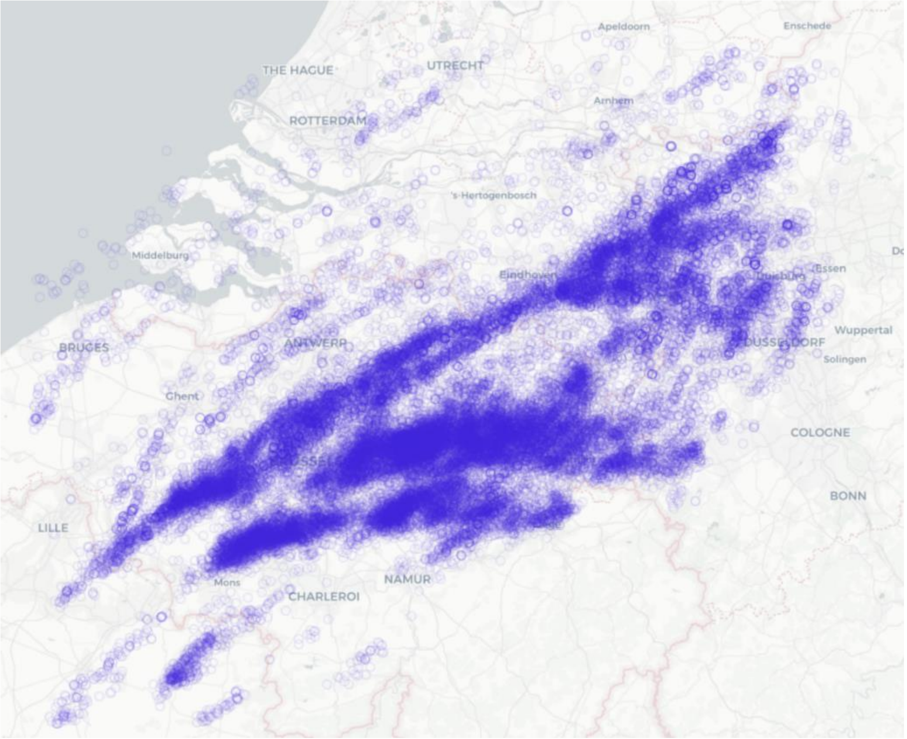
Lecture 07: Visual analytics of spatio-temporal data
Exercise description

Goals

- Get acquainted with two types of spatio-temporal data: spatial events and spatial time series.
- Get experience in performing spatio-temporal aggregation of events, including division of space and time. See how spatial time series are obtained through aggregation of spatial events.
- Understand two complementary views of spatial time series and possible ways to analyse the data with each of these views.

Data

The data records describe 53,702 lightning strike events that occurred over Europe on August 18, 2011 during the time interval from 13:30 till 17:30 (this is a subset of a set covering the whole day). Each record includes the geographic coordinates (longitude and latitude) and date-times of the events as well as some attributes.



Tasks

Study the spatio-temporal distribution of the strike events and describe how the thunderstorm evolved and moved over Europe by applying spatio-temporal aggregation to the events and analysing the resulting spatial time series.

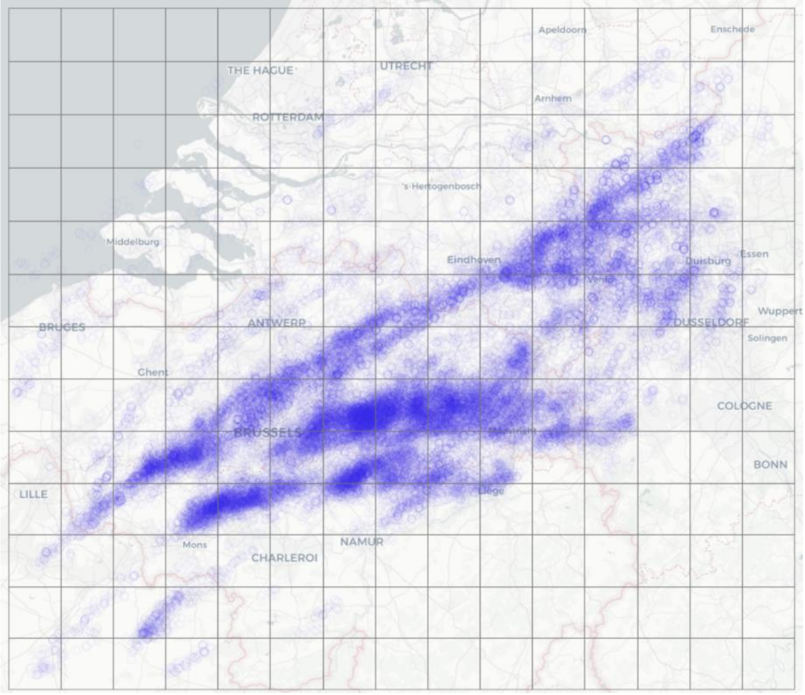
Spatio-temporal aggregation of the events

You partition the territory using a regular rectangular grid with the cell size 20x20 km.

Grid parameters:

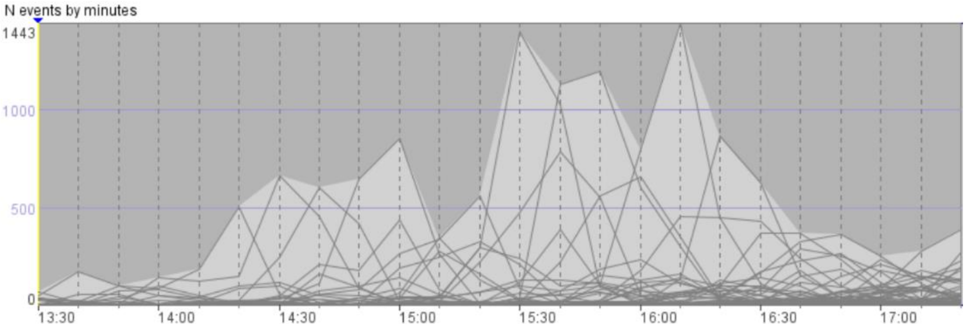
2.93 <= longitude <= 7.2287636 divided by 15 (number of columns)

49.95 <= latitude <= 52.28824 divided by 13 (number of rows)

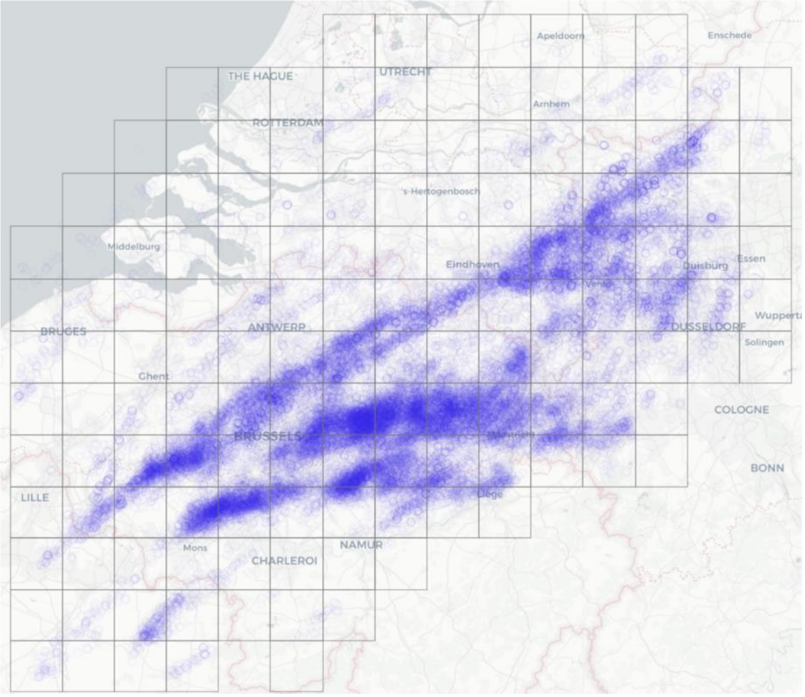


For the grid cells, you create time series of event counts by time intervals of the length of 10 minutes, i.e., you divide the time range of the data from 13:30 till 17:30 into 24 equal intervals. For your convenience (to save your efforts on transforming strings to datetime format and dividing the time into intervals), we have created an attribute "minute of the day" with values ranging from 810 to 1050. You can divide the events into time bins based on the values of this attribute.

After performing the aggregation, you will receive a set of time series, one per cell, similar to what is shown in the time graph below:

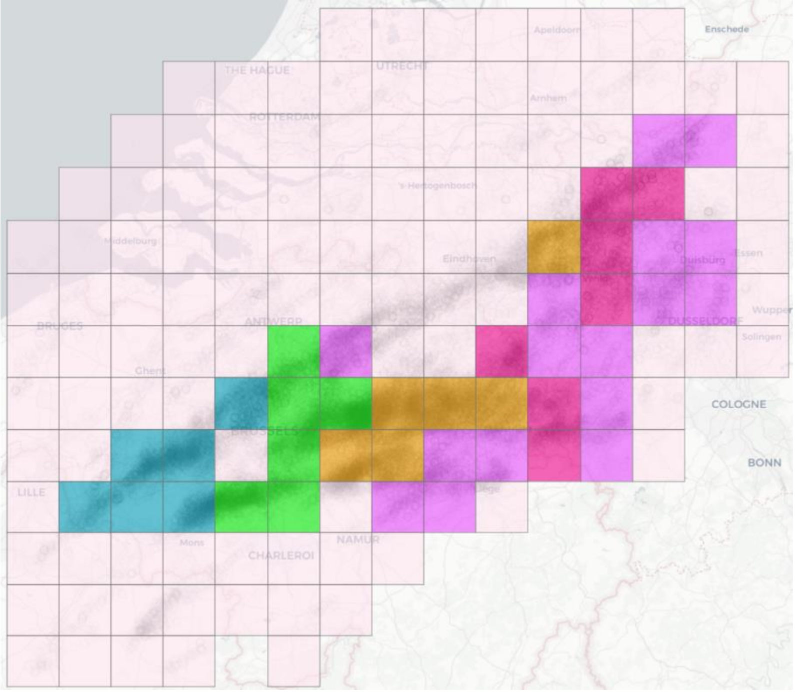


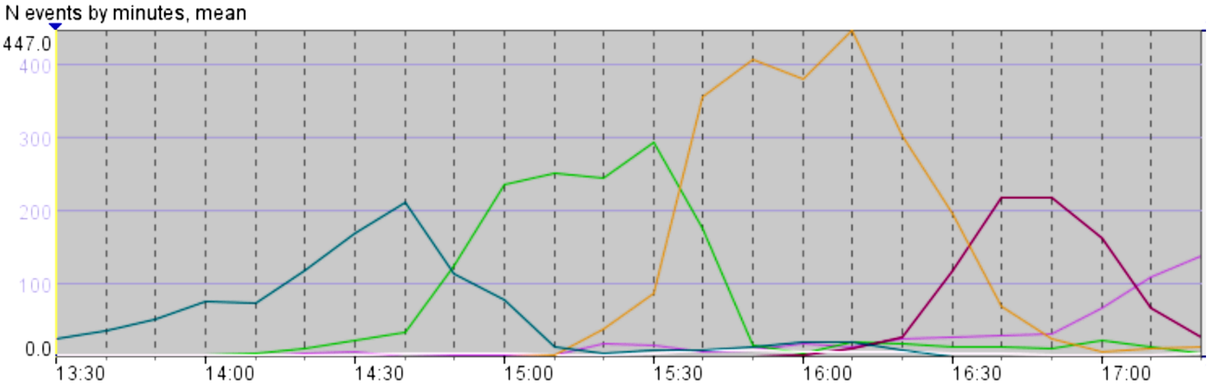
You also need to obtain the total counts of the events in each cell and remove the cells with no events from the further consideration by means of filtering:



Exploration of the spatial distribution of the local time series

You apply partition-based clustering (k-means) to the time series associated with the grid cells using Euclidean distance as the distance measure. You assign different colours to the clusters, paint the cells on the map in these colours, and observe the spatial distribution of the cluster members. You also generate representative time series for the clusters by computing the mean value for each time step. You try different number of clusters and find a good variant in terms of internal coherence of the clusters and interpretability of the spatial and temporal patterns.

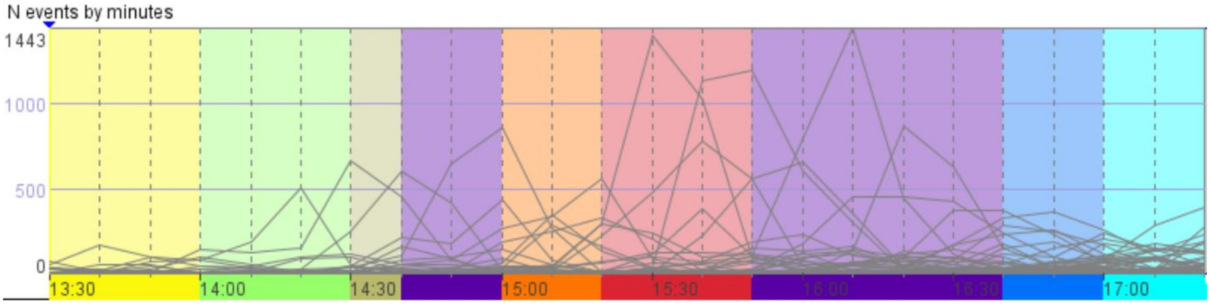


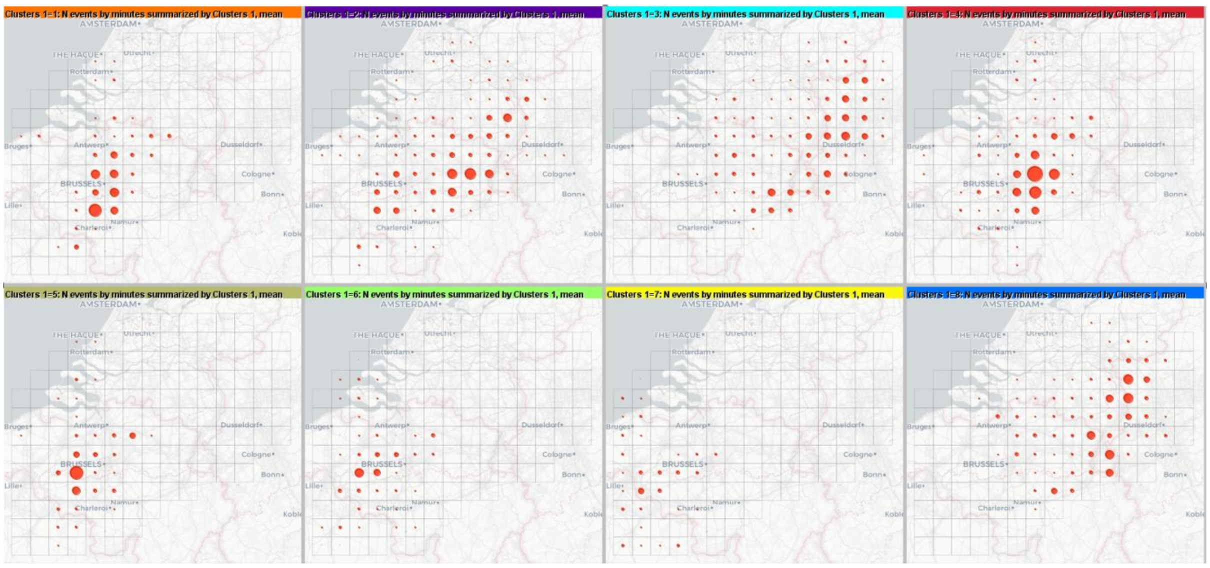


Exploration of the temporal evolution of the spatial distributions

Note: Clustering of the spatial distribution is not reflected in the Python notebook provided but is left for your own coding. The notebook includes transposing of the table with the time series, so that the rows in the resulting table correspond to the time steps and the columns to the cells. Clustering can be applied to the transposed table in the same way as to the original table. However, instead of clustering, the notebook includes creation of a 2D projection and representation of the evolution of the thunderstorm as a "trajectory" in the projection space. This demonstrates another possible approach to the exploration of the evolution of the phenomenon.

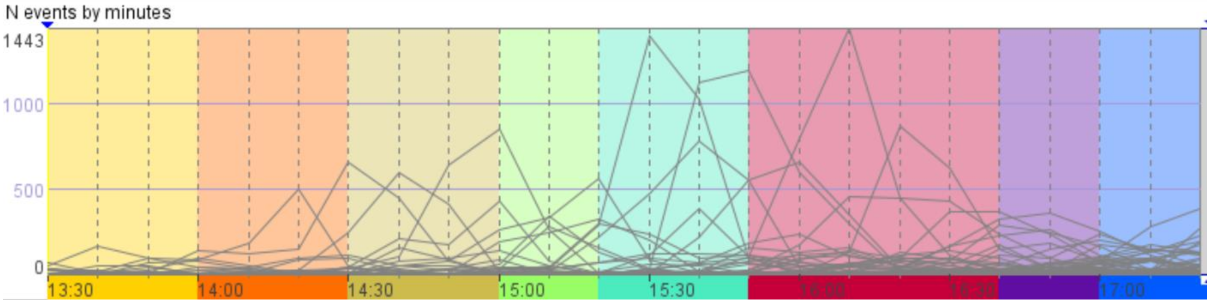
As a result of the spatio-temporal aggregation of the events, you have a table where the rows correspond to the grid cells, the columns to the time intervals, and the cells contain the counts. You need to transpose this table so that the rows correspond to the time steps and the columns to the grid cells (excluding the cells with no events). Now you apply the partition-based clustering to the rows of the transposed table, i.e., to the time steps. Again, you do the clustering several times for obtaining possibly clear and logical division of the time range 13:30-17:30. You compute the mean values for the cells and time clusters and observe the spatial distributions of the mean values corresponding to the time clusters.

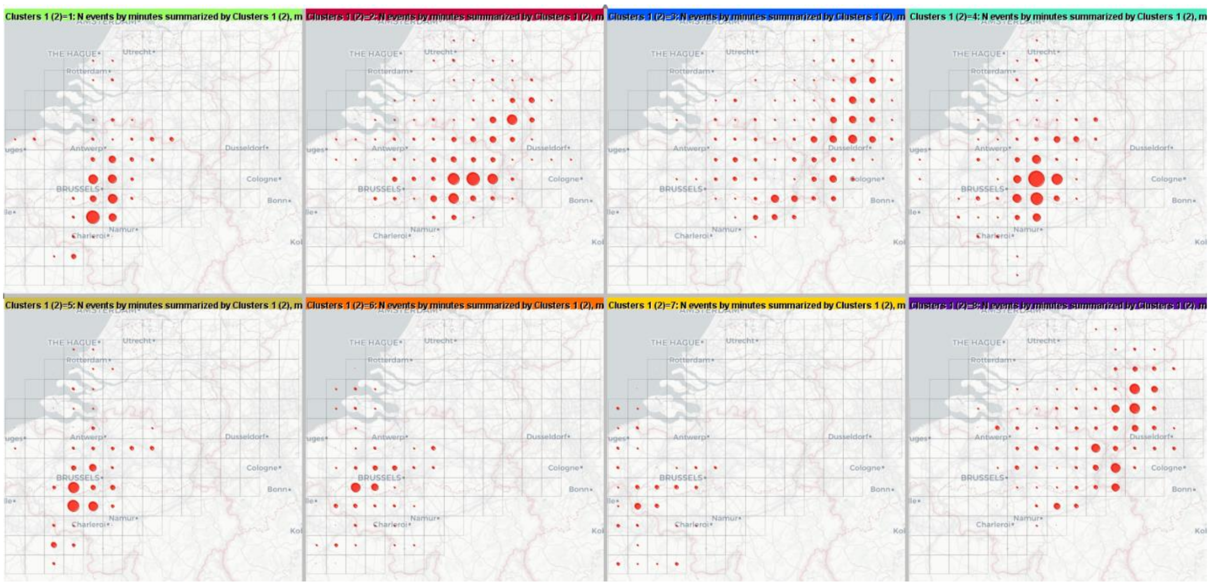



Clusters 1=1: N events by minutes summarized by Clusters 1, mean
Clusters 1=2: N events by minutes summarized by Clusters 1, mean
Clusters 1=3: N events by minutes summarized by Clusters 1, mean
Clusters 1=4: N events by minutes summarized by Clusters 1, mean
Clusters 1=5: N events by minutes summarized by Clusters 1, mean
Clusters 1=6: N events by minutes summarized by Clusters 1, mean
Clusters 1=7: N events by minutes summarized by Clusters 1, mean
Clusters 1=8: N events by minutes summarized by Clusters 1, mean
AMSTERDAM
THE HAGUE
Utrecht
Rotterdam
Antwerp
Bruges
BRUSSELS
Lille
Charleroi
Namur
Dusseldorf
Cologne
Bonn
Kobl

Please note that the maps above are not ordered chronologically. The same note refers to the next image with multiple maps.

You may wish to try progressive clustering, particularly, when you get a time cluster that is not temporally contiguous but consists of two or more parts. It is reasonable to select members of such clusters, as well as members of singletons (clusters consisting of single elements), and apply clustering to them, which will hopefully divide the selected subset in a better way.




Clusters 1 (2)=1: N events by minutes summarized by Clusters 1 (2), m
Clusters 1 (2)=2: N events by minutes summarized by Clusters 1 (2), m
Clusters 1 (2)=3: N events by minutes summarized by Clusters 1 (2), m
Clusters 1 (2)=4: N events by minutes summarized by Clusters 1 (2), m
Clusters 1 (2)=5: N events by minutes summarized by Clusters 1 (2), m
Clusters 1 (2)=6: N events by minutes summarized by Clusters 1 (2), m
Clusters 1 (2)=7: N events by minutes summarized by Clusters 1 (2), m
Clusters 1 (2)=8: N events by minutes summarized by Clusters 1 (2), m
THE HAGUE
Utrecht
Rotterdam
Antwerp
BRUSSELS
Charleroi
Namur
Dusseldorf
Cologne
Bonn

Questions

- How did the thunderstorm evolve over time (in terms of the spatial footprint and intensity)?

- What areas were highly affected by the storm (i.e., got many lightning strikes)? At what times were they affected the most?
- Which of the two complementary views of spatial time series is more suitable for answering each of the above questions?

Draft Python notebook

The provided Python notebook SpaceTime-aggregation-v.2019.10.14.ipynb performs basic data exploration and transformations needed for the task. The notebook also demonstrates how to perform clustering of locations according to similarities of corresponding time series. Try different numbers of clusters (the first line in cell [23]) and see its impact on the results. Compare time series for cluster centroids (cell [27]).

The notebook includes a code for creating a 2D projection of the time series according to their similarities (cell [22]) and assignment of colours to the cells based on the positions of their time series in the projection. Cell [21] contains a definition of the colour assignment function, and cell [22] includes assigning colours to the cells and using these colours in a map display. The idea is that similar items are located close to each other in the projection and therefore receive similar colours.

The same colour assignment function can also be used for assigning colours to clusters. For this purpose, the projection is applied to the cluster centroids.

After the code for the time series clustering, starting from cell [28], the notebook includes code for transposing the table with the time series so that the rows in the resulting table correspond to the time steps and the columns to the cells. Clustering can be applied to the transposed table in the same way as to the original table. However, instead of clustering, the notebook includes creation of a 2D projection of the value distributions over the cells (i.e., the rows of the transposed table) and representation of the evolution of the thunderstorm as a “trajectory” in the projection space. It shows application of two projection methods, PCA and MDS; the latter gives a clearer result with less over-plotting. Please note that large distances between consecutive positions of the trajectory mean large changes in the distribution, whereas small distances signify small changes.

Hence, the notebook covers only a part of the scenario described in the document and shown in the video using V-Analytics, but it includes some additional things for demonstrating other possible approaches to dealing with time series and multidimensional data.

You may try to do on your own:

- Assign colours to clusters based on their similarity. For this purpose you need to project centroids to 2D using MDS and then assign colours according to their positions using the function defined in the current notebook. This will resolve the problem of limited colour scale and enable visual perception of cluster similarities based on their colours.
- The notebook includes clustering of the cells according to their time series of event counts. Try to cluster the time steps according to the feature vectors representing the counts of the events in the cells. Visualize the centroids of the clusters (i.e., averaged per-cell values) in maps. Compare the clusters by subtracting their centroids and displaying difference maps.
- For both time-in-space and space-in-time clustering, analyse the sensitivity of the results to the parameters and examine the similarity of the cluster members to the centroids. For this purpose, for example, start with looking at frequency histograms of the distances to the centroids.

We suggest you to note your findings as comments in the notebooks you are using and to share your notebooks with the changes and notes you have made in the moodle forum.

We wish you a successful and fruitful fulfilment of the exercise.