An index-based system for early alerts of potential zoonotic disease outbreaks

Jaqueline S. Angelo[1*], Livia Abdala[1], Douglas A. Augusto[1], Eduardo Krempser[1], Marcia Chame[1],

1 Oswaldo Cruz Foundation (Fiocruz), Rio de Janeiro, RJ, Brazil

* Corresponding author
Email: jaqueline.angelo@fiocruz.br

Abstract

The Information System on Wildlife Health (SISS-Geo) is a free platform designed for the real-time collection of georeferenced data on wildlife health and environmental conditions via mobile devices. It serves as a collaborative tool, enabling health professionals, researchers, environmental managers, and the general public to report information about wildlife health occurrences directly within the system. Since its launch in 2014, SISS-Geo has been successfully applied in supporting decision-making during significant wildlife health events in Brazil. In this paper, we introduce a dynamic alert system based on a Multi-Attribute Zoonotic Alert Index (Z-Alert) to enhance disease outbreak detection and response in wildlife. The proposed approach improves the current alert system of the SISS-Geo platform by integrating clustering techniques with multi-objective optimization. The clustering stage allows to group SISS-Geo records in a way that reflects both spatial and temporal variations, capturing relevant patterns for epidemiological surveillance. Each identified cluster is assigned a numerical alert index, the Z-Alert index, reflecting its criticality level, based on optimally weighted attributes such as the percentage of dead animals, temporal interval, and geographical spread. The system offers customization alert thresholds, allowing health managers to adapt the alert system to their specific needs. To validate the model, we used historical yellow fever case data from Brazil's Ministry of Health, achieving over 94% accuracy in identifying critical clusters. By enabling the prioritization of prevention and investigation actions, the system strengthens the responsiveness of health managers, optimizing resource allocation and promoting smart economic strategies for outbreak containment.

Introduction

It is widely recognized that most emerging zoonotic diseases—those naturally transmissible from animals to humans—originate in wildlife and are driven by human activities like land-use changes, urbanization, and globalization. A recent review highlights that approximately 75% of all emerging infectious diseases are of zoonotic origin [1]. The effective surveillance and disease prevention requires a collaborative, multidisciplinary approach to understand disease ecology and conduct risk assessments. In this context, wildlife monitoring becomes essential for various sectors of society, especially for public health. Collecting and analyzing data on wildlife populations can provide valuable insights to guide preventive actions and control zoonotic disease transmission. The One Health approach is particularly relevant in this context, as it recognizes the interconnection between human, animal, and environmental health [2, 3]. This integrated perspective is especially important in a country like Brazil, where

environmental and epidemiological complexities require comprehensive strategies for disease surveillance systems [4].

The effective implementation of One Health principles requires innovative technologies that facilitate real-time, high-quality data collection and support decision-making through efficient information analysis and management. To address this need, the Brazilian Wildlife Health Information System, SISS-Geo (*Sistema de Informação em Saúde Silvestre Georreferenciado (in Portuguese)* [5, 6] was created in 2014 by the Institutional Platform for Biodiversity and Wildlife Health at Oswaldo Cruz Foundation (Fiocruz) to support collaborative monitoring.

SISS-Geo (avaliable at `https://www.biodiversidade.ciss.fiocruz.br` or `https://sissgeo.lncc.br`) is a digital platform that enables the real-time collection of georeferenced observations on wildlife and environmental conditions via mobile devices, with active public participation. It integrates the registration of occurrences, expert diagnoses, and computational tools for data analysis, prediction, visualization, and real-time alerts for wildlife-related health events. Since its launch, SISS-Geo has proven to be a valuable resource for supporting decision-making during major wildlife health events in Brazil, such as Yellow Fever (YF) outbreaks [7–9], by accelerating information dissemination, providing accurate geographic locations, and simplifying response efforts.

The One Health Joint Plan of Action (2022–2026) highlights the SISS-Geo as an example of an integrated surveillance system that supports early detection and response to zoonotic threats through multisectoral collaboration [10]. This positions SISS-Geo in alignment with the global call to implement the One Health Early Warning and Response System (OH-EWRS) approach for addressing emerging zoonotic diseases [11].

In Brazil, the YF virus is endemic to the Amazon region, eventually reemerging into epidemic waves [8]. While currently confined to South and Central America and parts of Africa, its potential for geographic expansion, raises concerns for public health and cross-border transmission [12]. This underscores the urgent need to develop early warning and response systems capable of detecting outbreaks in their initial stages and enabling timely, coordinated interventions. By enabling the real-time collection of high-quality data, the platform provides critical information to government agencies, the health sector, and society at large—even in some of the most remote regions of the country—thereby strengthening surveillance and response efforts across multiple levels.

One of the most strategic components of SISS-Geo's features is the alert system, which notifies health managers about reported cases. Currently, the system follows fixed monitoring rules. For instance, if a non-human primate (NHP) death or illness is reported, an email is automatically sent to health managers registered on the platform along with information about the record. While this enables managers to make timely decisions regarding the actions to be taken, it is unable to detect levels of risk, as fixed rules fail to account for regional variability and complexity. Without an appropriate prioritization mechanism, it becomes more challenging to direct efforts to the most impactful cases, compromising the efficiency of investigation and intervention measures.

The identification and prioritization of actions in critical health regions are essential for the effective allocation of resources in disease prevention and control efforts. Such an approach prevents the waste of supplies and personnel in low-risk areas, focusing efforts where there is the greatest need. Timely detection of diseases through surveillance, combined with effective implementation of targeted interventions, can significantly reduce the scale, severity, and economic impact of responses to outbreaks [13].

In this study, we aim to leverage data from the SISS-Geo to investigate the potential of animal-related information as an early warning signal for disease outbreaks that may later affect humans. Our focus is on NHPs, which play a key role as sentinels for YF circulation. However, the proposed methodology is not limited to the yellow fever context, as the SISS-Geo system also contains records related to other infectious

diseases, such as rabies and avian influenza. By analyzing spatial and temporal patterns of NHP cases, we seek to identify regions that exhibit early signs of viral activity. Using NHPs as indicators of environmental health risks enables earlier detection and supports timely public health responses before human infections are reported.

As part of the efforts to improve surveillance strategies, we propose an index-based alert system that incorporates regional information to accurately reflect the criticality of events. By creating a Multi-Attribute Zoonotic Alert Index (named Z-Alert), we aim to identify not only individual records but also regions with potential risks to human and animal health. The Z-Alert proposal is part of a broader effort to strengthen national surveillance strategies, especially following the expansion of YF into extra-Amazonian regions starting in 2014, which highlighted the need to enhance wildlife health surveillance in Brazil. In this context, tools such as SISS-Geo and the environmental suitability models developed by the Institutional Platform for Biodiversity and Wildlife Health (Pibss), in partnership with the Yellow Fever Modeling Group (GruMFA), have become central elements for anticipating risk areas, guiding vaccination efforts, and supporting the definition of ecological transmission corridors. Both have been officially incorporated into national response strategies, including epidemiological bulletins, technical notes [15–17] issued by the General Coordination for Arbovirus Surveillance of the Department of Health Surveillance and Environment of the Ministry of Health (CGARB/SVSA/MS), and the Contingency Plan for Public Health Emergencies – Yellow Fever (2nd edition) [18, 19].

Unlike the current rule-based model, the proposed system generates alerts based on clustered data analysis, considering multiple observations recorded in SISS-Geo. By integrating attributes derived from the platform's georeferenced records, such as the number and proportion of dead animals, event duration, and geographic extent, the system identifies areas of greater concern. We combine clustering techniques—to capture spatial and temporal disease occurrence patterns—with an index-based alert system that measures alert severity. Such an approach enhances the capacity for early detection and response within the context of zoonotic disease surveillance, representing an evolution of the current rule-based notification scheme of SISS-Geo. With the aid of a data clustering technique and a multi-objective optimization process, the zoonotic alert system will integrate multiple attributes to estimate the health risk level associated with an event. We also propose customized alert thresholds that dynamically adjust the number of alerts/notifications generated by the system, ensuring that only the most relevant situations are reported to decision-makers.

In this paper, we aim to:

- establish an index-based system that reflects a level of risk and provides a more refined assessment of potential health risks. The alert system is capable of distinguishing varying levels of risk by incorporating multiple regional attributes.

- allow health managers to customize the alert system, enabling them to prioritize alerts and optimize investigation and intervention efforts according to their specific needs and realities. In a vast country like Brazil, where environmental, epidemiological, and logistical conditions vary widely, such flexibility is essential for efficient resource allocation.

- utilize clustering methods to capture spatial and temporal patterns of disease occurrence, enabling a more comprehensive analysis of outbreak dynamics.

- apply a multi-objective optimization process to balance key criteria, ensuring that alerts are accurate and minimize false negatives to prevent the non-identification of critical events while reducing false positives to avoid overloading verification systems.

- finally, but not limited to, strengthen the integration of human, animal, and environmental health data to improve zoonotic disease surveillance and prevention.

The system's effectiveness relies heavily on the accuracy of its prediction model in correctly identifying both true positives (alerts) and true negatives (non-alerts). Missing an alert condition (false negative) can have serious consequences for wildlife, the environment, and human health. Conversely, false positives could overwhelm the limited network of laboratories and experts responsible for verifying alerts. Thus, finding the right balance between classifying alerts and non-alerts is crucial to ensuring that the system effectively prioritizes critical regions without overloading resources. In addition, the customization of alert thresholds plays a key role in this trade-off, as stricter limits reduce false positives but increase the risk of missing real cases (less sensitivity), while more relaxed limits enhance detection at the cost of more false alerts (more sensitivity).

The Zoonotic Alert System

The methodology for generating the Z-Alert index integrates data management, a clustering technique, and multi-objective optimization. The proposed index-based alert system utilizes regional information to quantify the criticality of an event. Our clustering approach aims to identify broader spatiotemporal patterns of group of data points (a cluster). In this context, an event is defined as a cluster representing a phenomenon, based on occurrence patterns in both space and time. For each cluster formed, a numerical value is assigned, the Z-Alert index, reflecting its criticality, in which high values indicate high-risk regions. Fig 1 illustrates the main steps in the development of the zoonotic alert system.

Fig 1. The zoonotic alert system. Main steps of the zoonotic alert system.

It is important to highlight that, although the national surveillance system recommends investigating a single dead or illness NHP, the alert index does not replace this approach—it complements it. While isolated cases are critical for early detection, clusters of NHP deaths often indicate that transmission is already underway. By clustering such occurrences and quantifying their attributes through the alert index, it becomes possible to prioritize areas where the risk is escalating, even when individual reports may seem sparse or dispersed.

Initially, data obtained from SISS-Geo are preprocessed according to the analyses of interest (section Data preprocessing). To extract regional information, a clustering technique will be applied, allowing similar instances to be grouped and organized into clusters (section Clustering data). After clustering, new attributes will be extracted from the records to characterize the clusters in greater detail (section Cluster characterization). In the optimization step (section Multiobjective optimization problem (MOP)), a multi-objective optimization process will be defined, parameterized by the cluster attributes, with the goal of obtaining an alert index that can effectively differentiate between critical and non-critical clusters. The optimization process will assign weights to each attribute, determining its relevance in calculating the Z-Alert index. Finally, the index is assigned to each cluster (section The Z-Alert Index), classifying them according to their level of criticality. Health sector managers will be able to define a threshold level for receiving alerts, which can be adjusted according to their needs and preferences. Clusters identified as critical will be highlighted and an automatic email will be sent to managers, informing them of the regions that require priority investigation.

A key advantage of the alert system proposed is its ability to provide a real-time risk signal, even in the absence of laboratory confirmation of circulating pathogens among

animals. By leveraging spatio-temporal patterns extracted from the data, the index enables early identification of clusters with a high probability of being critical. This proactive approach offers an important decision-support tool for public health management, allowing preventive actions to be taken before zoonotic pathogens are confirmed, thus potentially reducing the time between emergence and response.

Data preprocessing

The first step is the preprocessing of SISS-Geo's data. The data collected are described in Table 1.

Table 1. Data collected from SISS-Geo's platform.

Field	Descriptor field
Record	Identifier, Longitude, Latitude, Observation data (ISO 8601), State, Municipality, Source of location and Precision.
Record status	"Verified by a specialist", "audited" and "valid".
Animal	Identifier, Type, Observed quantity, Situation and Behavior.
Final classification	Disease and Classification.

In the preprocessing stage, only records obtained via GPS or with locations explicitly provided by the user were selected, on top of having locations errors within 100m to ensure the spatial reliability of the data used for cluster formation. After data collection and filtering, each record was geocoded to its corresponding municipality using latitude and longitude coordinates, based on the official dataset provided by the Brazilian Institute of Geography and Statistics (IBGE) (available at `https://geoftp.ibge.gov.br/organizacao_do_territorio/malhas_territoriais/malhas_municipais/municipio_2022/Brasil/BR/BR_Municipios_2022.zip`). This information is necessary for validating the results presented in section Validation of the Z-Alert Index.

Clustering data

Clustering techniques are designed to identify instances or groups of data that belong to the same phenomenon. The idea is to separate data with similar characteristics and assign them to a group or cluster. Given the dynamic nature of the phenomenon of interest, such as the occurrence of disease outbreaks—including YF—these events exhibit spatial and temporal movement patterns [9]. In this context, clustering enables the identification of groups of records that share these dynamic characteristics, supporting a more refined analysis of disease spread.

Several clustering algorithms can be found in the literature, the most traditional of which are available in the Scikit-learn open-source machine learning library. Fig 2 (sourced from `https://scikit-learn.org/stable/modules/clustering.html#clustering`) presents a comparative analysis of many of these methods applied to synthetic datasets with distinct cluster shapes. The patterns relevant to this study resemble those in the second and fourth rows of the figure, which depict curved clusters and clusters distributed along narrow bands. Among the evaluated methods, Spectral Clustering, DBSCAN, HDBSCAN, and OPTICS produced results most consistent with the expected cluster structure.

From these, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [20, 21] was selected due to its ability to identify clusters of arbitrary shapes, its robustness to noise and outliers–data points that significantly deviate from the overall

distribution–and its minimal parameter requirements. Additionally, unlike many clustering methods, DBSCAN does not require prior specification of the number of clusters, which is particularly advantageous in exploratory analyses where the underlying structure is unknown.

Fig 2. Clustering techniques. Comparative analyses of clusters techniques from Scikit-learn library.

The DBSCAN algorithm

The algorithm work by grouping points based on the density of neighbors within a specified radius (*eps*). It starts by selecting an unvisited point and checking whether it is a "core point". If it is a core point, a new cluster is created, and all points within the *eps* radius are added to it. The process continues by expanding the cluster through neighboring points, checking their neighbors recursively until no further points satisfy the criteria defined by *eps* and the minimum number of points required to form a cluster. The algorithm then selects another unvisited point and repeats the process until all points have been visited. The parameters used in DBSCAN for the clustering step were set as follows:

eps = 500. Defines the neighborhood radius around a point, where two points are considered neighbors if the distance between them is less than or equal to *eps*. This value directly influences the size and density of the clusters.

min_samples = 1. Represents the minimum number of points required within the *eps* radius for a point to be considered a core point and capable of expanding a cluster.

metric = 'precomputed'. Specifies the distance metric used to calculate the proximity between points. In the experiments, instead of computing distances with a specific metric, the method directly receives a (square) matrix of precomputed distances between the points.

The choice of a high *eps* is justified by the need to generate clusters that follow the temporal and spatial definitions given by the precomputed distance matrix. Meanwhile, the minimum number of points required to form a cluster was set to 1 due to the nature of the SISS-Geo records, where isolated geographical points can be observed.

The distance matrix assigned to DBSCAN was generated by combining two matrices: (i) a spatial distance matrix D_S, which considers the geodesic distance between points. The geodesic distance is calculated in a three-dimensional spherical space as the shortest path along the curved surface of the planet. This distance is computed based on the latitude and longitude of the points, and (ii) a temporal distance matrix D_T, which indicates the time in days between records. Note that both D_S and D_T are square matrices, which is a prerequisite of the *metric* parameter.

The spatial and temporal constraints are considered when computing both matrices, with the temporal and spatial limits configured according to the experiments conducted (see section Results and Discussion). The matrices D_S and D_T are computed as follows. Let $D_S(i,j)$ be the geodesic distance between points i and j. If $D_S(i,j) \geq L_{D_S}$, where L_{D_S} is the maximum spatial limit allowed for clustering points, a high value is assigned to indicate that these points cannot be grouped. Otherwise, $D_S(i,j)$ is assigned the actual distance between i and j. Similarly, let $D_T(i,j)$ be the temporal distance between instances i and j. If $D_T(i,j) \geq L_{D_T}$, where L_{D_T} is the maximum temporal limit allowed for clustering, a high value is assigned to indicate that these instances

cannot be grouped. Otherwise, $D_T(i,j)$ is assigned the time interval (in days) between i and j.

Since D_S and D_T have different scales (space versus time), both were normalized (values were scaled in the interval $[0,1]$) to prevent large values from distorting the clustering process. The limits L_{D_T} and L_{D_S} are defined by the user according to the experiments performed. Finally, the total distance matrix D_{total} used as input for DBSCAN is computed as

$$D_{total} = D_S + D_T$$

As a result of the clustering process, we obtain the clusters to which each record belongs, respecting the predefined temporal and spatial limits. It is important to highlight that the temporal and spatial limits are established for pairs of points, meaning that the generated cluster is not necessarily restricted to these limits.

Cluster characterization

After the clustering step, new information is assigned to each cluster to support the construction of the Z-Alert index. The newly extracted attributes are presented in Table 2.

Table 2. New attributes extracted from clustering.

Attribute	Description
n_death	Number of dead animals.
n_animal	Total number of animals involved; sum of live, sick and dead animals.
ini_date	Cluster start date, corresponding to the oldest record.
end_date	Cluster end date, corresponding to the most recent record.
interval	Time interval (in days); difference between the dates of the earliest and most recent records.
n_rec	Number of records.
freq_rec	Frequency of records; number of records divided by *interval*.
freq_death	Frequency of dead animals; number of dead animals divided by *interval*.
freq_animal	Frequency of total animals; number of animals involved divided by *interval*.
perc_death	Percentage of dead animals; number of dead animals divided by *n_animal*.
extention	Maximum distance between all points in the cluster.
n_conf	Number of animals confirmed to have a disease of interest (e.g. YF).
geocode	Cluster geocoding.

The cluster geocode is determined based on the geocodes of the records it comprises. If a cluster contains records from different municipalities (i.e., with distinct geocodes), it is duplicated in the database to ensure that each entry maintains the correct association with its respective municipality. This information is only necessary to validate the results (see in section Validation of the Z-Alert Index).

Multiobjective optimization problem (MOP)

Optimization refers to the study of problems in which the goal is to minimize or maximize a cost function, also known as the objective function, by selecting values for the function's variables within a feasible set. In other words, among all possible solutions in the search space, we seek the one that best meets the desired criteria. In a single-objective optimization problem, only one function of interest is optimized. In this study, however, we aim to optimize multiple objective functions, mathematically formulated as [22]

$$\begin{array}{ll} \text{Minimize} & F(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_k(\mathbf{x})] \\ \text{subject to} & \mathbf{x} = (x_1, x_2, ..., x_n) \in \mathcal{S} \end{array} \tag{1}$$

where $k(\geq 2)$ is the number of objectives to be optimized, $\mathcal{S}$ is called the feasible set, F is the vector of objective (function) values with $f_i : \mathbb{R}^n \to \mathbb{R}$, and $\mathbf{x} \in \mathbb{R}^n$ is the vector of decision variables. If $\mathbf{x} \in \mathcal{S}$, then $\mathbf{x}$ is a feasible solution; otherwise, it is infeasible. The image of the feasible set, denoted by $\mathcal{Z}(= F(\mathcal{S}))$, is known as the feasible objective set. The elements of $\mathcal{Z}$ are the objective functions with $\mathbf{z} = (z_1, z_2, ..., z_k)^T$, where $z_i = f_i(\mathbf{x})$ for all $i = 1, ..., k$ are objective values. If an objective function f_i is to be maximized, it is equivalent to minimize the function $-f_i$.

Due to the conflicting nature of the objective functions, which may also be in different units, it is not possible to find a single solution that is optimal for all objectives simultaneously. Instead, in MOP the aim is to identify a set of solutions, known as efficient or compromise solutions, that achieve the best possible trade-off among all objectives. This set of solutions is called the *Pareto-optimal solutions set*.

Assuming a problem in which all objectives should be minimized, a solution $\mathbf{x} \in \mathcal{S}$ dominates another solution $\mathbf{x}' \in \mathcal{S}$ (written as $\mathbf{x} \prec \mathbf{x}'$), when

$$f_i(\mathbf{x}) \leq f_i(\mathbf{x}') \quad \forall i \in \{1, ..., k\} \quad \text{and} \quad \exists j \in \{1, ..., k\} : f_j(\mathbf{x}) < f_j(\mathbf{x}') \tag{2}$$

i.e., the solution $\mathbf{x}$ is no worse than $\mathbf{x}'$ in all objectives and better in at least one. Thus, all possible pairwise solutions can be compared in order to find which solutions are non-dominated. Finally, the set of non-dominated solutions is called the *Pareto-optimal set* and the corresponding image in the objective space defines the *Pareto-optimal Front*, representing the best collection of solutions found for the problem.

To obtain a representative set of solutions, in multi-objective optimization two orthogonal goals are desired: (i) find a set of solutions as close as possible to the Pareto-optimal front, and (ii) find a set of solutions as diverse as possible along the Pareto-optimal region [23]. Fig 3 illustrates a hypothetical Pareto front where objective f_1 and f_2 must be minimized. Converging to a set of solutions that are close to the true optimal and sparsely spaced in the Pareto-optimal region are the two main goals of MOPs.

Fig 3. Two main goals of MOPs. Desired aspects of a Pareto front.

Solution method

The simplest technique for solving MOPs involves aggregating the objectives using a weighted sum, transforming the MOP into a single-objective one. For each function f_i, a weight w_i is assigned such that $0 \leq w_i \leq 1$ for all $i = 1, ..., k$, with $\sum_{i=1}^{k} w_i = 1$. The transformed MOP is formulated as follows:

$$\begin{array}{ll} \text{Minimize} & \sum_{i=1}^{k} w_i f_i(\mathbf{x}) \\ \text{subject to} & \mathbf{x} = (x_1, x_2, ..., x_n) \in \mathcal{S} \end{array} \tag{3}$$

If all weight coefficients are positive or if the problem has a unique solution, then every solution found in (3) will be a Pareto-optimal solution [22]. The main advantage of this technique lies in its computational simplicity. However, obtaining the full set of Pareto-optimal solutions requires solving the optimization problem multiple times with different weight configurations. Once the Pareto set is obtained, the solutions are presented to the decision-maker, who selects the one that best aligns with their preferences. This approach was used to optimize the attribute weights that form the proposed Z-Alert index (in section The Z-Alert Index).

Optimizing cluster attribute weights

The alert system currently implemented in SISS-Geo automatically generates an email whenever a dead or sick NHP is recorded. Since we aim to create an alert that

anticipates the identification of phenomena of interest (e.g., a disease outbreak), only clusters containing more than one record were considered for the optimization process. This decision is motivated by the nature of the data registered in SISS-Geo, where most of the generated clusters—approximately 81% (see section Results and Discussion)—contain only a single record (*unit clusters*). If all of these clusters were included, the Z-Alert index would be biased toward unit clusters, which provide little regional information about the analyzed attributes compared to clusters with multiple records.

The goal of the Z-Alert index is to generate meaningful insights from data groups rather than individual records, enabling an accurate determination of a cluster's criticality in relation to potential phenomena of interest. Thus, in the optimization phase, only clusters with more than one record were considered, and among these, only those containing confirmed cases of a disease of interest were selected. In the analyses conducted in section Analyzing the Pareto Front, we considered YF as the disease of interest. Clusters with confirmed cases were classified as *critical clusters*, while clusters without confirmed disease cases were classified as *non-critical clusters*. The formulated MOP aims to optimize two objectives simultaneously:

(i) The mean of the critical clusters, which needs to be maximized, and

(ii) The variance of the critical clusters, which needs to be minimized.

The idea is to maximize a function that represents the combination of these two objectives. This ensures that the alert index effectively distinguishes critical clusters from non-critical ones while maintaining uniformity among the critical clusters. The objective function that we aim to maximize for optimizing the clusters' attribute weights is

$$F = w \cdot f_1(Z_Alert, n_death) - (1 - w) \cdot f_2(Z_Alert) \tag{4}$$

where Z_Alert (defined in the next section) are the indices of critical clusters, $f_1(Z_Alert, n_death)$ represents the mean of the indices of critical clusters, weighted by the number of dead animals (n_death), and $f_2(Z_Alert)$ represents the variance of the indices of critical clusters. The parameter $w \geq 0$ controls the importance of maximizing f_1, while $(1 - w)$ controls the importance of minimizing f_2, ensuring that $w + (1 - w) = 1$. Objectives f_1 and f_2 are computed as

$$f_1(Z_Alert, n_death) = \frac{\sum_{i=1}^{nc} Z_Alert[i] \cdot n_death[i]}{\sum_{i=1}^{nc} n_death[i]} \tag{5}$$

$$f_2(Z_Alert) = \frac{1}{nc} \sum_{i=1}^{nc} (Z_Alert[i] - \mu)^2 \tag{6}$$

where nc is the total number of critical clusters and μ is the mean of Z_Alert of critical clusters. By maximizing the function (4), we encourage the attribute weights to adjust the index according to both combined objectives.

The Z-Alert Index

For the construction of the alert index, let M be the attribute matrix of the clusters, composed of the columns

$$M = [n_rec, interval, freq_n_rec, freq_a_animal, extention, perc_death, n_death]$$

and let $p \in \mathbb{R}^m$, where m is the number of variables, be the corresponding weight vector, which characterizes the importance of each attribute in computing the index. Due to

the differences in scale among the variables, a data standardization process was applied to ensure that all variables contribute equally to the optimization step. The adopted standardization is based on the *Z-score* normalization, which transforms the data into a distribution with a mean zero and a standard deviation of one, calculated as

$$Z = \frac{X - \mu}{\sigma} \tag{7}$$

where Z is the normalized value, X is the original value of the variable, and μ and σ are the mean and the standard deviation of the variable, respectively.

Finally, for each cluster $i = 1, ..., n$, where n is the total number of clusters, the Z_Alert index is calculated as

$$Z_Alert[i] = \sum_{j=1}^{m} M[i, j] \cdot p[j] \tag{8}$$

where $p[j]$ is the weight associated to variable j, and $M \in \mathbb{R}^{n \times m}$ is the matrix in which column j represents the value of the j-th variable and line i represents the i-th cluster.

At this stage, each cluster will receive a Z-Alert index, reflecting its level of criticality. Clusters with low index values are considered lower risk, while those with high index values indicate greater risk. However, the interpretation of this level of criticality may vary depending on the reality of each region, under the management of health departments. To accommodate this flexibility, the health manager can define a threshold level for receiving alerts, called *threshold_level*, which can be adjusted according to their needs and preferences.

The Threshold Level

The threshold level establishes a threshold for cluster criticality, directly influencing the number of alerts/notifications generated and ensuring that only the most relevant situations are reported to the manager. Considering that NHP mortality records depend on human analyses and may be subject to under-reporting, the alert index is designed to mitigate this by the aggregation of multiple attributes to provide a more robust signal. By calibrating the threshold appropriately, the system reduces the likelihood of overreacting to low-risk events while still capturing emerging patterns that may not be apparent from individual reports alone.

To define this threshold in an objective and adaptable manner, we use the distribution of the Z-Alert indices and perform the cutoff based on quartiles. Quartiles are statistical measures that divide an ordered set into four equal parts, each containing 25% of the observations. In the proposed scheme, quartiles are flexible, allowing the health manager to set the most appropriate threshold level for their reality. For instance, by choosing $threshold_level = 0.9$ as the limit, only the top 10% most critical clusters will generate alerts, ensuring greater focus on high-risk regions. If a broader criterion is needed, the manager can opt for a lower quartile, increasing the number of received notifications. This approach provides greater flexibility and customization, allowing the system to adapt to different regional demands.

In the next section, the computational experiments will be presented, where different performance metrics were used to analyze the quality of the obtained solutions, evaluating the effectiveness of the Z-Alert index in distinguishing between critical and non-critical clusters and its impact on prioritizing notifications.

Results and Discussion

In this section, we validated the results obtained by the Z-Alert Index using data from the Brazilian Ministry of Health (BMoH), which contains records of YF cases from 05

May 2014 to 31 December 2024 [24]. The initial date coincides with the first valid record from NHPs in SISS-Geo, while the final date reflects the ministry's latest update on YF case occurrences in Brazil. The following criteria were considered for data collection:

Animal type: Marmoset and monkey.

Inclusion period: from 05 May 2014 to 31 December 2024.

Record status: "Verified by a specialist", "audited" and "valid".

Collection date: 04 April 2025.

As a result of this search, we obtained 6831 records. These data were filtered, considering only records obtained via GPS or explicitly provided by the user, with an accuracy of 100m, resulting in 4821 records for analysis. After data collection and filtering, each record was assigned the geocoding of the municipalities based on latitude and longitude information. This geolocation step was necessary only for validating the results, as the data provided by the BMoH reports YF cases at the municipal level.

Figs 4 and 5 present the clusters generated after the clustering step, using the limits $L_{D_S} = 1$ km and $L_{D_T} = 30$ days. This process resulted in 3508 clusters, of which 2841 were unit clusters, representing approximately 81% of the total clusters. In the next stage, cluster characterization was performed, where new attributes were generated, as presented in section Cluster characterization.

Fig 4. Data clustering. Illustration of all clusters obtained, where unit clusters are colored in gray circles.

Fig 5. Data clustering. Emphasizing only non-unit clusters.

Analyzing the Pareto Front

As described in section Optimizing cluster attribute weights, only clusters with more than one register were considered for the optimization process, and among these, only those containing confirmed YF records were included. Thus, for the attribute weights optimization process, 264 records grouped into 38 clusters were considered. To maximize the objective function F described in equation (4), the SLSQP (Sequential Least Squares Programming) method implemented in the SciPy library (`https://scipy.org/`) was used. The bounds on the variables are $0 \leq p_j \leq 1$ for all $j = 1, \ldots, m$, constrained by $\sum_{j=1}^{m} p_j = 1$. Different values of w were tested for generating the Pareto front. The parameter w controls the priority between the two optimization objectives:

- Higher values of w assign greater weight to the mean of critical clusters indices, emphasizing the maximization of alerts in high-risk regions.
- Lower values of w assign more importance to the variance of critical cluster indices, aiming to reduce value dispersion and ensure a more balanced distribution of alerts.

The tested values were

$$w = [0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1]$$

Additionally, we evaluated a scenario in which no optimization process was used for weight assignment. In this case, the weights were evenly distributed among all variables, assuming that each attribute has the same importance in constructing the Z-Alert Index. The SLSQP method was executed 11 times, once for each value of w, with $w = 0$

for the first experiment, $w = 0.1$ for the second, and so on. The non-optimized experiment was also considered, totaling 12 experiments. Each execution resulted in different values for the analyzed objectives. Fig 6 presents the obtained Pareto front, where objective f_1 is maximized and f_2 is minimized. The values from 1 to 12 indicate the solutions obtained, representing the experiments conducted.

Fig 6. Pareto front. Pareto front obtained.

From these results, we can conclude the following points:

- The resulting front exhibits the expected behavior for a multiobjective optimization problem, where there is a trade-off between the two objectives [22]. As f_1 improves (increases), f_2 worsens (also increases). The red-connected points represent the Pareto front, meaning they are non-dominated solutions where no objective can be improved without compromising another. Note that uniformly distributing the values of w does not necessarily result in a uniformly distributed set of solutions on the Pareto front, as observed in the figure.
- The solution marked as 12 does not belong to the Pareto front, indicating that it is dominated by another solution. This means that assigning uniform weights to the attributes does not produce an efficient solution, reinforcing the importance of optimization in determining the weights.
- Solutions 9, 10, and 11, which correspond to $w = 0.8, 0.9, 1$, produced the same results. This suggests that the maximization of f_1 reaches a limit, where further variations in w no longer affect model performance.
- The best solution depends on the application's priority. If the goal is to only maximize f_1, a point further to the right on the curve should be selected. If minimizing f_2 is more important, the leftmost point is preferable. If the objective is to maintain a balanced index for critical clusters without excessively increasing variance, the optimal point should be near the middle of the curve, suggesting that experiment 6 is the best alternative, as it keeps f_1 reasonably high while significantly improving f_2.

Analyzing the attributes' weights

The attribute weights obtained in each solution are presented in Table 3. This table allows for a more detailed analysis of the relationship between attribute weights and objective values. From these results, we can highlight the following points:

- Solution 12 (highlighted in the table) that applies uniform weights without optimization, presents a uniform distribution of weight values leading to a dominated solution in the objective space. This outcome highlights the importance of weight optimization, as it allows the alert index to better differentiate between clusters by emphasizing attributes that are more strongly associated with critical clusters. By assigning weights based on their contribution to relevant patterns, the optimized solution achieves improved performance in identifying high-risk areas.
- Solutions 9, 10, and 11 prioritize only the *n_death* attribute, assigning 100% of the weight to it. These experiments achieve the best value for f_2, indicating that *n_death* is the most relevant attribute. However, since objective f_1 worsens significantly, these solutions are not suitable if a trade-off must be maintained.

- Solution 7 and 8 prioritize the *n_death* attribute but also introduce weights for *freq_rec* and *extention*. The *perc_death* attribute also shows relevance in experiment 7.

- Solutions 5 and 6 generate good trade-off solutions between f_1 and f_2. Both indicate that *perc_deaths* plays a crucial role in balancing the objectives, with *n_death* being the second most important attribute. Other attributes assigned weights in these experiments include *n_rec*, *interval*, *freq_rec*, and *extention*.

- Solutions 1 to 4 prioritize the minimization of f_2, where *perc_death* emerges as the most relevant attribute, followed by *interval*, *freq_animal*, and *n_rec*.

Table 3. Assignment of attribute weights and objective values in each experiment.

Sol.	*n_rec*	*interval*	*freq_rec*	*freq_animal*	*extention*	*perc_death*	*n_death*	f_1	f_2
1	0.007	0.140	0.000	0.097	0.000	0.756	0.000	0.655	0.027
2	0.041	0.127	0.002	0.061	0.000	0.768	0.000	0.732	0.032
3	0.046	0.120	0.016	0.000	0.000	0.761	0.056	0.935	0.069
4	0.048	0.101	0.018	0.000	0.005	0.688	0.141	1.195	0.157
5	0.040	0.067	0.024	0.000	0.032	0.581	0.256	1.551	0.353
6	0.028	0.020	0.033	0.000	0.070	0.432	0.416	2.050	0.769
7	0.000	0.000	0.055	0.000	0.109	0.185	0.651	2.775	1.673
8	0.000	0.000	0.034	0.000	0.039	0.000	0.927	3.495	2.968
9	0.000	0.000	0.000	0.000	0.000	0.000	1.000	3.611	3.257
10	0.000	0.000	0.000	0.000	0.000	0.000	1.000	3.611	3.257
11	0.000	0.000	0.000	0.000	0.000	0.000	1.000	3.611	3.257
12	0.143	0.143	0.143	0.143	0.143	0.143	0.143	1.839	0.977

Observe that the attributes *n_death* (number of dead animals) and *perc_death* (percentage of deaths) capture complementary aspects of disease severity. While the absolute count provides a direct measure of mortality events, the latter highlights the lethality in proportion to the affected population, thereby enabling a more robust characterization of epidemiological risk. The presence of *perc_death* seems to be crucial for determining the Z-Alert Index, as it appears to influence both objectives in a more balanced and controlled manner than *n_death*. This outcome is expected, given the structure of the objective functions, which are formulated based on critical clusters. In these clusters, animals are confirmed to be infected with the YF virus and are therefore likely deceased. The other attributes show a certain degree of importance, appearing in a balanced way in most of the experiments.

Validation of the Z-Alert Index

In this section, the Z-Alert Index is validated using data from the BMoH containing records of YF cases within a predetermined period. The goal is to verify whether the alerts generated by the index correspond to the cases recorded in the BMoH database. Validation will be carried out through the geocoding of municipalities and the analysis of the event occurrence period. To confirm an alert, we consider a 30-day window from the date of a confirmed case in the BMoH database. For instance, if the system generates an alert for a municipality on January 1, 2020, we check whether there is a confirmed YF case for that municipality in the BMoH database between that date and January 31, 2020, to confirm the alert in the corresponding cluster.

To assess the performance of the index, we use a confusion matrix [25]. The confusion matrix is a widely used tool to visualize the performance of classification

algorithms. It organizes the results in a table that details the relationship between the model's predictions and the actual observed values, as shown in Table 4.

Table 4. Confusion Matrix.

	Predicted Positive	**Predicted Negative**
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

In this study, the class defined as *actual* represents the data from the BMoH database containing confirmed YF cases. A municipality is classified as positive if a confirmed case of the disease is present, whether in humans or non-human primates. Thus, all records in the BMoH database correspond to positive cases. The class defined as *predicted* represents the alerts generated by the system. A cluster is classified as positive if it triggers an alert and negative otherwise. Based on this, the classification categories are defined as follows

- True Positive (TP): The system generates an alert, and the corresponding municipality has a confirmed YF case within the same period.
- False Negative (FN): The system does not generate an alert, but the corresponding municipality has a confirmed YF case within the same period.
- False Positive (FP): The system generates an alert, but the corresponding municipality does not have any confirmed YF cases within the same period. Notice that this is not a quite reliable metric due to probable under-reporting; in other words, the metric tends to be pessimistic.
- True Negative (TN): The system does not generate an alert, and the corresponding municipality also does not have any confirmed YF cases within the same period. Analogously, this metric may be impacted by under-reporting.

Based on this classification, various performance metrics were employed to evaluate the proposed alert system. The metrics used are described below:

- Precision: measures the accuracy of positive predictions, defined as: $Prec = \frac{TP}{(TP+FP)}$
- Sensitivity: is the true positive rate, i.e., the proportion of correctly classified positive predictions, calculated as: $Sen = \frac{TP}{(TP+FN)}$
- F1-Score: corresponds to the harmonic mean between Precision and Sensitivity, expressed as: $F1 = \frac{2(Pr \cdot Sen)}{(Pr+Sen)}$
- Specificity: is the true negative rate, i.e., the proportion of correctly classified negative predictions, defined as: $Spec = \frac{TN}{(TN+FP)}$
- Accuracy: represents the overall proportion of correct predictions, given by: $Acc = \frac{(TP+TN)}{(TP+TN+FP+FN)}$
- Generalization: measures the model's ability to correctly identify confirmed YF cases in relation to the total confirmed cases recorded in SISS-Geo, calculated as: $Gen = \frac{TP}{total_n_conf}$

We evaluated the Z-Alert index for all clusters with more than one record whose municipalities and event periods match the information in the BMoH database. In this experiment, 492 clusters were assessed using different threshold levels, which define the criticality limit of the clusters (as detailed in section The Z-Alert Index). Table 5 presents the results of performance metrics for different threshold levels $(0.8, 0.85, 0.9, 0.95,$ and $0.99)$ for the non-dominated solutions obtained in experiments 1, 6, and 11 (see section). These solutions were selected as the most representative based on the prioritization of objectives: solution 1 emphasizes minimizing f_2, solution 6 aims to balance f_1 and f_2, and solution 11 focuses on maximizing f_1. In table 5, TL indicates *threshold_level*, Alert is the number of alerts, NonAL represents the number of non-alerts, and PAL corresponds to the proportion of alerts.

Table 5. Performance metrics for the analysis of the Z-Alert Index for experiments 1, 6, and 9 presented in Fig 6.

Sol.	TL	Alert	NonAL	PAL	TP	TN	FP	FN	Prec	Sen	F1	Spec	Acc	Gen
1	0.8	349	143	0.71	329	15	20	128	0.94	0.72	0.82	0.43	0.70	9.14
6	0.8	364	128	0.74	343	14	21	114	0.94	0.75	0.84	0.40	0.73	9.53
11	0.8	387	105	0.79	366	14	21	91	0.95	0.80	0.87	0.40	0.77	10.17
1	0.85	341	151	0.69	321	15	20	136	0.94	0.70	0.80	0.43	0.68	8.92
6	0.85	355	137	0.72	335	15	20	122	0.94	0.73	0.83	0.43	0.71	9.31
11	0.85	297	195	0.60	278	16	19	179	0.94	0.61	0.74	0.46	0.60	7.72
1	0.9	230	262	0.47	220	25	10	237	0.96	0.48	0.64	0.71	0.50	6.11
6	0.9	229	263	0.47	217	23	12	240	0.95	0.47	0.63	0.66	0.49	6.03
11	0.9	232	260	0.47	219	22	13	238	0.94	0.48	0.64	0.63	0.49	6.08
1	0.95	132	360	0.27	126	29	6	331	0.95	0.28	0.43	0.83	0.32	3.50
6	0.95	135	357	0.27	127	27	8	330	0.94	0.28	0.43	0.77	0.31	3.53
11	0.95	116	376	0.24	109	28	7	348	0.94	0.24	0.38	0.80	0.28	3.03
1	0.99	24	468	0.05	22	33	2	435	0.92	0.05	0.09	0.94	0.11	0.61
6	0.99	27	465	0.05	25	33	2	432	0.93	0.05	0.10	0.94	0.12	0.69
11	0.99	26	466	0.05	24	33	2	433	0.92	0.05	0.10	0.94	0.12	0.67

From the results presented in Table 5, we can highlight the following points:

- Proportion of Alerts (PAL) and non-alerts: As the threshold level increases (from 0.8 to 0.99), the proportion of alerts decreases significantly, while the of non-alerts increases. This indicates that, with stricter thresholds, fewer clusters are classified as alerts. For example, at the 0.8 threshold, the PAL ranges from 0.71 to 0.79, while at the 0.99 threshold, PAL drops to 0.05.

- False Negatives (FN): Notice that the worst-case scenario occurs with the false negatives, as this indicates that the system failed to generate an alert for municipalities with confirmed YF cases. As the threshold level increases, false negatives also increase, because with higher thresholds, the model classifies fewer clusters as alerts.

- Precision (Prec): Precision is consistently high across all experiments and threshold levels, ranging from 0.92 to 0.96. This indicates that when a cluster is classified as an alert, there is a high probability that it is a true alert with a confirmed YF case.

- Sensitivity (Sen): Sensitivity decreases as the threshold level increases. For instance, at the 0.8 threshold, sensitivity ranges from 0.72 to 0.80, while at the 0.99 threshold, it drops to 0.05. This shows that, with stricter thresholds, the model detects fewer alerts with confirmed YF cases, since fewer alerts are generated.

- F1-Score (F1): The F1-Score, which balances precision and sensitivity, decreases as the threshold level increases. At the 0.8 threshold, F1 ranges from 0.82 to 0.87, while at the 0.99 threshold, it drops to $0.09 - 0.10$. This indicates that, with stricter thresholds, the balance between precision and sensitivity worsens.

- Specificity (Spec): Specificity increases with higher threshold levels, ranging from 0.43 (at the 0.8 threshold) to 0.94 (at the 0.99 threshold). This indicates that, with stricter thresholds, the model is better at correctly identifying non-alerts.

- Accuracy (Acc): Accuracy decreases as the threshold level increases. For example, at the 0.8 threshold, accuracy ranges from 0.70 to 0.77, while at the 0.99 threshold, it drops to $0.11 - 0.12$. This reflects the trade-off between detecting alerts and non-alerts.

- Generalization (Gen): Generalization also decreases as the threshold level increases, ranging from $9.14 - 10.17$ (at the 0.8 threshold) to $0.61 - 0.69$ (at the 0.99 threshold). This happens because, with stricter thresholds, the model classifies fewer clusters as alerts, resulting in fewer true positives. Here, it is interesting to note the trade-off between Generalization and Precision. Even though SISS-Geo database contains significantly fewer confirmed YF cases (185 records considered in the optimization process) compared to the BMoH database (3323 records), precision remains high across all experiments and threshold levels, indicating the model's high ability to generate an alert with a confirmed YF case.

Figs 7-9 illustrate the alert-generating clusters identified in solutions 1, 6 and 11 using a threshold level of 0.99. The color intensity reflects the alert index value, with darker shades indicating higher levels of risk associated with the clusters. The bottom right image in Fig 7 shows a cluster with records located along the border between two municipalities, Mafra and Itaiópolis in Santa Catarina state. In these situations, the alert is generated for both municipalities to ensure comprehensive coverage and response. In all figures, the effect of clustering the points according to the predefined spatial and temporal constraints (see in section The DBSCAN algorithm) is clearly visible, as the events are grouped into coherent regions that reflect both geographic proximity and timing, facilitating more accurate detection and analysis of potential outbreak patterns.

Fig 7. Alert-generating clusters. Result from solution 1 showing 24 alerts at threshold level 0.99.

Fig 8. Alert-generating clusters. Result from solution 6 showing 27 alerts at threshold level 0.99.

Fig 9. Alert-generating clusters. Result from solution 11 showing 26 alerts at threshold level 0.99.

By analyzing the results by solutions, a consistent trend is observed: as the threshold level increases, the number of alerts decreases, leading to higher specificity but lower sensitivity and F1-scores. Some other conclusions can be pointed out:

- In solution 1, which prioritizes the minimization of false positives, performance is characterized by consistently high precision (≥ 0.92) across thresholds. At TL = 0.8, this configuration achieves a sensitivity of 0.72 and an F1-score of 0.82, but

these values deteriorate as the threshold increases, with sensitivity dropping to 0.05 and F1-score to 0.09 at TL = 0.99. On the other hand, specificity improves from 0.43 to 0.94. This configuration is particularly suited for applications where false alerts carry a high cost, and precision is more important than sensitivity.

- Solution 6 seeks a balanced optimization between the objectives and demonstrates stable and robust performance. At TL = 0.8, it attains a sensitivity of 0.75, precision of 0.94, and F1-score of 0.84. As TL increases, performance degrades more gradually compared to the other solutions. This configuration provides a good compromise between detection capability and reliability, making it appropriate when both avoiding false positives and capturing true cases are important.

- Solution 11, which emphasizes maximizing true positives, yields the highest F1-score of 0.87 and sensitivity of 0.80 at TL = 0.8, while maintaining high precision (0.95). However, specificity remains low (0.40), indicating a trade-off where a higher number of false positives is tolerated to ensure greater coverage of true cases. This setting is ideal in scenarios where missing true alerts has a greater cost than issuing false ones.

When comparing results by threshold level, lower TLs (0.8–0.85) result in a higher percentage of alerts (PAL = 0.72–0.79), higher sensitivity (0.73–0.80), and stronger F1-scores (0.83–0.87), but also more false positives. Conversely, higher TLs (0.95–0.99) drastically reduce the number of alerts (PAL = 0.05–0.27), improving specificity (up to 0.94 at TL=0.99) while severely compromising sensitivity (as low as 0.05) and F1-score (as low as 0.09). These higher thresholds are most appropriate when it is critical to avoid false alarms, even if most true cases are missed. Overall, Table 6 presents some conclusions based on threshold level setting, indicating the recommended threshold level (Rec. TL) for each solution.

Table 6. Conclusions based on threshold level setting

Objective	Rec. TL	Best solution	Justification
Maximize real cases detection	0.8	11	High Sensitivity (0.80), best F1 (0.87), strong Generalization.
Best balance across metrics	0.85	6	Few more alerts, but best sensitivity and F1 scores.
Minimize false positives	0.9+	1 or 6	High Specificity, low values of Alerts and FP, but reduced sensitivity.
Extremely conservative alerting	0.99	Any (similar metrics)	Very low false alerts, but at the cost of missing almost all real cases.

From a practical perspective, defining the threshold level is essential, as it directly impacts the number of alerts that health managers will receive. Since this threshold can be adjusted to align with the specific needs and context of each region, it is important to ensure that only the most relevant situations are notified. Setting a higher threshold–resulting in fewer alerts–could be more appropriate in regions with high vaccination coverage and low human mobility, where the risk of disease spread is significantly reduced.

Based on the results, the solution that prioritizes the first objective (maximizing f_1) is more effective at identifying real cases, although it may also produce a higher number of false positives. If a balance between detection and precision is required, aiming to reduce extreme variations, the solution seeking a balance between f_1 and f_2 stands out as the most appropriate alternative. On the other hand, if the priority is to minimize

false alerts, even at the risk of missing some real cases, the solution that maximizes f_1 becomes the most suitable choice.

By observing the precision metric, it is possible to conclude that the model has a high ability to generate alerts for clusters with confirmed YF cases, even though this confirmation information is not present in the SISS-Geo database. The optimization process for setting the attribute weights was based solely on data from 38 clusters, which together account for 185 records with confirmed YF cases. This result indicates the model's strong potential for generalization, successfully identifying critical clusters beyond those used during the optimization process. Notably, the alert index was also able to highlight high-risk clusters that did not contain confirmed cases, including some composed solely of live animals. In such cases, the alert system complements the current SISS-Geo rule-based approach, which would not have issued an alert under its fixed criteria.

It is also important to highlight the model's limitations, which are directly related to the records provided by SISS-Geo. The effectiveness of the clustering process and the accuracy of the Z-Alert Index are highly dependent on the completeness, precision, and reliability of these records. Moreover, potential gaps in health surveillance, such as delays or failures in reporting confirmed YF cases or other diseases of interest, as well as the limited adoption of SISS-Geo in certain regions of the country, reduce the amount of available data. These issues hinder the model's ability to detect meaningful patterns and to support the timely prioritization of preventive and investigative actions.

Conclusion

The development of a multi-attribute zoonotic alert index (the Z-Alert Index) represents a significant advancement in epidemiological surveillance integrated within SISS-Geo. By adopting dynamic analyses based on clustering and optimization, the proposed system overcomes the limitations of the current model, enabling proactive identification of critical regions and efficient resource prioritization. Experiments with real-world data demonstrated the index's ability to detect clusters associated with YF cases with high precision and adaptability to different scenarios. The flexibility to adjust parameters (such as spatial/temporal limits and threshold levels) ensures that the system can be customized according to regional specificities, diseases of interest and public health strategies.

Although the alert index showed strong performance in identifying clusters potentially associated with yellow fever virus circulation, its applicability to other zoonoses is constrained by data availability. While SISS-Geo also contains records related to rabies and avian influenza, these are considerably less frequent compared to yellow fever-related reports, which may limit the model's sensitivity and generalizability when applied beyond the yellow fever context.

To address these limitations, future work will focus on expanding the model's application to additional zoonoses—such as rabies and avian influenza—by both obtaining more data related to these diseases and integrating complementary environmental variables. Additionally, the optimization process will be continuously refined to reduce false negatives and enhance early warning capabilities across a broader range of epidemiological scenarios.

It is worth highlighting that regions with efficient monitoring and control strategies tend to attract more investments in technology innovation, and healthcare infrastructure. These benefits reinforce that investing in the identification and prioritization of critical regions is not only a public health concern but also a sound economic strategy. The proposed approach underlines the importance of collaboration between technology, data science, and healthcare management to address complex

challenges at the interface of humans, animals, and the environment.

Acknowledge

The authors thank the financial support provided by the Brazilian agencies Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) (grants 380656/2024-8 and 445493/2023-2).

References

1. Sharan M, Vijay D, Yadav JP, Bedi JS, Dhaka P. Surveillance and response strategies for zoonotic diseases: a comprehensive review. Science in One Health. 2023;2:100050. doi:https://doi.org/10.1016/j.soh.2023.100050.

2. Sinclair J. Importance of a One Health approach in advancing global health security and the Sustainable Development Goals. Revue scientifique et technique (International Office of Epizootics). 2019;38(1):145–154. doi:10.20506/rst.38.1.2949.

3. Mackenzie JS, Jeggo M. The One Health Approach—Why Is It So Important? Tropical Medicine and Infectious Disease. 2019;4(2). doi:10.3390/tropicalmed4020088.

4. Gadelha Jr LMR, de Siracusa PC, Dalcin EC, da Silva LAE, Augusto DA, Krempser E, et al. A survey of biodiversity informatics: Concepts, practices, and challenges. WIREs Data Mining and Knowledge Discovery. 2021;11(1):e1394. doi:https://doi.org/10.1002/widm.1394.

5. Chame M, Barbosa HJC, Gadelha LMR, Augusto DA, Krempser E, Abdalla L. SISS-Geo: Leveraging Citizen Science to Monitor Wildlife Health Risks in Brazil. Journal of Healthcare Informatics Research. 2019;3(4):414–440. doi:https://doi.org/10.1007/s41666-019-00055-2.

6. Chame M, Abdalla L, Pinter A, Romano APM, Krempser E, Ramos DG, et al. Primates in SISS-Geo: potential contributions of mobile technology, health surveillance and citizen science to support species conservation in brazil. Neotropical Primates. 2020;26(2):80–89. doi:10.62015/np.2020.v26.42.

7. R CEC, da SGA, de J QA, das N T JK, de A LB, de CCAM, et al. Evaluation of SISS-GEO application as a tool for monitoring reservoir animals of the Yellow Fever Virus (YFV) in Brazil. Electronic Journal Collection Health. 2020;12(9):e3664. doi:https://doi.org/10.25248/reas.e3664.2020.

8. Andrade MS, Campos FS, Oliveira CHd, Oliveira RS, Campos AAS, Almeida MABd, et al. Fast surveillance response reveals the introduction of a new yellow fever virus sub-lineage in 2021, in Minas Gerais, Brazil. Memórias do Instituto Oswaldo Cruz. 2022;117:e220127. doi:10.1590/0074-02760220127.

9. Giovanetti M, Pinotti F, Zanluca C, et al. Genomic epidemiology unveils the dynamics and spatial corridor behind the Yellow Fever virus outbreak in Southern Brazil. Science Advances. 2023;9(35). doi:10.1126/sciadv.adg9204.

10. FAO, UNEP, WHO, WOAH. One Health Joint Plan of Action (2022-2026). Working together for the health of humans, animals, plants and the environment; 2022. Available from: `https://doi.org/10.4060/cc2289en`.

11. Hassan OA, de Balogh K, Winkler AS. One Health early warning and response system for zoonotic diseases outbreaks: Emphasis on the involvement of grassroots actors. Veterinary Medicine and Science. 2023;9(4):1881–1889. doi:https://doi.org/10.1002/vms3.1135.

12. Malik S, Pandey I, Kishore S, Sundarrajan T, Nargund SL, Ghosh A, et al. Yellow fever virus, a mosquito-borne flavivirus posing high public health concerns and imminent threats to travellers – an update. 2023;109(2):134–137. doi:10.1097/JS9.0000000000000086.

13. Vrbova L, Stephen C, Kasman N, Boehnke R, Doyle-Waters M, Chablitt-Clark A, et al. Systematic Review of Surveillance Systems for Emerging Zoonoses. Transboundary and Emerging Diseases. 2010;57(3):154–161. doi:https://doi.org/10.1111/j.1865-1682.2010.01100.x.

14. Possas C, de Oliveira RL, Tauil PL, de Paula Pinheiro F, Pissinatti A, da Cunha RV, et al. Yellow fever outbreak in Brazil: the puzzle of rapid viral spread and challenges for immunisation. 2018;113(10):1–12. doi:10.1590/0074-02760180278.

15. Ministério da Saúde Secretaria de Vigilância em Saúde. Boletim Epidemiológico: situação epidemiológica da febre amarela no Monitoramento 2019/2020. Boletim Epidemiológico. v. 51, n. 01, p. 1-9, 2020; 2020. Available from: `https://www.gov.br/saude/pt-br/centrais-de-conteudo/publicacoes/boletins/boletins-epidemiologicos/edicoes/2020/boletim-epidemiologico-vol-51-no-01/@@download/file/boletim-epidemiologico-svs-01.pdf`.

16. Ministério da Saúde Secretaria de Vigilância em Saúde. Boletim Epidemiológico: situação epidemiológica da febre amarela no Monitoramento 2019/2020. Boletim Epidemiológico. v. 51, n. 46, p. 8-19, 2020; 2020. Available from: `https://www.gov.br/saude/pt-br/centrais-de-conteudo/publicacoes/boletins/boletins-epidemiologicos/edicoes/2020/boletim_epidemiologico_svs_46.pdf/view`.

17. Ministério da Saúde Secretaria de Vigilância em Saúde. Boletim Epidemiológico: situação epidemiológica da febre amarela no Monitoramento 2020/2021. Boletim Epidemiológico. v. 52, n. 31, p. 20-31, 2021; 2021. Available from: `https://www.gov.br/saude/pt-br/centrais-de-conteudo/publicacoes/boletins/boletins-epidemiologicos/edicoes/2020/boletim-epidemiologico-vol-51-no-01/@@download/file/boletim-epidemiologico-svs-01.pdf`.

18. Ministério da Saúde Secretaria de Vigilância em Saúde e Ambiente. Nota Informativa nº 35/2024-CGARB/DEDT/SVSA/MS: atualização do cenário epidemiológico da febre amarela e resultados da Oficina do Grupo de Modelagem de Febre Amarela (GRUMFA). Brasília: Ministério da Saúde, 2024; 2024. Available from: `https://www.gov.br/saude/pt-br/assuntos/saude-de-a-a-z/f/febre-amarela/notas-tecnicas-e-informativas`.

19. Ministério da Saúde. Plano de Contingência para respostas às emergências em saúde pública - Febre Amarela 2° ed. atualizada. Brasília, DF; 2021. Available from: `https://www.gov.br/saude/pt-br/centrais-de-conteudo/publicacoes/publicacoes-svs/febre-amarela/plano_contingencia_emergencias_febre_amarela_2_ed-1.pdf/@@download/file/plano_contingencia_emergencias_febre_amarela_2_ed.pdf`.

20. Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. KDD'96. AAAI Press; 1996. p. 226–231.

21. Schubert E, Sander J, Ester M, Kriegel HP, Xu X. DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. ACM Trans Database Syst. 2017;42(3). doi:10.1145/3068335.

22. Miettinen K. Nonlinear Multiobjective Optimization. Norwell, Massachusetts: Kluwer Academic; 1999.

23. Deb K. Multi-Objective Optimization using Evolutionary Algorithms. John Wiley & Sons, Chichester; 2001.

24. Ministério da Saúde. Febre Amarela em humanos e primatas não-humanos - 1994 a 2025; 2025. Available from: `https://opendatasus.saude.gov.br/dataset/febre-amarela-em-humanos-e-primatas-nao-humanos`.

25. Ting KM. In: Sammut C, Webb GI, editors. Confusion Matrix. Boston, MA: Springer US; 2017. p. 260–260.

Supporting information

S1 Fig. The zoonotic alert system. Main steps of the zoonotic alert system

S2 Fig. Clustering techniques. Comparative analyses of clusters techniques from Scikit-learn library

S3 Fig. Two main goals of MOPs. Desired aspects of a Pareto front

S4 Fig. Data clustering. Illustration of all clusters obtained, where unit clusters are colored in gray circles

S5 Fig. Data clustering. Emphasizing only non-unit clusters

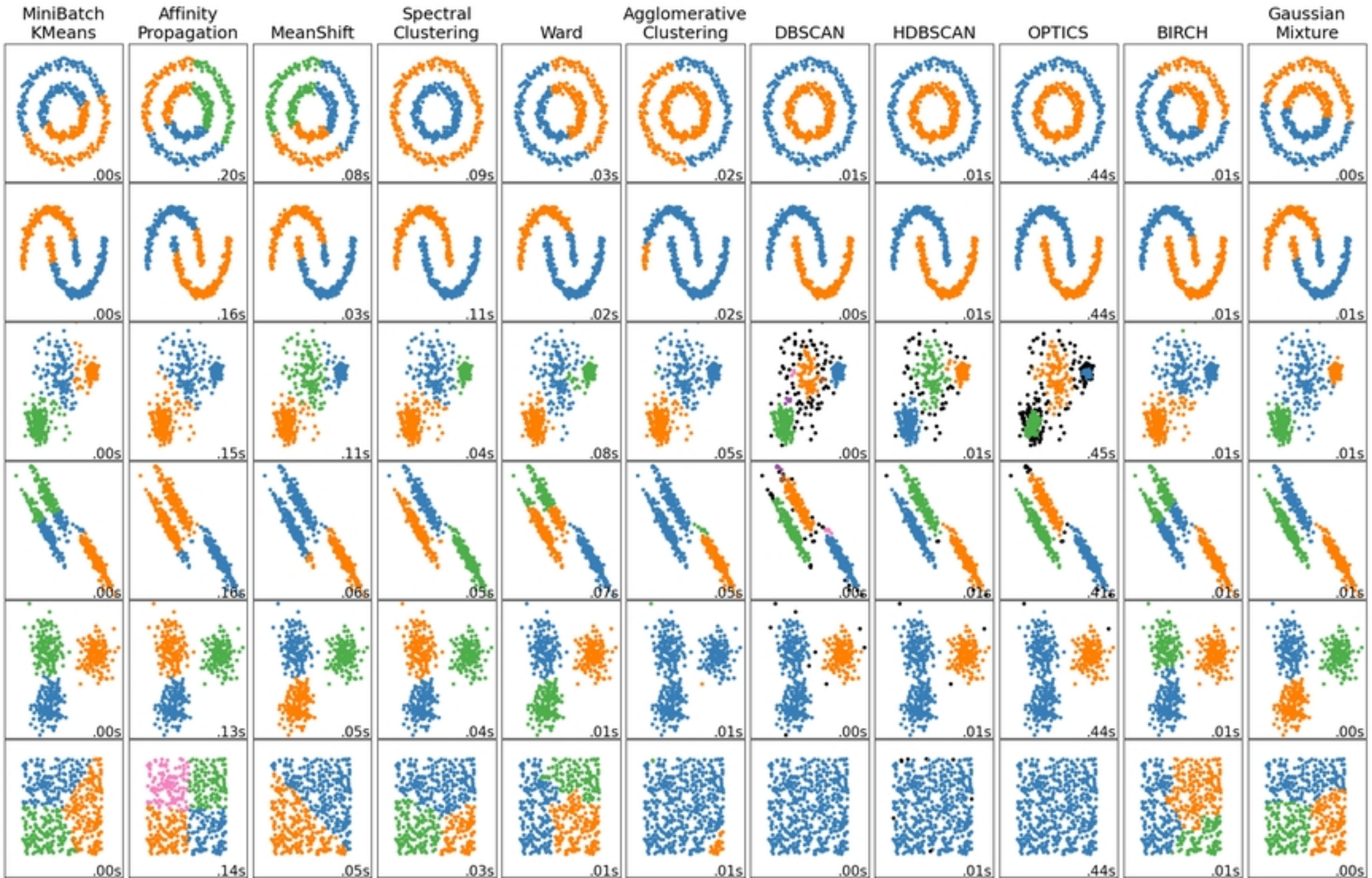
S6 Fig. Pareto front. Pareto front obtained

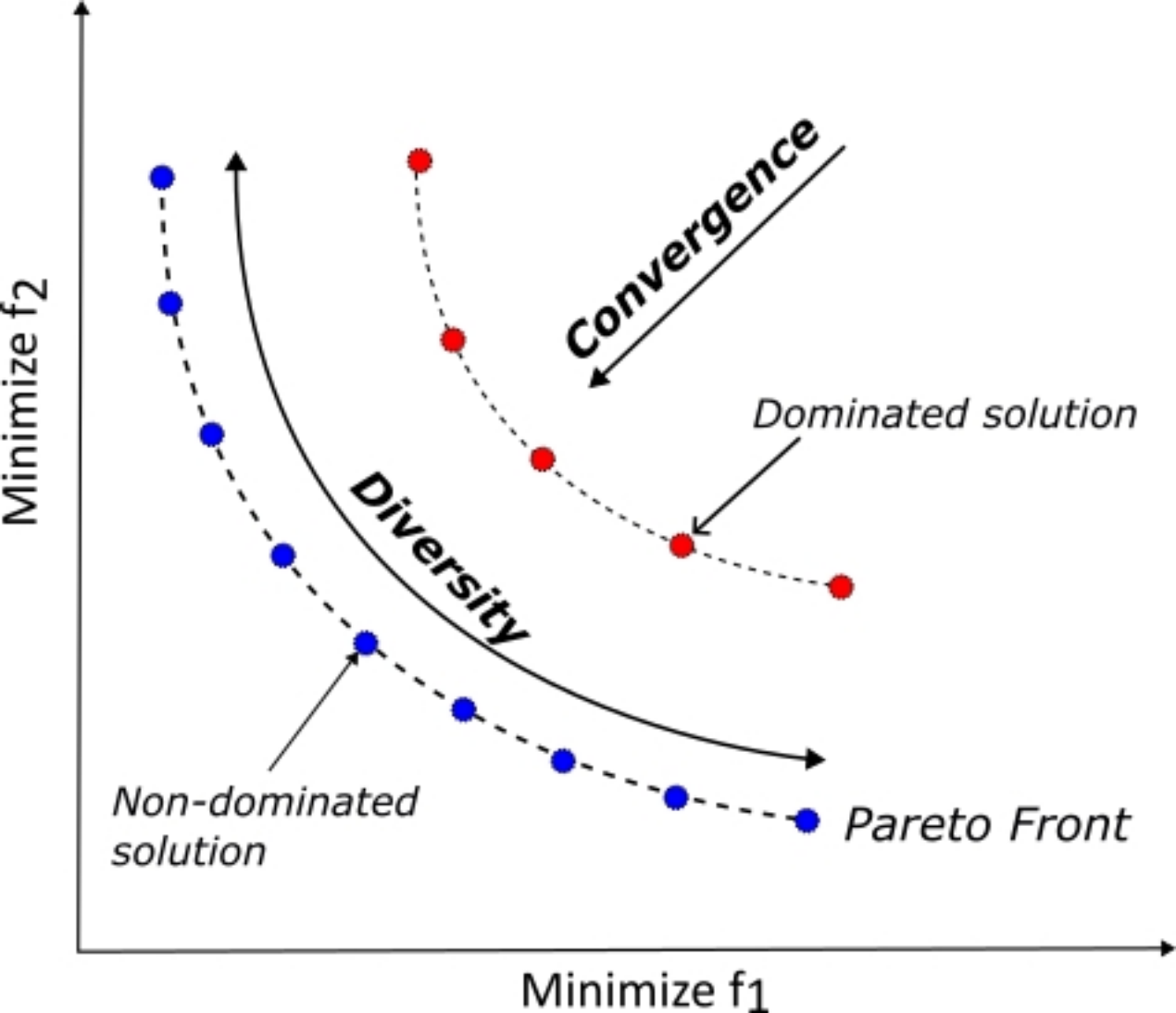
S7 Fig. Alert-generating clusters. Result from solution 1 showing 24 alerts at threshold level 0.99

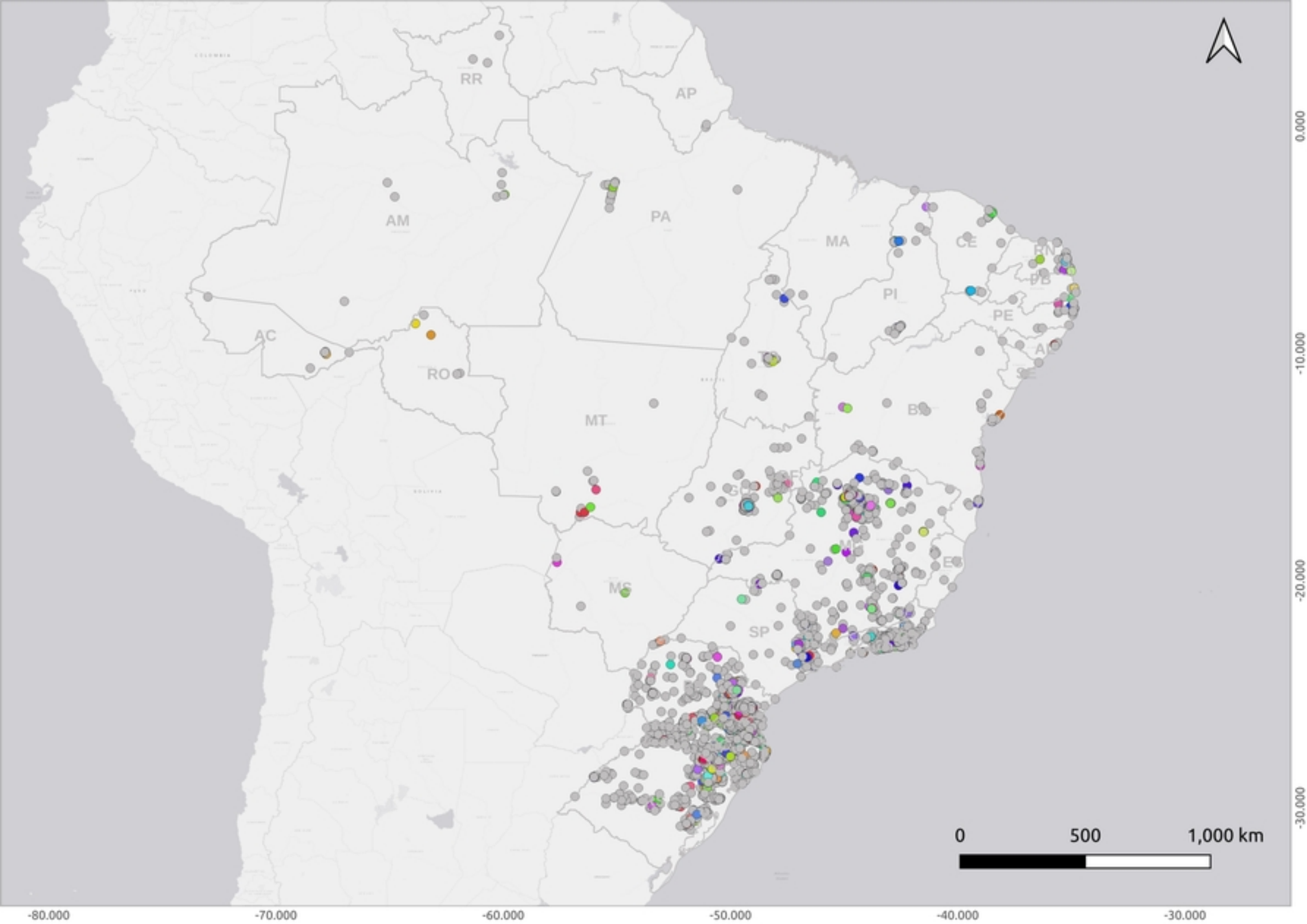
S8 Fig. Alert-generating clusters. Result from solution 6 showing 27 alerts at threshold level 0.99

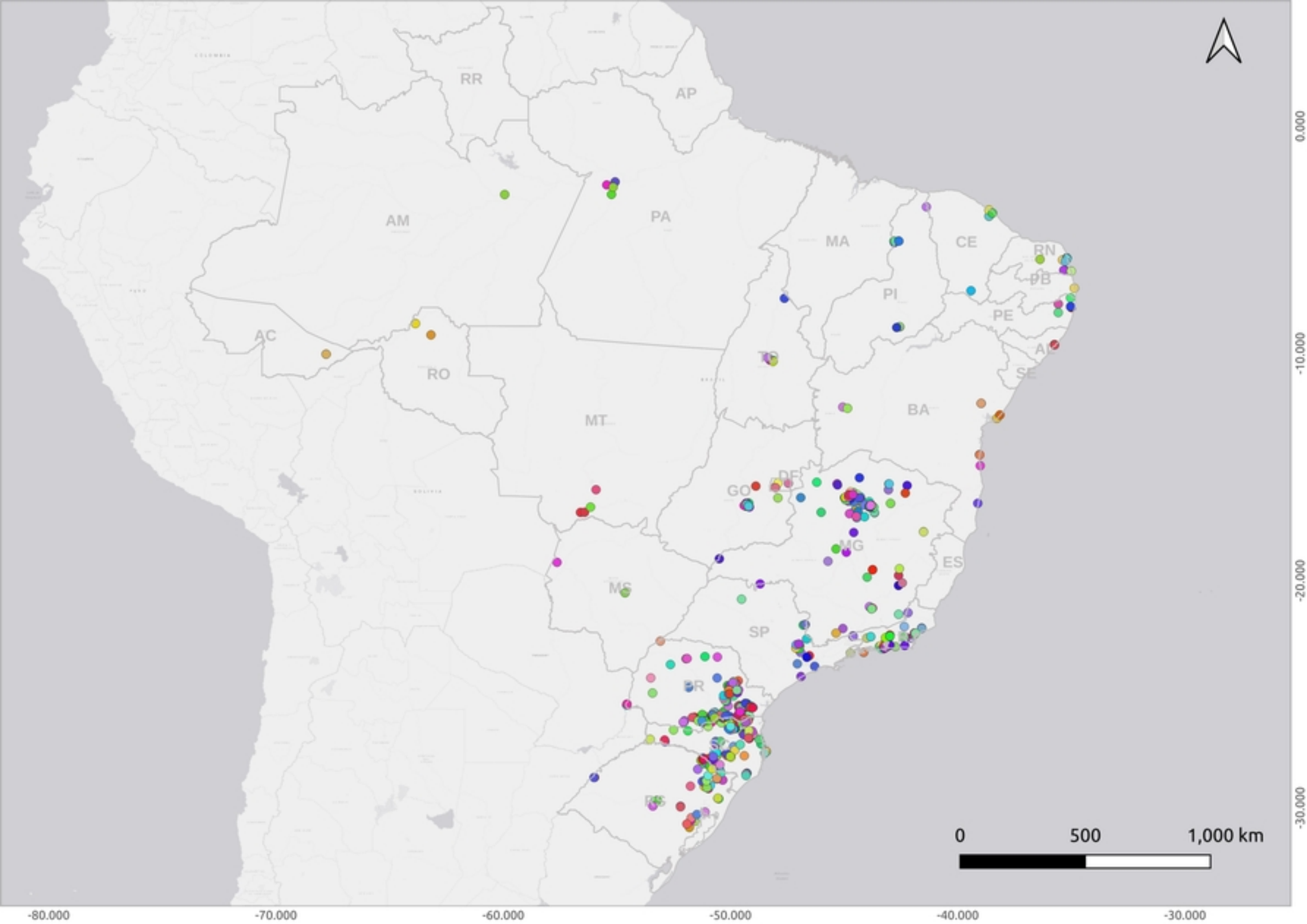
S9 Fig. Alert-generating clusters. Result from solution 11 showing 26 alerts at threshold level 0.99

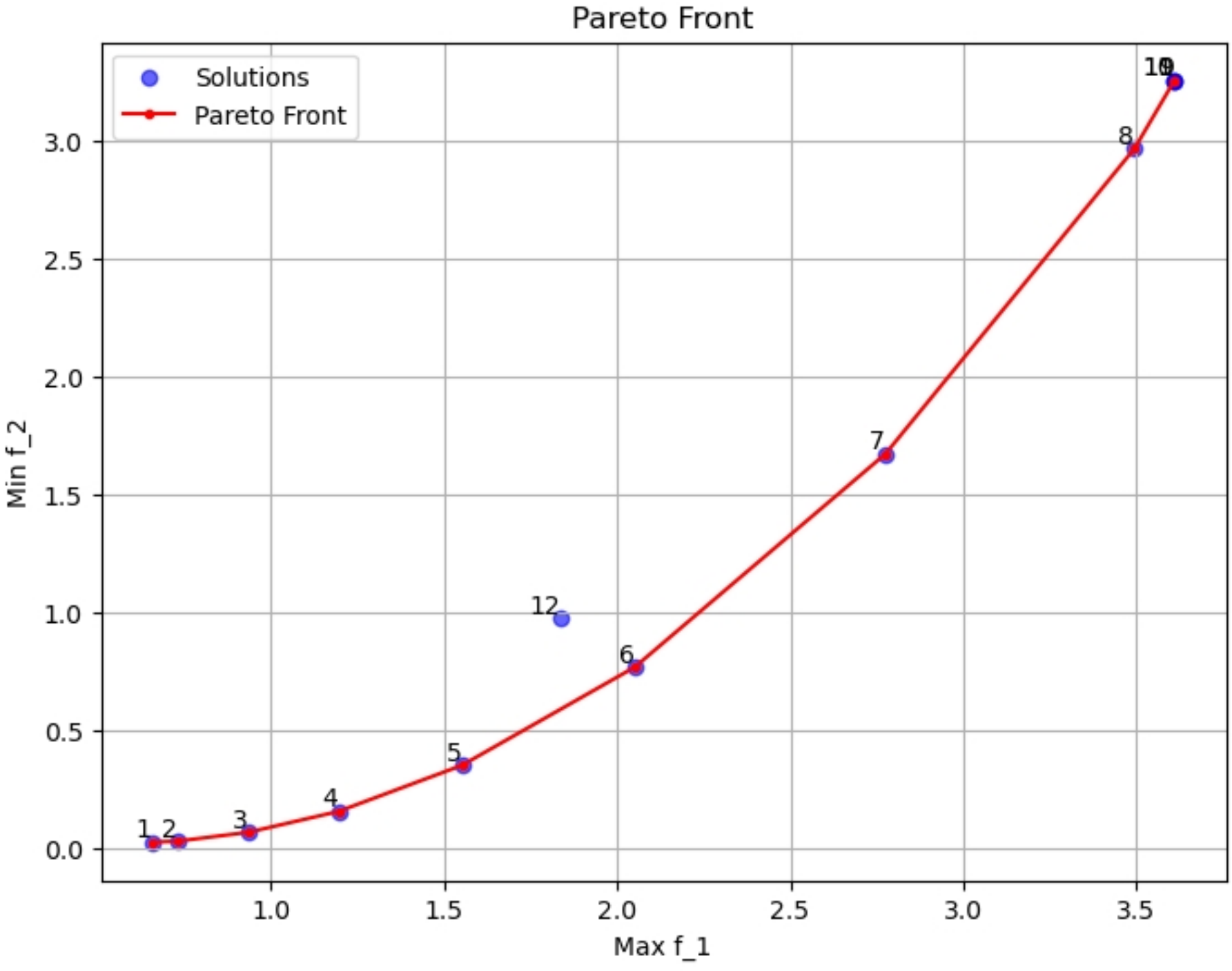
SISS-Geo
Preprocessing
Clustering data
Cluster characterization
Multiobjective optimization
Z-Alert Index
Sending email

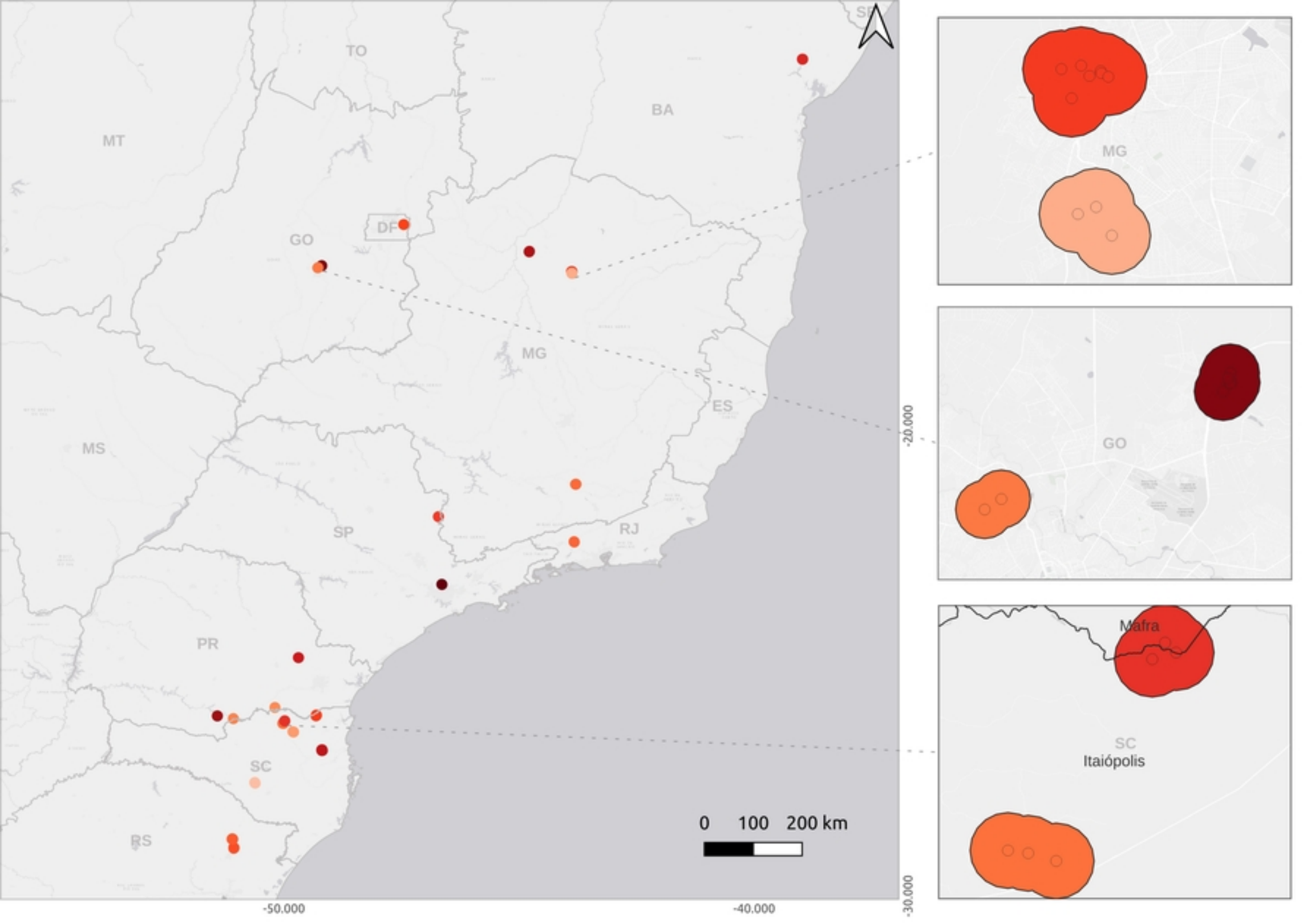

MiniBatch KMeans
Affinity Propagation
MeanShift
Spectral Clustering
Ward
Agglomerative Clustering
DBSCAN
HDBSCAN
OPTICS
BIRCH
Gaussian Mixture
.00s .20s .08s .09s .03s .02s .01s .01s .44s .01s .00s
.00s .16s .03s .11s .02s .02s .00s .01s .44s .01s .01s
.00s .15s .11s .04s .08s .05s .00s .01s .45s .01s .01s
.00s .16s .06s .05s .07s .05s .00s .01s .41s .01s .01s
.00s .13s .05s .04s .01s .01s .00s .01s .44s .01s .00s
.00s .14s .05s .03s .01s .01s .00s .01s .44s .01s .00s

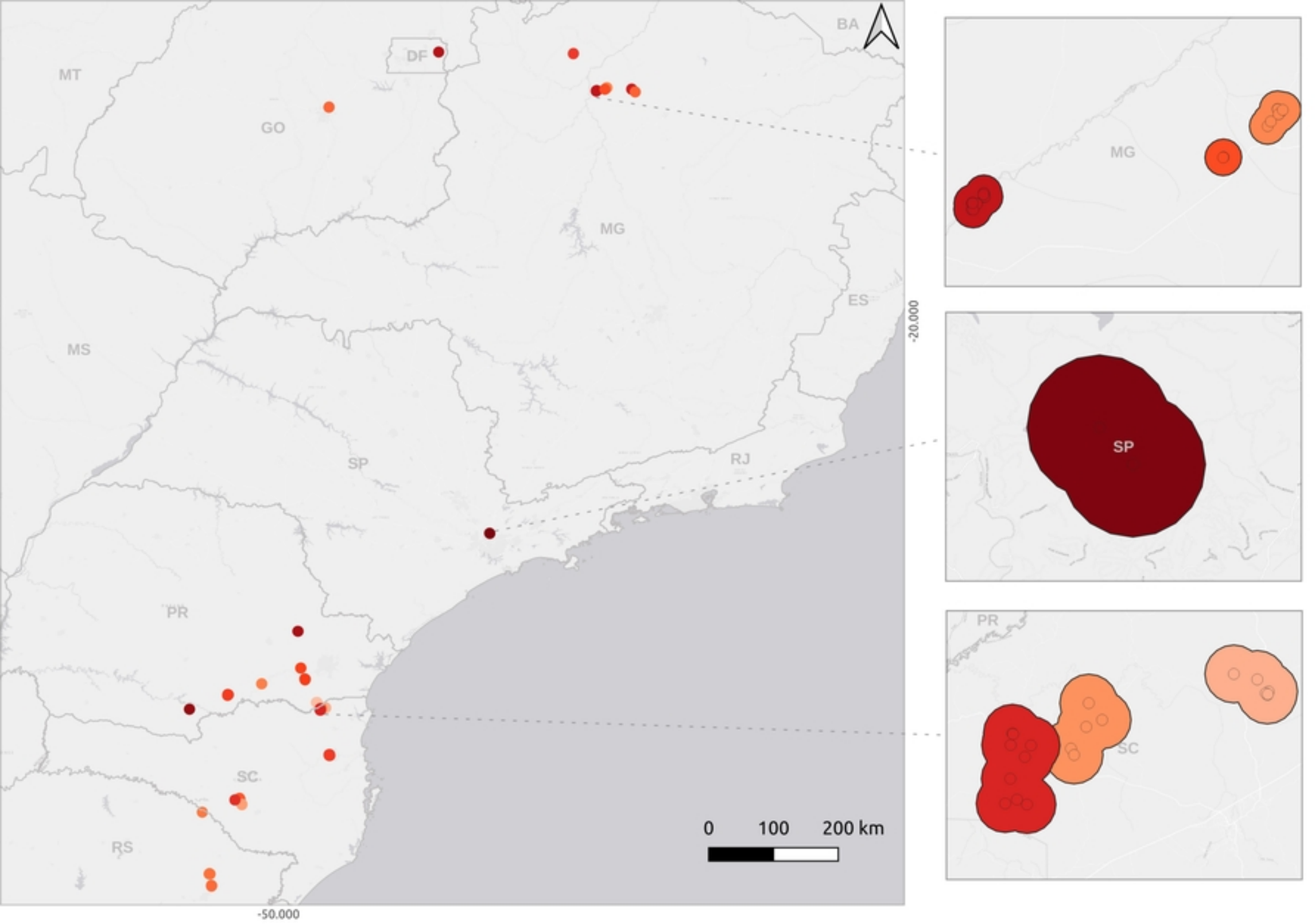

Minimize f2
Convergence
Dominated solution
Diversity
Non-dominated solution
Pareto Front
Minimize f1

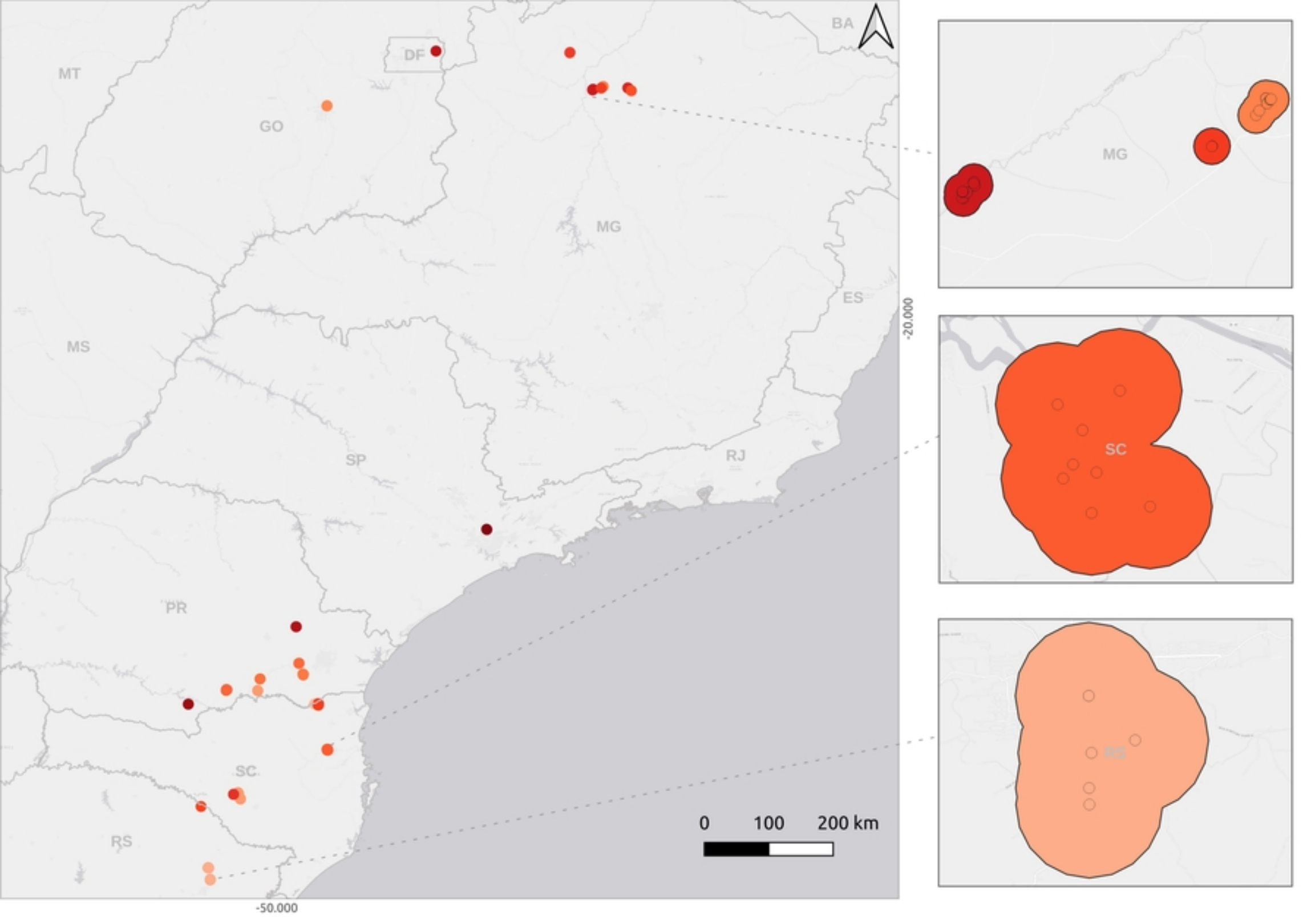

RR
AP
AM
PA
MA
CE
RN
PB
PI
PE
AL
SE
AC
RO
MT
BA
GO
DF
MG
ES
MS
SP
0
500
1,000 km
0.000
-10.000
-20.000
-30.000
-80.000
-70.000
-60.000
-50.000
-40.000
-30.000


RR
AP
AM
PA
MA
CE
RN
PB
PI
PE
AL
SE
AC
RO
TO
BA
MT
GO
DF
MG
ES
MS
SP
PR
RS
0
500
1,000 km
0.000
-10.000
-20.000
-30.000
-80.000
-70.000
-60.000
-50.000
-40.000
-30.000


Pareto Front
Solutions
Pareto Front
Min f_2
Max f_1
0.0
0.5
1.0
1.5
2.0
2.5
3.0
1.0
1.5
2.0
2.5
3.0
3.5
1
2
3
4
5
6
7
8
9
10
12


TO
BA
MT
DF
GO
MG
ES
MS
SP
RJ
PR
SC
RS
0 100 200 km
-50.000
-40.000
-20.000
-30.000
MG
GO
Mafra
SC
Itaiópolis


BA
DF
MT
GO
MG
ES
MS
SP
RJ
PR
SC
RS
0
100
200 km
-20.000
-50.000
MG
SP
PR
SC


BA
DF
MT
GO
MG
ES
MS
SP
RJ
PR
SC
RS
0
100
200 km
-20.000
-50.000
MG
SC
RS