2018 25th International Lightning Detection Conference &
7th International Lightning Meteorology Conference
March 12 - 15 | Ft. Lauderdale, Florida, USA

Python in Lightning Detection Network Data Analysis

P. Sarajcev and H. Matijasevic
Department of Power Engineering
University of Split, FESB
Split, Croatia
petar.sarajcev@fesb.hr

***Abstract*—This paper describes usage of several third-party Python (programming language) libraries, which could be seen as very useful for easy and efficient manipulation and analysis of data produced by the lightning detection networks (LDNs). Python is a high level interpreted language with extensive library of third-party packages and APIs. Several of its libraries, popular within the data science community, could be seen as very useful for efficient analysis of data produced by LDNs, such as: geographical and probability density distribution of lightning strikes, lightning flash-cell identification, its movement tracking, and nowcasting.**

***Keywords*—lightning; LDN; machine learning; Python; data analysis; clustering; nowcasting**

I. Introduction

Lightning detection networks, which nowadays span entire continents, provide extremely valuable information concerning the lightning phenomena, which can be of particular importance in several very different sectors of industry, e.g., electric power utilities (overvoltage and lightning protection, insulation coordination), insurance companies (risk analysis, insurance claims), aviation, weather bureaus, and even in different civil sectors (severe weather warnings, fire prevention, safety hazards at public events), [Cummins et al. , 1998; Hunt et al., 2016].

The basic products of these networks (of lightning detection sensors) can be seen as a time series of "geotagged" data objects with various additional features. For example, data produced by the LINET network [Betz et al. (2008), Betz et al. (2009)], which is operated by the German company nowcast GmbH (www.nowcast.de) and covers a European continent, features (among others) following attributes: (i) exact date-time stamp of lightning capture (down to the microsecond level), (ii) longitude and latitude of the strike position (with accuracy of three decimal places), (iii) lightning current amplitude, (iv) polarity of the lightning strike, (v) type of strike (cloud-to-cloud or cloud-to-ground), (vi) emission height of CC strokes, (vii) location detection accuracy information, etc. This data is available in several different formats (XML, TSV), can be retrieved using different protocols (FTP, SFTP, HTTPS), can be compressed or not, and can have different co-ordinate projections (via the selectable desired EPSG-ID).

This data can be analyzed from several different perspectives, such as [Campos et al., 2007; Kuk et al., 2010; Lakshmanan and Smith, 2009; Pedeboy et al., 2016; Poelman et al., 2017; Peters and Meng (2013); Qiu et al., 2013; Vasconcellos et al., 2006; Sarajcev, 2016]: (a) long-term yearly, seasonal, and diurnal lightning activity over a particular region of interest, (b) geographical density distribution of lightning strikes (i.e. ground flash density maps), (c) statistical cumulative distribution function of lightning-current amplitudes for a particular region and/or time-frame, (d) positive-to-negative lightning ratio for a particular region and season, (e) lightning flash-cell identification, (f) flash-cell movement tracking, and (g) nowcasting. Treatment of the data is predicated on the analysis for which it is intended, but should be as easy and efficient as possible. For example, analysis (a) is concerned primarily with the time-related data features, while analysis (b) is concerned primarily with a geographic-related features. Hence, efficient manipulation of time-series data is seen as a necessary precondition for analysis (a), as is manipulation of geospatial data for the analysis (b). Furthermore, analysis (e) uses clustering (and outlier detection) of lightning data, which falls under a machine learning domain [Betz et al., 2008; Pedeboy et al., 2016)]. It also makes use of convex hulls, which need to be manipulated easily in order to implement analyses mentioned under (f) and (g).

Python has a powerful arsenal of libraries (https://pypi.python.org) which can be of use for these very different tasks, such as (alphabetically ordered): basemap, datetime, fiona, folium, geopandas, geos, hdbscan, matplotlib, numpy, pandas, pickle(shelve), requests, scikit-learn, scikit-image, scipy, seaborn, shapely, statsmodels, and many others.

A brief introduction to the application of some of these libraries, for several aspects of LDN data processing, will be demonstrated in the paper, as follows: (1) seasonal, weekly, daily, and diurnal lightning activity, (2) kernel density estimation of lightning-current amplitudes statistical probability distribution, (3) bivariate kernel density estimation of geographical lightning distribution, and (4) clustering for lightning flash-cell identification for lightning tracking and nowcasting.

II. Analysis of Lightning Data

A detailed information related to the lightning activity over the European continent can be obtained from the LINET network, operated by the German company nowcast GmbH, where datasets for different regions and spanning different time-windows can be purchased on-demand. One typical small dataset has been obtained, for research purposes only, covering a rectangular area of some 400 km^2 of the Adriatic hinterland on the Croatian side and spanning a time-window of one calendar year. The mentioned area contains within it four individual wind farms, with forty wind turbines in total.

Table I present an example of lightning data obtained from LINET network, where type "1" means cloud-to-ground strike (CG) and type "2" indicates cloud-to-cloud strike (CC). A polarity of each strike is indicated with the amplitude (kA). Additional data related to the location detection accuracy and other features (height of CC strikes) is also available.

TABLE I. Example of Lightning Data Parameters

Timestamp	Longitude	Latitude	Type	Amplitude
2014-01-19 18:44:05.767	15.901	43.696	1	-23.2
2014-01-19 18:44:06.176	15.938	43.702	2	5.9
2014-01-19 18:51:20.268	15.894	43.743	1	-116.6
2014-01-19 18:53:58.817	15.901	43.778	1	9.9
2014-01-19 23:13:53.933	15.918	43.674	2	-5.1

A. Seasonal, weekly, and diurnal lightning activity

Using the "pandas" (https://pandas.pydata.org) Python library, carrying out complex analysis of the presented dataset becomes a straightforward task. Pandas has a powerful set of features for dealing with time series. Furthermore, "geopandas" package (http://geopandas.org) combines powerful features of the pandas library with the geometric features of the "shapely" Python library, enabling efficient manipulation of geospatial information contained within the lightning dataset.

Fig. 1 presents a "violin plot" of the aggregated seasonal CG lightning activity for the analyzed region. It can be seen that the winter and summer lightning have very different statistical distributions, and furthermore, that statistics of positive and negative lightning are quite different (as would be expected). It has been found that the proportion of positive lightning is far higher than 10 % that is often assumed. At the same time, Fig. 2 presents weekly aggregated lightning data for the whole year. Figures are prepared using the "matplotlib" (www.matplotlib.org) library.

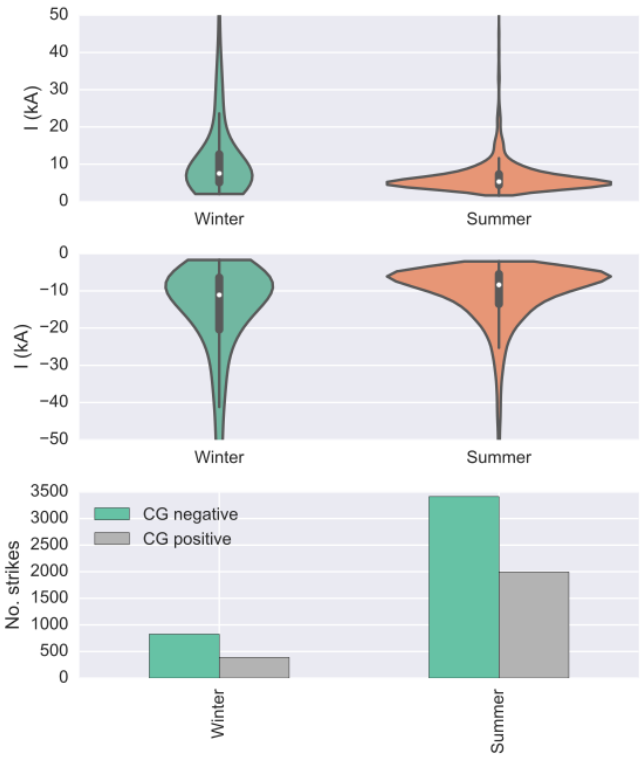

Fig. 1. Violin plot of the seasonal lightning activity.

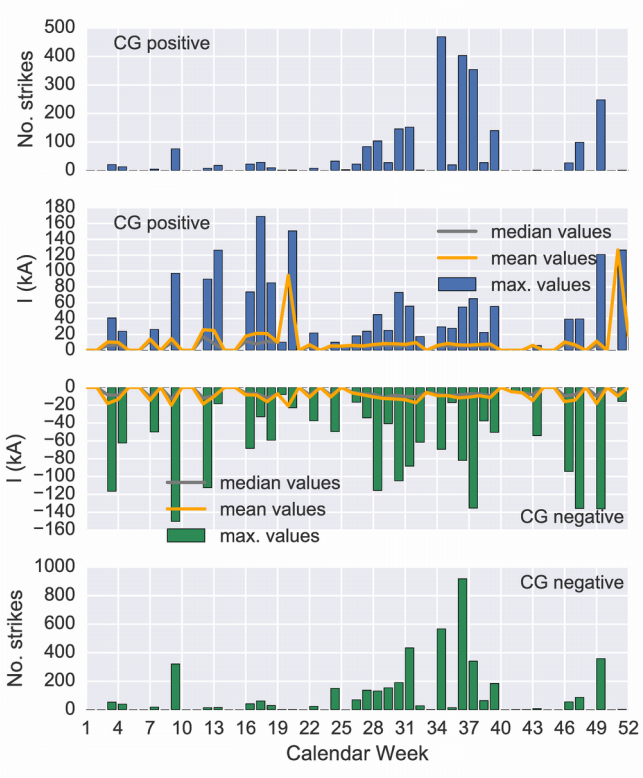

Fig. 2. Weekly aggregated lightning activity.

It is interesting to notice that this dataset features more than a dozen days with very high lightning activity, although they would still count toward a single thunderstorm day. One particular date (2014-09-01) features over 1200 CG lightning strikes, distributed over several rather narrow time intervals with extreme CG lightning activity. As an example, Fig. 3 presents a "stem plot" of the 10-minute interval of very intense lightning activity. Diurnal pattern exhibits rather short time intervals with bursts of lightning activity (both CG and CC), interspersed with quiet intervals.

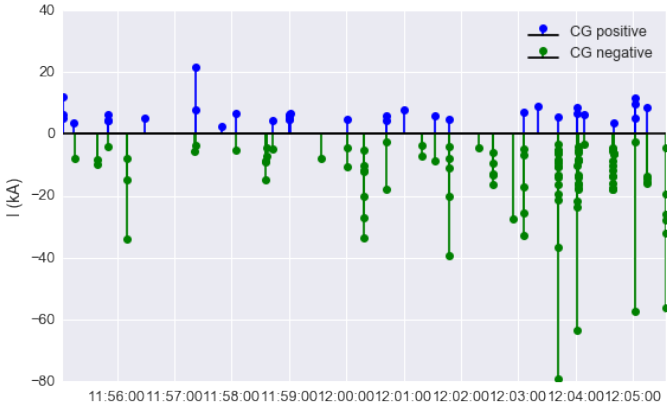

Fig. 3. Stem plot of the10-minute lightning activity.

B. *Geographical and probability density distributions*

Recorded geographical locations (latitude and longitude) of each CG lightning strike can be employed for the purpose of obtaining a detailed local distribution density of lightning strikes, which relates directly to the lightning risk. This has been tackled here using the bivariate kernel density estimation (using Gaussian kernels) in spherical geometry (i.e. using haversine distance), and mapping the ensuing distribution by means of the Mercator projection. The "scikit learn" Python library (http://scikit-learn.org/stable/index.html) has been used for the bivariate KDE procedure, along with the "basemap" library (https://matplotlib.org/basemap/) for the subsequent Mercator projection. The result of this procedure is the very detailed map of local distribution density of CG lightning strikes, graphically depicted in Fig. 4. It has been noted that the lightning pattern follows terrain features quite well, even with this small dataset.

Negative CG lightning amplitudes from the dataset has been fitted with the Log-Normal distribution and the result is graphically depicted in Fig. 5. It includes histogram of the data, obtained PDF, along with a "probability plot" which holds the same information as the better known "QQ plot". The "statsmodels" (http://www.statsmodels.org) and parts of the "Scipy" (www.scipy.org) Python libraries have been used for that purpose. It can be seen from the probability plot that the Log-Normal distribution is in-fact not a good fit for the data. In addition, it has been found that the median of this Log-N distribution is far lower than the 30 kA, which has often been assumed (see IEC 62305 for more information). In fact, it has been shown on several occasions [Franc et al., 2017; Holler at al., 2009] that the median of the Log-Normal distribution of lightning data is significantly below the 30 kA level.

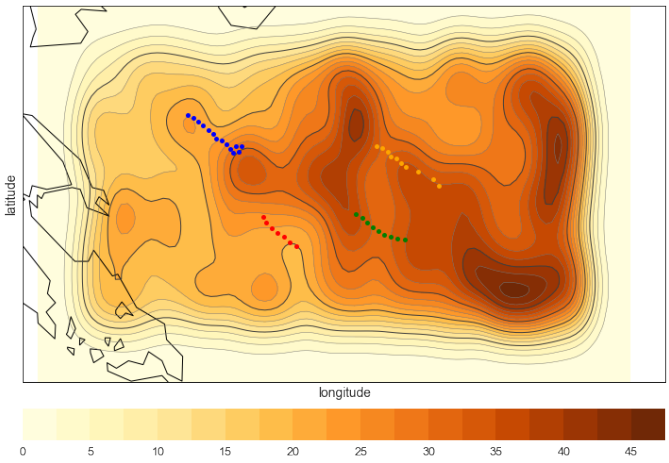

Fig. 4. Bivariate kernel density estimation of geographical distribution of negative CG lightning activity.

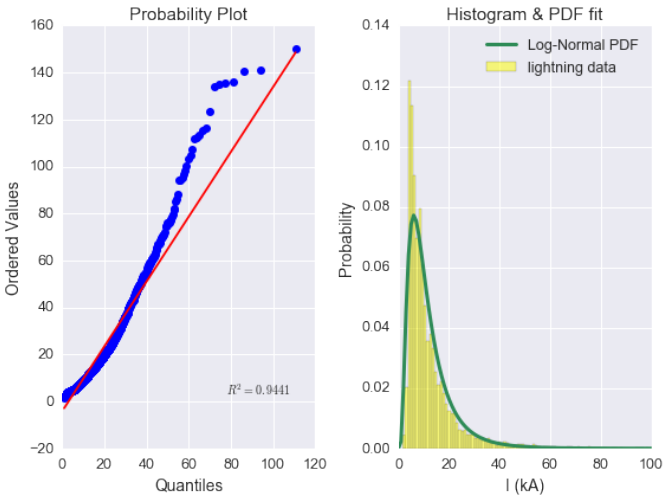

Fig. 5. Fitting Log-N distribution of negative CG lightning amplitudes.

C. *Wind turbine lightning incidence*

Wind turbine (WT) lightning incidence, in terms of its attractive area to lightning, can be established, either by using the "striking distance" or the "attractive radius" concepts. In general, attractive radius has a smaller magnitude than the corresponding striking distance, which can be of importance for establishing the WT lightning incidence. Using a distance measure (i.e. orthodromic distance) between the WT position and the position of each lightning strike (both defined in terms of the longitude and latitude), it can be determined if the lightning strikes within the attractive area of the WT, which counts as a lightning strike. An example of the statistical frequency distribution (of geographical locations) of lightning strikes to a particular WT (130 m tall on a 300 m tall hill, in terms of distances in meters and incident angles), is graphically presented in Fig. 6 as a "windrose plot" centered at the WT position [Sarajcev et al., 2016]. This distribution might give an indication on the principal direction of the lightning incidence, establishing the prevailing direction of the thunderstorm movement (e.g. for the purpose of shielding mast positioning).

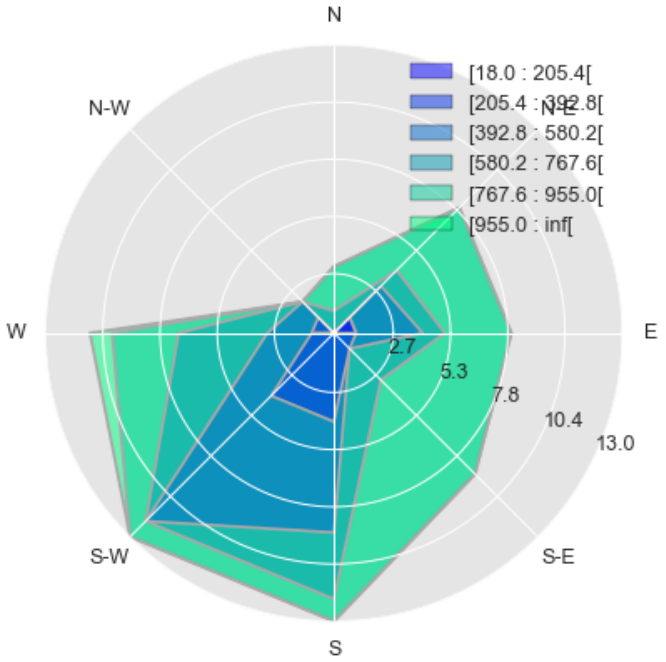

Fig. 6. Windrose plot of the geographical distribution of WT incident lightning strikes.

D. Clustering for lightning flash-cell identification

The problem of lightning flash-cells detection, viewed from the perspective of using exclusively the lightning activity data, presents itself as the problem of finding time-and-space data clusters, which falls under the unsupervised machine learning domain [Goodman, 1990; Peters and Meng, 2013; Vasconcellos at al., 2006]. The "scikit learn" Python library (http://scikit-learn.org/stable/index.html) has a very powerful arsenal of clustering algorithms, from a simple K-Means algorithm to the very sophisticate DBSCAN algorithm (including its hierarchical HDBSCAN version). Furthermore, clustering algorithms can execute in parallel on multi-core processor architectures, can operate in different metric spaces, can automatically eliminate outliers, and offer fine-tuning of hyper-parameters using (randomized) grid search and cross-validation.

Lightning flash-cells detection starts by first aggregating time-series lightning data in 10-minute intervals, and then searching for spatial clusters within these time windows. Large ratio of CC to CG lightning within these moving windows is a strong indicator of lightning flash-cell formation. Fig. 7 presents an example of the time-series lightning data analysis with a 10-minute moving windows. Spatial, geometric clusters of longitude-latitude lightning data, which can be identified, are very often irregular in shape and of varying density. Hence, sophisticate clustering algorithms are needed here, those which can cope well with irregular shape and varying density of clusters. The HDBSCAN algorithm can be seen as very useful in detecting and identifying lightning flash-cells as clusters of data. Fig. 8 presents an example of three clusters which have been identified using a HDBSCAN algorithm on a particular 10-minute interval lightning data.

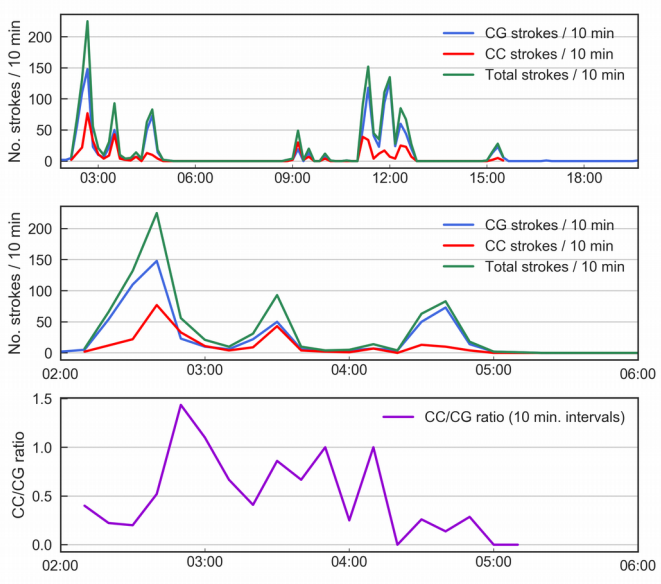

Fig. 7. Time-series lightning data in a 10-minute moving window.

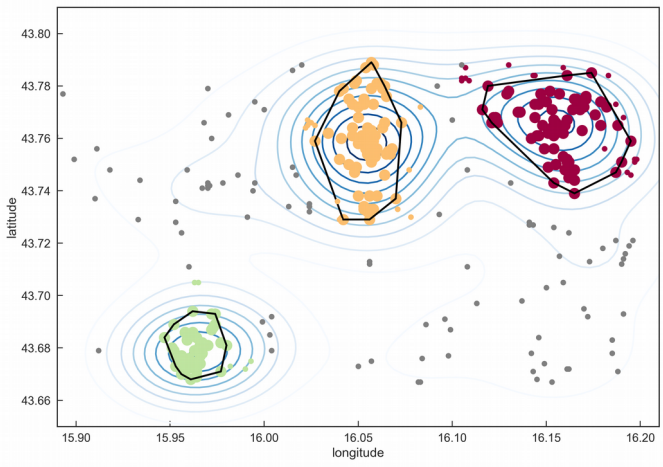

Fig. 8. Clustering lightning data using HDBSCAN algorithm.

It can be seen from the Fig. 8 that not all data points (i.e. lightning strikes) have been associated with all clusters. This is a strong feature of the HDBSCAN algorithm as such, which automatically removes noise and data outliers. Also, lightning flash-cells have been identified from the clusters of data using the "convex hull" algorithm and visually presented in Fig. 8 as well. The "shapely" Python library has a useful set of functions (http://toblerity.org/shapely/manual.html) for creating convex hulls and manipulating different planar shapes (intersection, union, translation, interaction, etc.).

Another feature of the hierarchical algorithms is the so-called dendrogram, which can visually represent subtle hierarchical structure that can be found within the geographical lightning data. Fig. 9 presents a "dendrogram plot" of the data from the Fig. 8. It can be seen that different number of clusters emerge at different distance measures, and that three clusters can be distinctly identified (from the length of the stems).

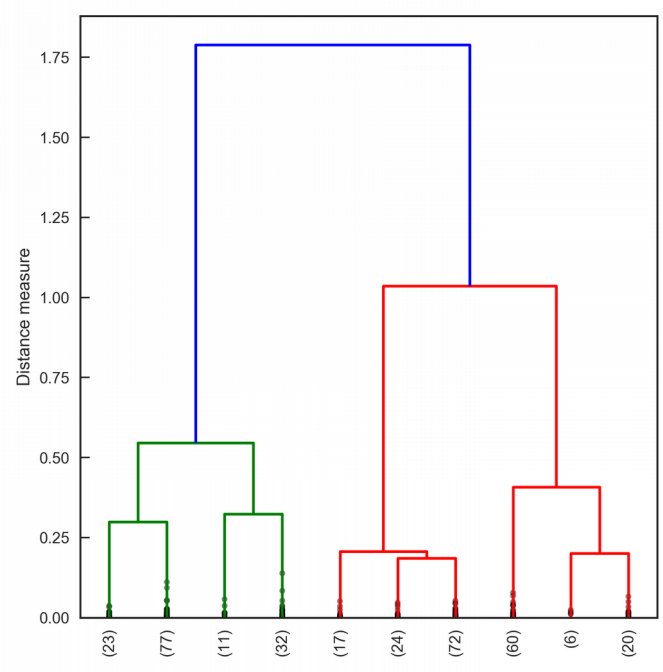

Fig. 9. Dendrogram plot of the lightning data from Fig. 8.

Geographical distributions of lightning strike locations (within the time windows of interest), which is important for the lightning flash-cell identification, often exhibit subtle hierarchical structures, particularly when very large areas are scrutinized [Campos and Pinto, 2007]. This structure within the data can be visualized using the dendrograms and clusters can be positively identified (even the structure within the main clusters can be visualized and associated with particular distance measures). Dendrograms are readily available from the extensive "Scipy" Python library (www.scipy.org), along with additional functions for hierarchical data analysis and clustering. When the information from the dendrogram is further combined with, e.g. "silhouette score" analysis, it can provide valuable insights into the structure of the flash-cells and their mutual interactions.

E. Lightning flash-cell tracking and nowcasting

Tracking of lightning flash-cells is a very complex task, which involves identifying cells between successive time windows, as well as allowing for the cells to split and merge, to die (i.e. disappear), and for new cells to be born [Dixon and Wiener, 1993; Johnson at al., 1997; Steinacker et al., 2000; Kalinis et al., 2005; Betz et al., 2009]. This can be rather difficult, and different algorithms have been proposed by various authors, with varying degrees of both complexity and success rate in tracking the flash-cells. A path of the cell movement is usually formed by connecting the centers of the convex hulls, which geometrically represent flash-cell in each time window [Peters and Meng, 2013].

Nowcasting of flash-cells, on the other hand, is concerned with predicting their position in the near future (i.e. during the several successive time windows), using the existing data from the past. In other words, nowcasting is the short-term forecasting of the flash-cell movement and position [Johnson et al., 1997; Betz at al., 2008; Peters and Meng, 2013]. Several algorithms have been proposed for this task as well, having various degrees of complexity and success rate.

One simple algorithm for lightning flash-cells nowcasting has been proposed in the work of Peters and Meng [2013], which has been applied here. No attempt at detection of flash-cell split and merge conditions has been attempted. Furthermore, a time-weighted linear least-squares regression (WLS) analysis has also been performed, and its results superimposed on the identified flash-cell path, for the forecasting of the direction of the possible flash-cell track during nowcasting.

Fig. 10 presents results of the tracking and nowcasting of a single flash-cell, using the algorithm from Peters and Meng [2013] and the WLS analysis. A combination of several Python libraries has been employed in producing Fig. 10. The shaded cone in Fig. 10 presents the most probable location of the flash-cell center for a future 10-minute time window (dark shaded cone is obtained with a single standard deviation of the predicted position, while light shaded cone takes into account two standard deviations). Red line is obtained by connecting centers of the convex hulls (red dots), from several successive time windows, where each convex hull represents the same flash-cell in different time instants. Area of the convex hull and its density changes with time, but there is no split or merge of the cell (it has a stable life with a distinct path). Solid blue line identifies a path from the WLS analysis of recorded positions of convex hull centers, while dash and dash-dot paths provide 95% confidence and prediction intervals, respectively. In the WLS analysis, more recent cell positions are given higher weights than the older ones (i.e. exponential weighting of data in the time domain).

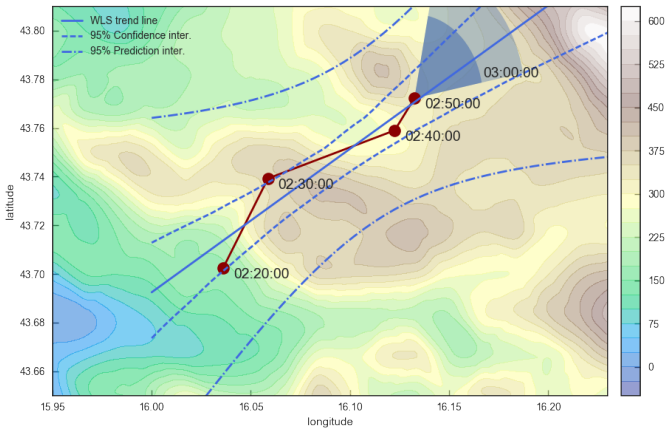

Fig. 10. Tracking and nowcasting of lightning flash-cell.

F. Interactive lightning data visualization

Interactive visualization and analysis of lightning data can be easily accomplished using the "folium" Python package (https://github.com/python-visualization/folium), that is based on the extensive "leaflet" JavaScript library. Folium API provides access to GeoJSON objects and shapefiles (popular geospatial vector data format for geographic information system software).

III. Conclusion

This paper briefly introduced application of several third-party Python (programming language) libraries, which could be seen as useful for easy and efficient manipulation and analysis of data produced by the lightning detection networks (LDNs). Python is a high level interpreted language with extensive library of third-party packages and APIs. At the time of this writing, Python ecosystem features over 125,000 packages (according to the PyPI repository) and is increasing rapidly. Its open source nature, very generous licensing terms, beautiful syntax, flexibility, and adaptability (support for both functional and object-oriented programming), along with a wide adoption by the data science community, resulted with its extremely rapid (and even unprecedented) growth in both user and code bases. Several of its libraries, popular within the larger data science community, could be seen as very useful for efficient analysis of data produced by LDNs. These include libraries for time-series data analysis, libraries for geographical and geometrical data analysis, libraries for advanced statistical modeling, data mining, and building sophisticate machine learning models (unsupervised clustering domain). It also includes libraries for creating beautiful and interactive visualizations of lightning data, as well as different kinds of specialized plots, such as: box and violin statistical plots, stem plots, polar plots, windrose plots, probability and QQ plots, histograms, dendrograms, cluster visualizations, density maps, cartographic projections of data, and other kinds of sophisticate data visualizations. It further includes libraries for specialized data access and manipulation, such as: HDF5, GeoJSON, and Shapefiles. Finally, it provides APIs for accessing Amazon, Spark, and Hadoop clusters, as well as for harnessing power of NVIDIA graphics processors with a CUDA platform (using PyCUDA). This enables streamlined access and fast processing of massive amounts of data.

Appendix

This paper is accompanied by a Jupyter Notebook which contains Python source code. Notebook features analysis of lightning activity, geographical lightning density distribution, kernel density estimation of amplitudes probability distribution, wind farm lightning incidence analysis, clustering analysis for lightning flash-cells identification, flash-cell tracking, and nowcasting. It can be freely accessed at the following link: https://nbviewer.jupyter.org/github/sarajcev/linet-lightning/blob/master/lightning.ipynb

References

Betz, H. D., K. Schmidt, P. Laroche, P. Blanchet, W. F. Oettinger, E. Defer, Z. Dziewit, and J. Konarski (2009), LINET - An international lightning detection network in Europe, Atmospheric Research, 91, pp. 564-573.

Betz, H. D., K. Schmidt, W. P. Ottinger, and B. Montag (2008), Cell-tracking with lightning data from LINET, Advances in Geosciences, 17, pp. 55-61.

Campos, D. R., and O. Pinto Jr. (2007), Investigation about the intensity and location of the maximum cloud-to-ground lightning flash density in the city of Sao Paulo, IX Intl. Symposium on Lightning Protection, Foz do Ignacu, Brazil

Cummins, K. L., E. P. Krieder, and M. D. Malaone (1998), The U.S. National Lightning Detection Network™ and Applications of Cloud-to-Ground Lightning Data by Electric Power Utilities, IEEE Trans. on Electromag. Compat., 40(4), pp. 465-480.

Dixon, M. and G. Wiener (1993), TITAN: Thunderstorm identification, tracking, analysis, and nowcasting - A radar-based methodology, Journal of Atmospheric and Oceanic Technology, 10(6), pp. 785-797.

Franc, B., N. Stipetic, I. Uglesic, K. Mesic, and I. Ivankovic (2017), Improvement of the correlation process in lightning location system, 13. HRO CIGRE Session, 5 – 8 November, Sibenik, Croatia (in Croatian).

Goodman, S. J. (1990), Predicting thunderstorm evolution using ground-based lightning detection networks, NASA Technical Memorandum TM-103521, NASA.

Holler H., H.-D. Betz, K. Schmidt, R. V. Calheiros, P. May, E. Houngninou, and G. Scialom (2009), Lightning characteristics observed by a VLF/LF lightning detection network (LINET) in Brazil, Australia, Africa and Germany, Atmospheric Chemistry and Physics, 9, pp. 7795-7824.

Hunt, H. G. P., K. J. Nixon, and J. A. Naude (2017), Using lightning location system stroke reports to evaluate the probability that an area of interest was struck by lightning, Electric Power Systems Research, DOI: https://doi.org/10.1016/j.epsr.2016.12.010 (in press).

Johnson, J. T., P. L. MacKeen, A. Witt, E. DeWayne Mitchel, G. J. Stumpf, M. D. Eilts, and K. W. Thomas (1997), The storm cell identification and tracking algorithm: An enhanced WSR-88D algorithm, Weather and Forecasting, 13, pp. 263-276.

Kalnis, P., N. Mamoulis, and S. Bakiras (2005), On discovering moving clusters in spatio-temporal data, in book: C. B. Medeiros, et al. (Eds.), Advances in Spatial and Temporal Databases, Springer, Berlin.

Kuk, B-J., J-S. Ha, H-I. Kim, and H-K. Lee (2010), Statistical characteristic of ground lightning flashes over the Korean peninsula using Cloud-to-Ground lightning data from 2004-2008, Atmospheric Research, 95, pp. 123-135.

Lakshmanan V., and T. Smith (2009), Data mining storm attributes from spatial grids, Journal of Atmospheric and Oceanic Technology, 26, pp. 2353-2365.

Pedeboy, S., P. Barneoud, and C. Berthet (2016), First results on severe storms prediction based on the French national Lightning Locating System, 24th Intl. Lightning Detection Conf. & 6th. Intl. Lightning Meteorology Conf., San Diego, California

Peters, S., and L. Meng (2013), Visual analysis for nowcasting of multidimensional lightning data, ISPRS International Journal of Geo-Information, 2, pp. 817-836.

Poelman, D. R., W. Shulz, R. Kaltenboeck, and L. Delobbe (2017), Analysis of lightning outliers in the EUCLID network, Atmospheric Measurement Techniques Discussions, pp. 1-25, DOI: 10.5194/amt-2017-150

Qiu, T., S. Zhang, H. Zhou, X. Bai, and P. Liu (2013), Application study of machine learning in lightning forecasting, Information Technology Journal, 12(21), pp. 6031-6037.

Sarajcev, P., J. Vasilj, and D. Jakus (2016), Monte-Carlo analysis of wind farm lightning-surge transients aided by LINET lightning-detection network data, Renewable Energy, 99, pp. 501-513.

Steinacker, R., M. Dorninger, F. Wolfelmaier, and T. Krennert (2000), Automatic tracking of convective cells and cell complexes from lightning and radar data, Meteorology and Atmospheric Physics, 72, pp. 101-110.

Vasconcellos, C. A., C. Beneti, F. Sato, L. C. Pinhero, and C. L. Curotto (2006), Electrical thunderstorms nowcasting using lightning data mining, 19th Intl. Lightning Detection Conf. & 1st Intl. Lightning Meteorology Conf, Tuscon, Arizona.