AGILE: GIScience Series, 5, 19, 2024. https://doi.org/10.5194/agile-giss-5-19-2024
Proceedings of the 27th AGILE Conference on Geographic Information Science, 4–7 Sept. 2024.
Editors: Alison Heppenstall, Mingshu Wang, Urska Demsar, Rob Lemmens, and Jing Yao.
This contribution underwent peer review based on a full paper submission.

Development of a tool to calculate distance for veterinary epidemiological applications

Mirco Cazzaro[1 2], Francesca Scolamacchia[1], Paolo Mulatti[1], and Nicola Ferrè[1]

[1]Istituto Zooprofilattico Sperimentale delle Venezie, Legnaro, Italy
[2]Department of Information Engineering, University of Padua, 35122 Padova, Italy

Correspondence: Mirco Cazzaro (mcazzaro@izsvenezie.it)

Abstract. The transmission of infectious diseases is intricately linked to spatiotemporal proximity. Spatial data used in veterinary epidemiology to depict the distribution of disease events typically encompasses pairwise distances between two locations. The Euclidean method is commonly employed to compute such distances. We have developed a tool capable of calculating pairwise distances, considering factors such as orography, the road network, or the land use, given a set of points representing farms or outbreak sites where samples have been collected. The outcome is a three dimensional object comprising six distinct pairwise matrix distances: Euclidean, Haversine, route, route with elevation, orographic/elevation and cost/friction. These distances offers opportunities for exploring novel approaches to integrating the spatial component into epidemiological applications.

Keywords. Veterinary epidemiology, distance, matrix distances

1 Introduction

Epidemiology entails the investigation of determinants, occurrence, and distribution of health and disease within a defined population across space and time. A determinant, fundamental to any epidemiological study, includes the inquiry of "where does the disease occur". Place can refer to an area where animals are housed or managed (e.g. farms), or the natural habitat of wild animal species. The most common method of representing a place is through a single point. For instance, a point might signify the location of the farmhouse where the disease occurred, or a site where an infected vector was found. Among the classical applications in veterinary epidemiology, the analysis of disease events clustering and disease spread modelling are particularly noteworthy. In these applications, spatial information is usually managed in terms of distance between places, calculated for each pair of points, allowing the determination of the distance from any point to other. This results in a square matrix (i.e. a two dimensional array) containing the pairwise distances between any two given places. To our knowledge, in veterinary epidemiology applications, spatial distances between places are generally calculated within the context of the Euclidean space. A Euclidean distance represents a direct line between two points and it is essentially the square root of the squared differences between corresponding elements of the rows (or columns) of the matrix.

In this preliminary analysis, we aim to introduce other types of pairwise distances between places applicable to the veterinary epidemiology domain. The objective is to calculate the distances based on orography, road networks, land use or a combination of these real-world characteristics.

Considering other types of elements in distance calculation, beyond the length of the line segment between two points, aims to incorporate other factors that may influence disease occurrence. Namely: orography to convey terrain slope effect due to the presence of relief, road networks to include human interaction, Haversine to obtain a more accurate measure for long distances, and land use information to incorporate the concept of friction distance (i.e. friction values are assigned based on abiotic factors that facilitate or hinder the pathogen movement).

To evaluate the contribution of orography, road networks, and land use factors in the distance calculation, to determine if significant differences exist among the identified types of distance, and to assess the impact of using a distance other than Euclidean for cluster identification, a research project has been defined.

The project is structured in three phases:

1. Development of a tool to calculate the Euclidean, Haversine, route/road network, orographic/elevation and cost/friction distances;

2. Creation of descriptive statistics, such as concordance statistic tests, and application of hotspot analysis (e.g. K-function) using the calculated distances with a series of epidemiological data;

3. Perform of spatial cluster analysis on a data set describing the occurrence of Tick Borne Encephalitis virus (TBEV) in goat's raw milk.

This study primarily focuses on the results of the first research phase: the development of the tool to calculate the six different distances.

2 Material and methods

The system comprises a series of Python® scripts, each designed for a specific type of distance calculation. These scripts are made accessible to users through a web application developed in pure PHP (Lerdorf and Tatroe, 2002) for the backend, and HTML, CSS, and JS (Flannagan, 1997) for the frontend.

The Python scripts (Van Rossum et al., 1995) use various libraries and models to compute distances, including:

- For computing Euclidean distances: numpy and pyproj libraries (Harris et al., 2020) are employed;
- For computing Haversine distances: numpy and math libraries are used;
- For computing route distances, either with or without elevation: the numpy library is used; alongside Graphhopper is used with an OpenStreetMap graph of the Veneto and Lombardy regions. Additionally, the request library is used to fetch distances among all pairs of points in the dataset;
- For computing elevation distances with DEM, libraries such as numpy, pyproj and rasterio are employed. The DEM model of the terrain utilized is the Tinitaly dataset from Istituto Nazionale di Geofisica e Vulcanologia (Tarquini et al., 2023) (INGV); specifically, the tiles referring to the Veneto and Lombardy regions, which have been merged together;
- For computing the friction elements distances, libraries such as numpy, geopandas and shapely are used. The Corine Land Cover (Cover, 2018) dataset from the Copernicus Program is utilized, with the vector dataset being employed.

The subsequent discussion will provide detailed descriptions about the system architecture and the data flow. The core of the proposed system consists of the aforementioned Python scripts, supplemented also by other software components to coordinate process activities, and ensure the system stability and robustness. As shown in Figure 1 the *Distance Evaluation Tool* server features a web interface exposed on port 80, enabling remote clients to access it. Upon a user uploading an ESRI shapefile containing points, the following actions occur sequentially:

1. The backend logic of the system adds the new points on the PostGIS database, replacing all previously stored points;
2. Following this operation, the backend logic invokes a bash script, responsible for sequentially executing all six previously described Python scripts, acting as a dispatcher;
3. Each script aims to produce an output matrix containing the respective type of distances for each pair of points. The matrix is then serialized in both CSV and JSON (Pezoa et al., 2016) formats;
4. All scripts fetch points from the PostGIS database where the backend logic previously uploaded the points;
5. The two scripts responsible for computing route distances utilized the Graphhopper service running on the same server as a daemon process. These scripts employ internal HTTP calls through the API offered by Graphhopper to fetch the distances corresponding to pairs of points;
6. Upon completion of a script execution, the output matrix in both formats is stored in a folder within the *htdocs* directory of the web server and made available to the user through the web application.

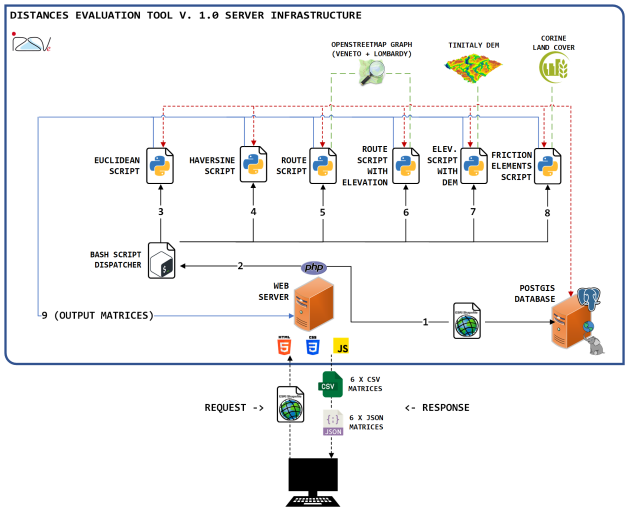

Figure 1. Distance Evaluation Tool System Architecture

Furthermore, it's worth mentioning that all scripts use libraries such as psycopg2 to fetch uploaded points from the PostgreSQL database, CSV and JSON-related libraries for serializing the matrices into files, and, for the last two scripts libraries enabling concurrent and parallel computing due to the substantial computational resources required.

3 Results

Users accessing the application utilize its web interface, which resembles the one depicted in Figure 2.

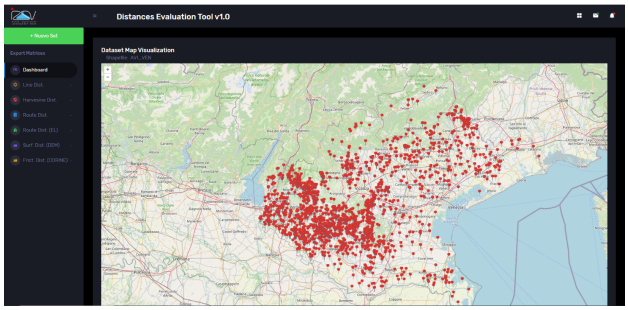

Figure 2. Web Interface of the Distance Evaluation Tool system

By clicking on "+ Nuovo Set" users can upload a new ESRI® shapefile, requiring both the .shp and .shx files to complete the operation. Points within the shapefile are then stored on the server within a PostGIS database table (PostGIS, 2013).

A map, displayed in the home section allows users to quickly visualize the uploaded points, facilitating verification of whether they belong to the dataset intended for processing.

Upon completion of the necessary computation time, the server generates two output files containing serializations in CSV format (as illustrated in Fig. 3) and JSON format for each of the six distinct Python scripts.

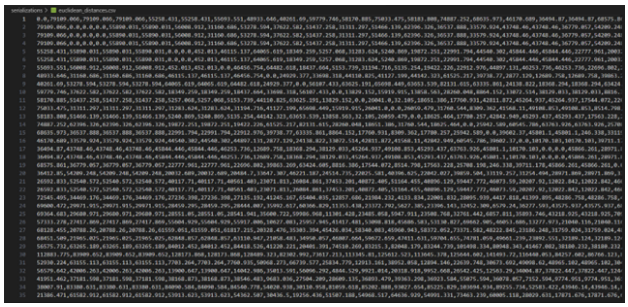

Figure 3. Example of a CSV matrix produced by the system

Finally, users can download the serializations by accessing the sidebar sub-menus corresponding to the distances they require.

4 Discussion and conclusion

We have successfully developed a system capable of computing various types of distances among all points within given datasets. As the system's purpose is to compute matrices, there exists a lower bound in terms of temporal performances in the order of $O(n^2)$, where n represents the number of instances in the dataset.

In nearly every script, we were able to halve the computing time required since the matrices are symmetric, meaning the distance from point A to point B is the same as from point B to point A, by assigning the same computed distance both in `dist_matrix[a,b]` and in `dist_matrix[b,a]`.

For line distances, Haversine distances, route distances, either with or without elevation, the cost is $\Theta\left(\frac{n^2}{2}\right)$. However, the real cost of a route distance computation is higher due to the involvement of additional processes such as http calls to service like Graphhopper.

For elevation distances with the DEM, the cost is $\Theta\left(\frac{n^2}{2} \cdot \frac{d}{10}\right)$, where d is the average Euclidean distance among all pairs of points. This computation is conducted over a 10-meters precise raster. The process involves sampling altitudes every 10 meters and calculating the overall distance by summing all segments that connects the three dimensional points.

For the friction elements distances, the cost appears to be $\Theta\left(\frac{n^2}{2} \cdot k\right)$, where k is the number of different vector features in the model, as the process involves analyzing how much each pair of points overlaps certain types of vector features.

The system's performance was tested on a dataset of 2050 points representing poultry farms located in the Veneto region. Table 1 displays the computational timings measured and estimated for two different hardware environments:

- **VM Server**: 4 GB of RAM, 2 physical cores of an Intel® XEON CPU, 80 GB HDD;
- **Lenovo® Thinkpad P15v**: 32 GB of RAM, Intel® Core i9-12900H 20 cores, 512 GB SSD M2 Micron® MTFDKBA512TFK.

Table 1. Computation timings (measurements and *estimates*)

Script	VM Server	Lenovo® Thinkpad
Euclidean	30 secs	3 secs
Haversine	35 secs	5 secs
Route	4 hrs	25 mins
Route with elevation	4 hrs	25 mins
Elevation with DEM	*10 hrs*	45 mins
Friction elements CLC *	*5 days*	12 hrs

* NB: Timing costs for friction element (CLC level no. 5, water) need to be multiplied by 5 to compute all the 5 levels, increasing them to 25 days or 60 hours, based on the system used.

As expected, both measurements and estimates are significantly lower in the second environment compared to the first one.

Clustering disease events in space and time can provide valuable insights into the disease process, aiding the development of disease control and prevention programmes. Previous research has underscored the significance of

proximity in disease transmission risk (Mulatti et al., 2010). Spatial models incorporating cost distance analysis are increasingly employed in assessing large mammal habitats or planning conservation and efforts (Bunn et al., 2000), (Anderson et al., 2022). However, there is a need for new tools to support the scaling up of interventions and enhance national surveillance capacity. Models incorporating different types of distances, as demonstrated in this study, can be integrated into an ensemble forecast framework, which considers all available information in predictions. This approach avoids the constraints and dependencies associated with single statistical or machine learning methods or limited data sources.

The development of a distances computation tool has numerous application in veterinary epidemiology:

1. **Animal Movements and Disease Spread**: modelling animal movements within a landscape, considering terrain, land use, and barriers, is crucial for designing effective disease control strategies.

2. **Vector-Borne Diseases**: identifying high-risk areas based on vector habitat suitability, proximity to water bodies, and landscape connectivity enables targeted intervention such as vector control measures.

3. **Disease Surveillance and Monitoring**: identifying key locations or pathways where animals come into contact, allows prioritization of surveillance efforts.

4. **Resource Allocation and Emergency Response**: during disease outbreaks or emergencies, like avian influenza, distance analysis aids in resource allocation and emergency planning by identifying critical pathways for disease spread. This facilitates implementing targeted control measures while minimizing economic and social disruptions.

5. **One Health Approaches**: integration of data from veterinary epidemiology, ecology, landscape ecology, and human health promotes interdisciplinary collaborations. Considering the interconnectedness of human, animal, and environmental health, enables a comprehensive understanding of disease dynamics and the development of holistic prevention and control strategies.

The research activities were conducted in the framework of a project financed by the Italian Health Ministry RC IZSVE 05/21 titled *"Indagine sulla prevalenza e distribuzione di patogeni trasmessi da zecche in allevamenti caprini e valutazione del rischio legato alla presenza del virus della Tick Borne Encephalitis (TBEV) nel latte crudo e in prodotti derivati. IXORISK"*.

References

Anderson, K., Cahn, M. L., Stephenson, T. R., Few, A. P., Hatfield, B. E., German, D. W., Weissman, J., and Croft, B.: Cost distance models to predict contact between bighorn sheep and domestic sheep, Wildlife Society Bulletin, 46, e1329, 2022.

Bunn, A. G., Urban, D. L., and Keitt, T. H.: Landscape connectivity: a conservation application of graph theory, Journal of environmental management, 59, 265–278, 2000.

Cover, C. L.: Copernicus Land Monitoring Service, European Environment Agency (EEA), https://doi.org/10.2909/71c95a07-e296-44fc-b22b-415f42acfdf0, 2018.

Flannagan, D.: JavaScript: The definitive guide, 1997.

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., et al.: Array programming with NumPy, Nature, 585, 357–362, 2020.

Lerdorf, R. and Tatroe, K.: Programming Php, " O'Reilly Media, Inc.", 2002.

Mulatti, P., Kitron, U., Jacquez, G. M., Mannelli, A., and Marangon, S.: Evaluation of the risk of neighbourhood infection of H7N1 Highly Pathogenic Avian Influenza in Italy using Q statistic, Preventive veterinary medicine, 95, 267–274, 2010.

Pezoa, F., Reutter, J. L., Suarez, F., Ugarte, M., and Vrgoč, D.: Foundations of JSON schema, in: Proceedings of the 25th international conference on World Wide Web, pp. 263–273, 2016.

PostGIS, P.: Spatial and Geographic objects for PostgreSQL, 2013.

Tarquini, S., Isola, I., Favalli, M., and Battistini, A.: TINITALY, a digital elevation model of Italy with a 10 meters cell size (Version 1.1)., Istituto Nazionale di Geofisica e Vulcanologia (INGV), https://doi.org/10.13127/tinitaly/1.1, 2023.

Van Rossum, G., Drake, F. L., et al.: Python reference manual, vol. 111, Centrum voor Wiskunde en Informatica Amsterdam, 1995.