZ types could be assigned to the corresponding geometries. Therefore, one may associate a confidence value to the LCZ type and check if filtering pixels (for raster data) or geometries (for vector data) accordingly to a confidence threshold has an impact on the way maps agree or disagree.

A comparison of a raster stream result and a GIS vector stream approach was proposed by Muhammad et al. (2022). This comparison relies on rasterising vector data. It uses several tools: QGIS, python scripts and SAGA GIS. As far as we know, the scripts and the automation of the method are not publicly available.

The need for automation of map comparison (both raster and vector), and specific features (like standard colors or legends for LCZs and sensitivity analaysis) justified the development of `lczexplore`.

Features

General workflow and processing steps

The `lczexplore` package is a set of R functions to import, visualize, group and compare LCZ classifications, even when they don't use the same spatial units to classify the area. These LCZ classifications can come from vector layers or raster layers (the latter will be vectorized by the dedicated import function)
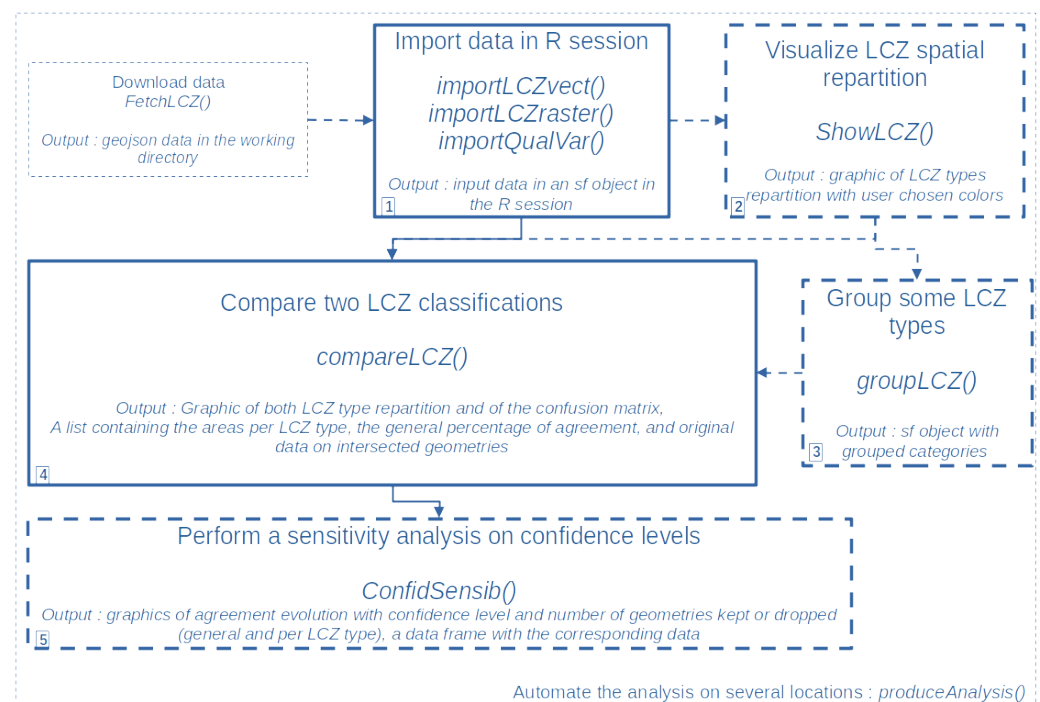

Figure 1: Workflow and main functions of the package

Figure 1 describes the general workflow of the package.

Main functions are presented in plain lines, the dashed boxes and arrows represent optional steps. These functions are presented in detail in the next section, and they allow the following steps of exploration:

1. The LCZ classifications (or any other qualitative variables) are imported from a file (geojson or shapefile format)
2. Each LCZ classification can then be visualized
3. Some LCZ levels may be grouped in broader categories
4. A pair of LCZ classifications (or qualitative variable maps) can then be compared:
   - a map of agreement/disagreement is produced,
   - the general agreement and a pseudo-kappa indicator of agreement are computed,
   - the summed surface of each LCZ type is computed for each classification,
   - a confusion matrix shows how the levels of one LCZ classification break up into the levels of the other
5. Influence of the level of confidence on the agreement between classifications is performed (sensitivity analysis)

All the steps of the analysis can be easily automated in R, for instance on several cities at a time.

Description of the main functions

The main functions are presented in this section. More details about their arguments and output can be found in the minimal example section.

From import to comparison

The `importLCZvect` function allows to import a LCZ classification from a geojson or a shapefile. Geometries and LCZ types are needed, but one can also optionally load unique identifiers and a confidence level for the LCZ type of each geometry.

The `importLCZraster` function allows the import from a raster map, and the user has to feed it a geotiff and a bounding box of the area of interest.

The import functions output objects of class `simple feature`, as defined by the openGIS consortium and as handled by the `sf` package, an R reference library for Geographical Information System Vector Data (Pebesma, 2018).

The `compareLCZ` function creates intersected geometries on which classifications either totally agree or totally disagree, as seen in the figure 2, and this prevents artificial rounding effects that may arise with a rasterisation approach.
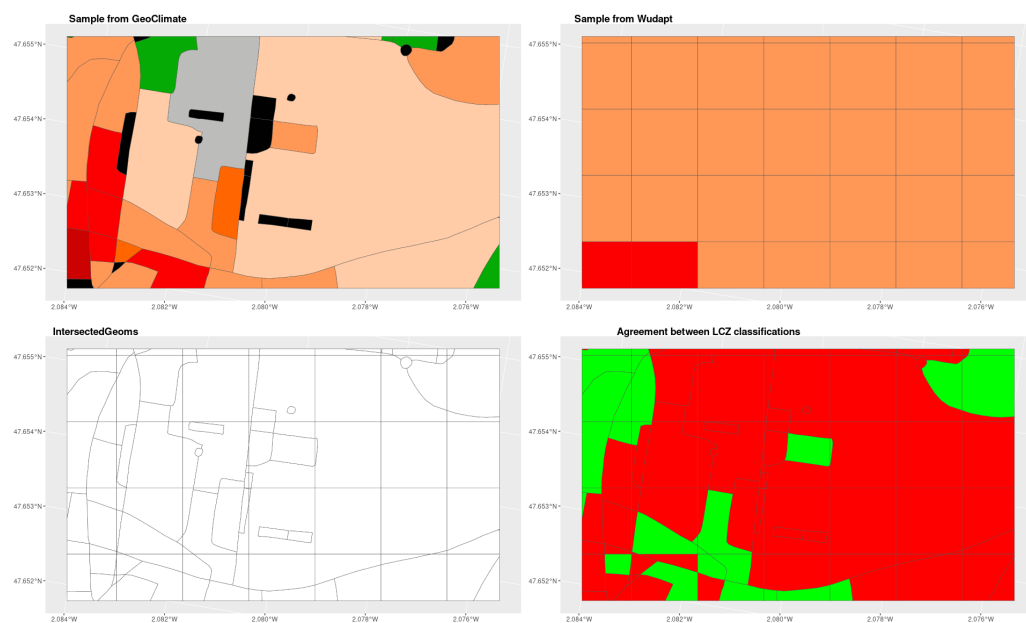
Figure 2: Intersecting geometries to have full agreement or disagreement

Next, the function computes:

- the area of each LCZ type for each classification,
- the percentage of area on which the two classifications agree, and a map of this agreement,
- a pseudo kappa statistics which evaluates agreement beyond how two random maps may agree,
- a confusion matrix of how the types of one LCZ classification break up in the types of the other, and a graphic of this, with an indicator of the summed area for each LCZ type (useful if the two classifications disagree on a type almost absent of the area).

The output of these functions are shown in the minimal example section.

With the standard representation, comparing LCZ maps is made easy by a default setting of legends and colors. The alter representation allows the user to deal with regrouped LCZ categories or any type of qualitative variables.
Levels can either be specified by the user or deduced from the data, colors can either be defined by the user or chosen from a random palette.

Class levels grouping

It is sometimes useful to group some LCZ types, for instance to create a broader category for all the urban LCZ types and another for landcape LCZ types, or to group the levels with similar estimated impact on an urban heat island intensity.

The groupLCZ function allows the user to specify the LCZ types one wants to group together, the names of the new resulting categories and their corresponding colors. One can then feed compareLCZ function these new groups, setting repr="alter", and specify desired levels and colors.
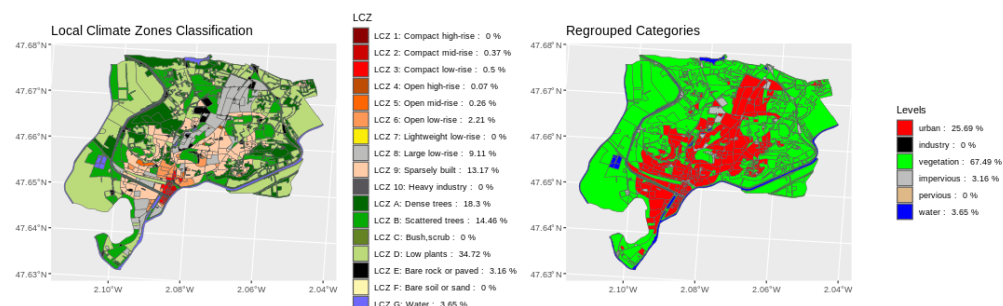
Figure 3: Grouping of LCZ types into larger categories

Sensitivity analysis

The Geoclimate algorithm adds a uniqueness value to the LCZ type it assigns to a spatial unit. It measures if another LCZ levels could have been assigned to this unit. Thus, it can be seen as a confidence value of the LCZ type.

The `lczexplore` package allows a sensitivity analysis according to this level of confidence, in order to answer the question: **does keeping only geometries with a higher confidence value make the degree of agreement between two classifications higher?**

This sensitivity analysis is performed considering all LCZ types and within each individual LCZ type.
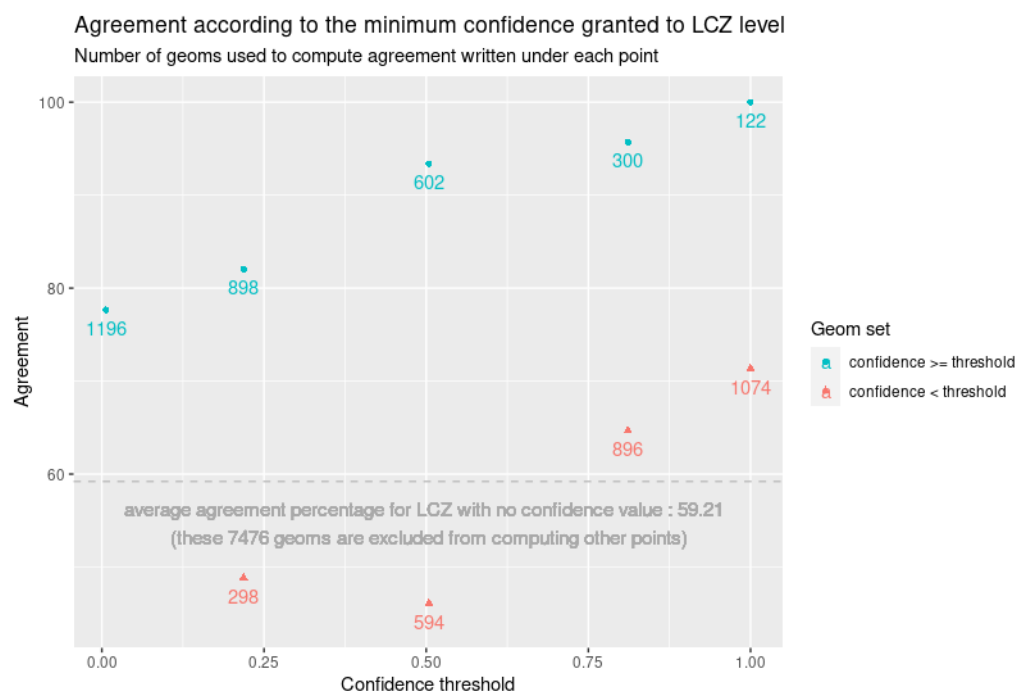
Figure 4: Sensitivity analysis according to confidence for all LCZ levels

The agreement between classifications for the geometries with a confidence level higher than the threshold, and their numbers, are plotted in blue. The agreement and the numbers of geometries under the threshold are plotted in magenta.

On this example, one can see that ditching geometries that have a confidence level lower than 0.5 increases the agreement to more than 90%. The curve then tends to flatten, and the number of kept geometries decreases a lot (from 602 to 122). One also needs to notice that on this example, most geometries didn't have a confidence value (7476 with a general agreement of 59.21%)

Import and explore categorical variable (other than LCZ classifications)

This package focuses on LCZ maps comparison, but more often than not, people working on LCZ maps also describe their area of interest with other categorical indicators. The workflow of comparison of LCZ maps can be used for any pair of maps of categorical variables, under certain limitations: - there must not be more than 36 levels for the categorical variable to explore - the associated geometries must be (multi) polygons or easily converted to them (typically, the package would not be suitable to compare road characterization), - the geometries must be topographically valid (this is also true for LCZ).

The `importQualVar` function allows the import of such variables on (multi-) polygons maps. It outputs an sf object that can be fed to the other main functions of the package (`showLCZ`, `compareLCZ`, `groupLCZ`…).

Coding implementation

`lczexplore` is an R package, all its specific functions are coded in R language. It relies on state-of-the art packages: - geographical computation requires the **sf** package for vector data and the **terra** package for raster data, - data management mainly requires the following packages: **dplyr, tidyr, forcats, rlang** and **methods** packages, - graphical production uses **ggplot2, grDevices, cowplot** and **RColorBrewer**, - tests need the **tinytest** package.

Every step corresponds to an R function (see the workflow on figure 1 for the name of the main functions).

Every function has an associated file for unitary testing.

A minimal example

Install and load

The lczexplore package can be downloaded from https://github.com/orbisgis/lczexplore and installed with the command install_github:

```r
library(devtools)
devtools::install_github("orbisgis/lczexplore", build_vignettes = TRUE)
library(lczexplore)
```

In order to discover the package, vignettes can be browsed via browseVignettes("lczexplore") or by calling them directly:

```r
vignette("lczexplore_en")
vignette("lczexplore_raster_vector")
vignette("lczexplore_alter")
```

Import the data

This example uses two LCZ classifications of French Redon city, produced using a single LCZ classification method (the GeoClimate workflow), with two different input data: OpenStreetMap and the 2.2 version of the French BDTopo.

```r
# Set the path to the example data folder created once you install the package

dirPath<-paste0(system.file("extdata",package="lczexplore"),"/")
dirPathOSM<-paste0(dirPath,"osm/2022/Redon")
dirPathBDT<-paste0(dirPath,"bdtopo_2_2/Redon")

# Import into an sf object the data produced with GeoClimate and
# the OpenStreetMap data (city of Redon)
redonOSM<-importLCZvect(
  dirPath=dirPathOSM,
  file="rsu_lcz.geojson", column = "LCZ_PRIMARY", geomID="ID_RSU",
  confid="LCZ_UNIQUENESS_VALUE")

# Import into an sf object the data produced with GeoClimate and
# the french BDTopo V2.2 data (city of Redon)
redonBDT<-importLCZvect(
  dirPath=dirPathBDT,
  file="rsu_lcz.geojson", column = "LCZ_PRIMARY",geomID="ID_RSU",
  confid="LCZ_UNIQUENESS_VALUE")
```

Visualize the data

To visualize a LCZ classification, use the showLCZ function. Setting repr to "standard" directly deals with standard values of LCZ levels and colors.

```r
# Plot the LCZ levels on the Redon Area using the default color set
# from the sf object produced by importLCZvect function
```

```
map1<-showLCZ(
    sf=redonOSM, column="LCZ_PRIMARY", repr="standard",
    title="LCZ Classification on the French City of Redon")
```

The result is a map of the Local Climate Zones on the area:
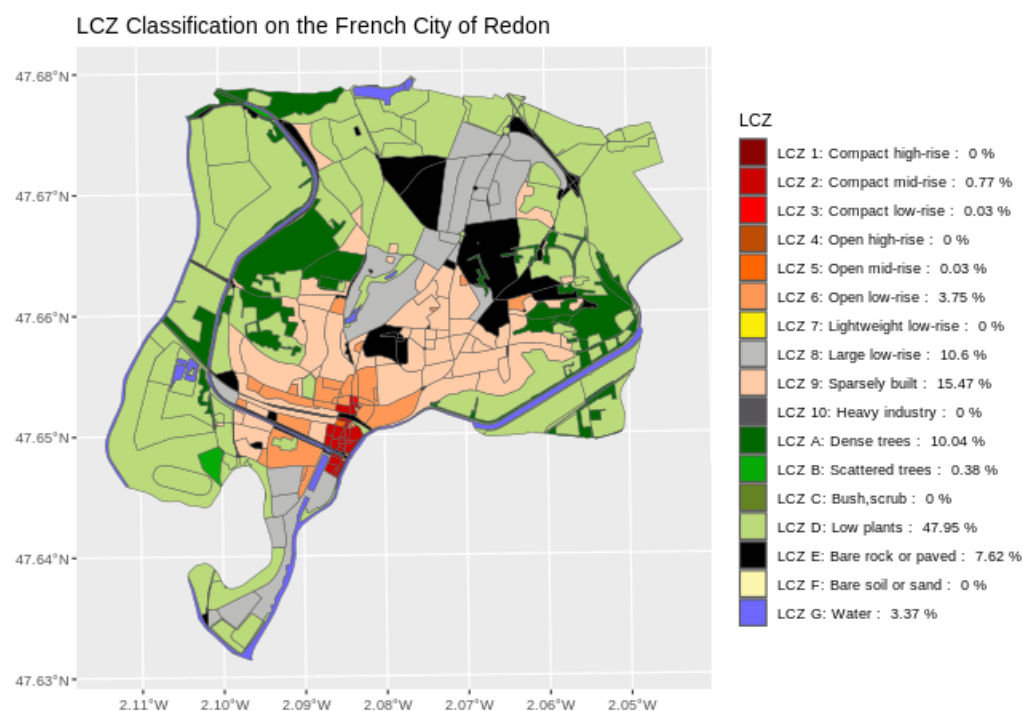
Figure 5: Local Climate Zones for Redon city based on the GeoClimate workflow applied to OSM data

Compare the two LCZ classifications

To compare the two loaded LCZ classifications, use the `compareLCZ` function.

- the `sf1` and `column1` arguments allow to specify the sf dataset of the first map, and the column in which the LCZ levels are stored (same for `sf2` and `column2`),
- the `wf1` and `wf2` arguments take strings to specify the workflows used to produce the input data.
- The `location` argument allows to name the area of interest covered by input datasets. The 3 previous arguments will be parsed in the legends of the graphics (except if `title` is specified) and the name of potential csv output files,
- when `exWrite=TRUE`, the data computed to create the confusion matrix will be written in a csv file in the working directory,
- if `saveG=TRUE`, the four plots created by the function will be written in a png file in the working directory.

```
# Compare how the BDTopo and the OpenStreetMap Data produce different classifications.
# The outputs are stored in a list.

comparison<-compareLCZ(sf1=redonBDT,column1="LCZ_PRIMARY", wf1="BDTopo v2.2",
            sf2=redonOSM, column2="LCZ_PRIMARY", wf2="OpenStreetMap",
            ref=1, repr="standard", exwrite=F, location="Redon")
```

All graphics are concatenated for a quick glance.
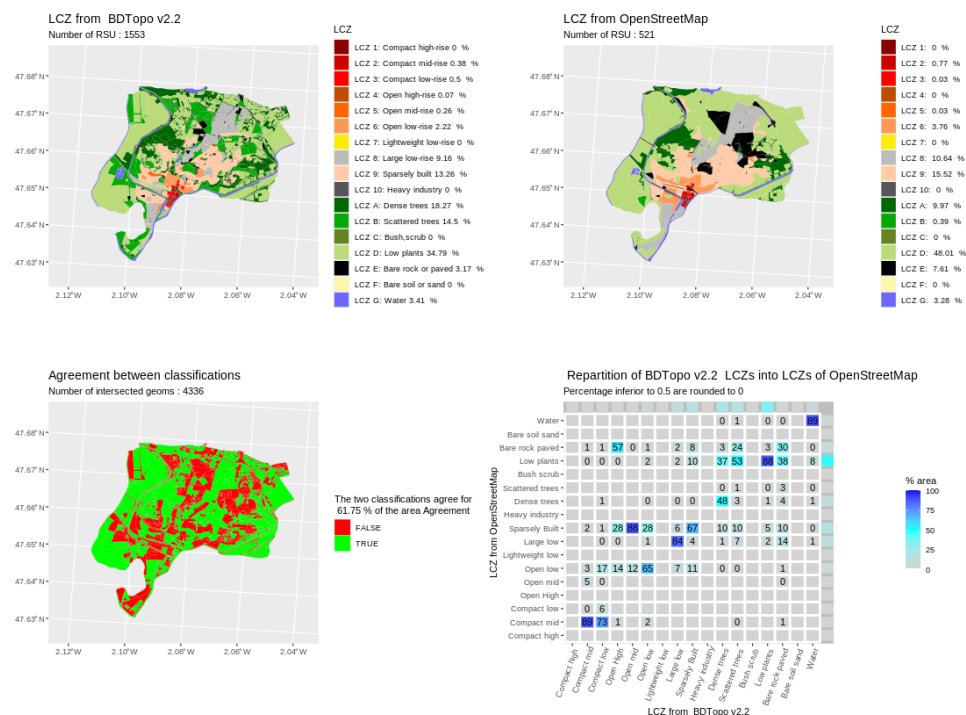
Figure 6: Local Climate Zones comparison for the French city of Redon based on the GeoClimate workflow applied to OpenStreetMap and BDTopo data

The first and second maps show the spatial distribution of the first and second LCZ classifications levels, respectively.

The third map shows where the two classifications agree or disagree.

The last graphic is a confusion matrix: how the LCZ types of the first classification break-up in those of the second, in percentage of the surface.

CompareLCZ outputs a list called `matConfOut` which contains:

- `$data`, intersected geometries, their identifiers and associated confidence value (if fed to the fonction), their LCZ type and their area,
- `$areas`, the summed area for each LCZ for both classifications,
- `$matConfLarge`, the confusion matrix,
- `$percAgg`, the general agreement of the two classification on the whole area (as a percentage of the global surface of the area),
- `$pseudoK`, the value of a heuristic pseudo Cohen's Kappa coefficient of agreement.

The value `pseudoK` corresponds to a Kappa, where the cells of the cross tabulation would be weighted by the percentage of the concerned surface.

One can see in a quick glance, that the BDTopo workflow leads to more and smaller Spatial Reference Units (1553 RSU) than the OpenStreetMap workflow (521 RSU). The third map shows that while the classifications tend to agree on wide vegetation areas, there seem to be more some discrepancies in the urban areas. The last graphics helps to refine this analysis: one can notice that both workflows quite agree for "compact mid" and "low plants" LCZ levels (89% and 88 % of the concerned areas). On the contrary, only 48% of the areas set to "dense trees" by BDTopo workflow are also set to "dense trees" by OpenStreetMap workflow, while 37% of these areas are set as "low plants."

In the same way, 73% of the areas set to "compact low" by BDTopo workflow are set to "compact mid" by OpenStreetMap workflow. This is coherent with the fact that building heights are often missing on OpenStreetMap and that the algorithm that GeoClimate uses to

predict them is less precise than the available information of BDTopo. One may read more in (Bernard et al., 2023) about differences between these workflows.

Study the impact of confidence on agreement

The `confidSensib` function makes a sensitivity analysis of the agreement according to the threshold of confidence above which one keeps the geometries of the compared maps. To perform this analysis, the confidence value and the identifier of each geometry must be fed to the comparison function. The $data slot is then passed to `confidSensib`. The `saveG` allows to set the path to a folder where resulting graphics will be saved.

The values of confidence for which the agreement will be computed are determined by the range of confidence values present in the data and by the number of points, set by the `nPoints` argument.

```
 # Path to the folder where compareLCZ stored output data file
# mainPath<-system.file("extdata", package = "lczexplore") (not executed to avoid useles

# Specification of geomID and confid columns for both dataset,
# as confidSensib needs them in the next step

comparison<-compareLCZ(
  sf1=redonBDT,column1="LCZ_PRIMARY", wf1="BDTopo v2.2",
  geomID1="ID_RSU", confid1="LCZ_UNIQUENESS_VALUE",
  sf2=redonOSM, column2="LCZ_PRIMARY", wf2="OpenStreetMap",
  geomID2="ID_RSU", confid2="LCZ_UNIQUENESS_VALUE",ref=1,
  repr="standard", exwrite=F, location="Redon")

testSourceFact<-comparison$data

# Sensitivityanalysis on this compareLCZ output

sensitAnalysis<-confidSensib(inputDf=comparison$data, filePath="",
  nPoints=5, wf1="bdtopo_2_2", wf2="osm",
  geomID1="ID_RSU", column1="LCZ_PRIMARY", confid1="LCZ_UNIQUENESS_VALUE",
  geomID2="ID_RSU.1",column2="LCZ_PRIMARY.1", confid2="LCZ_UNIQUENESS_VALUE.1",
  plot=TRUE, saveG="") #  one would use saveG = mainPath to write the results in the ma
```
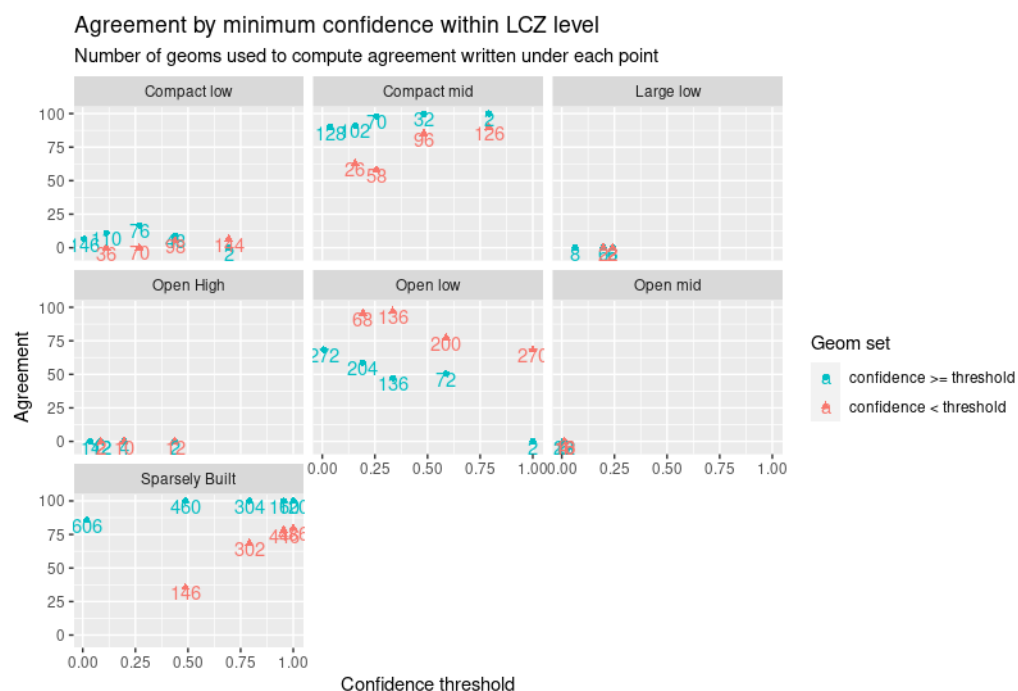
Figure 7 shows the resulting graphics for the agreement per LCZ type (types not present in the dataset are excluded).

Figure 7: Sensitivity analysis according to confidence by LCZ levels

For each LCZ type, the x-axis shows the minimum confidence threshold. The y-axis shows the agreement levels.
Each cyan point shows the average agreement between classifications for spatial units where the minimum confidence value is greater than the x-axis value.

As one wants to be sure the dropped units didn't have a greater agreement, they are showed as magenta triangles. For this particular example, it seems that for the "Open Low" LCZ type, the greater the confidence, the smaller the agreement!

This unexpected result is explained by the fact that in our case, the confidence qualifies the method, not the data. In this example, in the OSM dataset which is compared, the building height are mostly estimated using a statistical method (Bernard et al., 2022) as stated above.

Group some levels and perform the same analysis

The `groupLCZ` function aggregates chosen levels into broader categories. For each category, a vector is needed, which name is the name of the broader category to create and which values are the levels to be grouped in this new category.

```
redonOSMgrouped<-groupLCZ(redonOSM,column="LCZ_PRIMARY",
                          urban=c("1","2","3","4","5","6","7","8","9"),
                          industry="10",
                          vegetation=c("101","102","103","104"),
                          impervious="105",
                          pervious="106",
                          water="107",
                          colors=c("red","black","green","grey","burlywood","blue"))

redonBDTgrouped<-groupLCZ(redonBDT,column="LCZ_PRIMARY",
                          urban=c("1","2","3","4","5","6","7","8","9"),
                          industry="10",
```

```
                            vegetation=c("101","102","103","104"),
                            impervious="105",
                            pervious="106",
                            water="107",
                            colors=c("red","black","green","grey","burlywood","blue"))
```

By default, the column where the new categories are stored is named "grouped", but it can be set with the argument outCol.

It is the possible to perform the same analysis as on standard LCZ classifications, as shown in the following example:

```
map2<-showLCZ(redonOSMgrouped, column="grouped",repr="alter",
        title="Regrouped Categories for the city of Redon",
        LCZlevels = c("urban","industry","vegetation","impervious","pervious","water"),
        colors=c("red","black","green","grey","burlywood","blue"))
compareLCZ(sf1=redonOSMgrouped, column1="grouped", wf1="OpenStreetMap data",
            sf2=redonBDTgrouped, column2="grouped", wf2="BDTopo data",ref=1,
            repr="alter",exwrite=F,location="Redon",saveG="",
            LCZlevels = c("urban","industry","vegetation","impervious","pervious","water"
            colors=c("red","black","green","grey","burlywood","blue"))
```
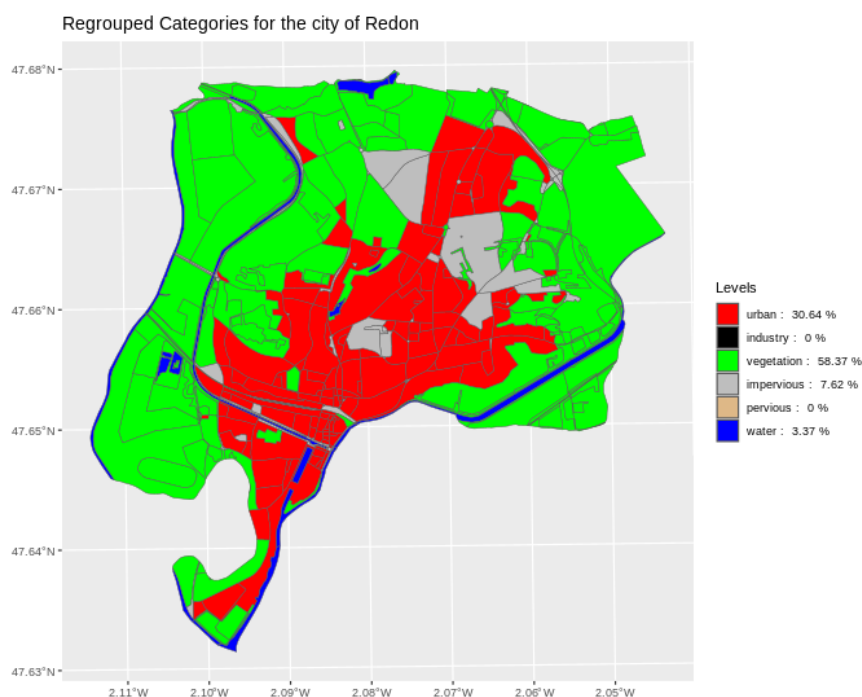

Figure 8: Map of grouped levels on the city of Redon

You can also map any qualitative variable using this grouped representation, as long as you specify the expected levels the same way.
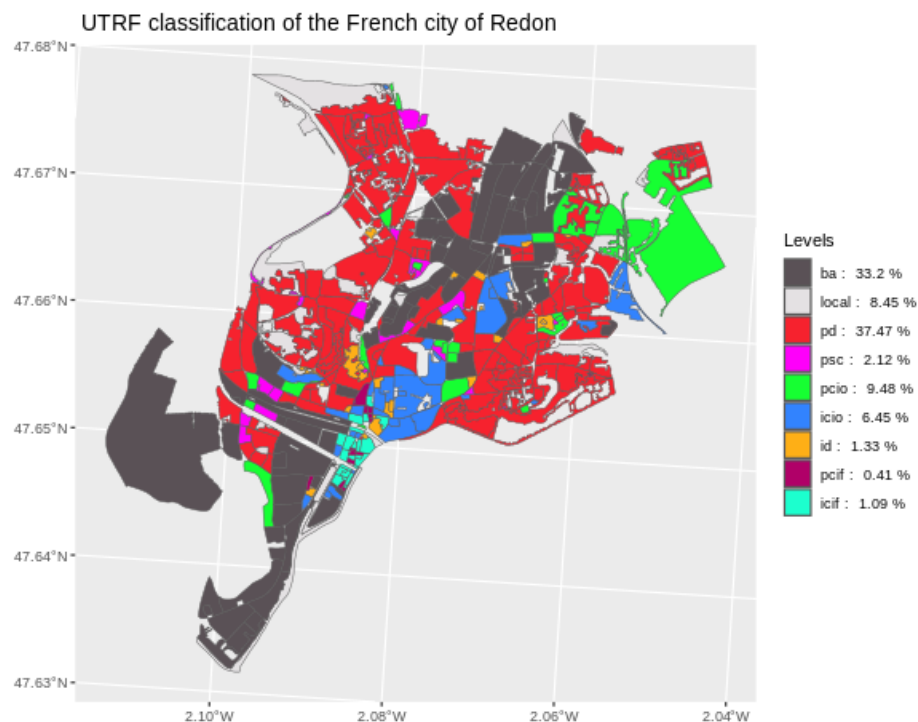
An example of qualitative variables

The function importQualVar allows you to import maps of qualitative variables on (multi)polygons. For instance, classifications of urban tissue called Urban Typolgy by Random Forest (Bocher et al., 2018) are available as examples in this package.

After the import, the usual functions can be used, as shown in the following code:

```
utrfRedonBDT<-importQualVar(dirPath=paste0(
  system.file("extdata", package = "lczexplore"), "/bdtopo_2_2/Redon"),
  file="rsu_utrf_area.geojson", column="TYPO_MAJ")

map3<-showLCZ(sf=utrfRedonBDT, column="TYPO_MAJ",repr="alter",
        title = " UTRF classification of the French city of Redon")
```


Figure 9: An example of some qualitative variable: Urban Typology by Random Forest (UTRF)

Figure 9 represents the UTRF classification of the French city of Redon, as computed by the GeoClimate workflow on BDTopo v2.2 data. The colors were not specified and therefore were randomly picked from a palette.

```
utrfRedonOSM<-
  importQualVar(
    dirPath=paste0(
      system.file("extdata", package = "lczexplore"),
      "/osm/2022/Redon"),
    file="rsu_utrf_area.geojson", column="TYPO_MAJ",
    geomID="ID_RSU", confid="UNIQUENESS_VALUE")

utrfComparison<-
  compareLCZ(sf1=utrfRedonBDT, column1="TYPO_MAJ",wf1=" UTRF BDTopo",
            sf2=utrfRedonOSM, column2="TYPO_MAJ", wf2 = " UTRF OpenStreetMap",
  location = " Redon",exwrite=FALSE,repr="alter")
  # Plot the confusion matrix of thes two classifications
```
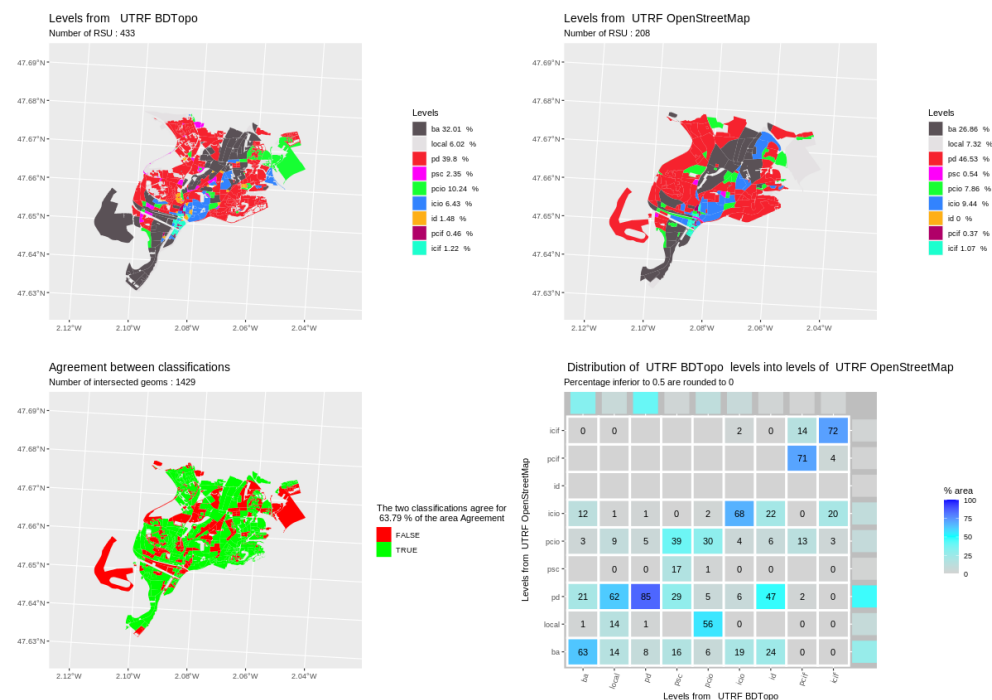
Figure 10: Example of comparison on a qualitative variable (UTRF)

Licensing, availability and contribution

lczexplore is licensed under the GNU Lesser General Public License (v3.0), with all source code stored at GitHub. In the spirit of honest and open science, requests and suggestions for fixes, feature updates, as well as general questions and concerns are encouraged via direct interaction with contributors and developers by filing an issue while respecting Contribution Guidelines.

Research projects involving lczexplore

The lczexplore package was developed thanks to the project PÆNDORA2 (Pour la gestion du confort estival: Donnees, Outils et Recherche-Action) (2022 -2025), funded by ADEME.

References

Bernard, J., Bocher, E., Gousseff, M., Leconte, F., & Le Saux Wiederhold, E. (2023). *A generic algorithm to automatically classify urban fabric according to the Local Climate Zone system: Implementation in GeoClimate 0.0.1 and application to French cities* [Preprint]. Climate and Earth system modeling. https://doi.org/10.5194/egusphere-2023-371

Bernard, J., Bocher, E., Le Saux Wiederhold, E., Leconte, F., & Masson, V. (2022). *Estimation of missing building height in OpenStreetMap data: A French case study using GeoClimate 0.0.1* [Preprint]. Climate and Earth system modeling. https://doi.org/10.5194/gmd-2021-428

Bocher, E., Bernard, J., Le Saux Wiederhold, E., Leconte, F., Petit, G., Palominos, S., & Noûs, C. (2021). GeoClimate: A geospatial processing toolbox for environmental and climate studies. *Journal of Open Source Software*, *6*(65), 3541. https://doi.org/10.21105/joss.03541

Bocher, E., Petit, G., Bernard, J., & Palominos, S. (2018). A geoprocessing framework to compute urban indicators: The MApUCE tools chain. *Urban Climate*, *24*, 153–174. https://doi.org/10.1016/j.uclim.2018.01.008

Ching, J., Mills, G., Bechtel, B., See, L., Feddema, J., Wang, X., Ren, C., Brousse, O., Martilli, A., Neophytou, M., Mouzourides, P., Stewart, I., Hanna, A., Ng, E., Foley, M., Alexander, P., Aliaga, D., Niyogi, D., Shreevastava, A., … Theeuwes, N. (2018). WUDAPT: An Urban Weather, Climate, and Environmental Modeling Infrastructure for the Anthropocene. *Bulletin of the American Meteorological Society*, *99*(9), 1907–1924. https://doi.org/10.1175/BAMS-D-16-0236.1

Clarke, J. F. (1972). Some effects of the urban structure on heat mortality. *Environmental Research*, *5*(1), 93–104. https://doi.org/10.1016/0013-9351(72)90023-0

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, *20*(1), 37–46. https://doi.org/10.1177/001316446002000104

Demuzere, M., Bechtel, B., Middel, A., & Mills, G. (2019). Mapping europe into local climate zones. *PloS One*, *14*(4), e0214474.

Demuzere, M., Hankey, S., Mills, G., Zhang, W., Lu, T., & Bechtel, B. (2020). Combining expert and crowd-sourced training data to map urban form and functions for the continental US. *Scientific Data*, *7*(1), 264. https://doi.org/10.1038/s41597-020-00605-z

Demuzere, M., Kittner, J., & Bechtel, B. (2021). LCZ Generator: A Web Application to Create Local Climate Zone Maps. *Frontiers in Environmental Science*, *9*, 637455. https://doi.org/10.3389/fenvs.2021.637455

Demuzere, M., Kittner, J., Martilli, A., Mills, G., Moede, C., Stewart, I. D., Vliet, J. van, & Bechtel, B. (2022). A global map of local climate zones to support earth system modelling and urban scale environmental science. *Earth System Science Data Discussions*, *2022*, 1–57. https://doi.org/10.5194/ems2022-83

Grigg, D. (1965). The logic of regional systems. *Annals of the Association of American Geographers*, *55*(3), 465–491. https://doi.org/10.1111/j.1467-8306.1965.tb00529.x

Hammerberg, K., Brousse, O., Martilli, A., & Mahdavi, A. (2018). Implications of employing detailed urban canopy parameters for mesoscale climate modelling: A comparison between WUDAPT and GIS databases over Vienna, Austria. *International Journal of Climatology*, *38*, e1241–e1257. https://doi.org/10.1002/joc.5447

Kotharkar, R., & Bagade, A. (2018). Evaluating urban heat island in the critical local climate zones of an indian city. *Landscape and Urban Planning*, *169*, 92–104. https://doi.org/10.1016/j.landurbplan.2017.08.009

Lowe, S. A. (2016). An energy and mortality impact assessment of the urban heat island in the US. *Environmental Impact Assessment Review*, *56*, 139–144. https://doi.org/10.1016/j.eiar.2015.10.004

Malys, L., Musy, M., & Inard, C. (2012). Microclimate and buildings energy consumption: Sensitivity analysis of coupling methods. *Eighth International Conference on Urban Climate and 10th Symposium on the Urban Environment*. https://doi.org/10.1080/17512549.2015.1043643

Monserud, R. A., & Leemans, R. (1992). Comparing global vegetation maps with the Kappa statistic. *Ecological Modelling*, *62*(4), 275–293. https://doi.org/10.1016/0304-3800(92)90003-W

Muhammad, F., Xie, C., Vogel, J., & Afshari, A. (2022). Inference of local climate zones from GIS data, and comparison to WUDAPT classification and custom-fit clusters. *Land*, *11*(5), 747. https://doi.org/10.3390/land11050747

Pebesma, E. (2018). Simple Features for R: Standardized Support for Spatial Vector Data. *The R Journal*, *10*(1), 439–446. https://doi.org/10.32614/RJ-2018-009

Quan, S. J., & Bansal, P. (2021). A systematic review of GIS-based local climate zone mapping studies. *Building and Environment*, *196*, 107791. https://doi.org/10.1016/j.buildenv.2021.107791

Stewart, I. D., & Oke, T. R. (2012). Local climate zones for urban temperature studies. *Bulletin of the American Meteorological Society*, *93*(12), 1879–1900. https://doi.org/10.1175/bams-d-11-00019.1

Visser, H., & De Nijs, T. (2006). The map comparison kit. *Environmental Modelling & Software*, *21*(3), 346–358.