2b. Intrinsic Analysis of Reference Data

This notebook analyses the quality of a user-provided reference bicycle infrastructure data set for a given area. The quality assessment is *intrinsic*, i.e. based only on the one input data set, and making no use of information external to the data set. For an extrinsic quality assessment that compares the reference data set to corresponding OSM data, see the notebooks 3a and 3b.

The analysis assesses the *fitness for purpose* (Barron et al., 2014) of the reference data for a given area. Outcomes of the analysis can be relevant for bicycle planning and research - especially for projects that include a network analysis of bicycle infrastructure, in which case the topology of the geometries is of particular importance.

Since the assessment does not make use of an external reference data set as the ground truth, no universal claims of data quality can be made. The idea is rather to enable those working with bicycle networks to assess whether their data are good enough for their particular use case. The analysis assists in finding potential data quality issues but leaves the final interpretation of the results to the user.

The notebook makes use of quality metrics from a range of previous projects investigating OSM/VGI data quality, such as Ferster et al. (2020), Hochmair et al. (2015), Barron et al. (2014), and Neis et al. (2012).

Familiarity required
For a correct interpretation of some of the metrics for spatial data quality, some familiarity with the area is necessary.

Data completeness

Network density

In this setting, network density refers to the length of edges or number of nodes per km2. This is the usual definition of network density in spatial (road) networks, which is distinct from the *structural* network density known more generally in network science. Without comparing to a reference data set, network density does not in itself indicate spatial data quality. For anyone familiar with the study area, network density can however indicate whether parts of the area appear to be under- or over-mapped.

Method

The density here is not based on the geometric length of edges, but instead on the computed length of the infrastructure. For example, a 100-meter-long bidirectional path contributes with 200 meters of bicycle infrastructure. With `compute_network_density`, the number of elements (nodes, dangling nodes, and total infrastructure length) per unit area is calculated. The density is computed twice: first for the study area for both the entire network ('global density'), then for each of the grid cells ('local density'). Both global and local densities are computed for the entire network and for protected and unprotected infrastructure.

Interpretation

Since the analysis conducted here is intrinsic, i.e. it makes no use of external information, it cannot be known whether a low-density value is due to incomplete mapping, or due to actual lack of infrastructure in the area. However, a comparison of the grid cell density values can provide some insights, for example:

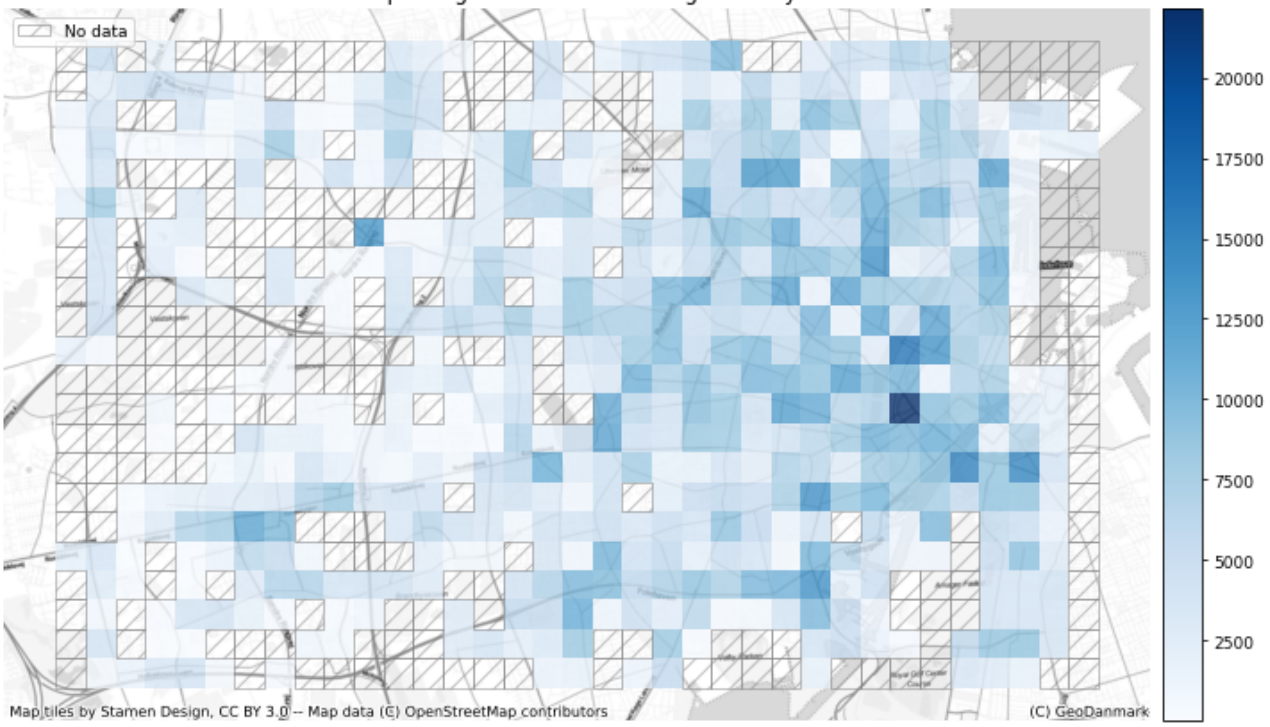
- lower-than-average infrastructure density indicates a locally sparser network
- higher-than-average node density indicates that there are relatively many intersections in a grid cell
- higher-than-average dangling node density indicates that there are relatively many dead ends in a grid cell

Global network density

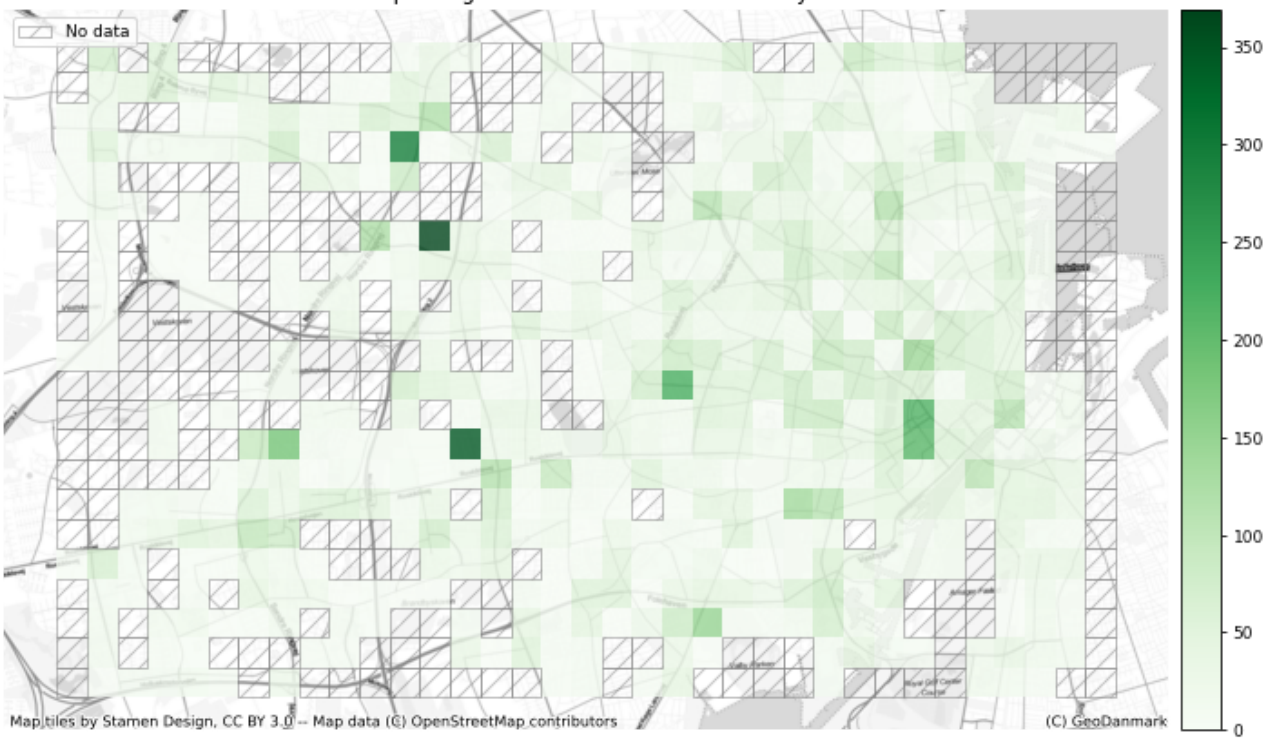
```
For the entire study area, there are:
- 3453.85 meters of bicycle infrastructure per km2.
- 22.74 nodes in the bicycle network per km2.
- 4.80 dangling nodes in the bicycle network per km2.
- 2998.80 meters of protected bicycle infrastructure per km2.
- 455.05 meters of unprotected bicycle infrastructure per km2.
- 0.00 meters of mixed protection bicycle infrastructure per km2.
```

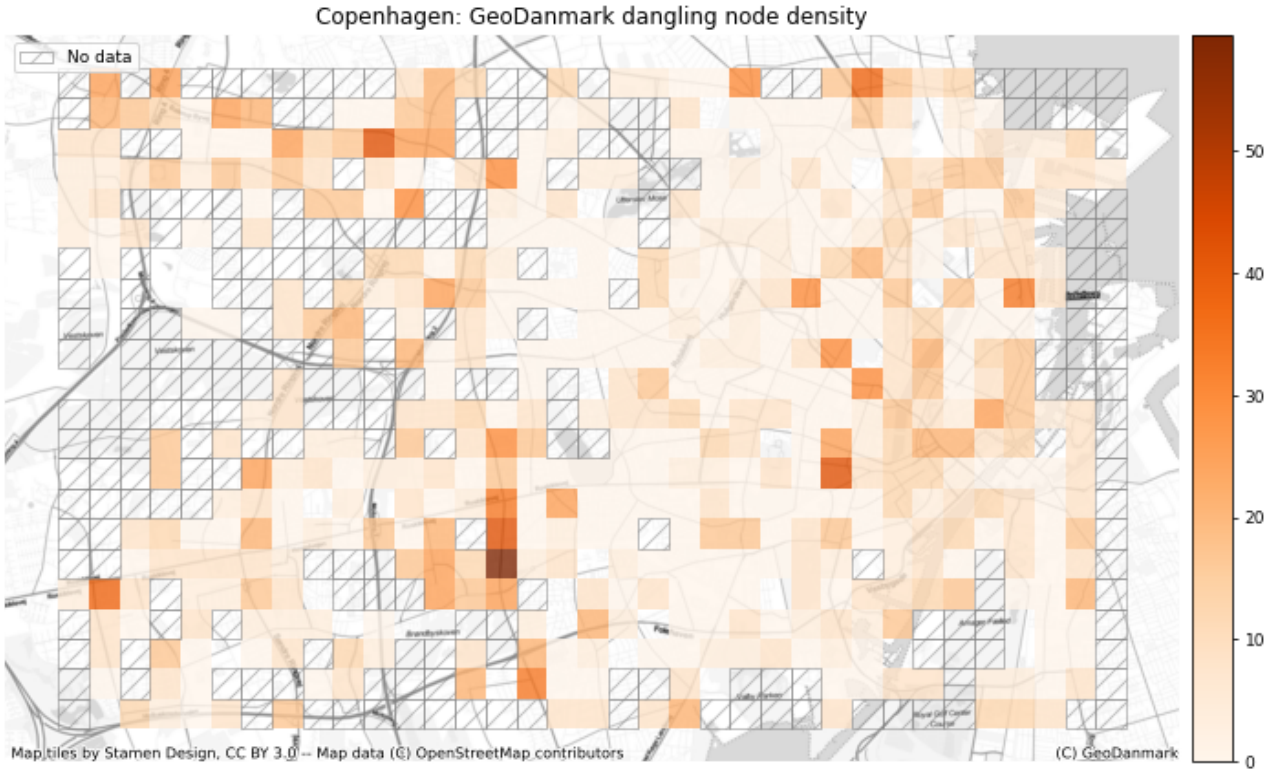
Local network density

Copenhagen: GeoDanmark edge density

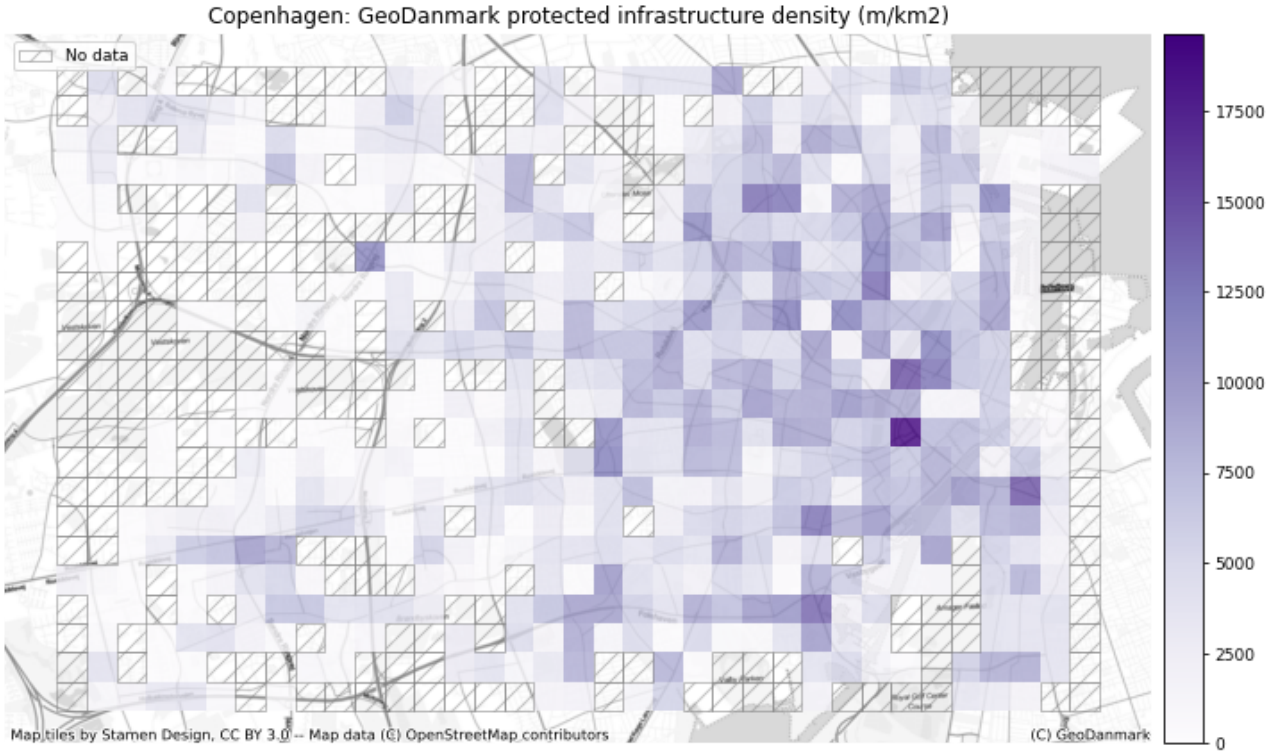


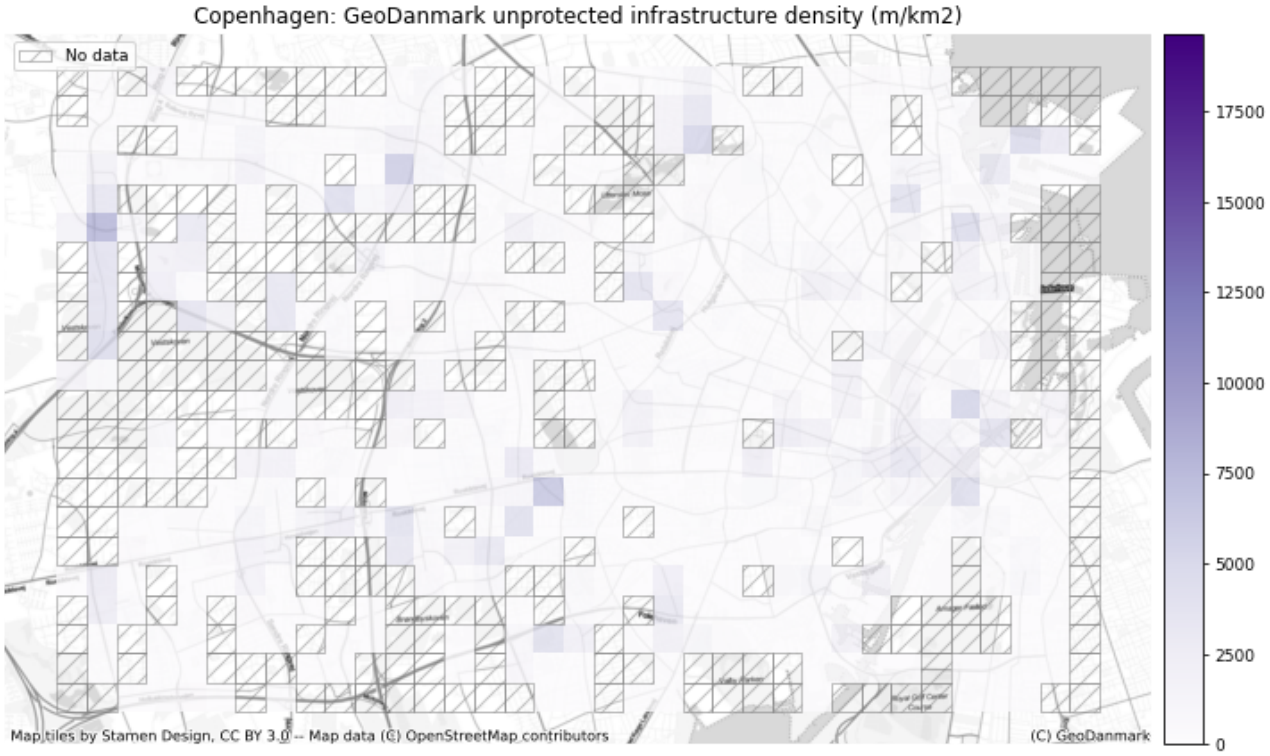
Copenhagen: GeoDanmark node density





Densities of protected and unprotected infrastructure:





Network topology

This section explores the geometric and topological features of the data.

These are, for example, network density, disconnected components, dangling (degree one) nodes; it also includes exploring whether there are nodes in close proximity, that do not share an edge - a potential sign of edge undershoots - or if there are intersecting edges without a node at the intersection, which might indicate a digitizing error that will distort any attempts at routing on the network.

Due to the fragmented nature of most networks of bicycle infrastructure, many metrics, such as missing links or network gaps, simply reflect the true extent of the infrastructure (Natera Orozco et al., 2020). This is different for car networks, where e.g., disconnected components could more readily be interpreted as a data quality issue.

Therefore, the analysis only takes very small network gaps into account as potential data quality issues.

Simplification outcome

When converting a set of geocoded linestrings (polygonal chains) to graph format, not all vertices (nodes) are of equal meaning. For geometry of the infrastructural element, all nodes are needed as an ordered list. For the topology of the network, however, only those nodes that are endpoints or intersection points with other edges are needed, while all other (so-called 'interstitial') nodes do not add any information. To compare the structure and true ratio between nodes and edges in a network, a simplified network representation which only includes nodes at endpoints and intersections, or where the value of important attributes changes, is required. Therefore, in the notebook `01_load_data` the bicycle network was simplified by removing all interstitial nodes from the graph object (retaining, however, the complete node lists in the geometry attribute of each edge). An additional advantage of

simplifying the network is the resulting substantial reduction of the number of nodes and edges, which makes computational routines much faster.

Comparing the node degree distribution for the networks before and after simplification is a quick sanity check for the simplification routine. Typically, the big majority of nodes in the non-simplified network will be of degree two; in the simplified network, however, most nodes will have degrees other than two. Degree two nodes are retained in only two cases: if they represent a connection point between two different types of infrastructure; or if they are needed in order to avoid self-loops (edges whose start and end points are identical) or multiple edges between the same pair of nodes.

As part of the simplification routine, in cases where there are several edges between the same pair of nodes ('parallel edges' or 'multiedges'), only one of the edges is retained. Within the routine, the number edges removed in this way are counted.

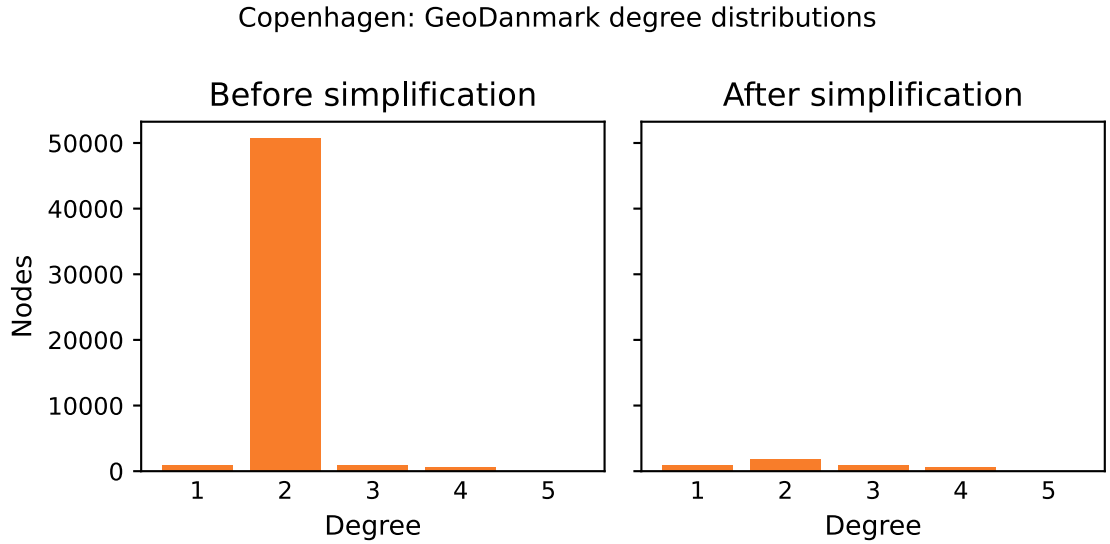
Method

The node degree distributions before and after simplification are plotted below.

Interpretation

Typically, the node degree distribution will go from high (before simplification) to low (after simplification) counts of degree two nodes, while it will not change for all other degrees (1, or 3 and higher). Further, the total number of nodes will see a strong decline. If the simplified graph still maintains a relatively high number of degree two nodes, or if the number of nodes with other degrees changes after the simplification, this might point to issues either with the graph conversion or with the simplification process.

```
Simplifying the network decreased the number of edges with 91.2% and the number of nodes with 9
2.2%.
```



Dangling nodes

Dangling nodes are nodes of degree one, i.e. they have only one single edge attached to them. Most networks will naturally contain a number of dangling nodes. Dangling nodes can occur at actual dead-

ends (representing a cul-de-sac) or at the endpoints of certain features, e.g. when a bicycle path ends in the middle of a street. However, dangling nodes can also occur as a data quality issue in case of over/undershoots (see next section). The number of dangling nodes in a network does to some extent also depend on the digitization method, as shown in the illustration below.

Therefore, the presence of dangling nodes is in itself not a sign of low data quality. However, a high number of dangling nodes in an area that is not known for containing many dead-ends can indicate digitization errors and problems with edge over/undershoots.

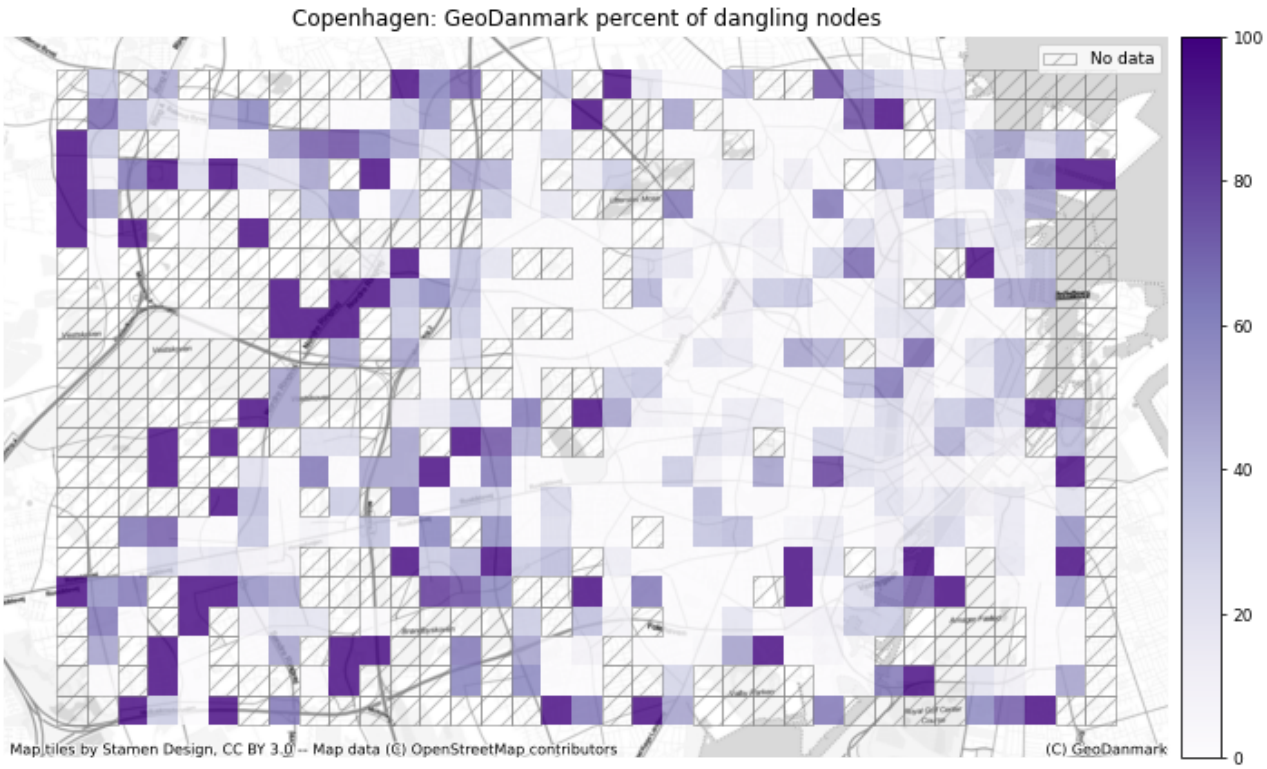
Left: Dangling nodes occur where road features end. Right: However, when separate features are joined at the end, there will be no dangling nodes.

Method

Below, a list of all dangling nodes is obtained with the help of `get_dangling_nodes`. Then, the network with all its nodes is plotted. The dangling nodes are shown in color, all other nodes are shown in black.

Interpretation

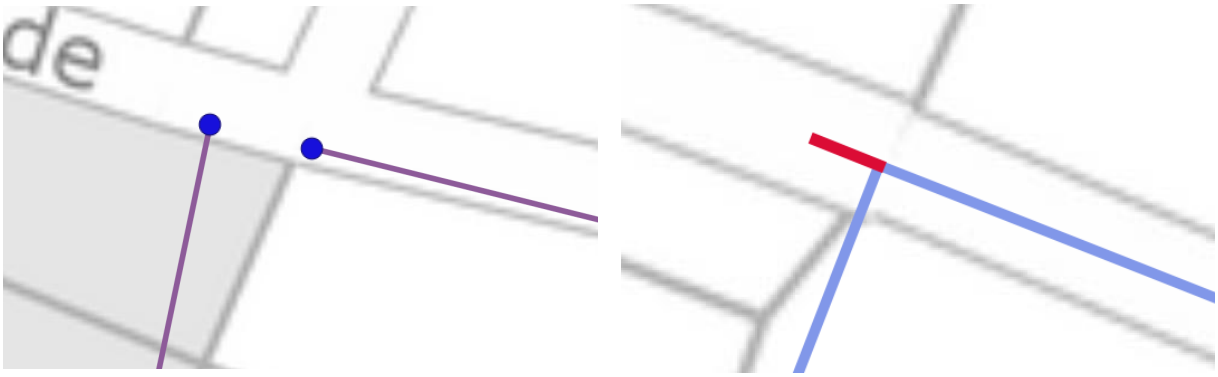
We recommend a visual analysis in order to interpret the spatial distribution of dangling nodes, with particular attention to areas of high dangling node density. It is important to understand where dangling nodes come from: are they actual dead-ends or digitization errors (e.g., over/undershoots)? A higher number of digitization errors points to lower data quality.




```
Interactive map saved at results/REFERENCE/cph_geodk/maps_interactive/folium_danglingmap_referen
ce.html
```

Under/overshoots

When two nodes in a simplified network are placed within a distance of a few meters, but do not share a common edge, it is often due to an edge over/undershoot or another digitizing error. An undershoot occurs when two features are supposed to meet, but instead are just in close proximity to each other. An overshoot occurs when two features meet and one of them extends beyond the other. See the image below for an illustration of an undershoot (left) and an overshoot (right). For a more detailed explanation of over/undershoots, see the GIS Lounge website.



Left: Undershoots happen when two line features are not properly joined, for example at intersection. Right: Overshoots refer to situations where a line feature extends too far beyond at intersecting line, rather than ending at the intersection.

Method

Undershoots: First, the `length_tolerance` (in meters) is defined in the cell below. Then, with `find_undershoots`, all pairs of dangling nodes that have a maximum of `length_tolerance` distance between them, are identified as undershoots, and the results are plotted.

Overshoots: First, the `length_tolerance` (in meters) is defined in the cell below. Then, with `find_overshoots`, all network edges that have a dangling node attached to them and that have a maximum length of `length_tolerance` are identifed as overshoots, and the results are plotted.

The workflow for over/undershoot detection below is inspired by Neis et al. (2012).

Interpretation

Under/overshoots are not necessarily always a data quality issue - they might be instead an accurate representation of the network conditions or of the digitization strategy. For example, a cycle path might end abruptly soon after a turn, which results in an overshoot. Protected cycle paths are often digitized in OSM as interrupted at intersections which results in intersection undershoots.

The interpretation of the impact of over/undershoots on data quality is context dependent. For certain applications, such as routing, overshoots do not present a particular challenge; they can, however, pose an issue for other applications such as network analysis, given that they skew the network structure. Undershoots, on the contrary, are a serious problem for routing applications, especially if only bicycle infrastructure is considered. They also pose a problem for network analysis, for example for any path-based metric, such as most centrality measures like betweenness centrality.

```
21 potential overshoots were identified with a length tolerance of 3 m.
11 potential undershoots were identified with a length tolerance of 3 m.
```

```
Interactive map saved at results/REFERENCE/cph_geodk/maps_interactive/overundershoots_3_3_refere
nce.html
```

Network components

Disconnected components do not share any elements (nodes/edges). In other words, there is no network path that could lead from one disconnected component to the other. As mentioned above, most real-world networks of bicycle infrastructure do consist of many disconnected components (Natera Orozco et al., 2020). However, when two disconnected components are very close to each other, it might be a sign of a missing edge or another digitizing error.

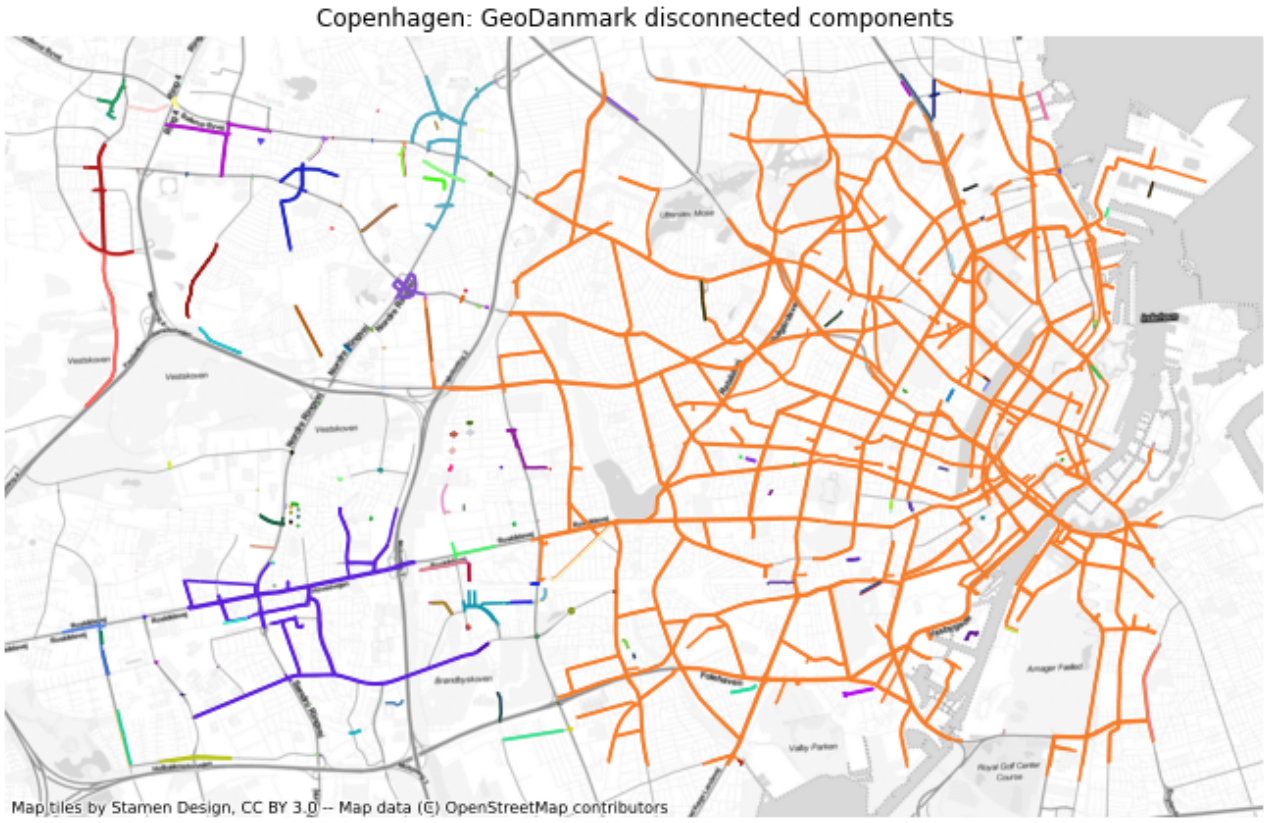
Method

First, with the help of `return_components`, a list of all (disconnected) components of the network is obtained. The total number of components is printed and all components are plotted in different colors for visual analysis. Next, the component size distribution (with components ordered by the network length they contain) is plotted, followed by a plot of the largest connected component.

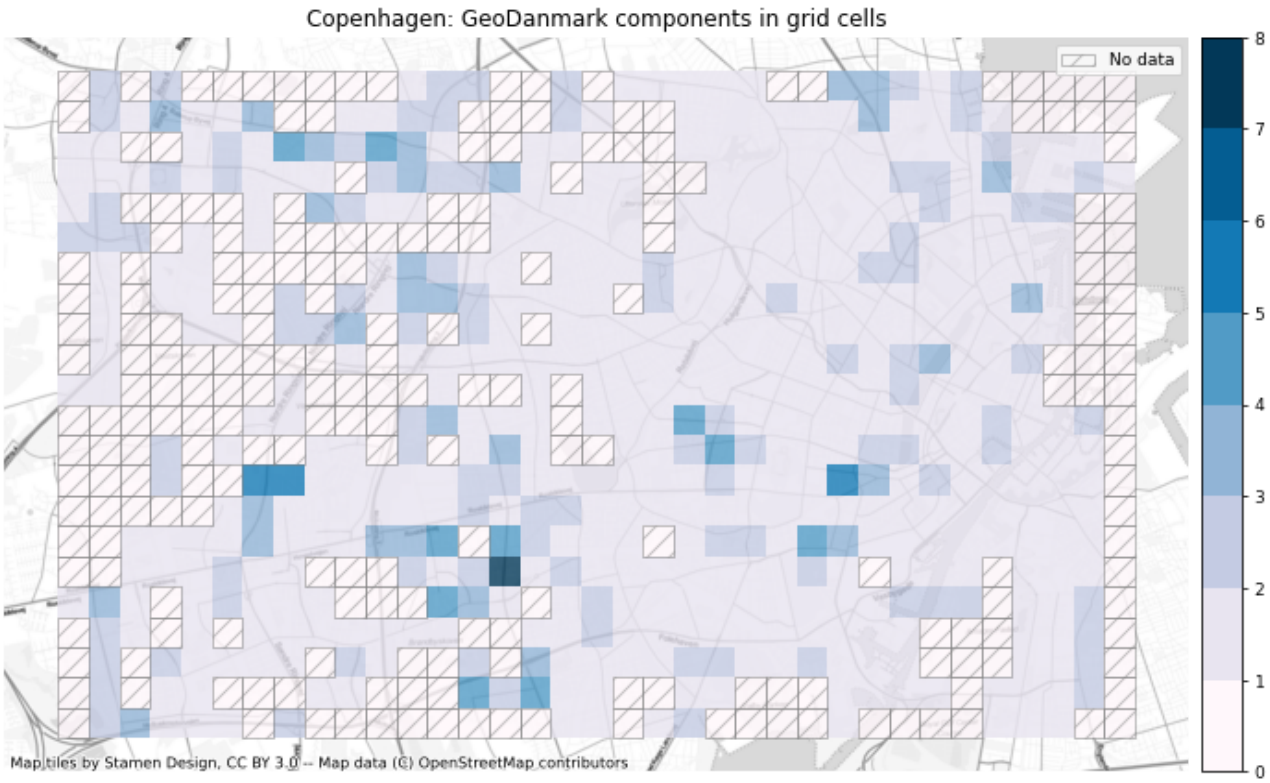
Interpretation

As with many of the previous analysis steps, knowledge of the area is crucial for a correct interpretation of component analysis. Given that the data represents the actual infrastructure accurately, bigger components indicate coherent network parts, while smaller components indicate scattered infrastructure (e.g., one single bicycle path along a street that does not connect to any other bicycle infrastructure). A high number of disconnected components in near vicinity of each other could indicate digitization errors or missing data.

`The network in the study area has 204 disconnected components.`



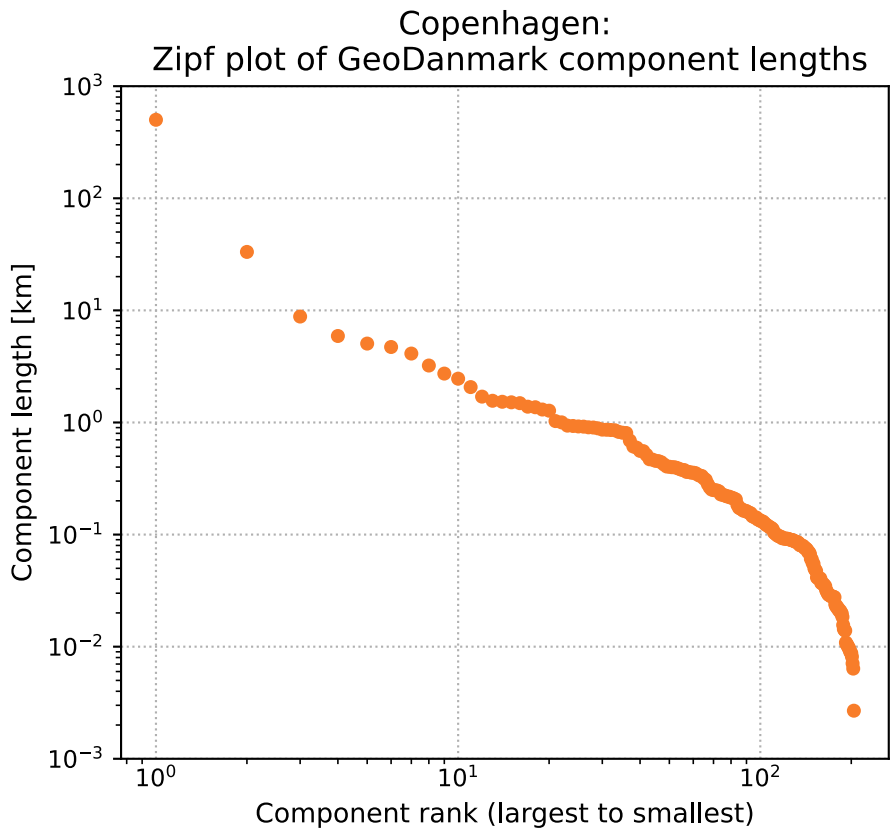
Components per grid cell



Component size distribution

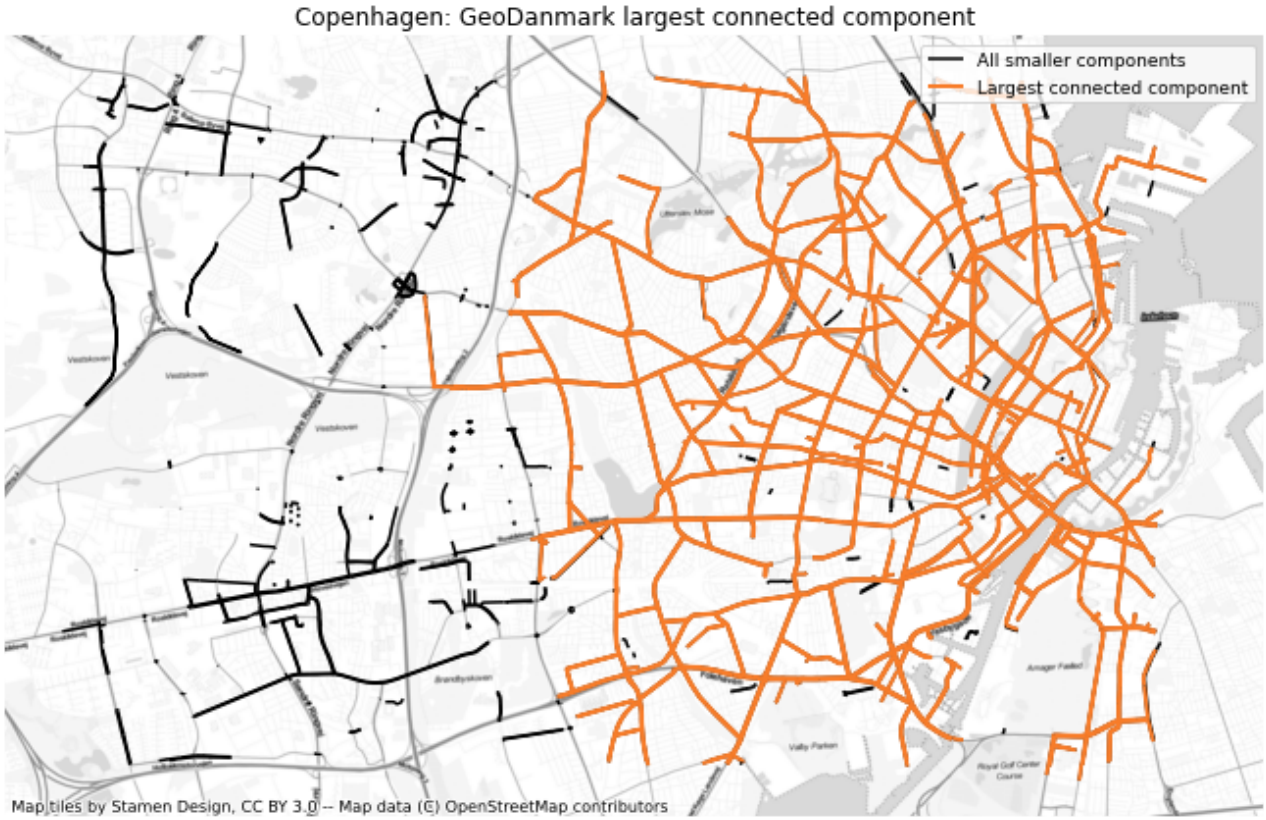
Many empirical distributions are skewed and often follow a power law, i.e. a straight line in a log-log plot, due to natural processes such as multiplicative network growth (Clauset et al., 2009). The network component size distribution (where size is length) can be visualized with a so-called Zipf plot, which plots the frequency of a component versus its rank (from largest to smallest). When a Zipf plot follows a straight line in log-log scale, it means that there is an much higher chance to find small disconnected components than expected by a distribution from an exponential family (like a normal distribution). This can mean that there has been no consolidation of the network, only piece-wise or random additions (Szell et al., 2022).

However, it can also happen that the largest connected component (the leftmost marker in the plot at rank 10^0) is a clear outlier, while the rest of the plot follows a different shape. This can mean that a consolidation *has* taken place, and that either a central planner has deliberately targeted to connect the network, or that the data are of high enough quality to have overcome many gaps.



The largest connected component contains 80.04% of the network length.

Largest connected component



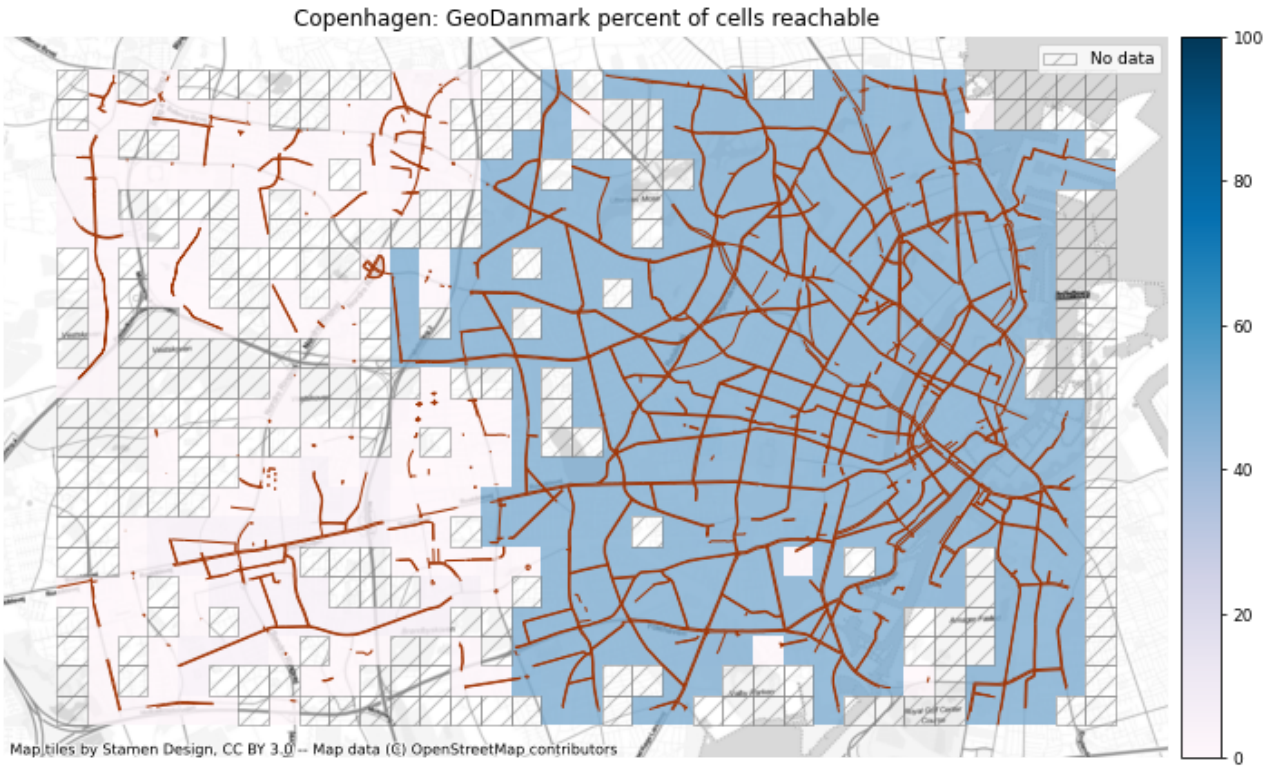
Missing links

In the plot of potential missing links between components, all edges that are within the specified distance of an edge on another component are plotted. The gaps between disconnected edges are highlighted with a marker. The map thus highlights edges which, despite being in close proximity of each other, are disconnected and where it thus would not be possible to bike on cycling infrastructure between the edges.

```
Interactive map saved at results/REFERENCE/cph_geodk/maps_interactive/component_gaps_10_referenc
e.html
```

Component connectivity

Here we visualize differences between how many cells can be reached from each cell. This is a crude measure for network connectivity but has the benefit of being computationally cheap and thus able to quickly highlight stark differences in network connectivity.



Summary

Intrinsic Quality Metrics - GeoDanmark Data

Total infrastructure length (km)	626
Protected bicycle infrastructure density (m/km2)	2,999
Unprotected bicycle infrastructure density (m/km2)	455
Mixed protection bicycle infrastructure density (m/km2)	0
Bicycle infrastructure density (m/km2)	3,454
Nodes	4,125
Dangling nodes	870
Nodes per km2	23
Dangling nodes per km2	5
Overshoots	21
Undershoots	11

Components	204
Length of largest component (km)	501
Largest component's share of network length	80%
Component gaps	52