Spatio-Temporal Modeling and Crisis Sensibility Analysis Using Social Media Data

Paulus Robert

2023-2024

Contents

1 Introduction

1.1 Motivation

The main motivation for this master's thesis is to be interdisciplinary, at the crossroads between geography, traditional statistics and machine learning. My goal to show that science is a domain of interconnections, like a web, rather than separate islands of knowledge. As a student with a strong background in practical spatial sciences like Earth sciences, I believe that the most fascinating aspect of statistics is its application to the world of science, rather than studying it in isolation. Space-time data embodies this philosophy, as it serves as a bridge between two distinct fields of knowledge. This thesis includes elements of data analysis, temporal and spatial statistics and machine learning. It aims not to explore each subject in depth, but to build a concrete and naturally flowing connection between diverse areas of research.

1.2 Abstract

This thesis aims to model social media user metrics, focusing on the number of users and crisis sensibility over time and space in the Philippines. We define a crisis event as any occurrence—be it natural, man-made, or a combination thereof—that precipitates a marked and abrupt alteration in the space-time dynamics of population distribution. In our case we will limit ourselves to typhoons the most important source of crisis events in the Philippines. We argue this spatio-temporal shift in population is likely due to crisis-related population displacement or network failure caused by adverse weather conditions over poor infrastructure. Crucially, in this study, the number of social network users distribution serves as a proxy measure for the overall Filipino population distribution. Utilizing social media data as a proxy for main population hinges on their differing natures: social media data offers real-time insights into people movements, capturing the immediate effects of trend, seasonality and crisis events over a population in space and time, unlike static measures such as population density that only provides a general, and often outdated, snapshot based on census data. This static data lacks the immediacy and flexibility required to effectively analyze crisis sensibility in time and space. With the widespread daily use of social media, we argue that these platforms provide an accurate reflection of the overall population. The opportunities offered by social media - crowd sourced data - are described in detail by (Dujardin et al. 2020), who refers to them as Mobile Phone data. Another key advantage of social media data is its production across various distinct geographical locations, which allows our research to be adaptable and reusable in diverse spatio-temporal contexts. For instance, (Jia et al., 2020) and (Maas, 2019) have used similar datasets for their research in the United States and India, respectively, demonstrating the versatility and applicability of this approach in different settings.

This thesis uses datasets provided by the social media company Meta that tracks the number of social media users in various parts of the Philippines, updating the count every eight hours. Those datasets are marked by strong spatial and temporal auto-correlation. These conditions pose both challenges and opportunities for sophisticated analytical approaches. The main objective of this thesis is to develop and evaluate data analysis and modeling techniques that account for spatial and temporal auto-correlations. This involves predicting social media user count in new times, new locations or both irrespective of potential crisis-events, and subsequently, to use these spatio-temporal forecasts as the foundation for estimating population sensibility to crisis events. The space-time models studied in this thesis aspire to balance simplicity in their covariates with the predictive power offered by spatio-temporally correlated data-points.

This thesis explores the convergence of distinct spatio-temporal modeling techniques from diverse research

domains, and the necessity to synergies those models into a unified analytical framework, showcasing and underlining the importance of interdisciplinary innovation in spatio-temporal data analysis and science in general[1].

1.3 Thesis structure

Our research is structured into three distinct sections: space-time data analysis and data preprocessing, space-time prediction of social media user count and space-time prediction of crisis event sensibility. Throughout this thesis, we will explore the notion that space-time problems can be decomposed into separate spatial and temporal components. Indeed most authors whether they come from econometrics (Hafner, 2020), (Beenstock & Felsenstein, 2007) or geostatistics (Bogaert, 1996), (Gneiting et al., 2006) argue that this decomposition is crucial for addressing joint space-time issues effectively.

Our thesis begins with a comprehensive definition and understanding of our dataset in spatial, temporal, and spatio-temporal contexts. We then proceed to the crucial step of data preprocessing, where we address and resolve the issues identified in the initial section as efficiently and statistically accurately as possible. This step ensures that the data is well-prepared and suitable for the subsequent modeling techniques. We will introduce measurements of spatial, temporal, and spatio-temporal autocorrelation, explaining why these measurements are the true bread and butter of autocorrelated data modeling and prediction. This foundational step ensures that the reader understands the presented dataset and its associated challenges setting the stage for effective modeling and prediction.

Building upon those steps, we proceed to define and apply spatio-temporal models that leverage established temporal and spatial modeling frameworks alongside an innovative probabilistic machine learning approach. This section paves the way for the practical application of these models to forecast social media user density across different locations and timeframes. For this purpose, we employ weakly stationary models, including box jenkins models rooted in econometrics (Fischer & Getis, 2010) and spatial interpolation techniques such as Kriging (Cressie, 1993) and its subsequent adaptations for spatio-temporal analysis (Cressie, 2019; Pebesma, 2023; Bogaert, 1996).

The final section of our thesis focuses on modeling crisis sensibility by leveraging our social media user count forecasts in the previous section and percent-change in the number of social media user relative to a baseline. By integrating these two components, we build a space-time dynamic crisis sensibility index (CSI), offering a robust and comprehensive tool for understanding and anticipating the impacts of typhoons in the Philippines. This index represents a potential end-use case of the developed modeling approach, demonstrating its practical application in assessing and managing societal risks in real world scenarios.

Another, though not less accurate, way to view the structure of this thesis is as being divided into three distinct sections: data analysis, statistics, and machine learning. Visual flow diagrams detailing the structure of this thesis can be found at the end, at figures 65, 66, 67. These are provided to help the reader navigate the content with greater ease.

1.4 Framework and notation for space-time Data

In this thesis, spatial data will be categorized into three types: point, polygon - geometries - and raster. Each representation has unique structures and modeling approaches, necessitating distinct notations for clarity.

[1]This idea of interconnection is found in a lot of different domains of science be it natural science (Capra, 1996), social science (Haraway, D. J. 2016) or even mathematics with its rich network theory

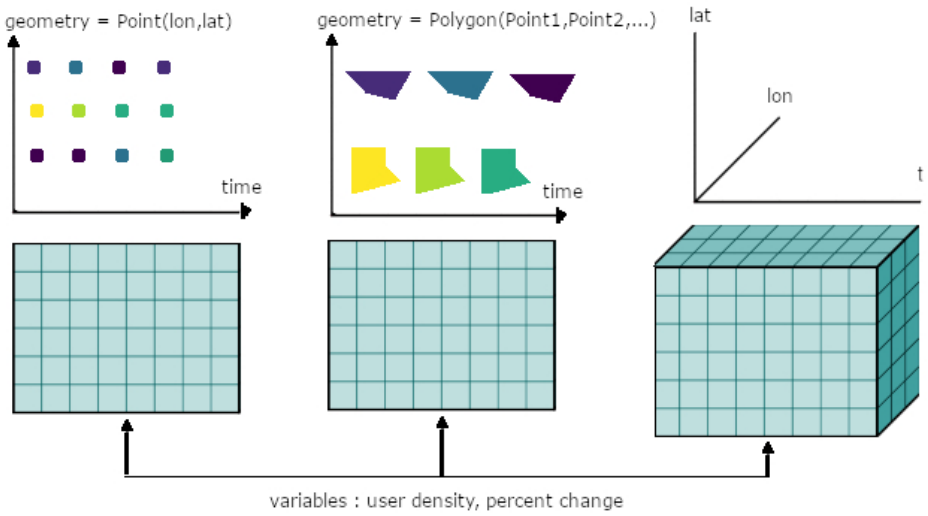


Figure 1: A visualization of the point, polygon, and raster data cubes with viridis colormap.

All spatial data will include a temporal dimension, thus we term the resulting spatio-temporal datasets as "data cubes".[2] A data cube is a multivariate dataframe with perpendicular axes acting as dimensions. Our three primary data structures are illustrated in Figure 1. Note that point and polygon data cubes are not true data cubes but data arrays as they aggregate spatial information directly into geometric (GIS) objects resulting in one less dimension for their global data structure.

Let us define a finite domain D over space and a finite domain T over time with $D \subseteq R^d$ and $T \subseteq R^1$. A space-time (point) location is defined as $(\boldsymbol{s}, t)$ and belongs to $D \times T$. $\mathcal{Z}$ is a space-time random field or a set of random variables defined as $\mathcal{Z} \equiv \left\{Z(\boldsymbol{s}, t) : (\boldsymbol{s}, t) \in D \times T, Z(\boldsymbol{s}, t) \in R^1\right\}$. In our datasets we only observe a finite set of random variables $Z(\boldsymbol{s}_i, t_j)$, with $i = 1, ..., I$ and $j = 1, ..., J$. An observation is defined as $z(\boldsymbol{s}_i, t_j)$, a realisation of $Z(\boldsymbol{s}_i, t_j)$.

A space-time polygon is defined by a set of spatial points and a time coordinate $(\mathbf{S}, t)$. Therefore an observation is defined as $z(\mathbf{S}_i, t_j)$.

A space-time raster cell defined by a spatial grid cell $\mathbf{g}$ and a time coordinate is represented as $(\mathbf{g}, t)$. Unlike a point, the grid of cell values must be regular in space and time. The regularity condition for a grid can be expressed as follows.

$$\|\mathbf{g}_{i+1} - \mathbf{g}_i\| = \Delta x \quad \forall i$$

$$\|t_{i+1} - t_j\| = \Delta t \quad \forall j$$

This regularity condition does not hold true for space-time point locations. Points could be missing in space, in time or both. Vector data and visualization is another (important) method to map spatial information, and there is a rich litterature[3] associated to it but this thesis will not cover it.

[2] Some practitioners in database theory also refer to these as cuboids (Chavalier et al., 2016); we use the term data cube to align with common terminology in GIS literature.

[3] The study of landmasses instability in Geology and wind direction for telemetry related data jut to cite examples in natural sciences.

2 Space-time data analysis

2.1 Data source

This thesis utilizes the *Data for Good* Meta (n.d.) service from Meta[4]. The data for good service describes itself as as a service which provides "real time data that can improve how we respond to real world crises", it requires data sharing agreements with Meta and is specifically engineered to monitor Meta user locations via activated location services on electronic devices, with data points collected at eight-hour intervals across locations during crisis events, as such the dataset is considered to be crowd-sourced. A key aspect of this study is that Meta considered COVID-19 to be a continuous crisis, providing us with comprehensive data spanning two years, from 2021 to 2022. This capacity to trace spatially the number of social media - in this case Meta - users before, during, and after significant crisis events, presents a unique opportunity to analyze the spatial and temporal dynamics of the Filipino population. Spatially, the data is organized through a quadkey system, adhering to the Bing Maps Tile System convention (Microsoft, n.d.), wherein each 13-digit quadkey denotes a tile with a 19.11 m^2 resolution. Figure 2 illustrates the system. This methodical arrangement results in a comprehensive grid, represented by Meta as a collection of points. Each point is the geometric center of a shape, specifically marking the center of a quadkey tile. Due to calculation efficiencies and/or system failures, the actual resolution of quadkeys varies in space and time. This variability results in a point grid that is not regular in either dimension, leading to empty spots in both space and time. Consequently, this irregularity makes it challenging to represent the original dataset as a raster data cube. This limitation will be addressed in later sections.

In this thesis, we exclude locations outside the Philippines national boundaries and those not on the mainland, including areas representing transportation and fishing activities at sea. To capture a comprehensive picture of pre- and post-crisis event dynamics, our analysis timeframe is divided into two sections. The first, pre-crisis section, covers a 1 month typhoon-free period from March 1, 2022, to March 31, 2022.

The second, in-crisis section, encompasses a series typhoons from 2020 to 2022. Those distinct spatial and temporal settings allow us to measure crisis events of varying intensity. Figure 3 illustrates the path as well as the intensity of the main typhoon events between 2020 and 2022, this thesis will focus on some of those typhoons as explained in section 7.

2.2 Variables definition

Let the variable *n-users*, represents the count of Meta users at a given location and a given time for a space-time random process $\mathcal{Z}$. The variable *n-users* is therefore the vector of all observations of interest in the space-time domain :

$$\boldsymbol{z} = (z(s_1, t_1), z(s_2, t_1), \ldots, z(s_i, t_1), \ldots, z(s_1, t_2), \ldots, z(s_i, t_j))' \tag{2.1}$$

Note that in this case, the notation infers that the space grid is regular. If not, the notation should be

$$\boldsymbol{z} = (z(s_{11}, t_1), z(s_{21}, t_1), \ldots, z(s_{i1}, t_1), \ldots, z(s_{1j}, t_j), \ldots, z(s_{ji}, t_j))' \tag{2.2}$$

since the number of spatial points would not be regular in time. In our case, the assumption of a regular grid doesn't hold true as some points are not always measured for at every space and time, but we will use the

[4]Meta is still commonly referred to as Facebook

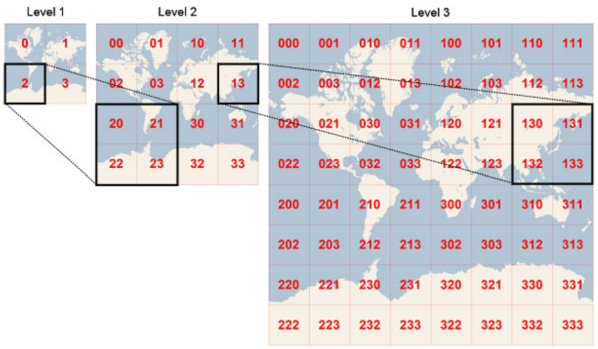


Figure 2: A visualization of the bing-tile system given by (Microsoft, n.d.)

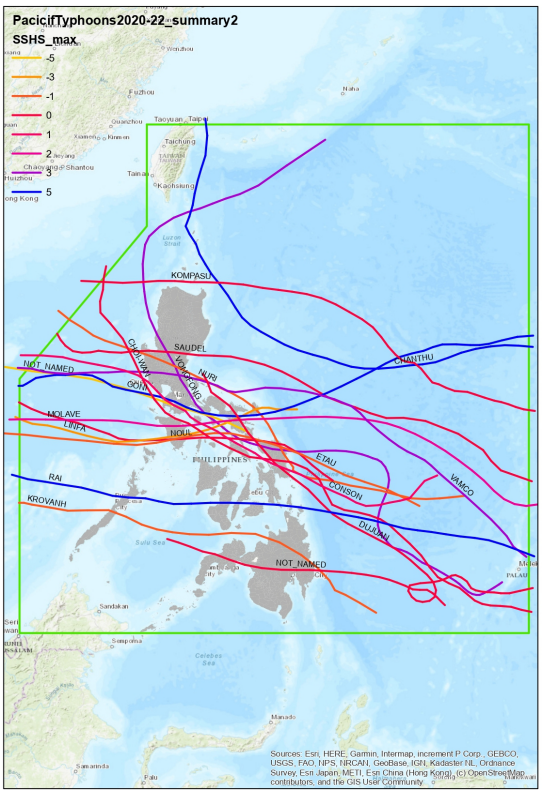


Figure 3: Main typhoons (crisis events) in the Philippines between 2020 and 2022 path and intensity

regular notation for the sake of notation clarity. A similar notational trick is performed in (Cressie, 2023).

The relative feature, *percent-change* $\boldsymbol{p}$, provides a valuable measure of the deviation between n-users and a baseline - also dynamic in space and time - established beforehand by Meta. This variable is particularly useful for analyzing crisis states and assessing recovery status over time and across different regions. Additionally, *date-time*, *ADM-NAME*, and *lat* and *lon* provide temporal and spatial key dimensions t_j and s_i for space-time indexing. The *date-time* feature adhere to the local Philippines time zone (UTC+8), and the x and y coordinates are expressed in longitude and latitude according to the WGS 84 coordinate reference system (CRS) [5]. The datatable representation of the space-time dataset is represented in figure 4.

A crucial observation is that Meta considers any time as reflecting observations from the 8 hours leading up to that time. Therefore, data marked with the time 08:00 actually pertains to observations made during the night (00:00 - 08:00). For clarity, timestamps are categorized as follows: 08:00 represents Nighttime, 16:00 represents Daytime and 00:00 represents Evening. To avoid confusion in the upcoming charts, please note that in some figures, the date may advance to the next day during the evening hours. This occurs because the time reaches 00:00, which is considered to be the the start of the following day.

An additionnal variable we will use later is the variable ***pop*** which represents the *population density* in the Philippines in space and time. Crucially, this variable is based on census data and is stable in time.

[5] A Coordinate Reference System (CRS) provides a framework for defining how spatial data points are positioned within a spatial reference. The WGS 84 CRS is a specific type of Geographic Coordinate System (GCS) that uses a three-dimensional spherical surface to define locations on the Earth.

date_time	n_users	percent_change	geometry	geometry_id	ADM_NAME
2021-12-31 16:00:00	1.30834	-17.2176	POINT..	12	Benguet
2021-12-31 16:00:00	1.54798	71.0262	POINT..	74	Surigao del Norte
2021-12-31 16:00:00	1.94611	67.7097	POINT..	45	Masbate
2021-12-31 16:00:00	2.07087	109.229	POINT..	27	Davao del Sur
2021-12-31 16:00:00	nan	76.8421	POINT..	5	Antique

Figure 4: Presentation of the features in the Meta population dataset

2.3 Data visualization

With a comprehensive understanding of the datasets used in this thesis and their construction, the next crucial step is to visually explore the data before proceeding with any preprocessing and modeling. Visual exploration provides initial insights, helps identify potential anomalies, and reveals underlying patterns within the data. Furthermore, when dealing with space-time data, it is often beneficial to separate the time and space dimensions to discern which one, if any, is more dominant. Some datasets may exhibit strong spatial variance and low temporal variance, while others may show strong temporal variance but low spatial variance. In this section, it is important to note that we will present the dataset without separating the crisis-state. The objective here is simply to evaluate the dataset as it is, without making any modifications.

2.3.1 Spatial analysis

(Cressie, 2023) defines defines two methods to ignore the time dimension of a space-time dataset. One is to fix an arbitrary time point and focus on the observations conditionally on that time, represented as $z(\boldsymbol{s}|t_j)$. The second method is temporal aggregation, where data is aggregated over the entire time dimension to obtain $\bar{z}(\boldsymbol{s})$ an hybrid approach, combines temporal aggregation with fixed time points. An example is daily aggregation where data is aggregated over each day and a specific daily measurement $\bar{z}_{\mathrm{d}}(\boldsymbol{s}|t_j)$ can then be selected. We illustrate both methods for comparison in the accompanying figure. The first approach examines a fixed time on January 1, 2022, from 08:00 to 16:00. The second approach aggregates the time dimension daily and shows a fixed time on January 1, 2022. The final approach aggregates all times over three months by computing the mean across all time steps. To enhance graph readability in regions with low values, we have capped the color map at the 90th percentile. This adjustment prevents the extremely high values in urban centers such as Cebu or Manilla from overshadowing the rest of the country, allowing for a clearer visualization of data in areas with low social media density. Examining Figure 5, we observe that the three figures appear nearly identical on a large scale. This indicates that, social media user density exhibits significantly lower temporal variance compared to spatial variance. Full or daily aggregation does not notably affect the spatial distribution of social media user density, suggesting a form of spatial ergodicity, a concept we will explore in section 4.3 as we analyze the space and time variance structures within this dataset for the number of Meta users.

To convince ourselves that those 3 plots are indeed different we can make the same operations on a spatial subset for example over the central islands of Bohol and Cebu 6. The point geometries and their irregularity in space then becomes apparent. We see with the full aggregation that some points have missing

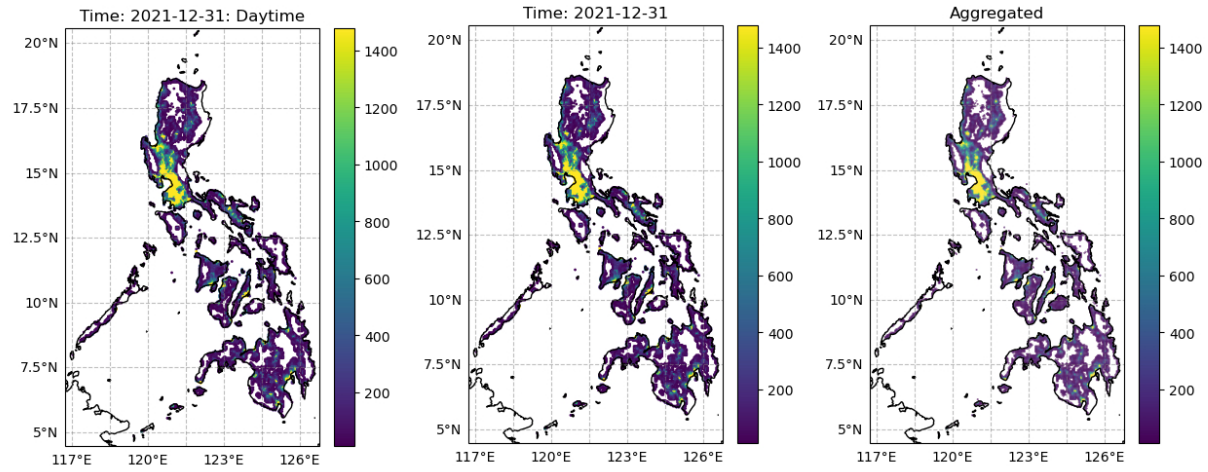


Figure 5: Meta user count over the Philippines (a) 01/01/2022 during the day (b) 01/01/2022 aggregated (c) fully aggregated

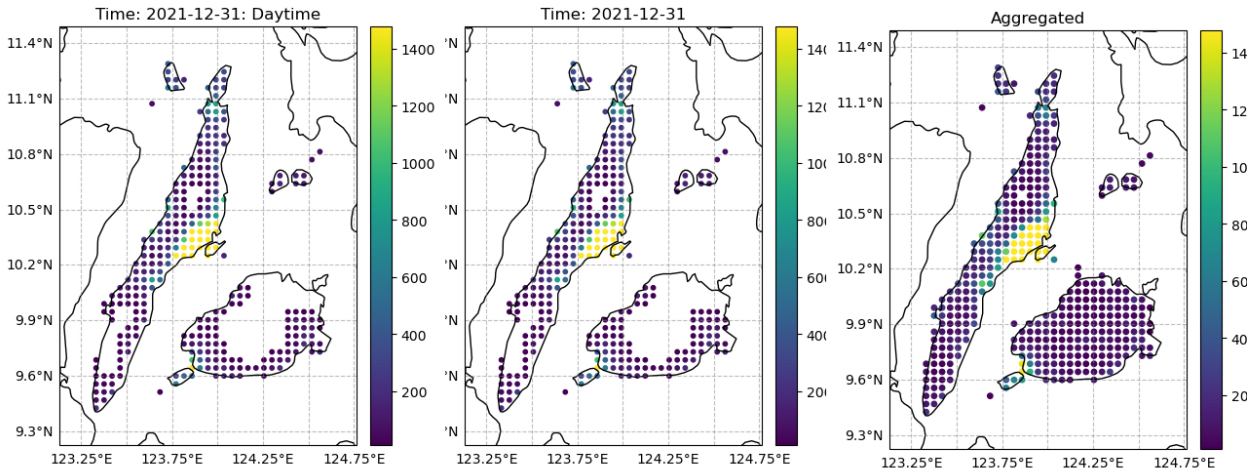


Figure 6: Meta user count over Bohol ans Cebu (a) 01/01/2022 during the day (b) 01/01/2022 aggregated (c) fully aggregated

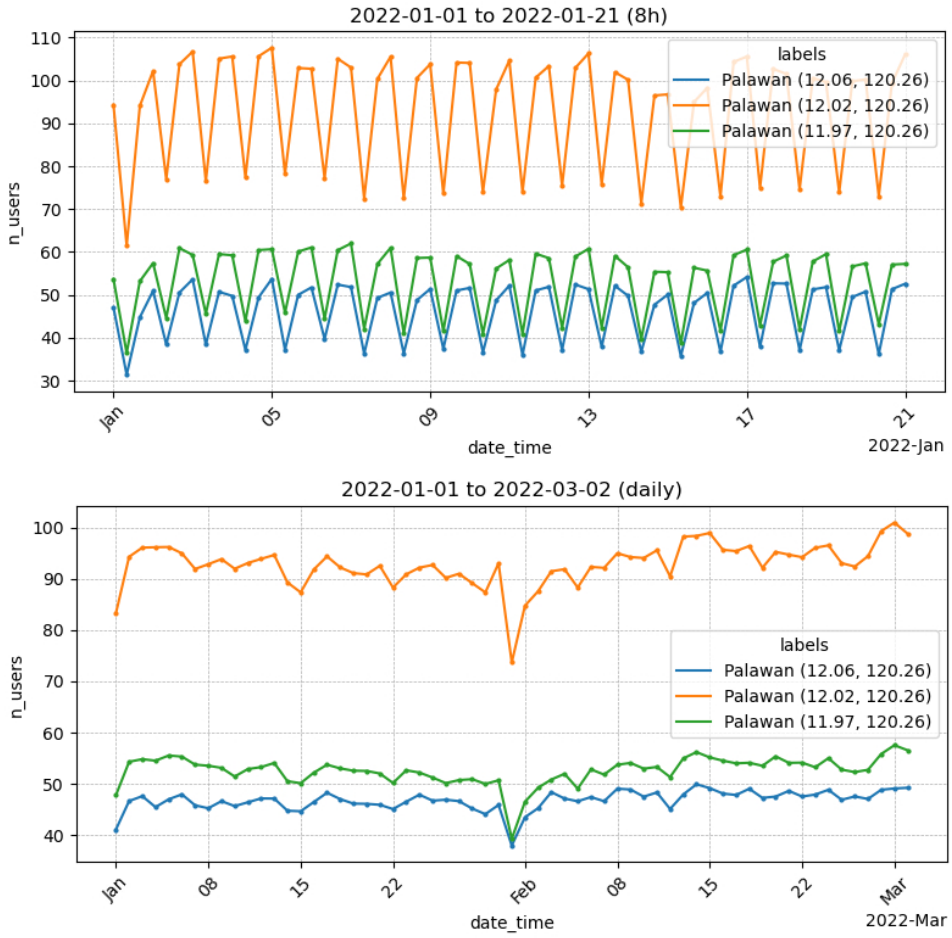


Figure 7: Meta user count over time for three point locations in Palawan (a) Every 8 hours (b) Every day. (Latitudes and longitudes are floored for readability).

values in time and space. We also observe that irrespective of the spatial scale, population density appears to be a primary determinant influencing Meta user count since we can very easily distinguish cities from the countryside such as Manila or Cebu.

2.3.2 Temporal analysis

In this thesis, the vast number of spatial locations (13,000) per time step limits the visualization of fixed spatial points $z(t|\boldsymbol{s_i})$ for detailed analysis. However, this method can still provide insights into the primary sources of temporal variation. The number of Meta users at 8-hour $z(\boldsymbol{s_i}|t)$ and at daily intervals $\bar{z}_{\mathrm{d}}(t|\boldsymbol{s_i})$, for three point locations on the western island of Palawan are illustrated in Fig 7. This "hourly" vs "daily" comparison across different point locations illustrates key temporal behaviors in the dataset. With 8-hour time steps, the data exhibits a distinct seasonal pattern characterized by a cycle that repeats every three time steps, corresponding to 24 hours. This seasonality is rooted in the daily lifestyle rhythms of users, as each 24-hour day is divided into three distinct 8-hour phases - nighttime, daytime, and evening. When examining the same data on a daily basis, the pronounced 24-hour seasonality is suppressed, revealing a less evident pattern that could correspond to a weekly (7-day) seasonality but appears to be weak. Additionally,

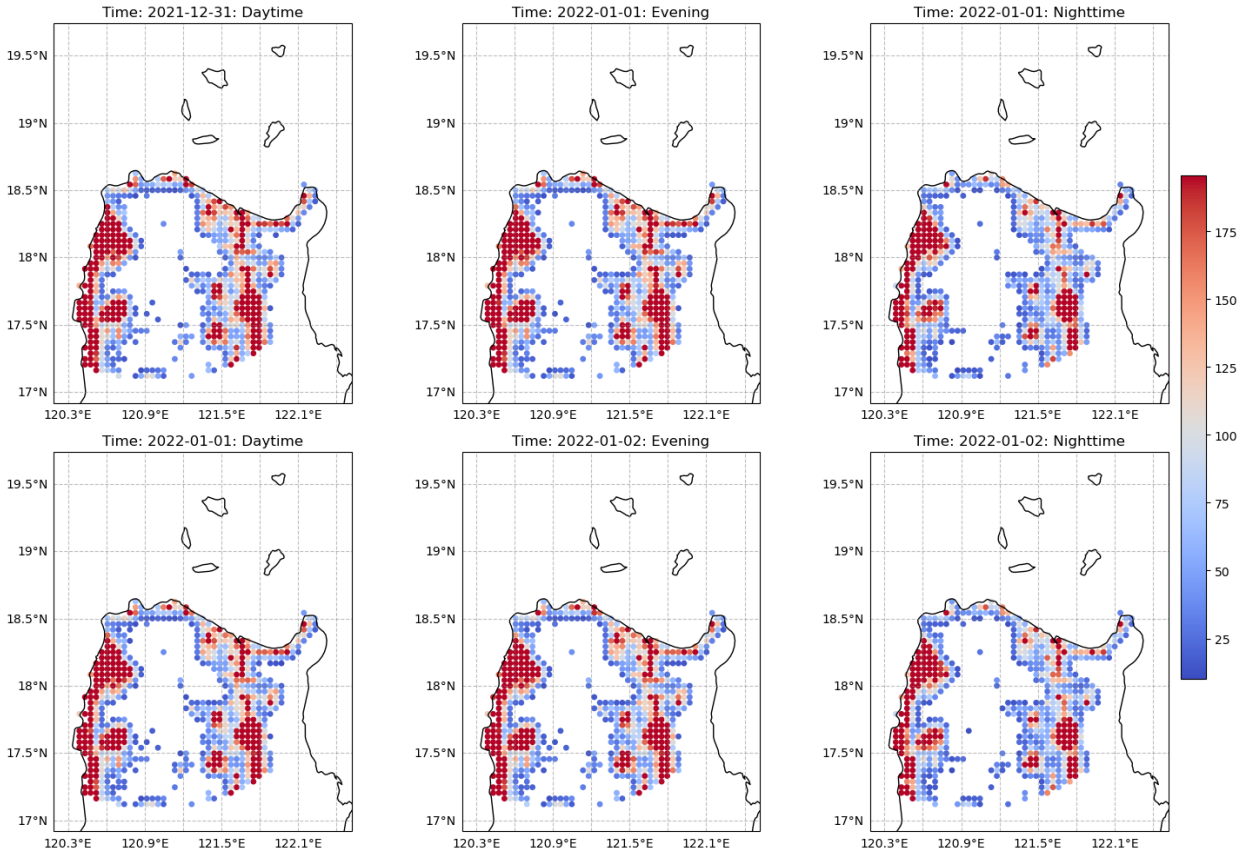


Figure 8: Meta user count every 8 hours around the province of Cagayan. The colormap is tweaked towards low values to see the changes during the nighttime.

the number of users drop at the end of January is likely due to a crisis-event, we can see it as an anomaly in regard to the seasonality assumption.

2.3.3 Space-Time analysis

According to (Cressie, 2023) a space-time dataset can be visualized as either a time series of maps or a cloud of time series.[6] However, due to the extensive size of this dataset—comprising 270 eight-hour time steps across 13,000 spatial locations—creating a complete space-time plot is impractical and would require interactive plotting software. Instead, we present a subset of the full dataset. This approach is demonstrated in Figures 8 and 9, where we focus spatially on the northern Philippines around the province of Cagayan and temporally on 300 randomly selected locations. The nighttime impact on the spatial locations becomes more evident. Additionally, the cloud of time series confirms that a space-time realization observed conditionally on space $z(t|\boldsymbol{s_i})$ is insufficient to describe the parameters of the space-time random field $\mathcal{Z}$. However, examining the time series of maps suggests that this insufficiency does not hold for space-time realizations conditionally on time $z(\boldsymbol{s}|t_j)$[7]. Those observations will have profound impacts in our modelling strategy.

[6]The author also mentions the use of Hovmöller plots, which fix a spatial coordinate and allow the creation of an image plot representing space-time data. We didn't use them in this thesis to stay coherent in our visualization logic but they are powerful space-time visualisation tools which should not be ignored.

[7]Excluding typhoon crisis events, which would obviously contradict this observation—this will be discussed in greater detail in Section7.

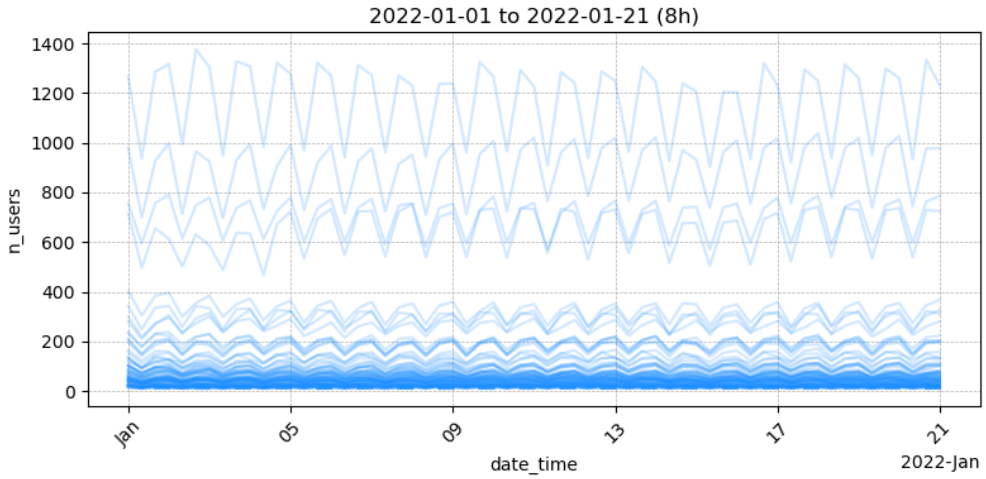


Figure 9: Meta user count for 300 random point locations during 3 months from January to March.

3 Data preprocessing

In this thesis we intend to forecast the number of Meta users $\boldsymbol{z}$ at new time $z(\boldsymbol{s}, t_0)$, new locations $z(\boldsymbol{s_0}, t)$ or both $z(\boldsymbol{s_0}, t_0)$. However in order to do so, it is necessary to tackle the problems we identified in in the analysis section. As we will discuss further in Section 4, the majority of traditional modeling approaches for autocorrelated data rely on the assumption that the random process is weakly stationary. This assumption of stationarity is crucial when considering our pre-processing step.

3.1 Definition of weak stationarity

In order to use auto-correlation into our models, it is clear that we will need to make assumptions on the structure of the spatio-temporal field $\mathcal{Z}$. Indeed, in order to build a model that takes into account auto correlated spatio-temporal neighbour observations and only those neighbours, $\mathrm{Z}(\boldsymbol{s}, t)$ is assumed to be at least weakly stationary. This means that following (Wikle et al., 2019)

$$E(Z(\boldsymbol{s}, t)) = \mu \quad \forall(\boldsymbol{s}, t) \tag{3.1}$$

$$Var(Z(\boldsymbol{s}, t)) = \sigma^2 \quad \forall(\boldsymbol{s}, t) \tag{3.2}$$

$$Cov(Z(\boldsymbol{s}, t), Z(\boldsymbol{s}, t) + k) = r_k \quad \forall(\boldsymbol{s}, t) \tag{3.3}$$

where r_k means the auto-covariance between two data points in space and time is only dependent on the lag k, in other words all random variables $\mathrm{Z}(\boldsymbol{s}, t)$ belonging the the random field $\mathcal{Z}$ share the same mean, variance and their covariance only depends on the spatio-temporal lag k between them but not on space and time itself. Naturally, those assumptions on space and time can be simplified on a temporal random field $\mathcal{Z}_T \equiv \left\{Z_T(t) : (t) \in T, Z_T(t) \in R^1\right\}$ or on a spatial random field $\mathcal{Z}_S \equiv \left\{Z_S(\boldsymbol{s}) : (\boldsymbol{s}) \in D, Z_S(\boldsymbol{s}) \in R^1\right\}$. In this case the stationarity conditions are the same but the covariance only depends on the time lag τ between two random variables or the spatial lag $\boldsymbol{h}$ between two random variables respectively. A lag is a measure of separation between related data points. For a purely temporal setting, τ is easy to define because the input space is 1 dimensional, it is the delay between two events in time and can be defined as the absolute difference

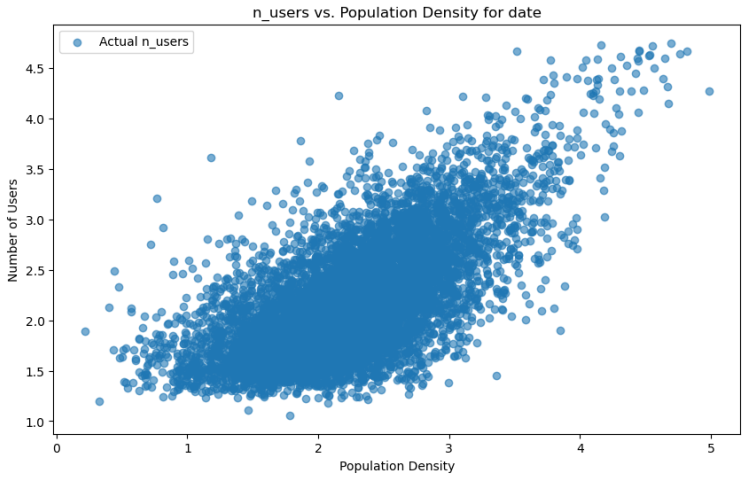


Figure 10: This scatter plot visualizes the relationship between the logarithm of Meta user counts and the logarithm of population density, showcasing a distinct positive correlation.

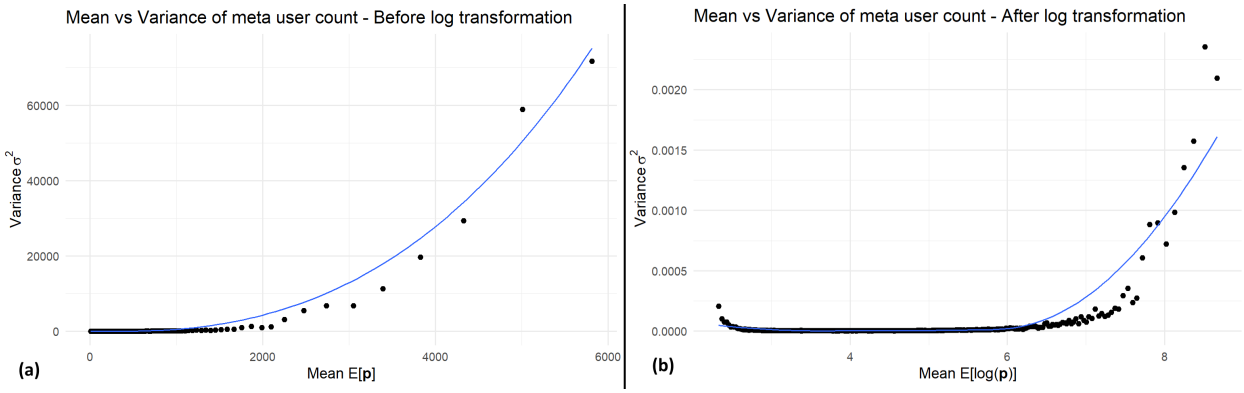


Figure 11: (a) The initial graph shows the variance increasing with the square of the mean, indicating heteroscedasticity, with higher user counts having greater variability (b) The post-log transform graph reveals a more uniform mean-variance relationship, greatly reducing heteroscedasticity and making the interaction more consistent across the data.

between two time points $|t1 - t2|$. In a higher dimensional input space, such as as the spatial setting, $\mathbf{h}$ is more complex. It refers to the distance or separation between two points in space but this separation will vary depending on the chosen metric for distance. For instance 2 spatial points could be 3 units apart using euclidean distance but 4 units apart using chordal distance[8]. Furthermore, in practice, two data points in space will rarely fall perfectly at a specified lag. Therefore, a buffer zone is necessary to account for slight variations or inaccuracies in their positions. A detailed review on spatial lag and its intricacies will be given in section 5.2.

3.2 Effect of population density and log-transform

In our previous section we were able to show that space time random process $\mathcal{Z}$ are in fact not stationary in space and/or in time. For instance, looking at maps it is very clear that the social media user count exhibits

[8]Chordal distance is the straight-line distance between two points on a sphere, useful for spherical coordinates such as the one found on earth

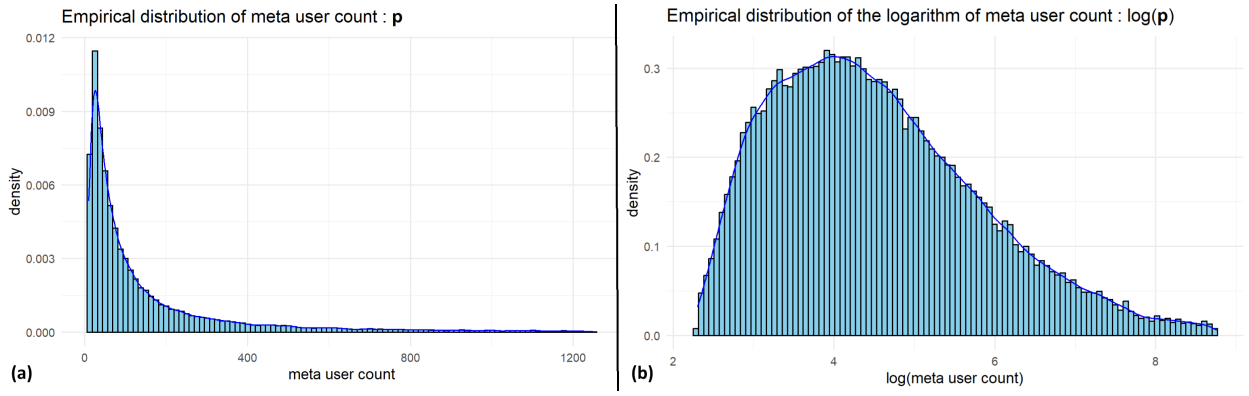


Figure 12: (a) The initial chart shows the empirical distribution of Meta user counts before transformation, with a significant right skew indicating lower user counts and a long tail towards higher values. (b) The chart after applying a logarithmic transformation shows a more symmetrical and interpretable distribution.

a strong correlation with population density. This effect is further illustrated using Figure 10. Consequently the number of Meta users suffers the same shortcomings as population density, with variability among regions with higher user counts potentially escalating to many orders of magnitude greater than those observed in regions with lower user counts. This phenomenon, known as heteroskedasticity, is depicted in Figure 11. In practice we experienced this problem when we defined colormaps and we had to manually tweak them to reflect low variability regions. Looking at figure 11 the variance appears to increase in proportion to the square of the mean, leading to significantly unstable variance for large observations $z(\boldsymbol{s_i}, t_j)$. In fact, the variance seems to grow proportionally to the square of the mean which means a log-transform would be the optimal transformation (Duke University, n.d.). An other crucial observation which supports this log transformation (15) can be found in figure 12. The initial chart (a) reveals that the empirical distribution of Meta user counts prior to transformation is characterized by a significant right skew. This skew indicates a concentration of lower user counts with a long tail extending towards higher values. In contrast, the subsequent chart (b) showcases the empirical distribution following the application of a logarithmic transformation to Meta user counts. This transformation results in a distribution that is more symmetrical and easier to interpret, aligning closer with the normality assumptions required for many statistical analyses (Wackerly et al., 2008). While the log-transform fixes second order stationary issues, it does not fix first order stationary issues. A solution to this problem will be explored during our modeling section.

Naturally, a log transformation can only be applied to positive values. In practice, this is not a problem for our dataset, as it has been artificially truncated. All values for the number of Meta users below or equal to 10 were removed by Meta due to privacy concerns. This truncation ensures that only positive values remain, making the log transformation applicable. However, this missing data represents a limitation, as it results in an under representation of lower user counts.

3.3 Missing values

The problematic of missing values in this dataset is complex and necessitates a detailed explanation because it will influence most of our model results in the following sections. As you can see from the mapping of $\boldsymbol{z}(s_i, t_j)$ in figures 5, and 8 a large portion of data (+-40%) is in fact missing in space in time or both. This missing data problem comes from two confounded but distinct sources. Firstly, Meta imposes a privacy clause which states that any Meta user count equal to or below 10 at a given space time location (s_i, t_j) is

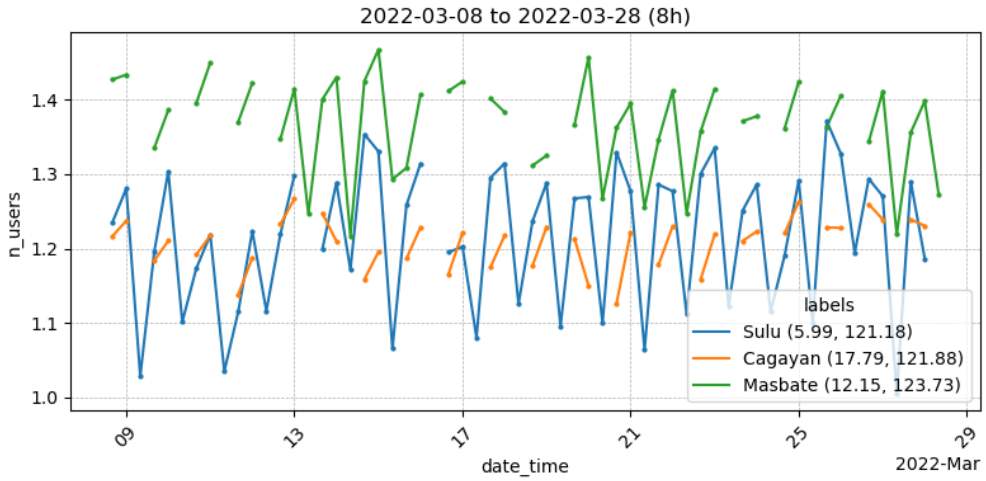


Figure 13: Meta user count over time for three point locations in Bohol every 8 hours. Desîte being above the truncation threshold, the system shows failure at measuring points.

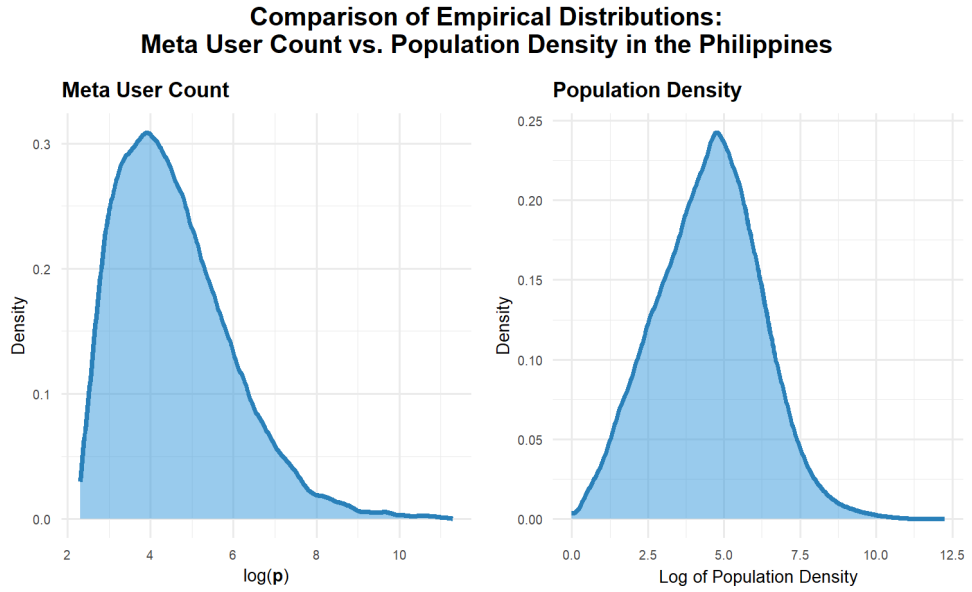


Figure 14: Comparison of the empirical distributions of $\log(\mathbf{z})$ with the logarithm-transformed population density across the Philippines. It reveals that the distribution pattern of population density does not exhibit abrupt spikes at lower density levels.

not reported. In short, the dataset is left truncated; we only observe data for which the social media user count $\mathbf{z}$ satisfies $z(\mathbf{s}_i, t_j) > 10$. This implies that, under typical non-crisis conditions, areas with low user counts (i.e., low population density) will not appear in the dataset. Secondly, missing data can also arise from errors in the system, which is a practical reality. This is particularly evident in Figures 6 and 13.

A critical choice that arises from those observations is to choose between those 2 hypothesis. Understanding this choice is crucial because it influences our replacement and modeling strategy. If we choose to prioritize low population density, we can easily address any missing values by replacing them with 0. While this method introduce an underestimation bias, it should not be too critical in low density locations likely have low user counts initially. Missing values related to system errors could however be seriously underestimated and the dataset will shift overall towards 0. However, if we choose to prioritize system failures, it means that missing data cannot be easily replaced. It is preferable to retain these as missing, maintaining the dataset's irregularity. This approach, however, may result in an overestimation bias, particularly in low population density areas.

Ultimately, the choice belong to us. Looking at 14. The examination of the empirical distribution of $\log(\boldsymbol{n})$, reveals that the probability density function near the threshold is quite low, which does not readily justifies the 40% data loss observed below truncation. This discrepancy could hint at hidden complexities in the data distributions, AKA a mixture between the two assumptions we have established. In practice, in this thesis, we will prefer the second assumption to the first as we argue that replacing missing values blindly is rarely good practice in data science. The most important idea to understand is that this will lead to overestimation bias for models predicting at low population density locations as well as an irregular grid in space and time.

3.4 Additional limitations

For privacy concerns, Meta doesn't only introduce truncation in the dataset, it also adds a constant noise to every measured user count (Maas, 2019). This approach, while enhancing privacy, introduces inherent limitations. Without details on the noise generation process, it's impossible to accurately denoise the observed values. Only Meta has the capability to address this challenge, either by applying denoising techniques directly or by sharing information about the noise generation function. Another significant limitation is the sheer spatial volume of observations, denoted which for point geometries is $270(t) \times 13000(\boldsymbol{s})$. While the abundance of information is valuable, it complicates the application of certain analytical methods. Techniques that require matrix inversions or the computation of covariance matrices become impractically complex with such a large dataset. In this thesis, we will often make predictions within a specific spatial subsets. As discussed in Section 4, utilizing all data points from the entire Philippines to predict a single location is unnecessary and likely excessive. This approach ensures more precise and relevant predictions for targeted areas.

3.5 Spatial aggregation on polygons

To address the spatial volume limitation previously discussed, we can perform spatial aggregation of the data $a(\boldsymbol{S}_i, t_j) = \frac{1}{N} \sum_{i=a}^{b} z(\boldsymbol{s}_i, t_j)$ where the sum is taken over all locations $\boldsymbol{s}_i$ that fall within the boundaries of an administrative boundary with points ranging from a to b and N represents the number of spatial units within the administrative boundary. This effectively transition the point geometry datacube to a polygon geometry datacube. Although this method will obscure spatial patterns during the aggregation process, we believe

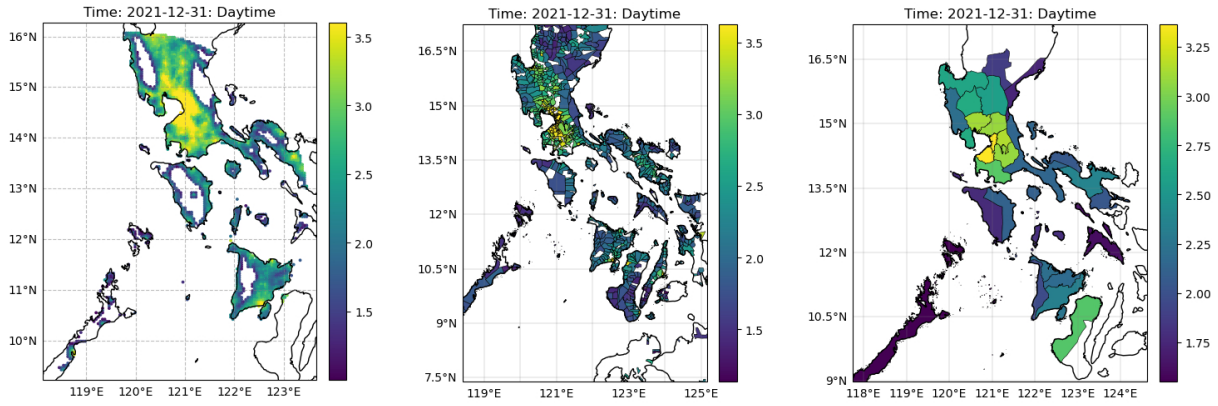


Figure 15: Analysis of the (log) number of meta users through different spatial aggregation levels.

that the dataset's inherent homogeneity minimizes this risk. In other words, because the dataset exhibits relatively consistent patterns across different areas, aggregating data should not eliminate significant spatial trends. This assumption is supported by figures 15. where we observe that the spatial patterns of both the base dataset $\boldsymbol{z}$ and the aggregated dataset $\boldsymbol{a}$ display similarities across different scales. Regions characterized by high and low densities remain distinctly identifiable in both datasets. This observation indicates that aggregating data at administrative level effectively preserves a satisfying amount of the underlying spatial patterns observed in the more granular, point-wise data. Furthermore, aggregating also enriches the data with more meaningful insights. By doing so, each aggregated location represents an entire administrative unit, transforming the previously point-specific user counts into spatially well defined measures of Meta users. This approach elevates the data from representing mere points in space to reflecting the social media usage within defined geographical boundaries. This method enhances the interpretability of the data, allowing for analyses that are more relevant to policy-making, social studies, and targeted interventions at the local level.

Aggregation over polygons comes however with drawbacks which are not negligible. Firstly, aggregation naturally follows the first missing data assumption, leading to potential underestimation and consequently introducing a global underestimation bias for the aggregated value of social media user count. This bias can vary in magnitude depending on the administrative unit and the quantity of missing data within it. For instance, aggregation over the province of Palawan may exhibit more pronounced bias compared to other provinces with different levels of missing data. Additionally, spatial aggregation is invariably influenced by the Modifiable Areal Unit Problem (MAUP) (Andresen, 2021), which impacts statistical results depending on the scale and zoning system used. Although the effect of MAUP may be reduced when calculating means over homogeneous data, it remains an important consideration in spatial analysis.

Naturally, aggregated values over polygons are still sensible to the population density variance issue underlined in sections 3.2. Therefore a log transform is still necessary, the log-transform is visible on figure 15.

4 Autocorrelation definition

In this thesis, our objective is to model a space-time random field $\mathcal{Z}$ with minimal use of covariates. Specifically, we aim to utilize only the space-time autocorrelation between each realization $\boldsymbol{z}(\boldsymbol{s}_i, t_j)$ of the random variables $\boldsymbol{Z}(\boldsymbol{s}_i, t_j)$. Consequently, it is essential to analyze the autocorrelation structures within our dataset,

as these will provide critical information for making accurate modeling decisions. As discussed in Section 2, decomposing space-time problems into their temporal, spatial, and spatiotemporal components is generally advantageous.

4.1 Auto-correlation impact on modeling strategy

In linear models - a simple parametric model - the dependent variable is explained by a linear combination of the independent variable and the error term is defined in this equation :

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma} = \sigma^2 \boldsymbol{I}) \tag{4.1}$$

$\boldsymbol{\varepsilon}$ is an $n \times 1$ vector of error terms assumed to be homoskedastic, meaning it has a constant variance, and a mean of 0, indicating that the error terms are, on average, unbiased. Under those assumptions, the Ordinary Least Squares (OLS) estimator

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y} \tag{4.2}$$

is the best linear unbiased estimator for $\boldsymbol{\beta}$ (Wackerly et al., 2008).

Unfortunately, as illustrated in figures 7 and 5, the assumptions of non-correlation for $\boldsymbol{\Sigma}$ will never hold true in temporal and spatial modeling. In both temporal and spatial analyses, observations are inherently dependent on their predecessors in time or their neighbors in space, leading to correlated errors that violate the independence assumption typically required in standard statistical models. Additionally, the concept of homoscedasticity often does not apply in real-world temporal and spatial datasets. Instead, these datasets exhibit volatility, where variance can fluctuate significantly over time or across different locations. Therefore it is clear that the most important difference with simple models and the biggest difficulty in temporal and spatial modeling is to account for $\boldsymbol{\Sigma}$ in the modeling effort. This involves developing models that can capture the correlated nature and their covariance, essential for making accurate predictions and understanding the underlying processes governing temporal and spatial phenomena.

In this thesis, we focus exclusively on modeling spatial and temporal auto-correlation, proceeding under the assumption that homoscedasticity—constant variance across both space and time—remains valid. Indeed, spatially, we've tackled heteroscedasticity by applying a log transformation, as detailed in details in Section 2.1. This method normalizes variance across spatial locations. Temporally, a parallel transformation has not been deemed necessary, partly because the log transformation over the space-time dataset is expected to alleviate temporal heteroskedasticity as well. Additionally, as illustrated through detailed visual analysis in Section 3, and excluding crisis events, the observations we made for this dataset temporal structure does not strongly support the presence of temporal heteroskedasticity. Thus, we assume constant variance over time and space for the transformed feature $\log(\boldsymbol{z})$. This is obviously not true for the mean, temporally due to seasonality and spatially due to an additional population density effect, but we will focus on this issue in our modelling section.

4.2 Covariance function and auto-correlation measurement

As we have explained above the main component of auto correlated data is their (auto)covariance represented by the $n \times n$ (auto)covariance matrix $\mathbf{\Sigma}$ where

$$\mathbf{\Sigma} = \begin{pmatrix} \mathrm{C}(x_1, x_1) & \mathrm{C}(x_1, x_2) & \cdots & \mathrm{C}(x_1, x_n) \\ \mathrm{C}(x_2, x_1) & \mathrm{C}(x_2, x_2) & \cdots & \mathrm{C}(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{C}(x_n, x_1) & \mathrm{C}(x_n, x_2) & \cdots & \mathrm{C}(x_n, x_n) \end{pmatrix}$$

A covariance function, also known as a kernel function, is a function that quantifies the degree to which two random variables change together. Formally, for two inputs x_1 and x_2 the covariance function $C(x_1, x_2)$ returns the covariance between these two values. For a function $C(x_1, x_2)$ to be a valid covariance function, it must satisfy certain properties the one of which are the most important:

- **Symmetry**: The covariance function must be symmetric, meaning $C(x_1, x_2) = C(x_2, x_1)$. This property reflects the fact that covariance between x_1 and x_2 is the same as the covariance between x_2 and x_1.
- **Non-negative Definiteness**: For any finite set of points $x_1, x_2, \ldots, x_n$, the covariance matrix $\mathbf{\Sigma}$ defined by $\Sigma_{ij} = C(x_i, x_j)$ must be positive semi-definite.

Naturally due to those constraints, a covariance function is not trivial to estimate ! When dealing with a weakly stationary spatial or temporal random field $\mathcal{Z}_t$ or $\mathcal{Z}_{\mathbf{s}}$, the autocovariance function between any pair of observations is only dependent on the lag between those points. Given two time locations and a time lag τ or two space locations and a space lag $\mathbf{h}$,

$$[9] C(z_t(t), z_t(t+\tau)) = C(\tau) = f(|t1 - t2|) \quad \text{and} \quad C(z_{\mathbf{s}}(\mathbf{s}), z_{\mathbf{s}}(\mathbf{s}+\mathbf{h})) = C(\mathbf{h}) = f(d) \tag{4.3}$$

With d : a user defined spatial distance.

It is possible to compute the sample auto-covariance function for a given lag by finding the sample auto-correlation or the sample semivariance at that lag. Auto-correlation is used in temporal statistics while semivariance is often preferred in spatial statistics for reasons we will explain in later sections.

The autocorrelation at lag τ is defined as:

$$\rho(\tau) = \frac{C(\tau)}{C(0)} \tag{4.4}$$

where $C(0)$ is the sample variance σ^2 of the time series (since $\tau = 0$). The sample estimate of the autocorrelation function is given by:

$$\hat{\rho}(\tau) = \frac{\frac{1}{N(\tau)} \sum_{t=1}^{N-\tau} \left(z_t(t) - \bar{z_t}\right) \left(z_t(t+\tau) - \bar{z_t}\right)}{\frac{1}{N} \sum_{t=1}^{N} \left(z_t(t) - \bar{z_t}\right)^2} \tag{4.5}$$

[9] In this thesis, we deliberately choose not to differentiate between C_τ and $C_{\mathbf{h}}$ and $C_{(\mathbf{h},\tau)}$ in practice to maintain notational clarity. We believe that the context provided by the lag is sufficient to convey the nature of the covariance function. This approach extends similarly to γ, ρ and their space-time counterparts.

The semivariance at lag $\mathbf{h}$ is defined as:

$$\gamma(\mathbf{h}) = \frac{1}{2}\mathbf{E}[(z_{\mathbf{s}}(\mathbf{s}) - z_{\mathbf{s}}(\mathbf{s}+\mathbf{h})^2] \tag{4.6}$$

The sample estimate of the semivariance is given by:

$$\hat{\gamma}(\mathbf{h}) = \frac{1}{2N(\mathbf{h})}\sum_{i=1}^{N(\mathbf{h})}\left(z_{\mathbf{s}}(\mathbf{s}_i) - z_{\mathbf{s}}(\mathbf{s}_i+\mathbf{h})\right)^2 \tag{4.7}$$

In practice, the semivariance can also be directly related to the autocorrelation function and the autocovariance function since one can show that [10]:

$$\gamma(\mathbf{h}) = C(0) - C(\mathbf{h}) \tag{4.8}$$

This set of equations or related forms can be found in (Bogaert, 1996) and (Cressie 2021). The charts displaying the estimates of autocorrelations and semivariance for a set of lags τ or $\mathbf{h}$ are referred to as correlograms and variograms, respectively. Examples of these charts can be found in figures 16 and 31.

By fitting a theoretical curve to these charts and under the assumption of weak stationarity, one can obtain an empirical estimator of the autocovariance function for any given lag using equations 4.4 and 4.8. This is precisely why, the weak stationarity assumption is necessary, whithout it one can not use those estimates to approximate the autocovariance function.

In practice, extending the sample estimator to $\mathbf{h}$ or $\tau \to \infty$ is infeasible due to the finite space/time domain of a dataset. However, this limitation is not critical because the covariance function is generally assumed to decrease with increasing lag. This assumption is based on the very scientific principle that nearby observations tend to be more similar to each other, and as the distance between observations increases, they become less correlated. This principle holds true for both spatial and temporal analyses and is known as Tobler's first law (of geography)(Waters, 2017). In practice this phenomenon is observed in figures 4.4 and 4.8. We observe that the autocorrelation drops to zero at a certain lag, while the semivariance reaches a plateau, known as its sill, where it no longer increases. These two observations are the same phenomenon, interpreted through different analytical methods.

Earlier in this paragraph, we subtly referred to autocovariance functions as kernels. This terminology reflects a profound connection between two distinct concepts, which enables us to extend beyond the weak stationarity assumption previously defined. This connection will be explored in much greater detail in Section 6. However, before delving into that, our modeling efforts going forward should always consider a weakly stationary random process for our measured space-time variables.

4.3 Space-time autocovariance function

So far we have defined autocovariance $C(\mathbf{h})$ and $C(\tau)$ for spatial or temporal random processes. However, we have not defined the autocovariance of a space-time random process which should depend on both $\mathbf{h}$ and τ.

A natural extension of both the space and time methods would be to consider a space-time lag $(\mathbf{h}, \tau)$ and a space-time autocovariance $C(\mathbf{h}, \tau)$. The lag or the separation between two space-time points could simply

[10] This relationship is proved in the Appendix A2

be related to an extension of the spatial distance in a 3 dimensional input space for instance

$$C(\mathbf{h},\tau) = C(z(\mathbf{s},\tau), z(\mathbf{s}+\mathbf{h}, t+\tau)) = f(\sqrt{||\mathbf{h^2}|| + \tau^2}) \tag{4.9}$$

However, (Bogaert, 1996) argues that this approach has two major drawbacks one conceptual $\mathbf{h}$ and τ having no natural equivalence as they are expressed in different units (a space-time distance in seconds should be different then one in minutes). The second, perhaps more importantly being that such a definition imposes a very specific structure on the space-time covariance. Indeed in practice, it is possible to apply the separability philosophy we used throughout this thesis to the space-time autocovariance function. Under the assumption that random variables $Z(\mathbf{s},t)$ of a space-time random field $\mathcal{Z}$ have the separability property $Z(\mathbf{s},t) = Z(\mathbf{s})Z(\mathbf{t})$, the autocovariance function $C(\mathbf{h},\tau)$ can then be expressed as $C(\mathbf{h},\tau) = C(\mathbf{h})C(\tau)$. This means according to (Flaxman,2015) that the generated space-time covariance matrix $\Sigma_{\mathbf{h},\tau}$ can be defined as the kronecker product between the spatial generated covariance matrix $\Sigma_{\mathbf{s}}$ and the temporal generated covariance matrix Σ_t :

$$\Sigma_{\mathbf{s},t} = \Sigma_{\mathbf{s}} \otimes \Sigma_t \tag{4.10}$$

In practice, this hypothesis will the main driver of section 6 (where we will deal with an innovative probabilistic space-time machine learning framework.

When the random variables $Z(\mathbf{s},t)$ can not be expressed as $Z(\mathbf{s},t) = Z(\mathbf{s})Z(\mathbf{t})$. We can still follow the precepts of the separability hypothesis by following (Bogaert, 1996) method. In that case the space-time autocovariance function is defined as

$$C(\mathbf{h},\tau) = \sigma^2\rho(\mathbf{h})\rho(\tau) \tag{4.11}$$

. This definition will require us to estimate the space-time variance $sigma^2$ as well as the purely spatial and purely temporal autocorrelation functions. This is studied in greater details in section 5.3.1.

5 Modelling in space and time

As we have done for previous sections, to understand space-time modeling, we apply the philosophy of separability. In practice, we have already defined most of the tools necessary for comprehending space-time modeling. However, we have reserved a few key concepts for this section. For space-time modelling we will focus exclusively on non-crisis state data since it is compatible with the stationarity assumption. The data in this section (outside exceptions) will therefore cover the typhoon free period of March 2022.

5.1 Purely temporal models

5.1.1 Base model

Time series modeling is rooted in the world of econometrics. Forgetting about the spatial dimension, the input space of time series data is inherently one-dimensional. This simplicity allows for straightforward discretization of the time axis, as it only involves dividing the continuous timeline into discrete intervals. Unlike multi-dimensional data, where discretization requires managing multiple axes and their interactions, time series data's single-dimensional nature makes it easier to handle. In practice, accounting for temporal autocorrelation in a discrete domain involves representing a temporal data point as a sum of past values, past error terms or both. This discrete representation is essential because it would not be feasible in a continuous

domain, where infinite precision would be required. This discrete approach is in fact foundational in time series under the famous Box-Jenkins methodology mostly known for its autoregressive integrated moving average (ARIMA) models[11]. Following (Brockwell, 2016) and (Shumway & Stoffer, 2011), an ARIMA(p,d,q) model is defined as

$$\phi(B)(1-B)^d X_t = \theta(B)\varepsilon_t \tag{5.1}$$

where:

- $\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p$ is the autoregressive (AR) polynomial of order p,
- $(1-B)^d$ is the differencing operator of order d used to deal with trend in the dataset,
- $\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q$ is the moving average (MA) polynomial of order q,
- B is the backshift operator, $BX_t = X_{t-1}$,
- X_t is the time series value at time t,
- ε is the error term at time t.

Expending an ARIMA(1,1,1) model we get :

$$X_t = (1+\phi_1)X_{t-1} - \phi_1 X_{t-2} + \varepsilon + \theta_1 \varepsilon t - 1 \tag{5.2}$$

What is important is that because ARIMA models use discrete correlated data points and/or error terms in the model, they essentially displace the modelling effort unto the mean and not the covariance. The vector of residuals ε should therefore follow the classical normality assumption (eq 4.1).

$$z_{t+1} = (1+\phi_1)z_t - \phi_1 z_{t-1} + \varepsilon + \theta_1 \varepsilon t \tag{5.3}$$

5.1.2 ACF and PACF

In Section 4.2, we discussed that a useful method for determining the autocovariance of a random process is to plot the Autocorrelation Function (ACF) or the variogram, and then fit the observed results to a rightly conditioned function . We emphasized that temporal statistics and by extension econometrics typically prefer the ACF over the variogram. This preference arises from the fact that there is a direct relationship between the ACF and an ARIMA(0,0,q) model. Specifically, one can show that the ACF is a robust method for determining the order q of the moving average terms in the model. There is, therefore, a direct relationship between the estimation of the autocovariance function and the degree of the moving average component in a time series model. This relationship is rooted in the fact that both approaches are fundamentally concerned with modeling autocorrelation structures within the data, the only difference being a matter of discretization. Specifically, for a moving average process of order q, the autocovariance function will exhibit non-zero values up to lag q and zero thereafter (we say that the ACF cuts off after q lags).

Moreover, in the context of ARIMA models, it is also crucial to introduce the Partial Autocorrelation Function (PACF). While the ACF helps in identifying the order of the moving average component q, the PACF is instrumental in determining the order of the autoregressive component p. The PACF measures the correlation between observations at lag k, after removing the effects of intermediate lags. In an ARIMA(p,0,0)

[11] A quick refresher on ARIMA models is provided on the Appendix A3

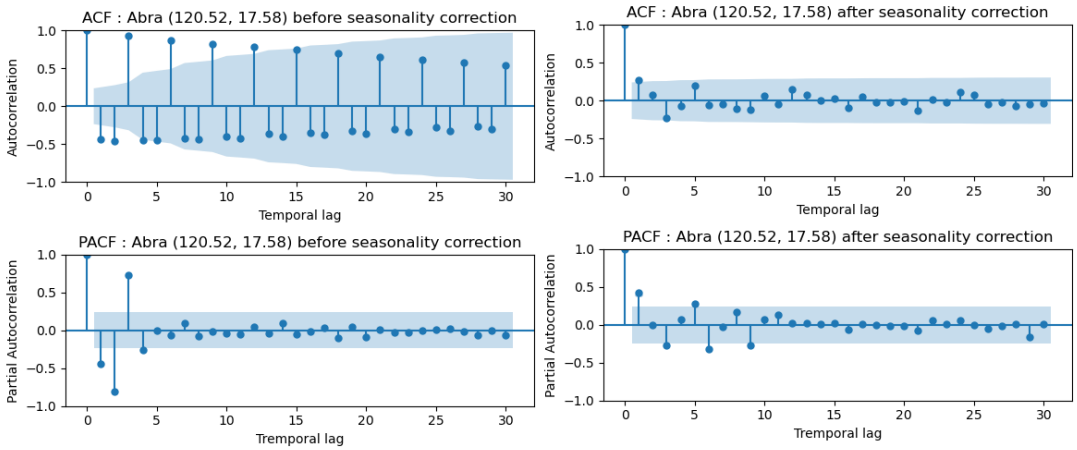


Figure 16: ACF and PACF of Meta user count for a given point in Abra. Blue confiden intervals are built to see if the autocorrelation at lag τ is significantly different to 0. Any value below the interval is considered to be equal to 0.

model, the PACF cuts off after lag p, providing a clear method to identify the number of autoregressive terms to include in the model.

In a more complex ARIMA(p,d,q) model, the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) alone are insufficient to fully determine the orders p and q of the model. This is because, unlike pure MA or AR processes, the ACF and PACF of an ARIMA model do not necessarily exhibit clear-cut cutoffs. Instead, they often exhibit a combination of exponential decay and sinusoidal patterns due to the mixed autoregressive and moving average components (Brockwell, 2016). However, despite these complexities, the ACF and PACF remain valuable diagnostic tools. The ACF can provide preliminary insights into the moving average components, as significant autocorrelations at certain lags suggest the presence of an MA process. Similarly, the PACF can help identify the autoregressive components, with significant partial autocorrelations at certain lags indicating the presence of an AR process.

Figure 16 shows the ACF and PACF of a random point location $\mathbf{z}(t|\mathbf{s}_i)$ in Abra. An evident observation is that seasonality is interfering with the results, which is expected since weak stationarity is not maintained. As discussed earlier in Section 3.2, we addressed the non-stationarity of variance in both space and time using a logarithmic transformation. However, we also anticipated non-stationarity in the temporal mean $\mu(t|\mathbf{s}_i)$. In fact, we illutrated in figure 7 that for 8-hour data, the datapoints exhibited a daily seasonality, as it disappeared after daily aggregation. Notably, we did not observe a temporal trend in the data, indicating that the primary source of non-stationarity is seasonality. Correcting the mean to achieve stationarity is essential for accurately utilizing ACF and PACF charts and for effective ARIMA modeling. When the mean exhibits seasonal patterns, the autocorrelation structures captured by the ACF and PACF are distorted, leading to misleading conclusions about the underlying processes. By removing or adjusting for seasonality, we can ensure that the mean remains constant over time and that second order as well as first order stationarity are respected, which is a fundamental assumption for many time series analysis techniques, including ARIMA modeling.

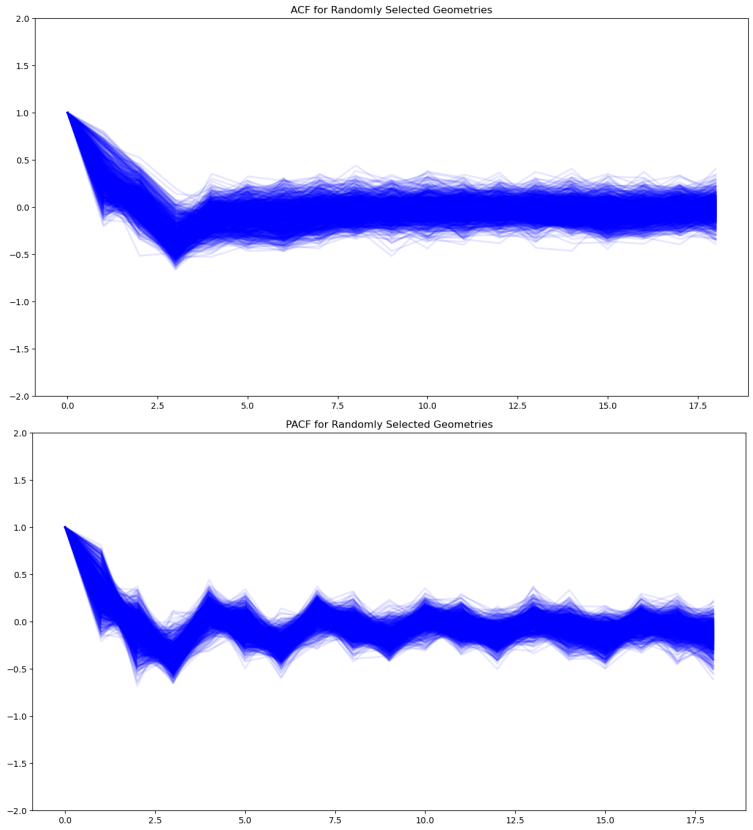


Figure 17: ACF and PACF of 1000 random locations.

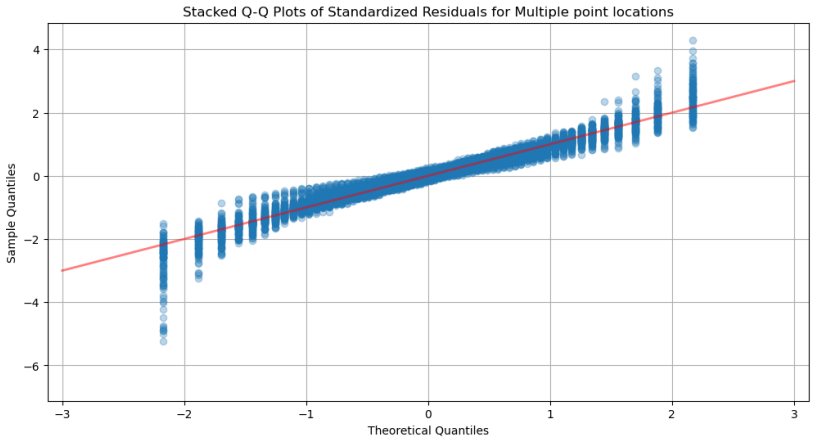


Figure 18: (Standardized) Residuals normality test across multiple locations

5.1.3 Dealing with seasonality : SARIMA models

There are multiple methods to remove seasonality in a time series. In our thesis, we will adhere to the ARIMA framework and employ seasonal differencing. Seasonal differencing can be described using the backshift operator B defined earlier. For a seasonal period s, the seasonal difference of order 1 can be expressed as (Brockwell, 2016):

$$\Delta_s y_t = (1 - B^s) y_t \tag{5.4}$$

The PACF and ACF of Δ_s is illustrated in figure 17 and 16 for all non missing points in the Philippines. The ACF and PACF now both display an exponentially decaying pattern, which is expected of stationary time series and suggests that the data no longer contains deterministic seasonal autocorrelation. By examining the confidence intervals for significance, we observe that both the ACF and PACF exhibit some lags that appear to be significantly different from zero. The autoregressive component however, as illustrated by the PACF, is more pronounced, suggesting that the model for these time series at specific locations will likely include at least one if not serveral autoregressive component $AR(p)$. Unfortunately, despite the application of differencing, a form of stochastic seasonality remains evident, particularly in the ACF plot, where significant autocorrelations persist at seasonal lags. This persistent stochastic seasonality indicates that the data still exhibit seasonal dependencies that are not fully removed by simple differencing. To address this stochastic seasonality, we can model the seasonality as its own ARIMA process. This leads to the formulation of the Seasonal ARIMA (SARIMA) model, which extends the ARIMA model to incorporate seasonal components. The SARIMA model is defined as (von Sachs R., 2024) : SARIMA$(p, d, q)(P, D, Q)_s$

In this model, the seasonal part $(P, D, Q)_s$ explicitly accounts for the seasonal structure in the data. Seasonal differencing of order D with period s is applied to remove the seasonal trend, while seasonal autoregressive (P) and moving average (Q) terms model the seasonal dependencies.

The final SARIMA model can be expressed as follows:

$$\Phi(B^s)\phi(B)(1 - B)^d(1 - B^s)^D y_t = \Theta(B^s)\theta(B)\varepsilon_t \tag{5.5}$$

5.1.4 Parameter estimation and model selection

Numerous techniques exist to guide parameter estimation and model selection in ARIMA modeling frameworks. However, the comprehensive examination of these approaches escapes the scope of this thesis which intends to focus on space-time modeling itself rather then time series data analysis specifically which is an incredibly rich domain of study. In our thesis, we use the SARIMAX model from the *statsmodels* library in Python. The method used by *statsmodels* for parameter estimation is Maximum Likelihood Estimation (MLE), which involves optimizing the likelihood function to estimate the model parameters that best fit the observed data. For model selection, we employ a grid search to identify the model with the smallest Bayesian Information Criterion (BIC) across a subset of 500 random point locations $\mathbf{s}_i$. Our initial guesses for the parameters $(p, d, q)(P, D, Q)$ for the SARIMA framework are based on the patterns observed in the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) from different point locations. We explore the parameter ranges $p \in \{0, 1, 2\}$, $d = \{0, 1\}$, $q \in \{0, 1\}$ for the non-seasonal components, and $P \in \{0, 1\}$, $D \in \{0, 1\}$, $Q \in \{0, 1\}$ for the seasonal components to find the optimal model in terms of BIC for a given time series $\mathbf{z}(t|\mathbf{s}_i)$. The result of our grid search is given in table 1. Uisng this table we determined that we determined that the optimal model for a given time series $\mathbf{z}(t|\boldsymbol{s}_i)$ in term of is the following : SARIMA(1,0,0)(0,1,1). While this model may not have the lowest BIC, it is a well-established

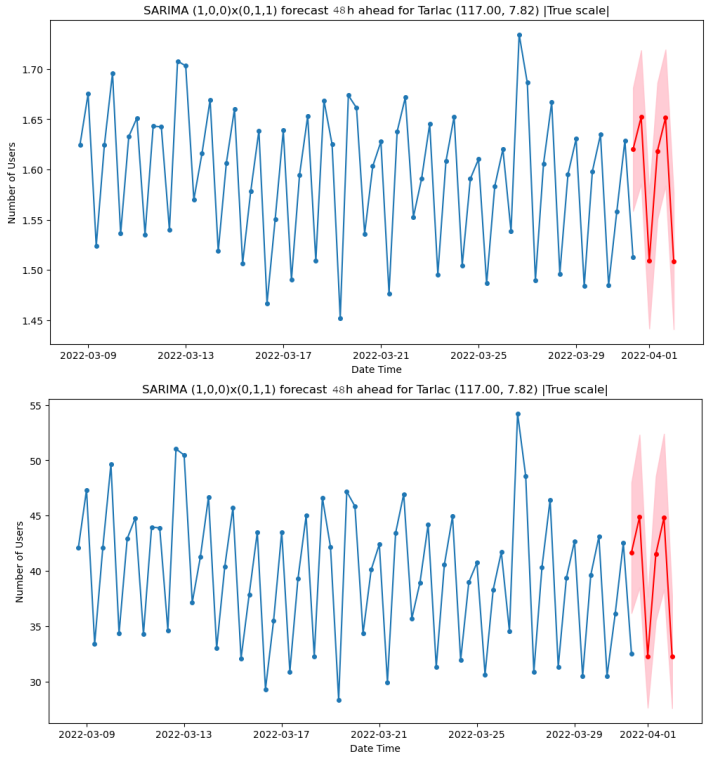


Figure 19: Forecasted [(a) log (b) true] meta user count (24h) for a given point in Tarlac.

practice in time series analysis (Von Sachs, 2024) to take a conservative approach when selecting models. In our case, the reduction in BIC from adding an extra parameter is minimal, suggesting that the simpler model remains a good choice.

(p,d,q)	**(P,D,Q)[3]**	**BIC**
(2,0,0)	(0,1,1)	-435.79
(1,0,0)	(0,1,1)	-435.13
(1,0,0)	(1,1,1)	-433.29
(0,0,2)	(0,1,1)	-430.18
(1,0,2)	(0,1,0)	-429.72

Table 1: Comparison of BIC values for different (p,d,q) and (P,D,Q)[3] configurations

We follow the assumption supported by figure 17 that all other locations follow a similar ARIMA model. Using this assumption we can therefore model and forecast Meta user count at any point locations. If the SARIMA model is well-conditioned, we expect the residuals to be normally distributed across all point locations. This expectation is confirmed in Figure 18, where the normal Q-Q plots display consistent patterns. This observation further supports the suitability of the SARIMA(1,0,0)(0,1,1) model.

5.1.5 Forecasting

Following a grid search process for a small subset of locations, we determined that the optimal model for $\mathbf{z}(t|\mathbf{s}_i)$ in term of BIC is the following : SARIMA(1,0,0)(1,1,1). Consequently, we employ this temporal model across all spatial locations to forecast the next 24 hours in April 2022 using the box jenkins procedure

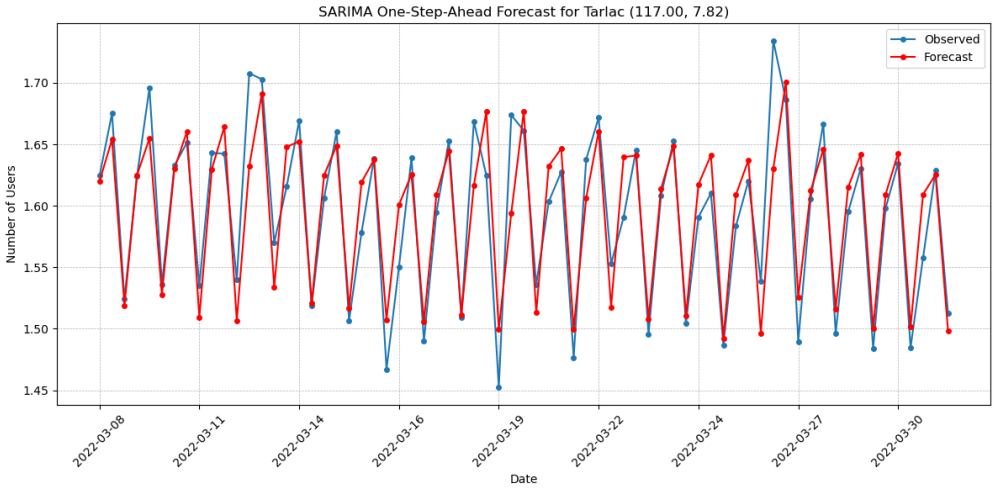


Figure 20: Forcasted VS actual log meta user count at a given point in Tarlac.

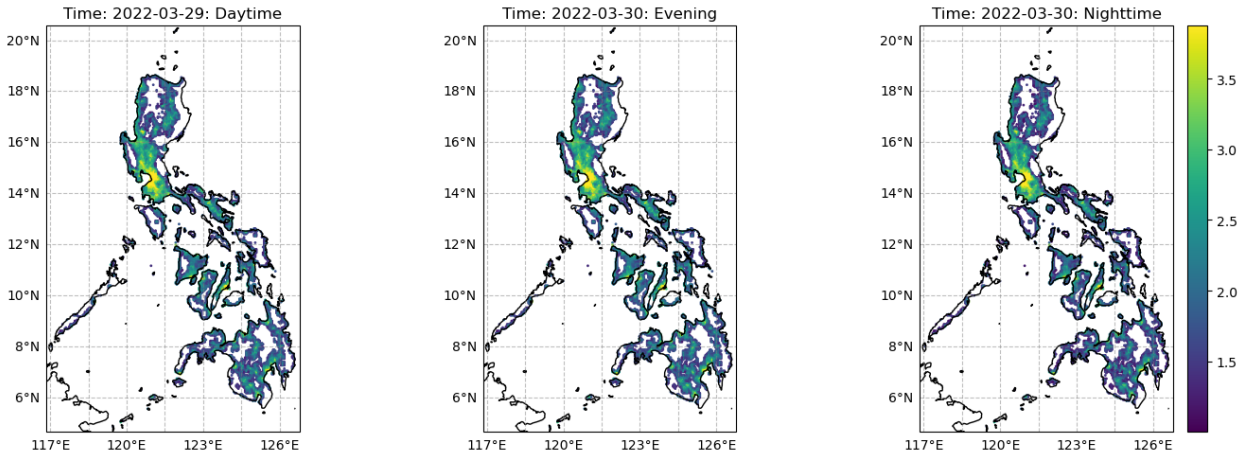


Figure 21: Forecasted log Meta user count in the Philippines for the 01/04/2022.

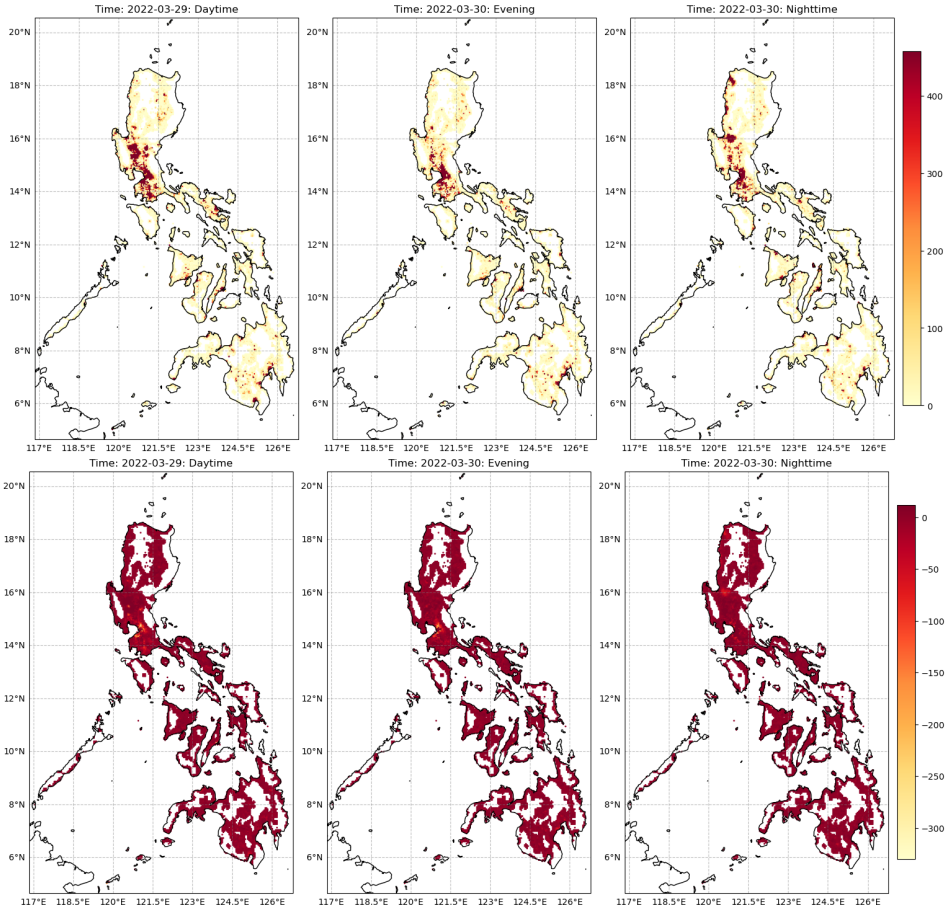


Figure 22: Squared residuals of the purely temporal SARIMA(1,0,0)(0,1,1) model.

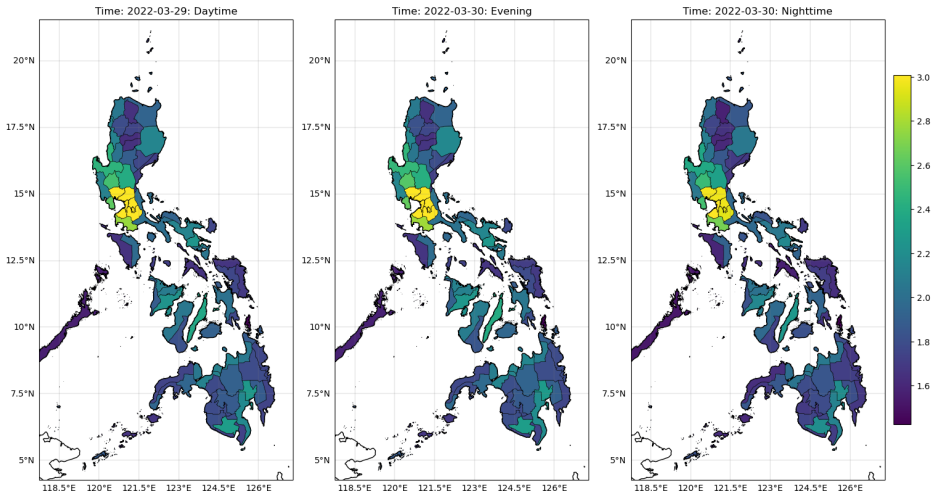


Figure 23: Municipalities forecasting of Meta user count during 24H

described in eq 5.3. The results of our forecasting process are presented for a specific location in Tarlac as well as for the entire Philippines, as shown in figures 19 and 20. As expected, the confidence interval widens over time, indicating increasing uncertainty in long-term predictions. Nevertheless, the model demonstrates a strong capability to capture the seasonal patterns accurately. To validate our model, we divided the original dataset into a training set and a test set. The test set consisted of 48 hours of observed data at the end of March. We used the training set to forecast the end of March and then compared the forecasted values to the test set after reversing the log transformation. We observe that the predicted values closely match the true observed values for the log number of users. Additionally, looking at figure 21, it is clear that the model effectively accounts for the spatial effect of population density, which is not surprising given that each time series has a different mean but a similar model structure and that every time series $\mathbf{z}(t|\mathbf{s}_i)$ was modeled separately.

After transforming the log observations back to their original scale, the analysis of the squared residuals, as shown in figures 22 , offers valuable insights. The squared residuals exhibit spatial and temporal variation, which appears to be correlated with population density. This correlation is not unexpected, given that Figure 12 indicates how the log transformation did not fully resolve the variance issue in areas with very high population density. Figure 22 (b) suggests that the residuals primarily result from an underestimation bias, indicating that the model tends to underestimate the true Meta user count in densely populated areas which is especially true in Manilla. We also remark that squared residuals vary across space and time, but do not show signs of significant degradation over time. This suggests a potential shift in population density over time, possibly due to population movement or migration related to work habits. The residuals are more pronounced in high population density areas, such as urban centers, during daytime hours. As the day progresses into the evening and night, these residuals appear to spread out towards lower population density and peri-urban locations. Regardless of the time of day, there is a consistent correlation between higher population density and increased prediction error.

5.1.6 Missing data limitations

Our proposed framework exhibits some important limitations, the most critical being its inability to handle missing temporal data. Time series with gaps, such as the one shown in Figure 6, cannot be properly

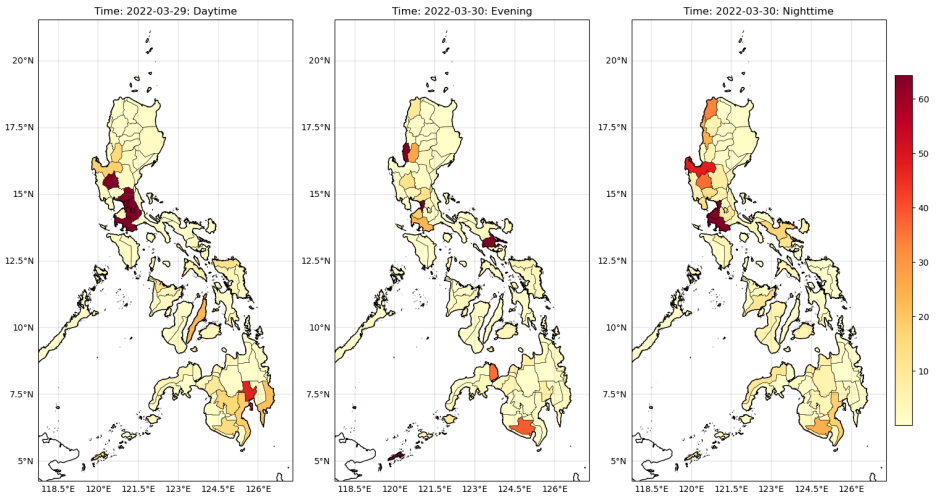


Figure 24: Municipalities forecasting residuals during 24H

differentiated nor can they be modeled using discrete temporal neighbours. In practice, the Box-Jenkins methodology always requires a regular, complete data by design.

Two potential solutions can address this issue. The first is to spatially aggregate the data into polygons, which reduces the number of empty locations but may come at the cost of some prediction accuracy. This approach is particularly advantageous as it also addresses the inherent challenge of large data volume. Predicting with the SARIMA model for each individual location requires making approximately 13,000 predictions—an effort that can be made significantly easier computationally through polygonal aggregation. Figures 23 and 24 illustrate this forecast for municipalities at the end of March and their associated residuals, using a SARIMA(1,0,0)(0,1,1)[3] model.

The results obtained after aggregation are comparable to those for point locations, but the forecasting process becomes significantly more efficient, reducing the analysis from 13,000 time series $\mathbf{z}(t|\mathbf{s}_i)$ to just 81 time series $\mathbf{a}(t|\mathbf{S}_i)$. This aggregation not only streamlines the computation but also tends to produce lower average residuals, as extreme values are smoothed out. However, this loss of granularity means that interpretation at specific point locations becomes impossible, shifting the focus to predictions for entire municipalities instead. While the decision to aggregate ultimately depends on the practitioner's objectives, aggregation is often chosen to simplify both the conclusions[12] and the computational process.

A second solution to the missing data problem involves leveraging the similarity between time locations rather than relying solely on direct temporal neighbors for modeling. This approach shifts from a discrete model that primarily focuses on modeling the mean to a continuous model that emphasizes modeling the covariance structure. By identifying and utilizing the autocovariance, this method offers a robust framework for handling empty time series. The detailed exploration of this approach will be the primary focus in Sections 5.2 and 5.3, where spatial statistics are discussed, and Section 6, which covers Gaussian processes and probabilistic machine learning.

[12]Observations on administrative entities usually offer a stronger basis for policy-making, as they aggregate data in a way that reflects the organization of real-world governance and resource allocation.

5.1.7 Space extension

It is possible to extend the Box-Jenkins methodology by incorporating econometric theories, such as those discussed in (Beenstock & Felsenstein, 2007), to include spatial neighbors in the forecasting process. This is achieved through the use of a Spatial Vector Autoregression (SVAR) model, where a spatial matrix W is incorporated to account for spatial autocorrelation. The SVAR model extends traditional VAR models by including terms that capture the influence of neighboring spatial units, thereby allowing for a more accurate forecasting process that considers both temporal and spatial dependencies. The SVAR equation can be represented as:

$$Y_t = A_1 Y_{t-1} + A_2 Y_{t-2} + \cdots + A_p Y_{t-p} + W Y_t + \boldsymbol{\varepsilon_t} \tag{5.6}$$

where Y_t is a vector of variables at time t, A_i are coefficient matrices, W is the spatial weight matrix, and $\boldsymbol{\varepsilon_t}$ is the error term vector. However, to fully understand this extension and ideas on how to build W, it is necessary to explore purely spatial models and spatial "forecasting," commonly referred to as interpolation.

5.2 Purely spatial models

5.2.1 Weighted sum

The main distinction between temporal and spatial random processes lies in the dimensionality of their input space. While theoretically possible to construct a spatial model analogous to a temporal model, significant challenges arise due to this difference. Consider a spatial model formulated in an autoregressive framework with spatial lags defined as $\boldsymbol{h_k} = \boldsymbol{s_i} - \boldsymbol{s_k}$:

$$z_{\boldsymbol{s}}(\boldsymbol{s_i}) = \phi_0 + \phi_1 z_{\boldsymbol{s}}(\boldsymbol{s_i} + \boldsymbol{h_1}) + \phi_2 z_{\boldsymbol{s}}(\boldsymbol{s_i} + \boldsymbol{h_2}) + \ldots + \phi_p z_{\boldsymbol{s}}(\boldsymbol{s_i} + \boldsymbol{h_k}) + \varepsilon_{\boldsymbol{s_i}} \tag{5.7}$$

However, this model encounters significant limitations due to the 2-dimensional nature of the input space $z_{\boldsymbol{s}}(\boldsymbol{s})$ there will not be two observation location pairs at exactly the same lag. Furthermore, such a model structure inherently assumes a regular spatial grid, which is rarely encountered in practical applications. Specifically, our analysis of the Meta user count data demonstrates that user locations do not conform to a regular grid pattern

A more realistic model would involve considering neighboring points irrespective of their spatial lag and predicting a point at a given location using a weighted sum approach. This can be expressed as:

$$\hat{z_S}(\mathbf{s} + i) = \sum_{i=1}^{n} \lambda_i z_S(\mathbf{s}_i) = \boldsymbol{\lambda}^T \boldsymbol{z_s} \tag{5.8}$$

The modeling effort is now focused on determining the optimal weights λ_i for the weighted sum approach.

5.2.2 Kriging

Kriging is a geostatistical interpolation[13] method that provides an optimal solution to the following problem: Given a set of spatial data points, how can we predict the value at a new, unobserved location in such a way that minimizes the prediction error ? Specifically, Kriging aims to find a set of weight that minimize the

[13]Interpolation is the equivalent of forecasting in the spatial world. Even if forecasting corresponds to interpolation, one might argue that in statistics there is no such things as true interpolation.

variance of the residuals (the differences) between the predicted value and the true, unobserved value at the new location. Kriging equations can be found in (Cressie, 1993), (Bogaert, 1996) and (Wikle et al., 2019).

$$\lambda_i = \arg\min(Var[Z_{\boldsymbol{s}}(\mathbf{s_0}) - \hat{Z}_{\boldsymbol{s}}(\mathbf{s_0})]) = \arg\min(C(\mathbf{0}) - 2\boldsymbol{\lambda}^T\boldsymbol{\sigma} + \boldsymbol{\lambda}^T\boldsymbol{\Sigma}\boldsymbol{\lambda}) \tag{5.9}$$

Where $\boldsymbol{\sigma}$ is the vector of autocovariance $C(Z_{\boldsymbol{s}}(\boldsymbol{s}_0), Z_{\boldsymbol{s}}(\boldsymbol{s}_i))$[14]

Naturally, a key constraint in this optimization is that the expected value of the prediction error is zero, ensuring that the predictions are unbiased.

$$E[Z_{\boldsymbol{s}}(\mathbf{s_0}) - \hat{Z}_{\boldsymbol{s}}(\mathbf{s_0})] = (\mathbf{X}_0' - \lambda\mathbf{X}')\boldsymbol{\beta} = 0 \qquad \text{with the mean } \mu = \mathbf{X}\boldsymbol{\beta} \tag{5.10}$$

A solution to this optimization with constraint is the following :

$$\begin{pmatrix}\lambda \\ \eta\end{pmatrix} = \begin{pmatrix}\boldsymbol{\Sigma} & \mathbf{X} \\ \mathbf{X}' & \mathbf{0}\end{pmatrix}^{-1}\begin{pmatrix}\boldsymbol{\sigma} \\ \mathbf{X}_0\end{pmatrix} \qquad \text{with } \eta \text{ the Lagrangian to force constraints.} \tag{5.11}$$

But this system simplifies to :

$$\begin{pmatrix}\boldsymbol{\Sigma} & \mathbf{1} \\ \mathbf{1}' & 0\end{pmatrix}\begin{pmatrix}\lambda \\ \eta\end{pmatrix} = \begin{pmatrix}\boldsymbol{\sigma} \\ 1\end{pmatrix} \qquad \text{with the mean } \mu = \boldsymbol{\beta}_0 \tag{5.12}$$

and finally to :

$$\boldsymbol{\lambda} = \boldsymbol{\Sigma}^{-1}\boldsymbol{\sigma} \qquad \text{with the mean } \mu = 0 \tag{5.13}$$

Those 3 solutions are called respectively in geostatistics simple, ordinary and universal kriging. Regression kriging is another term used when the practitioner incorporates covariates for modeling the mean in addition to spatial coordinates generally used in universal Kriging. Naturally, even if weak stationarity is respected, having a mean $\mu = 0$ is in practice quite rare. In order to simplify the Kriging system and have a stationary random process with a mean function $u(\boldsymbol{s}) = 0$ we need to take population density into accont since this variable is responsible of the spatial trend observed in the dataset and that it naturally encapsulates the spatial x and y dependency.

5.2.3 Dealing with Population density

Throughout this thesis, we have repeatedly demonstrated that the Meta user count is not consistent across different areas. Its mean and variance are influenced by population density. While we adjusted the variance using a log transformation, we did not adjust the mean because it wasn't needed for purely temporal models. To make the purely spatial processes $\mathcal{Z}_{\boldsymbol{s}}$ first-order stationary, we perform a GLS linear regression of the log Meta user count against the log population density in the Philippines[15].

$$log(Z(s|t_i)) = \beta_0 + \beta_1 log(\text{pop} + 1) + \varepsilon$$

This method helps us obtain residuals that should be first-order stationary with a mean of 0. However, as shown in figure 25, the mean is still slightly skewed to the right after the regression. This skewness is likely

[14] This result is demonstrated in the Appendix A4

[15] We arbitrarily add 1 to the population density where it equals 0 to avoid indefinite values at those points

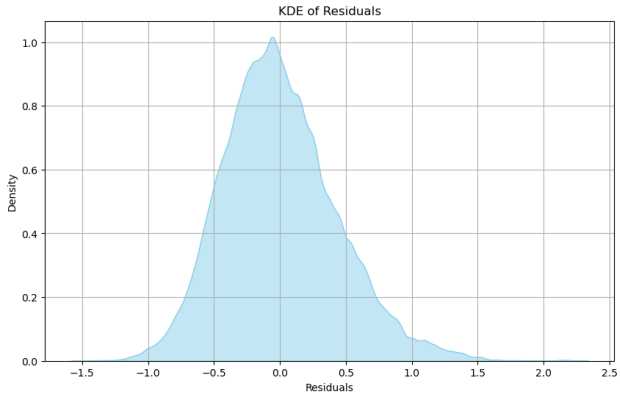


Figure 25: Population density residuals distribution, the mean appears slightly skewed.

attributable to persistent variance in regions with high population density, indicating that the logarithmic transformation was insufficient to fully normalize the variance. To address this issue and stay conservative we assume the mean is unknown but constant and we will therefore employ ordinary kriging.

5.2.4 Variogram

To find an estimate of the covariance matrix $\mathbf{\Sigma}$ we have established that we should fit an empirical variogram using the Matheron definition as outlined in equation 4.6. We also mentionned how different distance defintions could impact the variogram. However, we have not omitted to mention the specifics of variography. Indeed, in a multidimensionnal input space such as spatial settings, we know there will not be two observation location pairs at exactly the same lag. Thus, we need to group information about point pairs at similar distance together, to learn how similar their observed values are this process of grouping distance data together is called binning. A common binning strategy is to use evenly spaced bins up to the maximum distance value. However, in this thesis, we use k-means clustering on the distances. The centroids of the clusters are used to determine the upper edges of the lag classes, with each upper edge set to half the distance between two neighboring cluster centroids. K-means binning ensures a meaningful number of pairs for each lag and a natural grouping at the cost of non equidistant lag classes. The result of our binning strategy is visualized in figure 27. We observe that the classes vary in size, but most lags are well represented by the data. The first lags have fewer observations, which is expected because there are fewer points close to a given point compared to those further away (smaller spher of influence). Additionally, the pairwise difference increases with the number of lags, which aligns with geostatistical principles and Tobler's first law of geography.

Another crucial concept is the distinction between isotropy and anisotropy. The conventional method for calculating a variogram assumes that the covariance between observations depends solely on the distance separating them, making the system isotropic. In this approach, pairs of points are formed from all observation points, with the only restriction being a limiting distance beyond which pairing points becomes meaningless. However, this assumption doesn't always hold true. In real-world landscapes, processes can occur in an organized rather than random manner. This organization is often directional, resulting in stronger covariance in one direction compared to another, making the system anisotropic (in term of covariance). this property is illustrated for synthetic data in figure 28. Consequently, before forming lag classes, an additional step must be introduced to account for this directional dependence. The estimator should in that case be

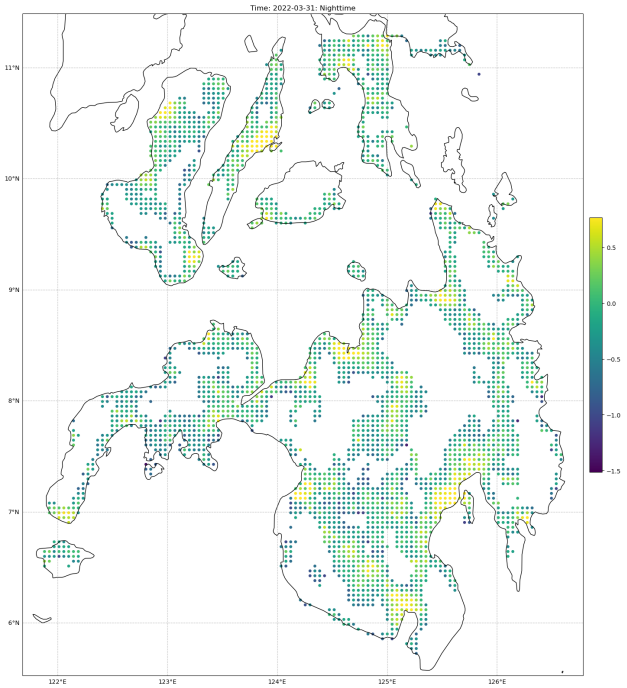


Figure 26: Log residuals in the south of the Philippines on the 31/03/2022.

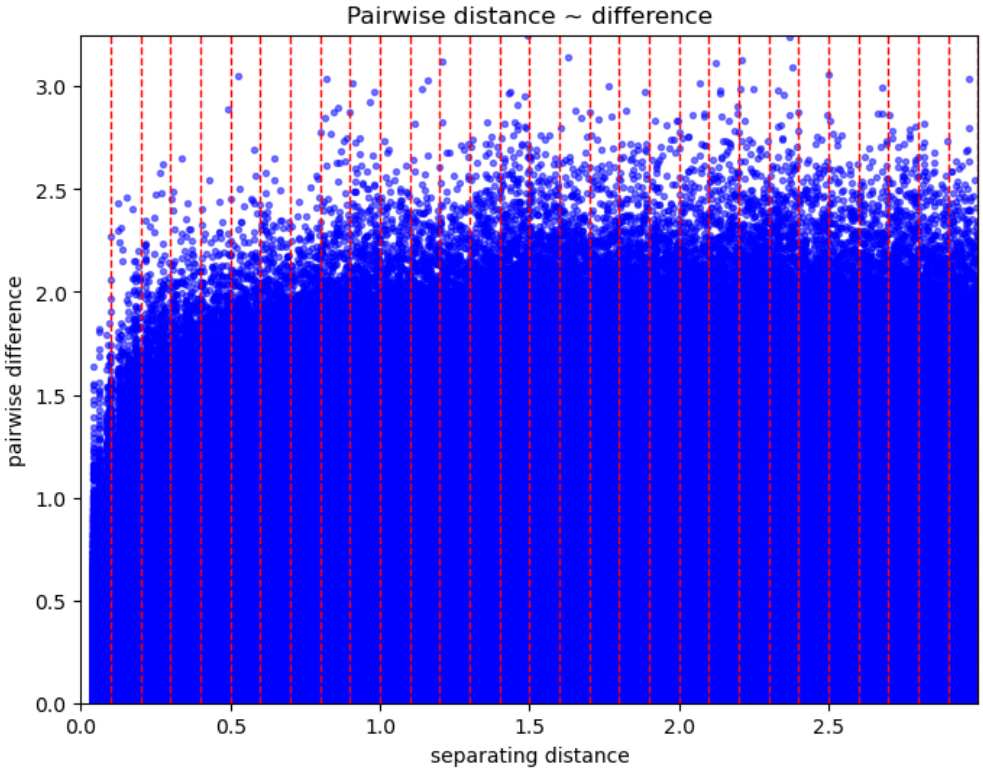


Figure 27: Kmeans binnings on pairwise distance vs pairwise difference.

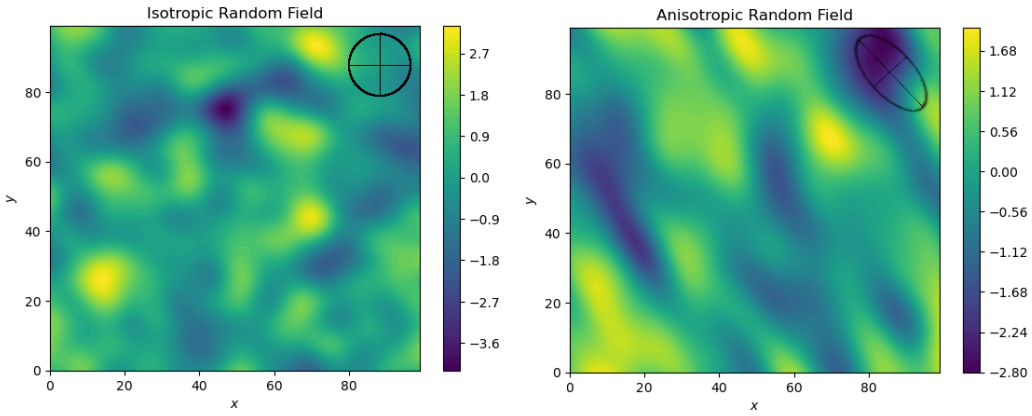


Figure 28: Synthetic Gaussian random field (a) isotopic (b) non-isotropic

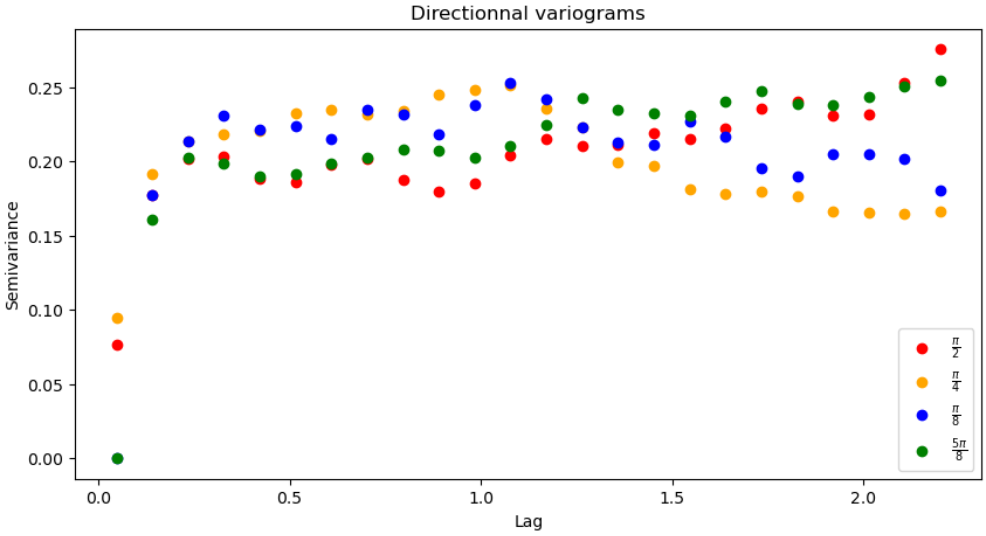


Figure 29: Directional variogram for the log residuals

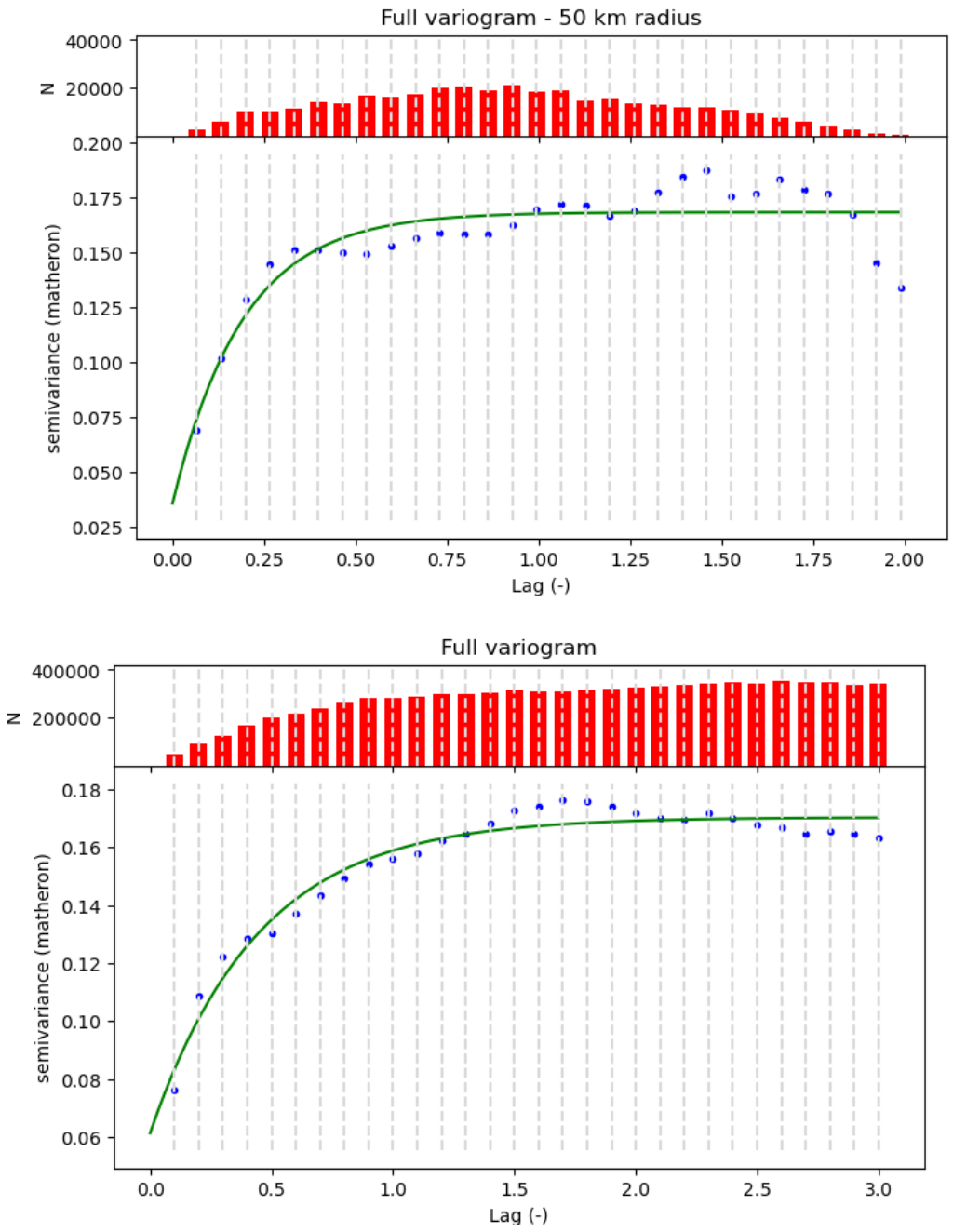


Figure 30: Variogram (a) 50 km radius (b) all points in the Philippines

defined as:

$$\gamma(h,\theta) = \frac{1}{2N(h,\theta)} \sum_{i=1}^{N(h,\theta)} [Z(x_i) - Z(x_i + h)]^2$$

Figure 29 shows directional variograms with a 25 degrees tolerance for θ. Notably the North and east directions are represented by the red and orange dots. While there are certainly some difference differences between semi-variance across different directions, we argue this difference does not appear important enough to warrant separate treatment. Consequently, we use a single variogram across all directions (fig 30). This decision is also motivated by the nature of our data. We do not expect the number of Meta users to exhibit anisotropy, as it is not influenced by processes such as wind patterns, water flow, or geological formations, which are on the contrary inherently anisotropic. We argue that most of stronger irregularities for higher lags are due to leftover non-stationarity in high population density urban locations.

In the purely temporal analysis in section 5.2, we used all temporal information to construct the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF). Following this logic, it might seem natural to use all spatial information to build the variogram. However, this approach is not necessary nor practical. The variogram estimator is an $O(n^2)$ algorithm meaning its computational complexity increases quadratically with the number of data points. Additionally, according to Tobler's First Law of Geography, it is unlikely that points separated by very large distances have any significant influence on each other at all. To prove this point we show the variogram of a spatial subset of observed points located within a 25 km radius from a center at coordinates (125, 7.5) in the south of the Philippines and the variogram of every observed point in the country (fig 30). We see that the variograms are very similar. Since we are dealing with a purely spatial process and are aware that temporal variation is low, we have arbitrarily chosen to fix the date and time to the 01/01/2022. The marker size on the subset map (26) is intentionally smaller than usual to accurately represent the point locations and to facilitate future spatial interpolation. While the reader might perceive a high density of points due to marker-size, this impression can be misleading.

5.2.5 Variogram models

In the purely temporal analysis, we did not fit the ACF or PACF directly; instead, we used them as indicators for potential models and subsequently employed a grid search algorithm to identify the best temporal model. However, we showed this approach is not directly applicable to spatial data. In spatial analysis, it is essential to model a variogram function $\gamma(\mathbf{h})$ to estimate the covariance function $C(\mathbf{h})$ accurately. The choice of the variogram model is crucial and significantly influences the interpolation results. Modeling a variogram in practice is challenging because the covariance matrix derived from the variogram must be positive definite to ensure valid and meaningful spatial predictions. In practice, geo-statisticians use function that are known to generate well conditionned covariance matrices. While studying those functions escape the scope of this thesis, table shows 4 of the most famous functions and figure shows the structural difference between those functions.

The covariance function $C(\mathbf{h})$ is associated to $\gamma(h)$ with equation 4.8. Figure shows how each of the 4 models fit our data, the adjusted r^2 for each variogram is shown in table. The overall variance of the spatial random field, also called the sill, is illustrated with the black dotted line, while the range - the blue dotted line - represents the distance beyond which the covariance between points reaches 0. The sill and the range are respectively represented by c, a in the variogram model presented in table 2. In practice the Matérn model is the best performing model in term of pseudo r^2 as shown on table 3 however due to its fitting similarity with the exponential model, we will choose the later one as it much easier model to unpack and

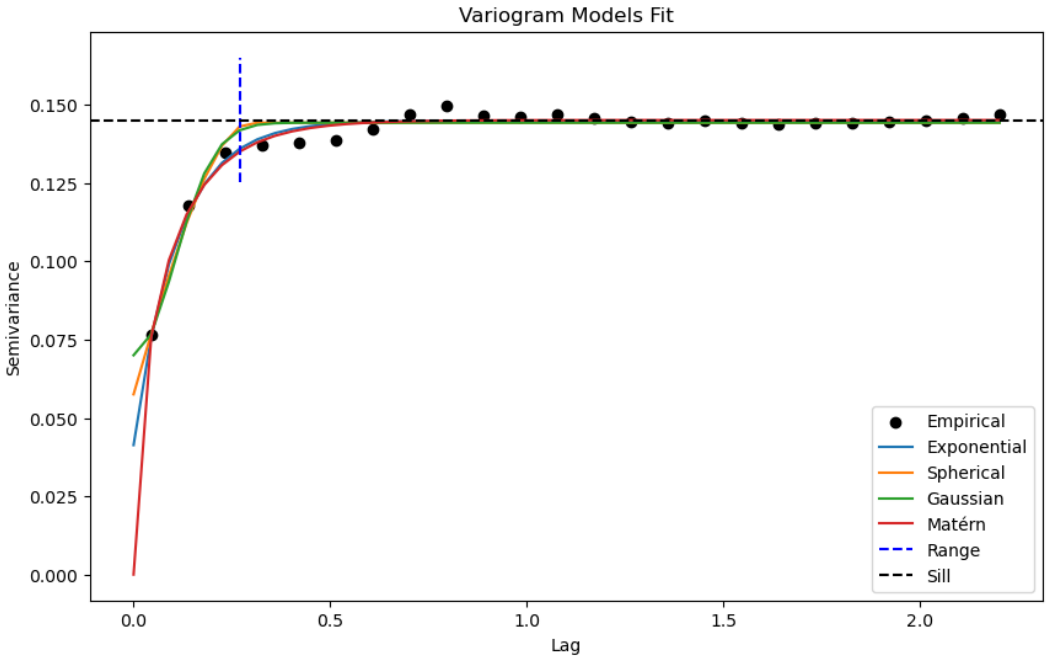


Figure 31: Fitted variogram models for the spatial subset data on the 01/01/2022

Model Name	**Variogram Function $\gamma(\mathbf{h})$**
Gaussian	$c\left(1-\exp\left(-\left(\frac{h}{a}\right)^2\right)\right)$
Spherical	$\begin{cases} c\left(1.5\left(\frac{h}{a}\right)-0.5\left(\frac{h}{a}\right)^3\right) & \text{if } 0 \le h \le a \\ c & \text{if } h > a \end{cases}$
Matèrn	$c\left(1-\frac{1}{2^{\nu-1}\Gamma(\nu)}\left(\frac{h}{a}\right)^{\nu} K_\nu\left(\frac{h}{a}\right)\right)$
Exponential	$c\left(1-\exp\left(-\frac{h}{a}\right)\right)$

Table 2: Variogram and Covariance Functions for Different Models

describe. Our final variogram model is therefore

$$\gamma(h) = 0.104\left(1-\exp\left(-\frac{h}{0.11}\right)\right)+0.0413$$

where 0.0413 is called the "nugget effect." The nugget effect represents a discontinuity at the origin of the variogram, indicating measurement errors or spatial variation at distances smaller than the sampling interval. It accounts for the variance observed when the distance h approaches zero, which cannot be explained by the spatial structure alone.

5.2.6 Interpolation

Having defined our variogram model, we can now utilize it to solve the Kriging system in Equation 5.11. Indeed we can use the variogram model to find the exponential covariance function (figure 32) and we can use this covariance function to generate the finate $n \times n$ covariance matrix $\boldsymbol{\Sigma}$ and the vector of autocovariance at a new location $\boldsymbol{\sigma}$.The resulting covariance matrix, depicted in Figure 33, illustrates the behavior of the exponential covariance function, showing that neighboring points have gradually decreasing covariance as

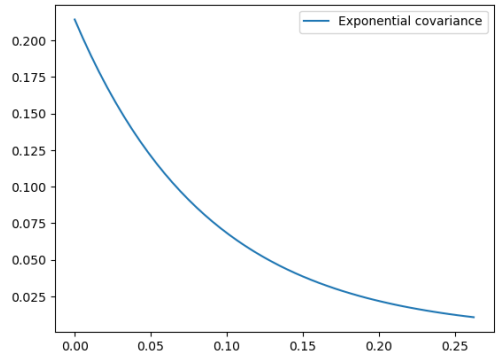


Figure 32: Covariance function obtained from the relation $C(\mathbf{0}) - \gamma(\boldsymbol{h})$

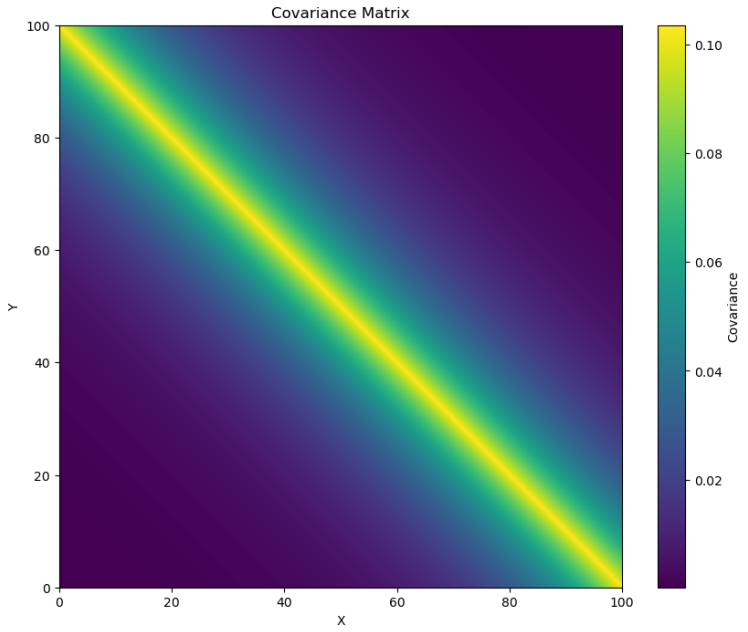


Figure 33: Covariance matrix obtained by sampling the covariance function.

Rank	Model Name	Pseudo-r2 Score
1	Matèrn	0.97979
2	Exponential	0.97715
3	Gaussian	0.91862
4	Spherical	0.91616

Table 3: Ranking by Pseudo-r2 Score

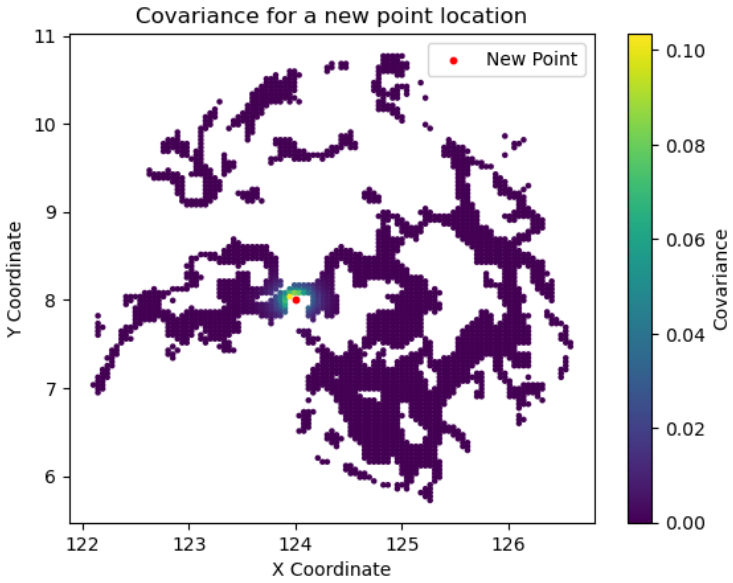


Figure 34: Covariance matrix obtained by sampling the covariance function.

distance increases, eventually approaching zero. This visualization is crucial as it demonstrates the natural relationship derived from stationarity assumptions and variogram modeling, despite the potential for infinite covariance forms. The auto-covariance vector $\boldsymbol{\sigma}$ is spatially represented in Figure 34 highlighting what weights $\boldsymbol{\lambda}$ will be set to higher values following equation 5.11.

Instead of solving the Kriging equation for a single point, we can extend it to a large set of new locations, enabling high-resolution predictions of log-residuals. Figures 35 - 38 illustrates this procedure for the southern Philippines. The model produces a smooth interpolation of log-residuals. By incorporating the effect of population density and reversing the log transformation, we achieve a precise spatial prediction of the True Meta user counts across the region as observed in figure 36. This prediction is robust because it incorporates population density, which is crucial for accuracy. Without accounting for population density, urban centers would appear as smooth, featureless areas, which is unrealistic. Geological or geographical features can significantly influence city shapes, making distance alone insufficient to describe Meta user counts, in practice this aspect was translated into the first-order non stationarity. Figures 36 and 37 compare predictions with and without population density considerations. We can see that omitting population density doesn't produce entirely inaccurate results; however, it causes urban centers to appear as unrealistically smooth circles. By looking at figure 38 we can see that our Meta user count interpolation is a good method to emphase urban centers and transport infrastructure. Naturally, the quality of the prediction is dependant on the resolution of the population density raster.

Using the Kriging variance equation 5.9, we can assess the prediction uncertainty in figure 39. As

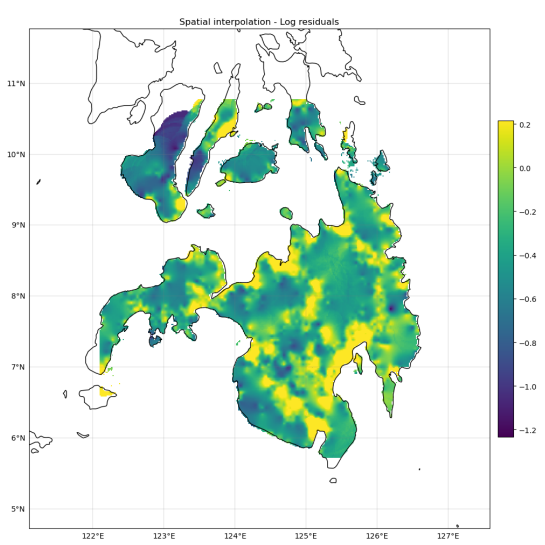


Figure 35: Spatial interpolation for the log residuals

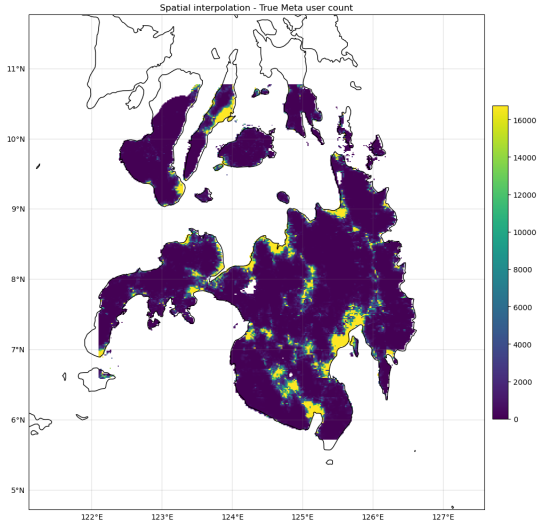


Figure 36: Spatial interpolation for the true meta user count

expected, the uncertainty increases in areas where data is sparse—in this case, regions with low population density that were affected by censoring. This demonstrates that, given a substantial amount of data, Kriging is an effective tool for interpolation, particularly in well-informed areas.

5.3 Space-time Kriging

We have forecasted the Meta user count at new times for fixed locations $z(\hat{\boldsymbol{s}_i}, t_0)$ and interpolated the Meta user count at new locations for a fixed time $z(\hat{\boldsymbol{s_0}}, t_j)$. A natural question is therefore how to predict the Meta user count at new times and new locations jointly. For example what if we want to predict the value $z(\hat{\boldsymbol{s_0}}, t_0)$ for a new unobserved point in space and time. A proposed space-time model is the stochastic additive model :

$$Z(\boldsymbol{s}, t) = Y(\boldsymbol{s}, t) + M_{\boldsymbol{s}}(\boldsymbol{s}) + M_t(t) + \mu(\boldsymbol{s}, t) \tag{5.14}$$

With $Z(\boldsymbol{s}, t)$ an space-time random variable, $Y(\boldsymbol{s}, t)$ a purely spatial stochastic component $M_{\boldsymbol{s}}(\boldsymbol{s})$, a purely temporal stochastic component $M_t(t)$ as well as a non stochastic space time mean function. We can already simplify this expression by considering that the mean function $\mu(\boldsymbol{s}, t)$ is zero. This assumption is reasonable, as demonstrated in our residuals analysis in Section 5.2.3, where we corrected for first-order stationarity using regression on population density, resulting in a mean of zero.

In relation to the autocorrelation structures we defined in section 4.2, a fully temporal stochastic component with a one dimensionnal input space can also be described by a covariance function, the only difference with Box Jenkins temporal model is that the stochastic process is not bounded to discrete timesteps anymore. This means we can build and fit temporal variogram models which in turn allow to build a covariance function to generate temporal covariance at any time lag.

If we build the temporal and spatial conditional variograms $\gamma_z(\boldsymbol{h}, t_j)$ and $\gamma_z(\boldsymbol{s}_i, \tau)$ with one dimensionnal and two dimensionnal input spaces we can isolate the effects of the purely spatial or purely temporal components. This can be expressed as:

$$\gamma_z(\boldsymbol{h}|t_j) = \gamma_y(\boldsymbol{h}|t_j) + \gamma_{M_t}(\tau) \quad \text{and} \quad \gamma_z(\tau|\boldsymbol{s}_i) = \gamma_y(\tau|\boldsymbol{s}_i,) + \gamma_{M_{\boldsymbol{s}}}(\boldsymbol{h})$$

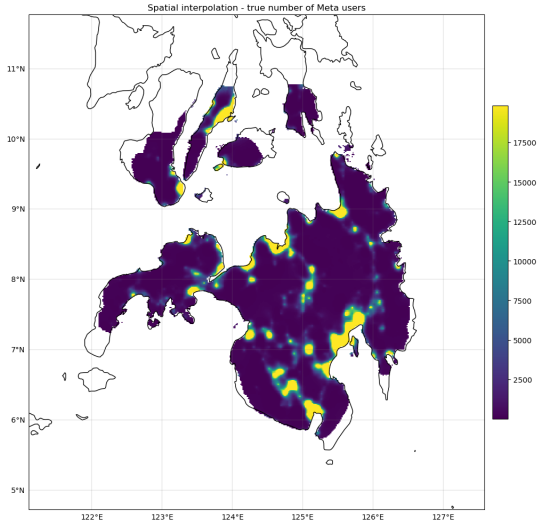


Figure 37: Spatial interpolation without accounting for population density

Figure 38: Urban and transport infrastructure in the philippines (google maps).

This observation is natural: when we fix either time or space in a space-time random process, the inherent variation is contained within the mean of the new conditional process (this was observed in section 2.3.3, figure **??**. As a result, this fixed value does not affect the variability and therefore can not appear in the variogram.

Conditional variograms are illustrated in Figures 40 and 41, based on 100 iterations at 500 random locations using seasonally differenced log residuals. The spatial variograms conditioned on time are consistent, indicating a low purely temporal stochastic component. Conversely, the temporal variograms conditioned on space show significant variation, suggesting a substantial purely spatial component. These results align with similar observations made in several figures in Section 3 when we specifically focused on temporal and spatial distribution. Indeed, when conditioned on time, most spatial processes appeared very similar, to the extent that we had to map - to distinguish any differences. In contrast, the time series associated to temporal processes conditioned on space exhibited completely different mean values. In that section, we anticipated that this might indicate ergodicity. A notion we can now fully define.

5.3.1 Ergodicity

The definition of ergodicity is mostly taken from (Bogaert, 1996) who describes the concept so well that it would be unwise to change his definition. (Bogaert, 1996) describes a random field as ergotic in term of space and time if "a realisation of a space-time random field observed conditionally on a fixed space or time location is representative of all the possible space-time realisations of this random field". This is obviously close to the truth for space conditioned on time but far from the truth for time conditioned on space. This means $M_t(t)$ is probably weak and the space-time model could be rewritten as

$$Z(\boldsymbol{s}, t) = Y(\boldsymbol{s}, t) + M_{\boldsymbol{s}}(\boldsymbol{s})$$

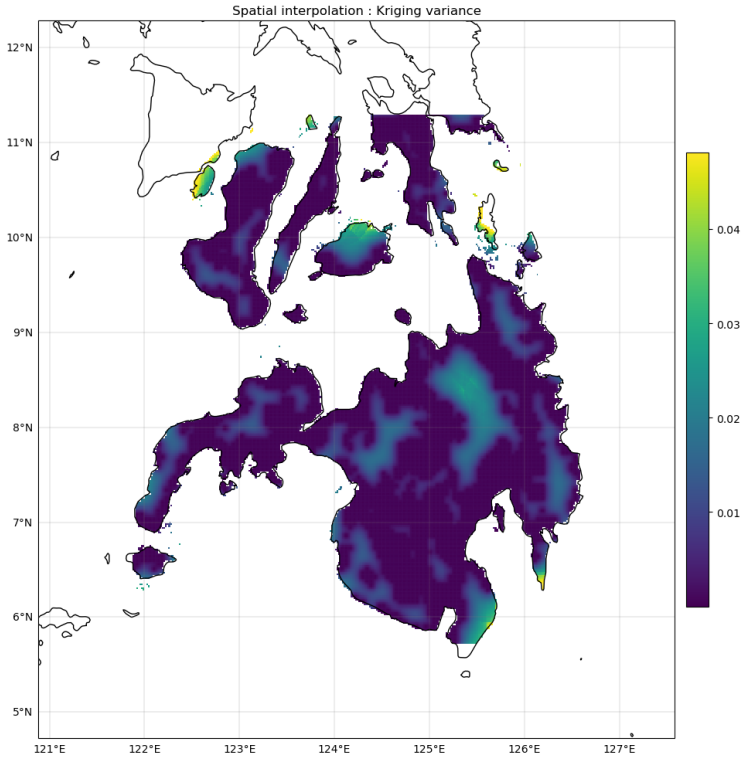


Figure 39: Kriging variance

The space-time random field $\mathcal{Z}$ is said to be ergotic along the space axis but not ergotic[16] along the time axis.

In direct relation to this ergodicity, the temporal variance of the conditional time series for $Z(t|\boldsymbol{s}_j)$ is naturally very small since the variance of $Z(t|\boldsymbol{s}_j)$ depends almost solely on the variance of $Y(t|\boldsymbol{s}_j)$ ignoring the (important) spatial variance of $M_{\boldsymbol{s}}(\boldsymbol{s})$.

This small temporal variance has a significant implication, as discussed in Section 5.1. It suggests that using other locations to predict values at a new time and known space $z(\boldsymbol{s}_i, t_0)$ may not be necessary for point locations. A fully temporal model, tailored to each specific point, could be sufficient due to the inherently low temporal variation of those specific points.

5.3.2 Space-time interpolation

With our space-time model defined, we can now proceed with spatio-temporal interpolation. This process is analogous to classical spatial interpolation. The main difference is the definition of that covariance matrix $\Sigma_{\boldsymbol{s},t}$ which is now dependent on both temporal and spatial lags $\boldsymbol{\tau}$,$\boldsymbol{h}$ and is therefore of dimension $ST \times ST$. We discussed in section 4.3 two methods to generate this space-time covariance matrix one assuming random field separability (eq 4.10) and one only assuming space-time covariance separability (eq 4.11). In the first case the method is straightforward. We can see the stochastic space-time component $Y(\boldsymbol{s}, t)$ as separable in space and time and the space-time autocovariance matrix $\Sigma_{\boldsymbol{s},t}$ as well as the space-time autocovariance vector $\boldsymbol{\theta}_{\boldsymbol{s},t}$ are computed using Kronecker products between their purely spatial and purely temporal parts. Those new space time autocovariance matrix and vector can be then used in the Kriging procedure described

[16] In practice : Insert non true ergodicity here

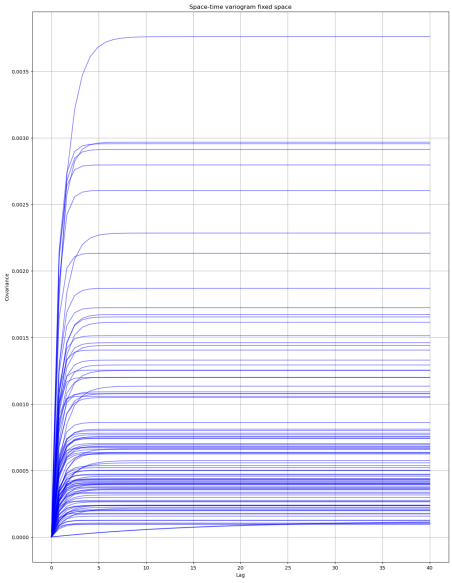


Figure 40: Temporal variograms conditionned on random space locations

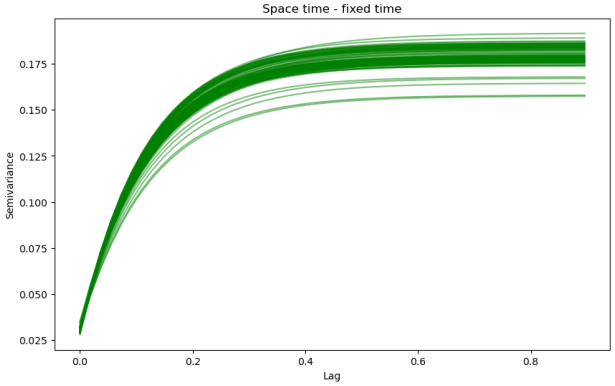


Figure 41: Spatial variograms conditionned on random temporal locations

in equation 5.11. In the second case, we don't see the space time stochastic component $Y(\boldsymbol{s}, t)$ as separable but we assume its covariance is according to equation 4.11. To find the spatial and temporal auto-correlation functions $\rho_{\boldsymbol{s}}(\boldsymbol{h})$ and $\rho_t(\tau)$ we take the average of all the conditional space - or time - variograms parameters and transform them later in their autocorrelation counterpart (eq 4.4). To find the space-time component variogram we assume ergodicity. The variogram $\gamma_z(\boldsymbol{h}|t_j)$ is therefore equal to

$$\gamma_z(\boldsymbol{h}|t_j) = \gamma_y(\boldsymbol{h}|t_j) + \gamma_{M_t}(\tau)$$

which corresponds approximatively to

$$\gamma_z(\boldsymbol{h}|t_j) = \gamma_y(\boldsymbol{h}|t_j)$$

This signifies the mean variance σ_z^2 (the sill) derived from the average of $\gamma_z(\boldsymbol{h}|t_j)$ is a good approximation of σ_y^2 the space-time variance needed in equation 4.11. With the three components necessary to determine covariance in space and time identified, we can readily apply the Kriging procedure. This process is illustrated in Figure 42, where we show the weights λ distribution in space and time.

Under this configuration both methods seem to lead to very similar weights which supports the assumption of a separable space-time random field. The Kriging operation's weights in space and time for two given time steps ($\tau = 0$,$\tau = 1$) are concentrated around neighbouring locations which is expected but reassuring. Moreover, respectively >99% of the weight is contained at lag $\tau = 0$, which means that both space-time model seems to heavily favour neighbouring locations in the present. This suggests that including temporal neighbors for new time and space prediction may be an overshoot for this dataset, a spatial prediction could be more then enough. A similar observation is made when predicting at a new time and locations with only past-time lags $\tau = 1$ and $\tau = 2$. Most of the weights are then condensed in $\tau = 1$.

Finally, when trying to predict new points in time for known spatial locations $\hat{z}(\boldsymbol{s_i}, t_0)$ similarly to what we have done in section 5.1, we see that the weights now clearly favoritise only one location which is the same location in the past. Spatial neighbors have virtually no weights at all. This shows that in that specific

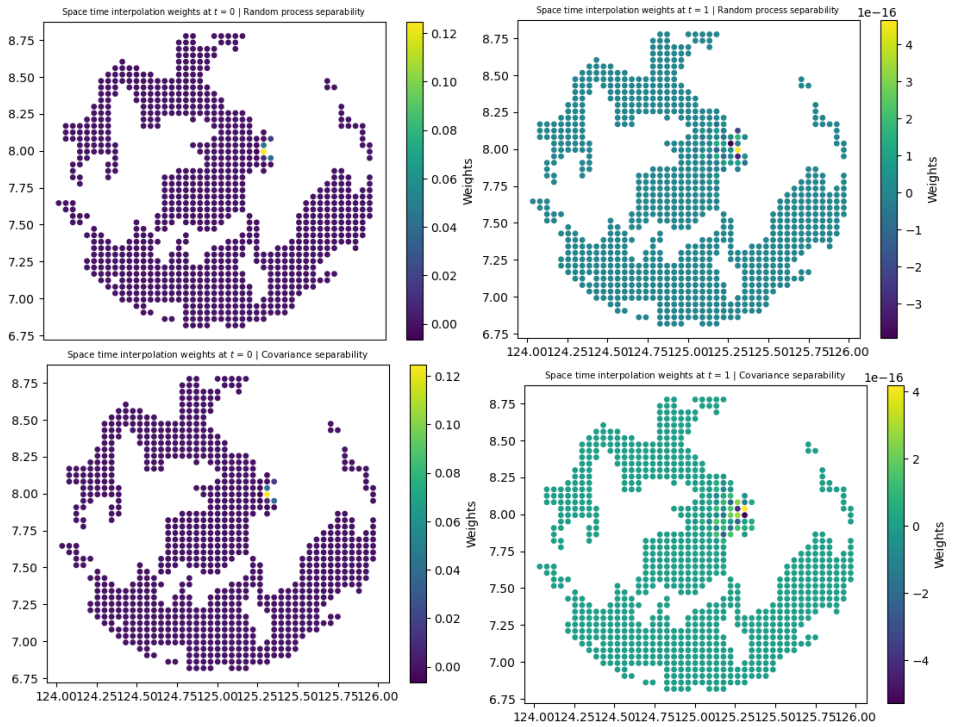


Figure 42: Space time distribution of Krige weights for the 2 separability hypothesis in the prediction of $\hat{z}(\boldsymbol{s_0}, t_0)$

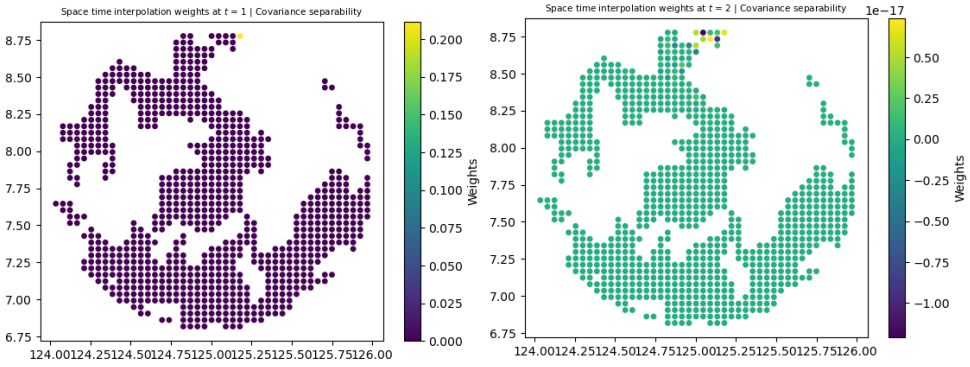


Figure 43: Space time distribution of Krige weights in the prediction of $\hat{z}(\boldsymbol{s_i}, t_0)$

case a temporal approach is preferred to a spatial one. In this section we do not show full krige results as they would be very similar from those already displayed in figures 36 and 37.

5.4 A review of temporal and spatial methodology

The fundamental distinction between temporal and spatial methodologies lies in their domains of application and underlying philosophical approaches. Temporal data analysis is historically rooted in econometrics, focusing on time series analysis and the dynamics of variables over time. In contrast, spatial data analysis is more closely associated with the applied natural sciences, where the focus is on understanding patterns and processes across geographical space. This divergence has led to the development of distinct modeling frameworks, each tailored to the specific characteristics of temporal and spatial data.

Temporal models typically address autocorrelation directly in the mean structure, often through hierarchical models or autoregressive processes, which are inherently discrete in nature. These models are well-suited for forecasting and they can easily incorporate exogenous variables since they are grounded in econometric theory. Space-time data is often treated as panel data which is the equivalent of a multivariate time series framework, where the primary goal is to predict future values and assess the relationships between different time-dependent variables. Practitioners like (Hafner, 2020) and (Beenstock & Felsenstein, 2007) often approach space-time data with a grid-based mindset , applying techniques that extend classical time series methods sich as the one seen in section 5.1 to multidimensional datasets with spatial weight matrices used to correct the sptially autocorrelated residuals. On the other hand, spatial models typically handle autocorrelation within the covariance structure, reflecting a continuous view of the world. Spatial practitioners, such as (Bogaert, 1996) and (Cressie, 1993), focus on spatial interpolation techniques like Kriging, which are designed to estimate values at unsampled locations based on the spatial correlation of nearby data points (section 5.3.1). In this framework, time is often treated as another continuous dimension, leading to the conceptualization of space-time as a random field with potentially infinite neighbors, rather than a finite, predefined set of points.

In econometrics, while forecasting is a significant goal, the primary focus often lies in parameter estimation and hypothesis testing. For instance, researchers might investigate how a government's increase in public spending in one region affects job creation not only within that region but also in adjacent areas. In contrast, spatial sciences often prioritize interpolation and spatial prediction, particularly when dealing with incomplete or sparsely sampled data. For example, scientists might collect samples of pollutants along a river or measure mineral concentrations in a geological deposit, and then use spatial models to estimate the distribution of these variables across unsampled areas. The goal here is to reconstruct the underlying process or field with a high degree of accuracy, providing a more comprehensive understanding of the spatial phenomenon being studied. Sparse data is a much more frequent problem in the spatial world then the temporal world since it is continuous by definition.

When considering these two modeling philosophies—temporal and spatial—they might initially seem disjointed and difficult to reconcile. In practice, econometric practitioners often undervalue the spatial dimension of their problems, while spatial science practitioners frequently overlook the temporal aspects. However, as we argue based on our analysis in our previous sections, this disconnect primarily stems from the distinct origins of these frameworks in different fields of study. Despite their differences, these methodologies are inherently connected through the broader field of autocorrelated data. We believe that recognizing this connection could pave the way for significant advancements in both disciplines, supporting a comprehensive, multidisciplinary approach to space-time problems.

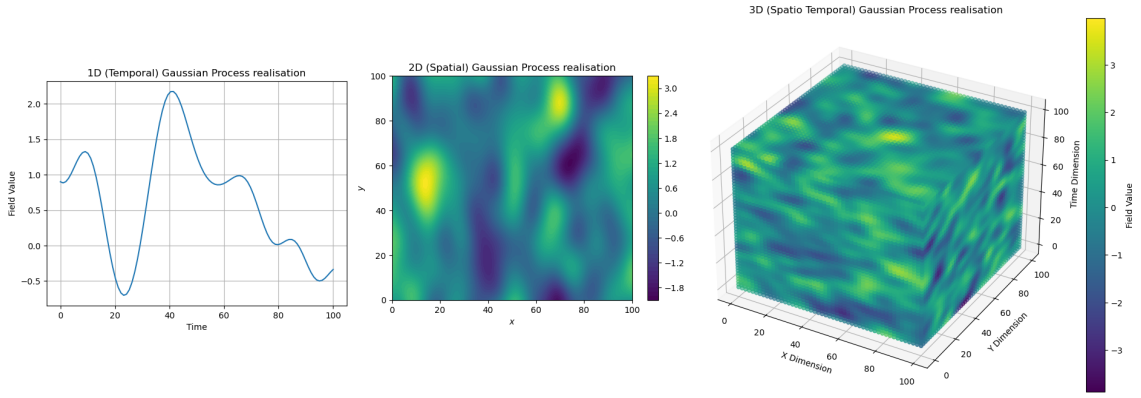


Figure 44: GP's realisations in space and time

5.4.1 Limitations and unification

In this section, we have highlighted the significant challenges that temporal models encounter when dealing with irregular, non-gridded data, particularly in the presence of missing values in time series. Additionally, the Box Jenkins methodology relies on parameter selection, a process that is often not straightforward and frequently depends on computationally expensive grid search algorithms. On the other hand, while spatial models do not face regularity issues, they present their own set of challenges. The estimation of the covariance function in spatial models is heavily dependent on variogram estimation—a process that is not only complex for non spatial statistics initiates but also highly sensitive to user-defined parameters and binning strategies. Furthermore, variogram estimation is computationally intensive, with an algorithmic complexity of $\mathrm{O}(n^2)$.

In Section 6, we introduce a novel machine learning methodology that elegantly addresses these challenges. Moreover, our approach **unifies** the modeling of spatial and temporal data **within a single coherent framework for auto-correlated datasets** overcoming the limitations of existing models and providing a robust solution for analyzing space-time data.

6 Machine learning in space and time

6.1 Introduction

In this section, we introduce a new machine learning model that serves as a partial remedy to the challenges associated with space-time modeling, while also providing a unifying framework for the analysis of all auto-correlated datasets, spatial, temporal or others. The concepts presented here are inspired by the recent groundbreaking work of (Flexman, 2022) and (McElreath, 2020). Importantly, this section serves as an introduction rather than an in-depth exploration of the proposed solution the later of which would probably justify a master thesis on its own. Gaussian process regression is a rapidly evolving hot topic in the world of Bayesian statistics.

6.2 Definition

Kriging is also sometimes considered to be a foundation of Gaussian process (GP's) regression in scientific literature, in fact (Marinescu, 2024) defined formally this relation in his recent article. GP's can be seen as a multivariate normal distribution extended over an infinite-dimensional function space with a mean

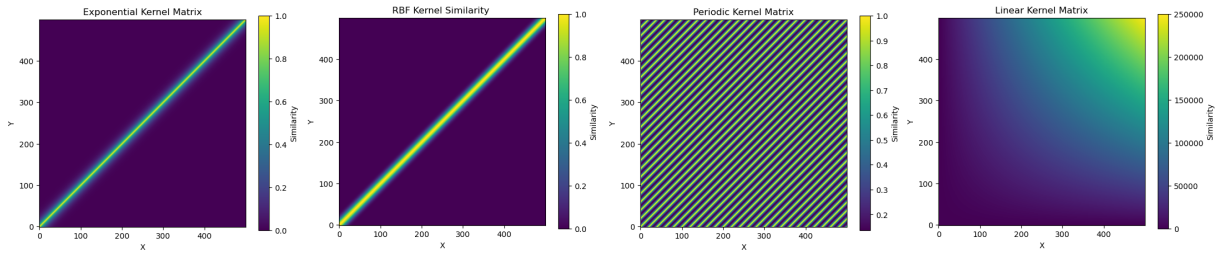


Figure 45: Kernel matrices visualisation

function $m(x)$ and a kernel $K(x_1, x_2)$. This sophisticated conceptual framework suggests that an observed dataset is not simply an assortment of discrete random realizations of random variables $Z(x_i)$. Instead, it constitutes a singular, partial realization from the vast continuum of functions encompassed by the GP. Each such realization at a finite set of points can be mathematically described by what is known as a marginal multivariate normal distribution from the infinite-dimensional Gaussian Process. This marginal distribution is defined as:

$$\begin{bmatrix} Z(x_1) \\ Z(x_2) \\ \vdots \\ Z(x_n) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(x_1) \\ m(x_2) \\ \vdots \\ m(x_n) \end{bmatrix}, \begin{bmatrix} \kappa(x_1, x_1) & \kappa(x_1, x_2) & \cdots & \kappa(x_1, x_n) \\ \kappa(x_2, x_1) & \kappa(x_2, x_2) & \cdots & \kappa(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ \kappa(x_n, x_1) & \kappa(x_n, x_2) & \cdots & \kappa(x_n, x_n) \end{bmatrix} \right) \tag{6.1}$$

More generally - in the infinite dimension case - a random field $\mathcal{Y}$ is assumed to follow a GP:

$$\mathcal{Z} \sim \mathcal{GP}(m(x), \mathbf{K}(\mathbf{x}, \mathbf{x}')) \tag{6.2}$$

where $\mathcal{Z}$ is any random field, $m(X)$ defines the mean function of the GP, providing the expected value of the process for each point in the associated input space and $\boldsymbol{K}(x, x')$ denotes the covariance function or kernel establishing the covariance between the function values at any two points, x and x' within the input space of $\mathcal{Z}$. This definition provides the missing puzzle piece, between the covariance function we covered in details in sections 4 and 5.2 and kernel theory. What we called covariance function, a function that generated valid covariance matrices, can also be called a kernel.

Under weak stationarity described in earlier sections a GP can be simplified and assumed to have a constant mean μ and a constant variance σ^2 with a covariance depending solely on the similarity between x and x' the GP can then be written as follows:

$$\mathcal{Z} \sim \mathcal{GP}(\boldsymbol{\mu}, \mathbf{K}(||\boldsymbol{k}||)) \tag{6.3}$$

Where the covariance function $\boldsymbol{K}$ is now explicitly shown to depend only the (Euclidean)[17] distance between points in the input space $||\boldsymbol{k}||$ and the mean function $\boldsymbol{\mu(x)}$ is constant. To account for anisotropy where the relationship between data points varies not only with their distance but also with other parameters θ distribution of $\mathcal{Z}$ could be easily extended using GPs as :

$$\mathcal{Z} \sim \mathcal{GP}(\boldsymbol{\mu}, \mathbf{K}(||\boldsymbol{k}||, \theta)) \tag{6.4}$$

[17]Other distance definition are possible. Common distance definitions are given in Appendix A1.

GP's are powerful tools; their definition highlights the flexibility and broad applicability of Gaussian Processes. GP's are in practice applicable to any type of random process as long as it is possible to make a measurement of distances or similarity between random observations. This is valid for temporal, spatial and spatio-temporal random processes $\mathcal{Z}_{\mathbf{s}}$, $\mathcal{Z}_t$, $\mathcal{Z}_{\mathbf{s},t}$ but also more exotic random process such as for instance phylogenetic data or natural language data (Flexman, 2022). Figure 44 shows examples of temporal and spatial gaussian processes samples with a constant mean function $\mu(x) = 0$.

6.3 Kernels

The essential modeling power of Gaussian Processes (GPs) lies in their kernels. We have already explored spatial kernels when we defined and illustrated variogram functions and covariance functions, as shown in Table 2 and Figure 34 in Section 5.2. Here, we will briefly formalize the addition of non-stationary kernels commonly used in temporal settings. A review of those kernels can be found in table 4 and figure. Kernel definitions can be found in (Rasmussen & Williams, 2006), (Hafner, 2024), (Corani et al., 2021).

Kernel Name	**Kernel Function** K
Exponential	$K(\mathbf{h}) = \sigma^2 \exp\left(-\frac{\lvert\mathbf{h}\rvert}{\ell}\right)$
RBF (Radial Basis Function)	$K(\mathbf{h}) = \sigma^2 \exp\left(-\frac{\lvert\mathbf{h}\rvert^2}{2\ell^2}\right)$
(Seasonality) Periodic	$K(\tau) = \sigma^2 \exp\left(-\frac{2\sin^2\left(\frac{\pi\lvert\tau\rvert}{T}\right)}{\ell^2}\right)$
(Trend) Linear	$K(t1, t2) = \sigma^2(t_1 t_2)$

Table 4: Kernel Functions for spatial and temporal random processes

The RBF kernel is the equivalent of the Gaussian covariance function we defined in section 5. This terminology will be used a lot in section 7.

6.4 Conditional properties of Gaussian processes

One very useful property of GP's is their conditional propoerties. Indeed, given a set of observations $\boldsymbol{x}$, a new input point x_0 can be easily predicted conditionally on $\boldsymbol{x}$. That is because marginal realisation of GP's are multivariate normal distributions which are proven to have simple conditional properties (Wackerly et al., 2008). As such we can see in (Rasmussen & Williams, 2006) that

$$\mu_0 = k(x_0, \mathbf{x})' K(\mathbf{x}, \mathbf{x})^{-1} [Z(\mathbf{x}) - m(\mathbf{x})] \tag{6.5}$$

$$\sigma_0^2 = k(x_0, x_0) - k(x_0, \mathbf{x})' K(\mathbf{x}, \mathbf{x})^{-1} k(x_0, \mathbf{x}) \tag{6.6}$$

$$Z(x_0) \sim N(\mu_0, \sigma_0^2) \tag{6.7}$$

These properties are powerful. As seen in equations 5.11 to 5.13, and equation 6.5, the conditional GP behaves like a weighted sum where the weights depend on kernel similarity. This means that, in practice, GPs operate similarly to Kriging, using similarities - covariances in Kriging - to assign weights to neighboring data points. Additionally, from equation 6.6, we observe that the variance of the conditional GP increases with the level of dissimilarity as the effect of the neighbouring observations decreases. This is a desirable characteristic, especially in forecasting, where the prediction variance should increase with time to reflect

growing uncertainty. Similarly, for spatial predictions, the prediction variance should increase for points that are far from other observations. Ultimately, the prediction variance will converge to the kernel similarity between the new observation and itself $k(x_0, x_0)$ - that is - the empirical variance of the dataset σ^2(table 4)[18].

6.5 Learning kernel parameters

An important question that remains is how to choose the parameters for these kernels which we call $\boldsymbol{\theta}$. In practice, this is a complex task, and we've already determined that methods like grid search or visually fitting correlograms and variograms are not ideal. Fortunately, **Bayesian statistics** offer a robust framework for parameter estimation in machine learning, particularly when applied to Gaussian processes. In Bayesian statistics, the goal is to update our beliefs about the parameters as we observe more data. This process is encapsulated by Bayes' theorem, which allows us to compute the posterior probability distribution of the parameters $\boldsymbol{\theta}$ conditionally on the data D (McElreath, 2020):

$$P(\boldsymbol{\theta}|D) \propto P(D|\boldsymbol{\theta})P(\boldsymbol{\theta}) \tag{6.8}$$

With $P(D|\boldsymbol{\theta})$ the likelihood of the data given the parameters and $P(\boldsymbol{\theta})$ the prior distribution reflecting our initial beliefs about the parameters. In practice, Bayesian models are closely related to Monte Carlo methods and simulations because the posterior distribution is often impossible to express in a closed form. Instead, advanced techniques such as Markov Chain Monte Carlo (MCMC) or NUTS (Abril-Pla et al., 2023) are used to iteratively sample from the posterior distribution by exploring the parameter space Θ. After running a sufficient number of iterations, these methods approximate the posterior distribution of the model parameters. Once this approximation is achieved, we can generate new, plausible observations by directly sampling from these posterior distributions. By cascading this sampling process, we can effectively approximate the true distribution of the underlying data. This approach is particularly useful for constructing confidence intervals, as it provides more than just a single estimated value for each parameter. Instead, it gives us a full probability distribution that covers a range of potential probable values for those parameters. GP's are especially well-suited to this approach due to their inherent flexibility and the ease with which they can be sampled. GPs also have natural conditional properties.

It's crucial for the reader to understand that when we sample from a GP, we are not merely obtaining a discrete set of values. Instead, each sample from a GP represents an entire function over the input space. To put this into perspective with a practical example: if we run 100 iterations, the resulting posterior chain doesn't have dimensions of 100×1 as it would in classical bayesian settings. Instead, it has dimensions of $100 \times N$ where N is the number of points in the GP sample. This process is illustrated in the accompanying figure for clarity. The figure shows how, with each iteration, the Gaussian Process function progressively becomes a better approximation of the observed data.

In practice, we could delve much deeper into Bayesian statistics and this seems like an exciting process, but doing so would largely extend beyond the scope of this thesis. We leave this opportunity for further exploration by another researcher, possibly in a future master's thesis. For those interested in a more detailed exploration, a substantial amount of material is available in (Flaxman 2015). In this thesis we will let the formidable modern python package *pymc* (Abril-Pla et al., 2023)[19]do most of the heavy lifting. In

[18]This is analogous to Kriging which should not come as a surprise, since GP's are an extension of Kriging.
[19]You can acess the package at https://www.pymc.io/welcome.html

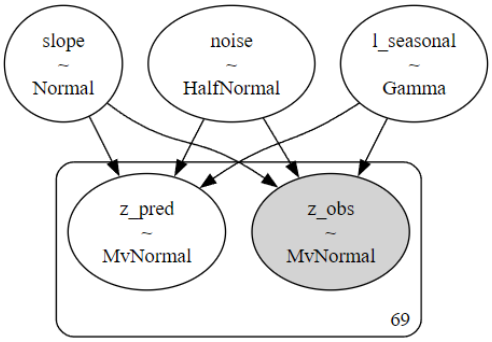


Figure 46: Temporal bayesian model structure

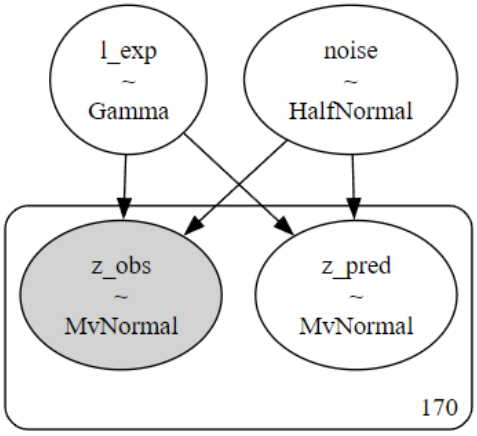


Figure 47: Spatial bayesian model structure

this section, the reader should grasp the key concept that optimal kernel parameters can be determined by assigning probability distributions to their parameters and iteratively refining them.

6.6 Space and time prediction

In practice, it is common to combine multiple kernels to get a better fit to the data (as discussed in Sections 4.3 and 5.3). Bayesian methods do not limit us to using a single kernel; in fact, they often benefit from a richer prior[20], allowing for more flexibility and improved modeling.

6.6.1 Temporal GP's

We obtain the log Meta user count time series for the aggregated municipality of Agusan del Sur $\mathbf{a}(\mathbf{t}|\mathbf{s_i})$, as defined in Section 5.1.6, and fit a temporal Bayesian model to this data. Adescription of the model can be found below and in figure 46 where :

$$Z_t(t) \sim \mathcal{GP}(m(t), k_{\text{periodic}}(t, t')) \tag{6.9}$$

- $m(t)$ is the mean function, typically modeled as a linear trend: $m(t) = \text{slope} \cdot t$

[20] We will generally use additive rules to build conjugate kernels such that $K1 = K2 + K3$ (Flaxmann, 2015)

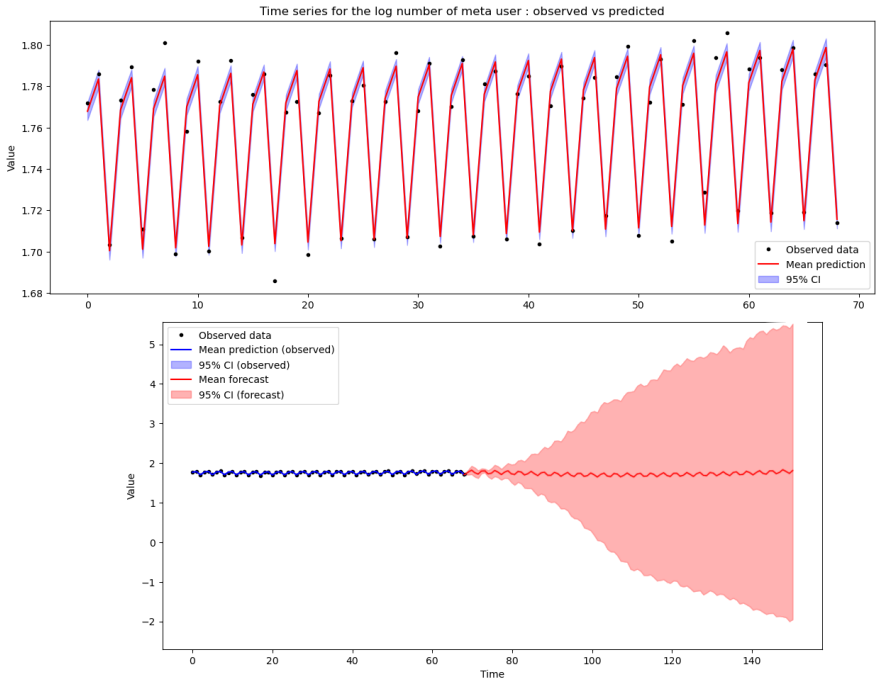


Figure 48: Gaussian process modeling and interpolation for temporal data.

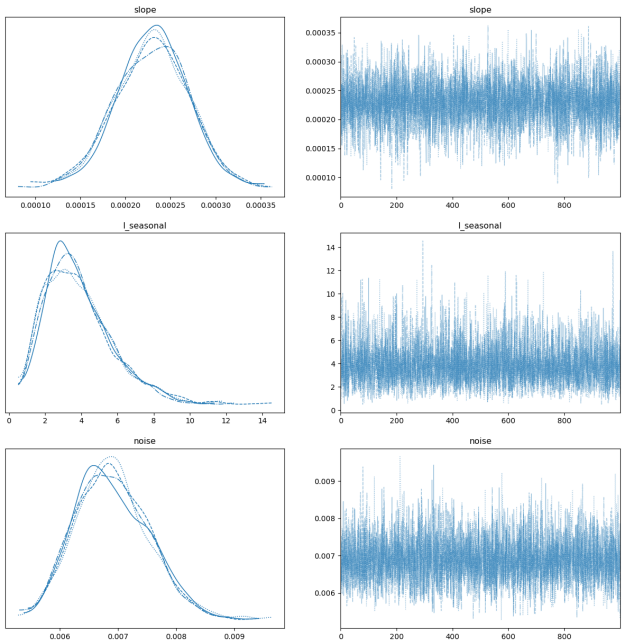


Figure 49: Trace plots indicating the convergence and distribution of the sampled parameters.

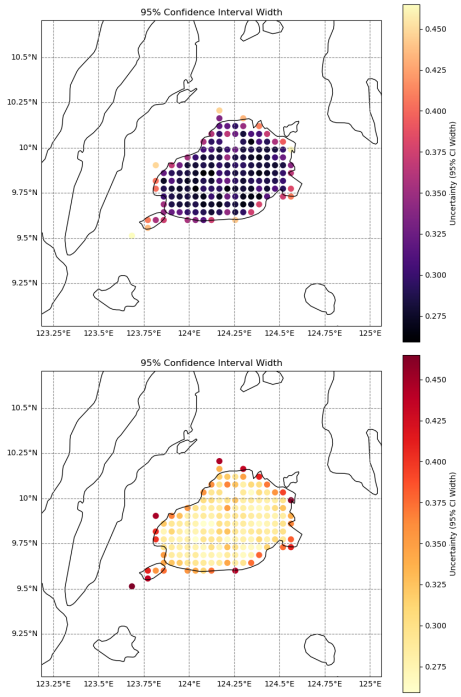


Figure 50: Gaussian process modeling for spatial data.

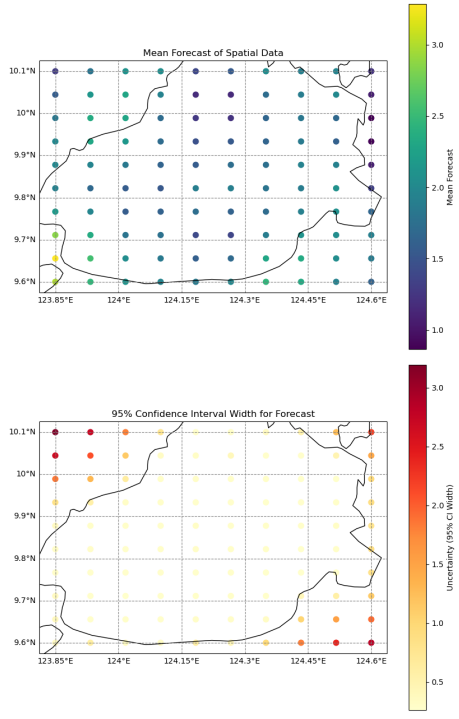


Figure 51: Gaussian process interpolation for spatial data.

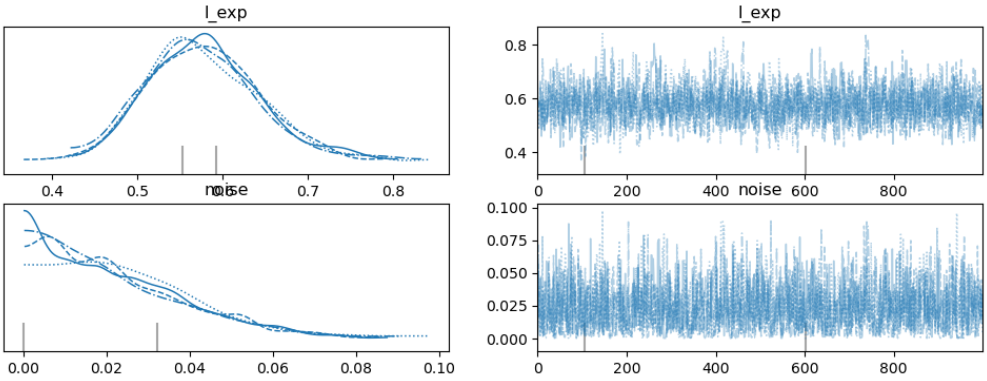


Figure 52: Gaussian process interpolation for spatial data.

Here, the slope is weak so we take as small prior, slope $\sim \mathcal{N}(0, 0.1)$.

- $k_{\text{periodic}}(t, t')$ is the periodic kernel with a gamma prior on l[21], $l_{\text{seasonal}} \sim \text{Gamma}(1, 1)$
- The predicted values z_{pred} follow a multivariate normal distribution: $z_{\text{pred}} \sim \text{MvNormal}(\mathbf{m}, \mathbf{K})$, where $\mathbf{m}$ is the mean vector and $\mathbf{K}$ is the covariance matrix as described above.
- A prior on the noise $\varepsilon \sim \text{HalfNormal}(\sigma)$ where σ is a small positive value is added to the prediction.

The predicted values $z_{\text{pred}}(t)$ are the outcomes from this temporal gaussian process, incorporating both the linear trend and the periodic structure.

Looking at figure 48, it is clear that the GP predictor is accurately modeling the data. The prediction variance increases with time as expected. Crucially, we did not need to make a single stationarity assumption to model this data and all parameters were automatically learned by the Bayesian framework. Figure 49 represents the parameters chain after convergence of the iterative process. We see that Bayesian modelling represent parameters as distribution over probable values rather than point-wise estimations.

6.6.2 Spatial GP's

Bayesian modeling for spatial Gaussian process are very similar to their temporal counterpart. As such we obtain the spatial distribution of the log Meta user count on the island of Bohol for the 30st of march $\mathbf{z}(\mathbf{s}|t_j)$. A description of the model can be found below and in figure 47 where :

$$Z_{\mathbf{s}}(\mathbf{s}) \sim \mathcal{GP}(m(\mathbf{s}), k_{\text{exp}}(\mathbf{s}, \mathbf{s}')) \tag{6.10}$$

- $m(\mathbf{s})$ is the mean function, typically modeled as a constant mean function over the spatial domain. We take a prior: $m(\mathbf{s}) \sim \mathcal{N}(0, 0.1)$.
- $k_{\text{exp}}(\mathbf{s}, \mathbf{s}')$ is the spatial exponential kernel, which models the covariance between points in space. The length scale parameter l_{spatial} governs the smoothness of the spatial process. A prior for this parameter is chosen as $l_{\text{spatial}} \sim \text{Gamma}(1, 1)$.
- The predicted values z_{pred} follow a multivariate normal distribution: $z_{\text{pred}} \sim \text{MvNormal}(\mathbf{m}, \mathbf{K})$, where $\mathbf{m}$ is the mean vector and $\mathbf{K}$ is the covariance matrix as described above.

[21] See definition in table 4.

- A prior on the noise $\varepsilon \sim \text{HalfNormal}(\sigma)$ where σ is a small positive value is added to the prediction.

Similarly to what we observed in the temporal GP model, we see in figure 50 that the spatial GP is accurately modelling the data. The modelling uncertainty increases with the number of neighbour locations where a location with less neighbour is more uncertain. This is a major difference with Kriging where prediction at known location was exact, such is not the case for GP modelling. This uncertainty is important in spatial models and bayesian statistics offer us an simple way to assess it. Spatial interpolation can also be performed using the conditional properties and we observe in figure 51 that once again points with less neighbour locations are more uncertain. Importantly, the forecasting uncertainty is more important then the modelling uncertainty. Figure 52 represents the parameters chain after convergence of the iterative process.

6.7 Main advantages

In this section, we summarize the main advantages of Bayesian machine learning using Gaussian Processes (GPs) in space-time settings:

- **Flexibility:** GPs are highly flexible models capable of describing temporal, spatial, or spatio-temporal processes with equal effectiveness. They can model any type of autocorrelated datasets, making them versatile tools for a wide range of applications.
- **Interpolation Capabilities:** Due to their conditional properties, GPs are excellent for interpolation. They allow for accurate predictions in regions where data may be sparse, leveraging the information from existing data points.
- **Confidence Intervals and Variance Estimation:** Bayesian frameworks naturally provide confidence intervals and prediction variance estimates, tasks that are traditionally challenging in classical statistical settings. This built-in capability is a significant advantage of using GPs within a Bayesian context over more traditional (likelihood maximisation) contexts.
- **Parameter Selection:** Unlike classical temporal and spatial statistical methods, Bayesian frameworks with GPs eliminate the need for manual parameter selection, which can be a complex and error-prone process. This automation leads to more reliable and consistent models at the condition that priors are well conditioned.
- **No Need for Stationarity Assumptions:** GPs do not require stationarity assumptions since they can incorporate these considerations directly within the prior distributions. This flexibility allows for the modeling of more complex, non-stationary processes, which is often difficult in traitionnal methods.

  Given these advantages, we argue GPs within a Bayesian framework represent a superior modeling strategy for space-time processes. They often necessitate however a much stronger computational power and a deep understanding of bayesian statistics to be exploited at their utmost potential.

7 Crisis state analysis

Having extrapolated the Meta user count in non-crisis states across various times and locations, and established the relationship between covariance functions and kernel theory, we now focus on the percent change in Meta user count **p** during crisis events, specifically typhoons. Detailed definitions of typhoon events and

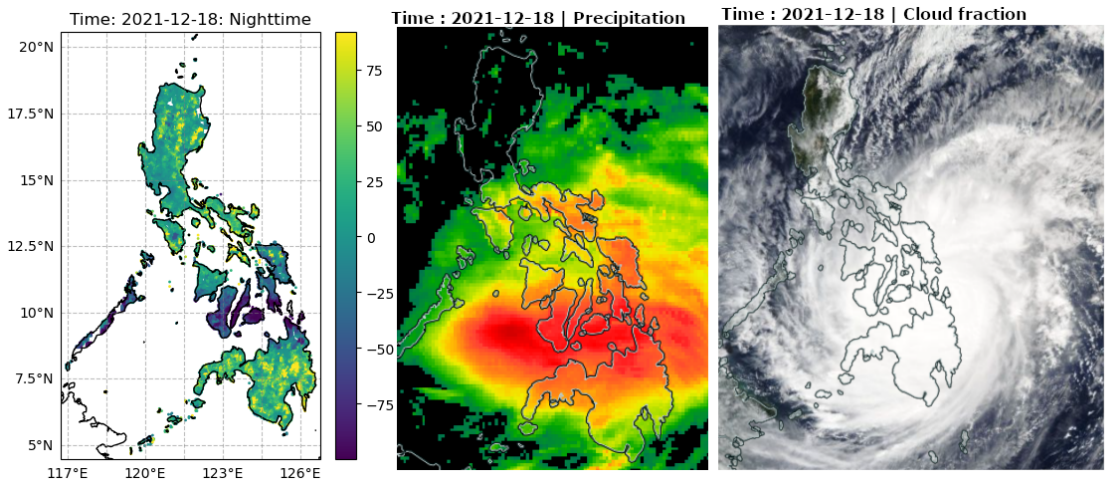


Figure 53: Typhoon RAI visualized with three different metrics: (a) Percent change in Meta users, (b) Precipitation [GPM IMERG], and (c) Cloud fraction [MODIS]. Source: NASA Earthdata.

the percent-change variable are provided in Sections 2.1 and 2.2. It is important to note that a significant percent change typically indicates an anomaly in user numbers, which could be attributed to factors such as network infrastructure failure, population displacement, casualties, or a combination of these during severe crisis events. We cannot distinguish the specific causes behind the low percent change; however, we argue that it remains a valuable proxy for the impact of a crisis on localities, an argument we will illustrate in the following subsections.

For this analysis, we specifically focus on Typhoon Rai, a Category 5 typhoon that impacted the Philippines in December 2021. This typhoon is particularly suitable for crisis state data analysis due to its high intensity and extensive spatial distribution, as it traversed the center of the archipelago from east to west and caused substantial dammage[22]. Consequently, the space-time analysis of this event is expected to yield significant insights into the dynamics of typhoon-related crisis events in the Philippines. Importantly, the variable we study, percent change, is not assumed to be stationary in time and space. On the contrary, during a crisis state, percent change is highly dynamic, fluctuating significantly across both spatial and temporal dimensions. Essentially, a typhoon can be viewed as a dynamic space-time event or shock that disrupts an otherwise stationary - space-time - random process.

7.1 Spatial analysis

Looking at maps of percent-change before during and after typhoon Rai at different scales in figures 53 and , 54 the spatial correlation between typhoon and the percent-change goes without saying. There is an obviously strong percent change decrease in areas that were impacted by the typhoon.

This observation can also be made for Typhoon Goni and Molave (fig 55) , but the anomaly area is smaller, making them less suitable for spatial analysis. This reduced impact may be due to these typhoons crossing highly developed areas like the Manila region, where the superior infrastructure mitigates the perceived effects of the typhoon. Consequently, the infrastructure's resilience in these regions lessens the observable impact on Meta user counts. Lesser category typhoons, such as Krovanh or Dujuan, do not appear to have a direct influence in terms of percent change at a country scale. This suggests that lesser category typhoons

[22]https://en.wikipedia.org/wiki/Typhoon_Rai

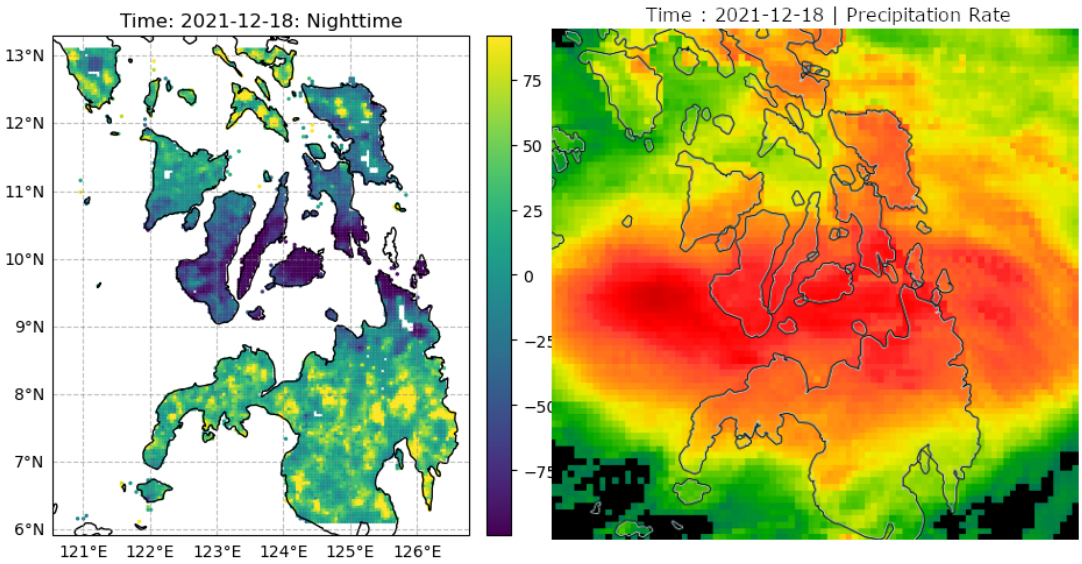


Figure 54: Typhoon RAI visualized with two different metrics and focused on the crisis area: (a) Percent change in Meta users, (b) Precipitation [GPM IMERG]. Source: NASA Earthdata.

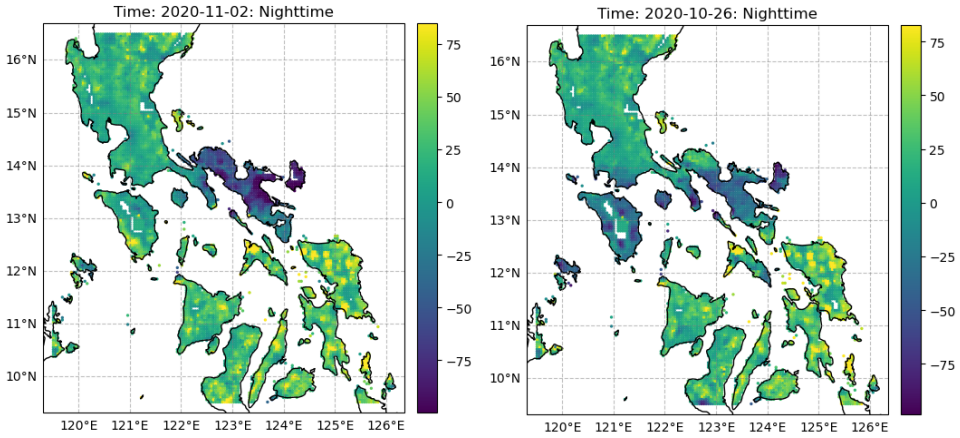


Figure 55: Meta user count percent change for Typhoons (a) Goni (b) Molave.

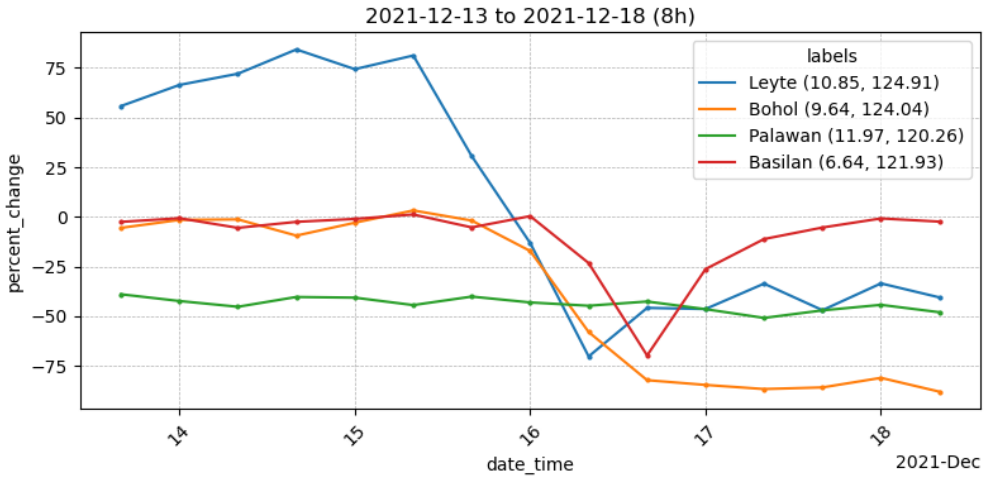


Figure 56: Typhoon Rai time series classes

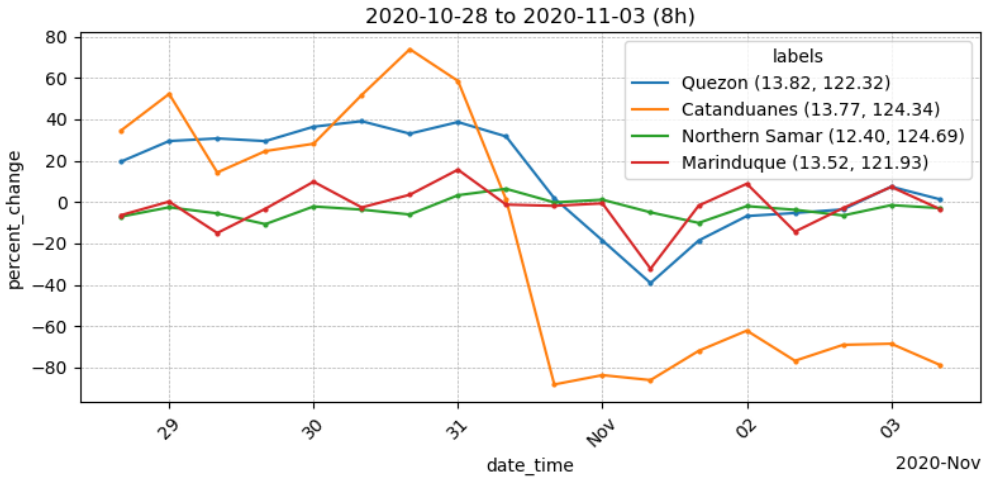


Figure 57: Typhoon Goni time series classes

likely induce changes on smaller local scales or are not strong enough to produce substantial changes in social media user activities. A similar observation can be made of the high category typhoon Kompasu which never hit the Philippines and consequently did not have an important impact.

7.2 Temporal analysis

Focusing on Typhoon Rai, we observe that the time series of Meta user percent change exhibits residual stochastic seasonality that cannot be removed by simply subtracting hourly observations from their corresponding baseline. We will later remove this seasonality altogether by performing daily aggregation.

By examining the time series data across various locations relative to the typhoon center, we reveal four distinct classes of behavior (fig 56). In the first class illustrated by a location in Palawan, the time series shows no significant decrease, indicating a location unaffected (or barely affected) by the crisis. In the second class such as the Basilan location, the time series is impacted by the typhoon, but the effect is short-lived, with values returning to normal within a day, suggesting a low impact. In the third class for instance in

Leyte, the time series exhibits a delayed return to normal, indicating a moderate but recoverable impact. Finally, in the fourth class, the time series in Bohol shows a severe impact, with no recovery even after three weeks, indicating a prolonged crisis state. Therefore, the percent change in Meta user count proves to be a valuable proxy for assessing **crisis impact** when analyzed through its spatial and temporal dynamics.

For Typhoons Goni and Molave (Fig 57), similar patterns are observed. However, not all time series fit neatly into one of the predefined classes, in practice some locations experience a blend of impacts, or behaviours that are hard to categorize. This problem highlights the limits in classifying every location purely by visual inspection.

To accurately assess and classify crisis recovery, we need to employ more refined unsupervised learning techniques. A crucial first step is to visualize the data in a reduced feature space, a process known as dimensionality reduction. Dimensionality reduction (DR) can help distinguish between different classes and provides a clearer understanding of the data structure in a lower-dimensional visual space. In the context of multivariate time series, such as space-time data, dimensionality reduction could involve condensing the time dimension into as few dimensions as possible. For instance, with approximately two weeks of data and 8 hours time steps, we could transform the data from a 16-dimensional temporal feature space $\mathbb{R}^{16}$ to a 2-dimensional embedding space $\mathbb{R}^2$ (Rauber et al., 2016), (You & Hung, 2020). We could also perform a dimensionality reduction in space; however, in our case, this approach would likely not yield useful results for identifying effective crisis recovery classes. We do not need to perform spatial dimensionality reduction to understand that there are essentially three spatial classes: crisis-related low percent change, non-crisis related high percent change at high variability locations, and non-crisis related zero percent change at low variability locations. While spatial dimensionality reduction can be beneficial in many contexts, such as analyzing pollution patterns across different regions (Wikle et al., 2019), it is not suitable for our study.

7.3 Transforming percent change

There are two key reasons for transforming the percent-change variable. First, it eliminates the need to account for seasonality when analyzing percent-change behavior over time. Including seasonality could lead our machine learning algorithm to capture seasonal fluctuations rather than changes in the overall mean of the temporal process, which is our primary focus. Therefore, we apply a daily temporal aggregation to the time series to obtain the aggregated measure $\bar{p_d}(t|\mathbf{s_i})$ (refer to Section 2.3.2 for temporal analysis).

Once we have this aggregated dataset, it is crucial to correct for variance. The percent-change values can be disproportionately large in areas with low population density, similar to the issue we encountered with the number of users $\mathbf{z}$ (see Section 3.2), but with the relationship inverted. Low population density leads to higher percent-change values because changes represent a larger portion of the population, while in high-density areas, these changes are more stable in percentage terms.

Since percent-change can range from -100 (indicating a complete absence of users) to positive values, a standard logarithmic transformation is not directly applicable. Instead, we use the transformation $log(\bar{p_d}(t|\mathbf{s_i}) + 100)$. This transformation improves the data distribution without affecting our primary goal, which is dimensionality reduction (and later classification) rather than visualization. Consequently, the transformation poses no significant issues.

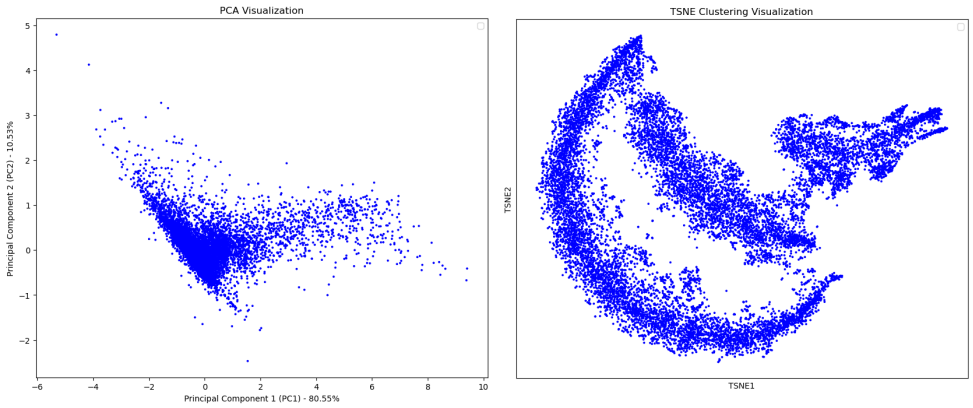


Figure 58: PCA vs t-SNE for time series DR

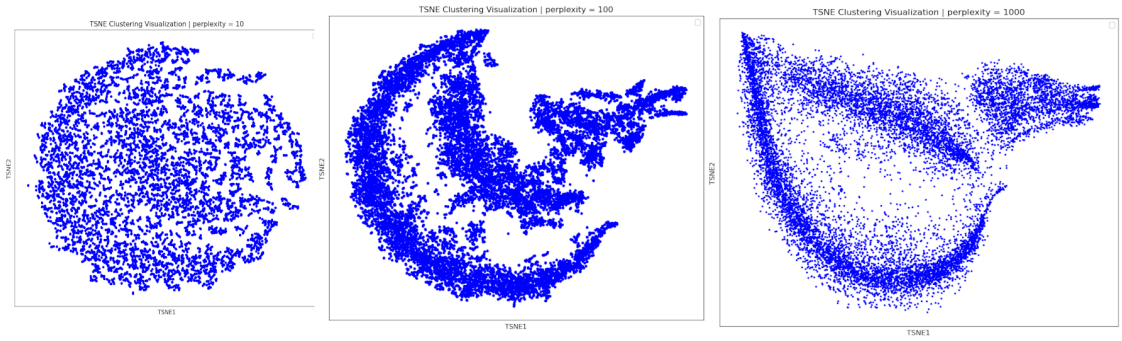


Figure 59: t-SNE visualization for different perplexity parameters.

7.4 Dimensionality reduction

The field of dimensionality reduction (DR) and clustering are vast and rapidly evolving, with numerous techniques worthy of extensive study from statistics to machine learning and deep learning. However, the goal of this master's thesis is not to compare various DR methodologies (even if it is hard to avoid as DR enthousiasts), but to apply DR in the context of space-time data analysis to gain new insights into the variable at hand. Specifically, we aim to reduce dimensionality in the temporal domain to classify time patterns and distinguish between different crisis recovery classes.

One of the most classical DR methods is Principal Component Analysis (PCA). Despite being over a century old, PCA remains a powerful and practical technique and is widely used in a lot of different fields which is why we mention it quickly. PCA works by transforming the data into a new reduced coordinate system (principal components), where the overall variance is preserved as much as possible. In our case we do not wish to preserve variance per say but we would instead like to preserve the local structure of the data in the reduced space. Indeed, time series that are close to each other in a 15-dimensional temporal space should stay close in a reduced 2-dimensional space to give us an idea of cluster arrangements and numbers. In practice, t-SNE (t-distributed Stochastic Neighbor Embedding), a local structure dimensionality reduction technique, often proves more effective for visualization than PCA due to its ability to preserve local structures (der Maaten, 2008). This advantage is illustrated in Figure 58, where we see that PCA very efficiently captures two sources of variation but fails to effectively separate the data, unlike t-SNE.

While a detailed explanation of PCA was not given as the technique is assumed to be known and does not play a significant role in the thesis, such is not the case with t-SNE which has a significant relation with our thesis emphasis on kernel modeling. As a local structure preservation technique, t-SNE naturally must evaluate the probability that two given points are neighbors. Naturally, a very good way to measure similarity to neighbours is to use the (RBF) kernel covered in section 6 where we established the relationship between the covariance function $C(k)$ and the Kernel K. The conditional probability of being a neighbor of a point i, $P_{j|i}$ in a high dimensional space can then simply be defined as (der Maaten, 2008):

$$P_{j|i} = \frac{\exp\left(-|x_i - x_j|^2/2l^2\right)}{\sum_{k\neq l}\exp\left(-|x_k - x_l|^2/2l^2\right)} \tag{7.1}$$

The sum in the denominator ensures that the probabilities sum up to 1, effectively normalizing the probabilities and forming a conditional probability distribution. The final join probability is defined as

$$P_{ij} = \frac{P(j|i) + P(i|j)}{2n} \tag{7.2}$$

The goal of t-SNE is to match the neighborhood probability in the high-dimensional space with the neighborhood probability in the lower-dimensional embedding space by minimizing the Kullback-Leibler (KL) divergence (eq 7.4). High-dimensional spaces suffer from the curse of dimensionality, where distances between points become less informative as they tend to become similar, making it challenging to preserve local structures when reducing dimensions. To address this, t-SNE uses a Student's t-distribution with one degree of freedom for the low-dimensional space. The heavy tails of the Student's t-distribution allow it to better handle the spread of data, ensuring that nearby points stay close while distant points are pushed further apart. This helps in preserving the local and global structure of the data. The joint probability Q_{ij} of points

i and j of being neighbors in the low-dimensional space is defined as:

$$Q_{ij} = \frac{\left(1 + \|y_i - y_j\|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \|y_k - y_l\|^2\right)^{-1}} \tag{7.3}$$

With the KL divergence defined as :

$$KL(P\|Q) = \sum_{i \neq j} P_{ij} \log\left(\frac{P_{ij}}{Q_{ij}}\right) \tag{7.4}$$

Importantly, minimizing this KL divergence, a probability distribution divergence metric, ensures that the low-dimensional embedding preserves the local structure of the high-dimensional data, effectively overcoming the challenges posed by the curse of dimensionality.

Naturally since t-SNE is dependent on RBF kernels to define $P_{j|i}$ and $Pi|j$ we will need to play with the width l of the kernel as well as the distance metric which in this context is assumed to be Euclidean. In practice, in the data cloud the density is not always the same at all locations so the width should change at for every point i. To manage this adaptation, we use a parameter called *perplexity*. Perplexity essentially controls the effective number of neighboring data points considered by the RBF kernel. A larger perplexity results in smoother/larger kernels across the board though the exact value of the kernel width will be dependant on local data density for a given point i, and conversely for a small perplexity. An illustration of the effect of perplexity is given in figure 59. We observe that a low perplexity value captures only very local structures and fails to embed the potential groupings of the high dimensional dataset. On the other hand, a high perplexity value includes too many points as neighbours, blurring natural groupings together, although this effect is somewhat mitigated by the *early exaggeration* hyperparameter.

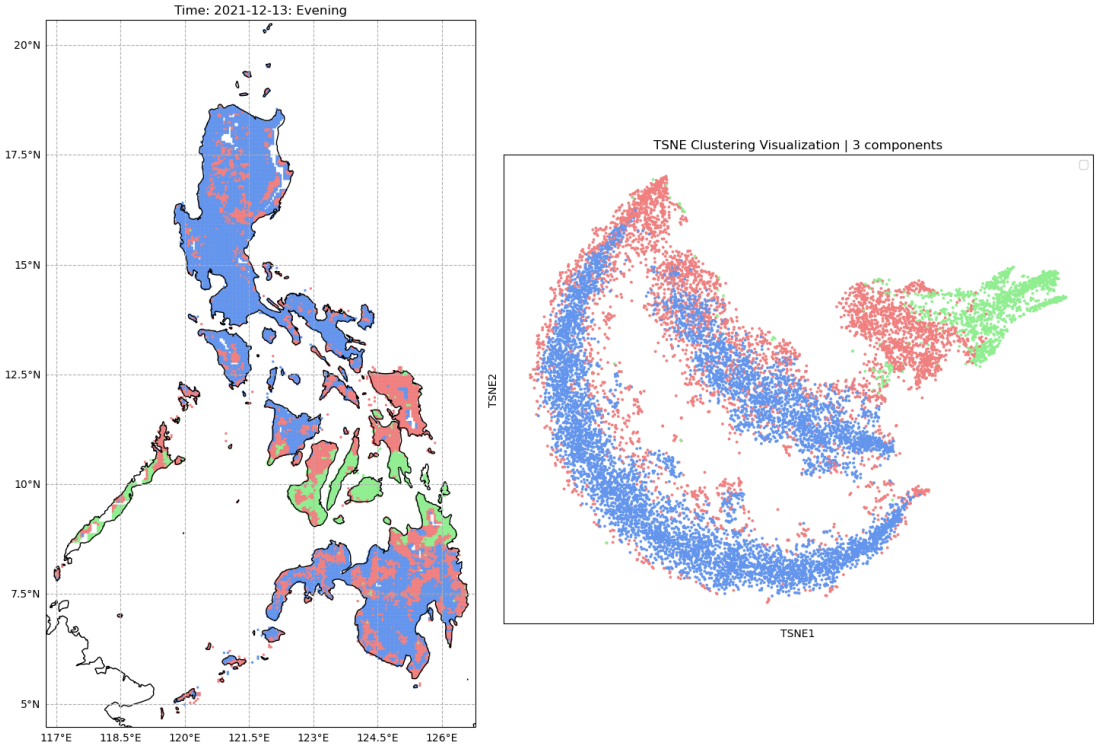
Early exaggeration is another hyperparameter that controls the separation between clusters in the early stages of the t-SNE optimization process. Adjusting early exaggeration can significantly impact the resulting visualization, making it easier to discern distinct groups in the data, in this thesis we keep it to its default levels[23] which is 12, since we deem default results sufficient.

7.5 Classification using gaussian mixtures

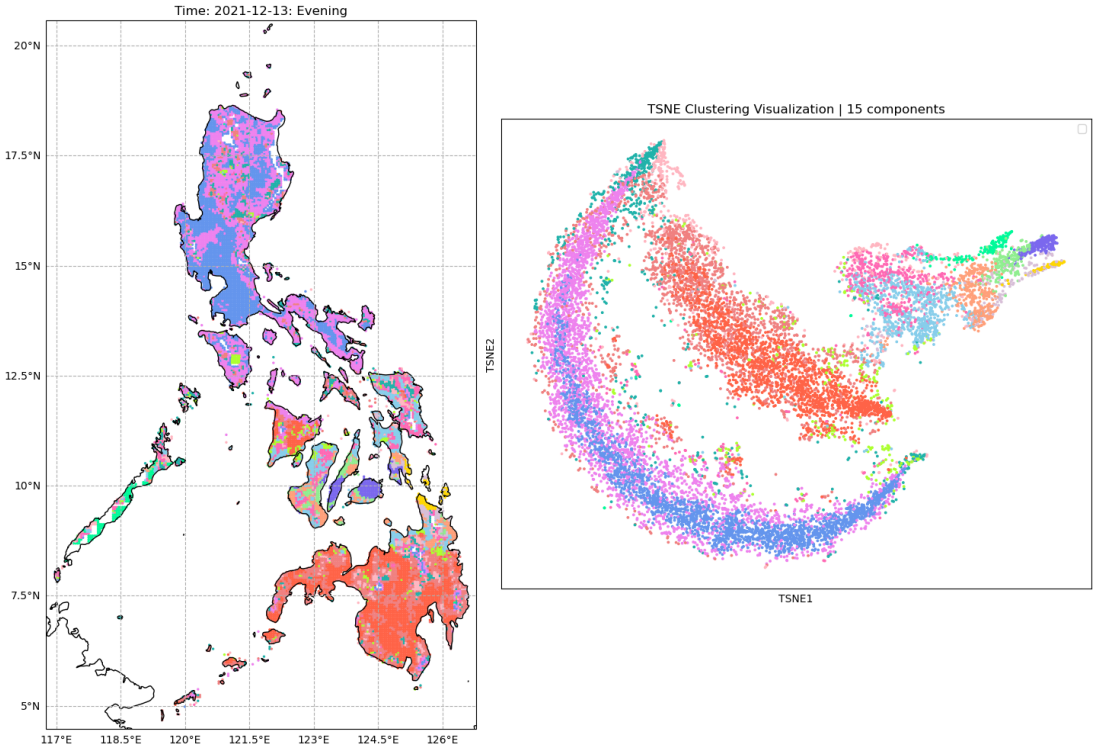
Now that we have a method to reduce dimensionality by focusing on local similarities, we seek a similar to find a good approach for classification. In practice we could very well use the vary famous k-means algorithm or its kernel equivalent which uses similarities instead of euclidean distances (Hafner, 2024). However, this approach makes hard predictions, assigning each data point to at most one class. Instead, we prefer a method where each point can belong to multiple classes with varying probabilities of membership. An excellent way to deal with this is to use Gaussian Mixtures. A Gaussian Mixture (GM) model assumes that the data is generated from a mixture of several (multivariate[24]) normal/gaussian distributions, each representing a different cluster. Each cluster k is represented by its own mean vector $\boldsymbol{\mu_k}$ its covariance matrix $\boldsymbol{\Sigma_k}$ and its mixing coefficient/class probability π_k the probability of the overall GM model is a weighted sum of these

[23] See python package sci-kit learn : https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html

[24] If the column space is bigger then 1.



(a) 3 mixture components



(b) 15 mixture components

Figure 60: Comparison of GMM cl using (a) mapping (b) t-SNE embeddings

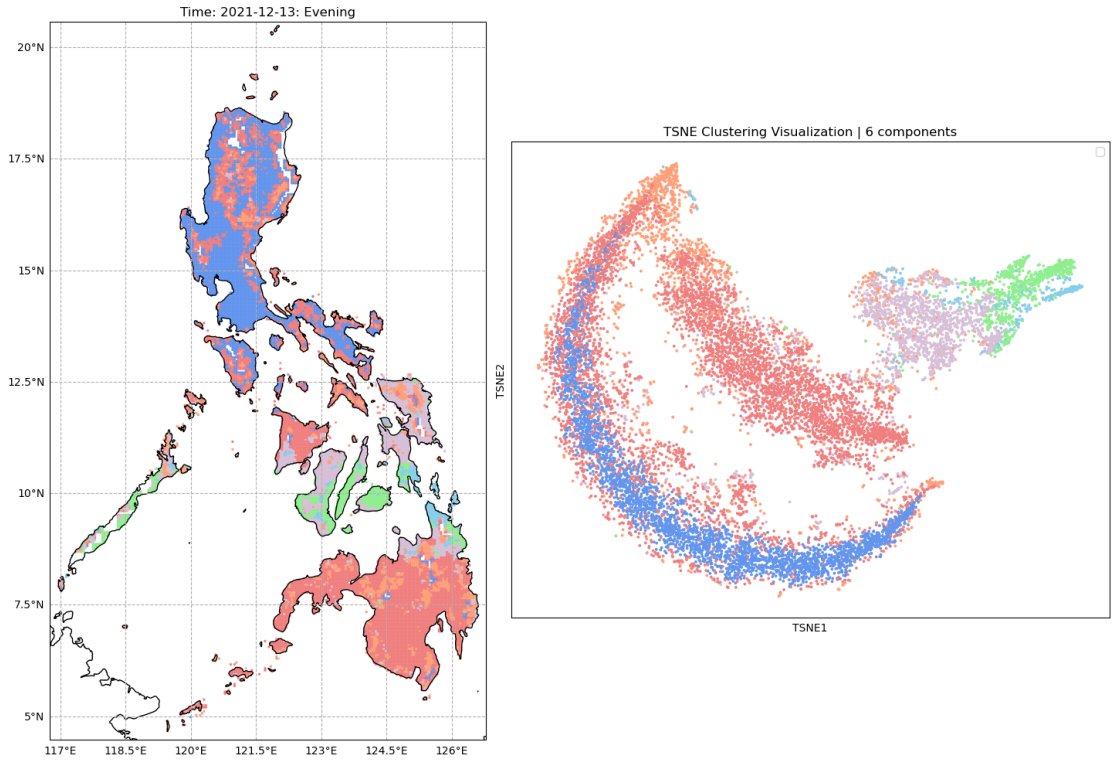


Figure 61: 6 mixture components

Gaussian component densities (Bishop, 2006):

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{7.5}$$

The responsibility γ_{ik} is the metric we use for classification. It is defined in a Bayesian[25] setting as the probabilities of a class conditionally on a data-point.

$$\gamma_{ik} = P(C_i = k \mid X_i = x_i) = \frac{P(X_i = x_i \mid C_i = k)P(C_i = k)}{P(X_i = x_i)} \tag{7.6}$$

Which is equivalent to :

$$\gamma_{ik} = \frac{\mathcal{N}(x_i \mid \mu_k, \Sigma_k)\pi_k}{\sum_{j=1}^{K} \mathcal{N}(x_i \mid \mu_j, \Sigma_j)\pi_j} \tag{7.7}$$

Crucially, the responsability γ_{ik} we use for classification also intervenes in an iterative expectation maximisation (EM) algorithm used to find the best parameters $\theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^{K}$. A comprehensive description of the EM algorithm and how it relates to GMM can be found in (Bishop,2006) and (Hafner, 2024). What is essential to understand is that once the parameters θ are fitted to the observed data, in our case the temporal dataset with T dimensions, a point is classified as belonging to a class using the highest responsability value found in equation 7.7.

One we have classified all points, we can show them directly over the t-SNE embeddings to study how

[25](Bishop, 2006) actually classifies gaussian mixture models as bayesian models since there is an intervention of the Bayes formula.

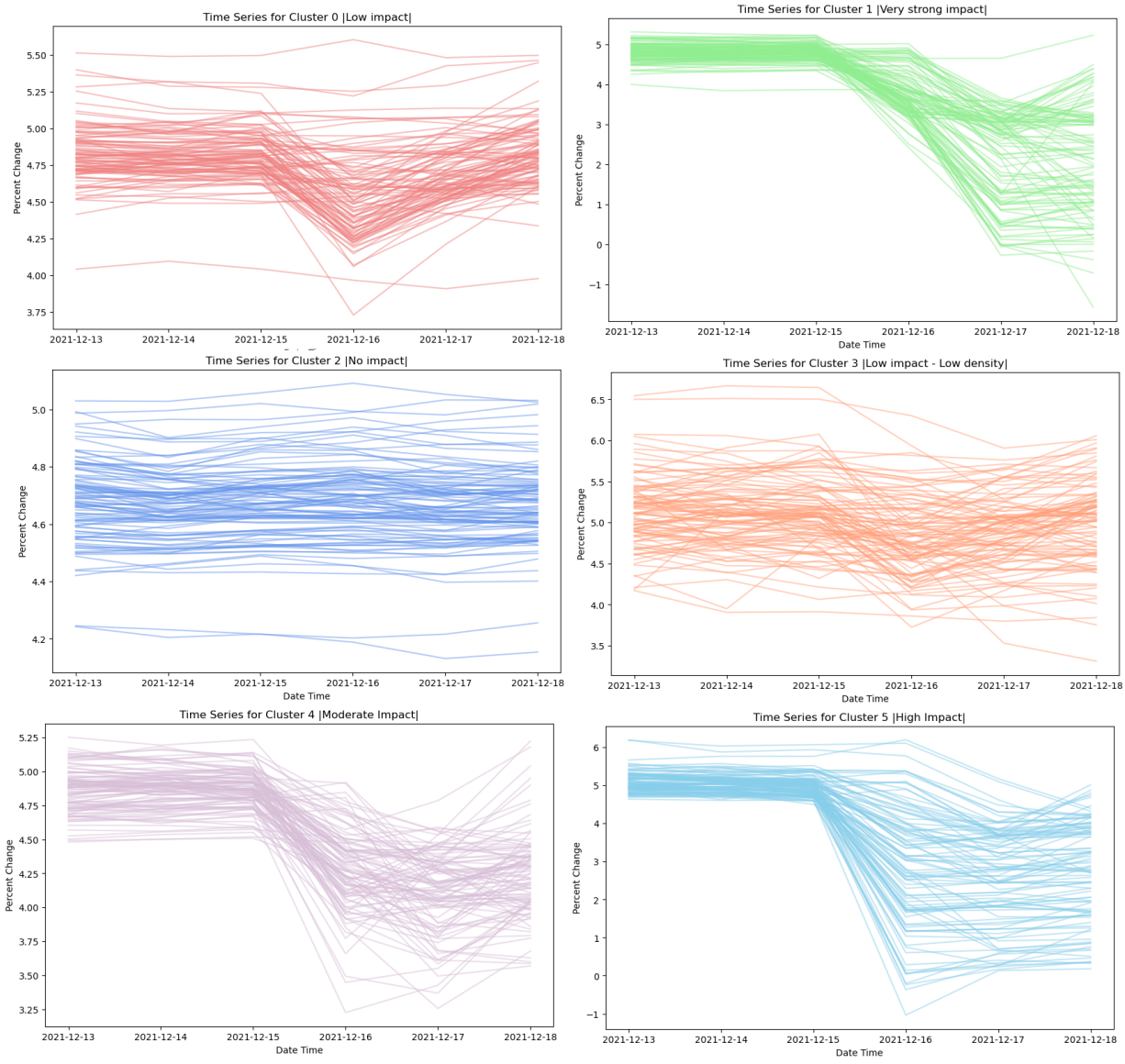


Figure 62: GMM classes : time series visualisation

the classes are behaving in time and if they appear to be well separated in a lower dimensionnal space. We can also map the classes directly over the Philippines to compare them with the percent change. We do it for different 2, 6 and 15 gaussian components in figures 60a to 60b. The reader should be careful, colors on the DR charts and mappings are not equivalent to the ones used earlier in figure 56 when we assed classes visually.

We observe looking at the clusters on the maps and embedding spaces that even with a relatively low number of mixtures, the model is able to capture space-time anomalies related to crisis events. In Figure 60a, where there are three classes, the green class is most likely associated with crisis events and likely represents the lowest recovery. However, the model struggles to distinguish between the intermediate recovery classes and normal low population density time series. This is not surprising because the anomaly at those locations for 3 mixture components is probably not prominent enough to stand out in the temporal space, so it tends to align more closely with time series that have generally lower values. Oppositely When the number of mixtures is too large such , as shown in Figure 60b, the model tends to overfit, especially in time series affected by crisis events. This suggests that many of these time series experience different crisis impacts locally, leading to significant variations in recovery behavior depending on the location. While a large number of mixture components may be too complex for a straightforward classification, they do reveal a substantial amount of features in the dataset that would otherwise be difficult to detect. For instance, it becomes clear from the map that a higher number of mixtures can distinguish between rural and urban settings, as well as road infrastructure. This result is very promising, as it indicates that time series data inherently carry demographic information. We could further explore this by analyzing each class in more detail, attempting to replicate them in other countries and/or crisis events settings, to see if parallels can be drawn. We argue this is a promising and exciting research prospect.

To keep a good classification power without sacrificing interpretability we use 6 mixture components in figure 61. We find on the t-SNE embeddings that this number of embeddings is the one which offers the best balance. Visualizing time series at each class allows us to understand how those classes behave in time and to put a formal name on them, such is done in figure 62.

- Class 0 and 3: **Low impact**
  - Exhibit a low, short-lived crisis impact.
  - The difference between these classes is mainly attributed to population density, as observed in map 61 and the time series.
- Class 1: **Very strong impact**
  - Shows a very strong crisis impact with no signs of recovery.
- Class 2: **No impact**
  - Experiences no crisis impact, indicating unimpacted locations.
- Class 4: **Moderate impact**
  - Undergoes a moderate crisis impact where values don't fall to zero but gradually recover.
- Class 5: **Strong impact**
  - Faces a high crisis impact with values dropping to zero, followed by a slow recovery.

7.5.1 Probability distribution interpretation

A significant property of gaussian mixture models is that they are not a classification model per se but instead a probability distribution model (specifically, a mixture of multivariate Gaussians). This has two profound implications:

- Data Generation: The fitted distribution can be used to generate new synthetic data. This means the GMM serves as a data generator, creating new data points that follow the same statistical properties as the original data.
- Interpolation and Imputation: Missing values can be found using conditional expectations within the GMM framework. This should allow Gaussian Mixture Models to solve interpolation and missing data issues. Whether or not this is a practical solution for temporal and spatial data is not explored in ths thesis.

7.5.2 Difference between GMM and GPs

Gaussian mixture models (GMMs) and Gaussian processes (GPs) may seem similar because they both involve multivariate normal distributions and can operate within a Bayesian framework due to their conditional properties. However, they are fundamentally different. GPs can be understood as a single, infinitely-dimensional multivariate normal distribution, effectively representing an entire function. In contrast, Gaussian mixtures are exactly what their name suggests—a mixture of multiple normal distributions, each with its own finite dimensionality.

7.6 Space-time sensibility index

To construct a Crisis Sensibility Index (CSI), we start by defining a mapping function $f : C \to I$, where C represents the set of crisis impact classes, and I denotes the corresponding numerical impact values. Given a well-defined set of crisis impact classes, we can assign numerical values that reflect the severity of the impact with a certain degree of confidence.

We propose using a power-law relationship to model the increasing impact more accurately. The mapping function is defined as:

$$f(C) = a \cdot (b^C - 1) \tag{7.8}$$

where $a = 1.5$ is a scaling factor that adjusts the overall magnitude of the impact values, and $b = 1.5$ is the base that determines the rate at which the impact value increases with the crisis impact class C. With these parameters, the mapping function produces the following values:

$$\begin{aligned}
f(\text{No impact}) &= 1.5 \cdot (1.5^0 - 1) = 0, \\
f(\text{Low impact}) &= 1.5 \cdot (1.5^1 - 1) = 0.75, \\
f(\text{Moderate impact}) &= 1.5 \cdot (1.5^2 - 1) = 1.875, \\
f(\text{High impact}) &= 1.5 \cdot (1.5^3 - 1) = 3.5625, \\
f(\text{Very high impact}) &= 1.5 \cdot (1.5^4 - 1) = 6.09375.
\end{aligned}$$

This approach provides a mathematical framework where the impact value increases non-linearly with the severity class, allowing nuanced representation of our class uncertainty (since lower impact classes will

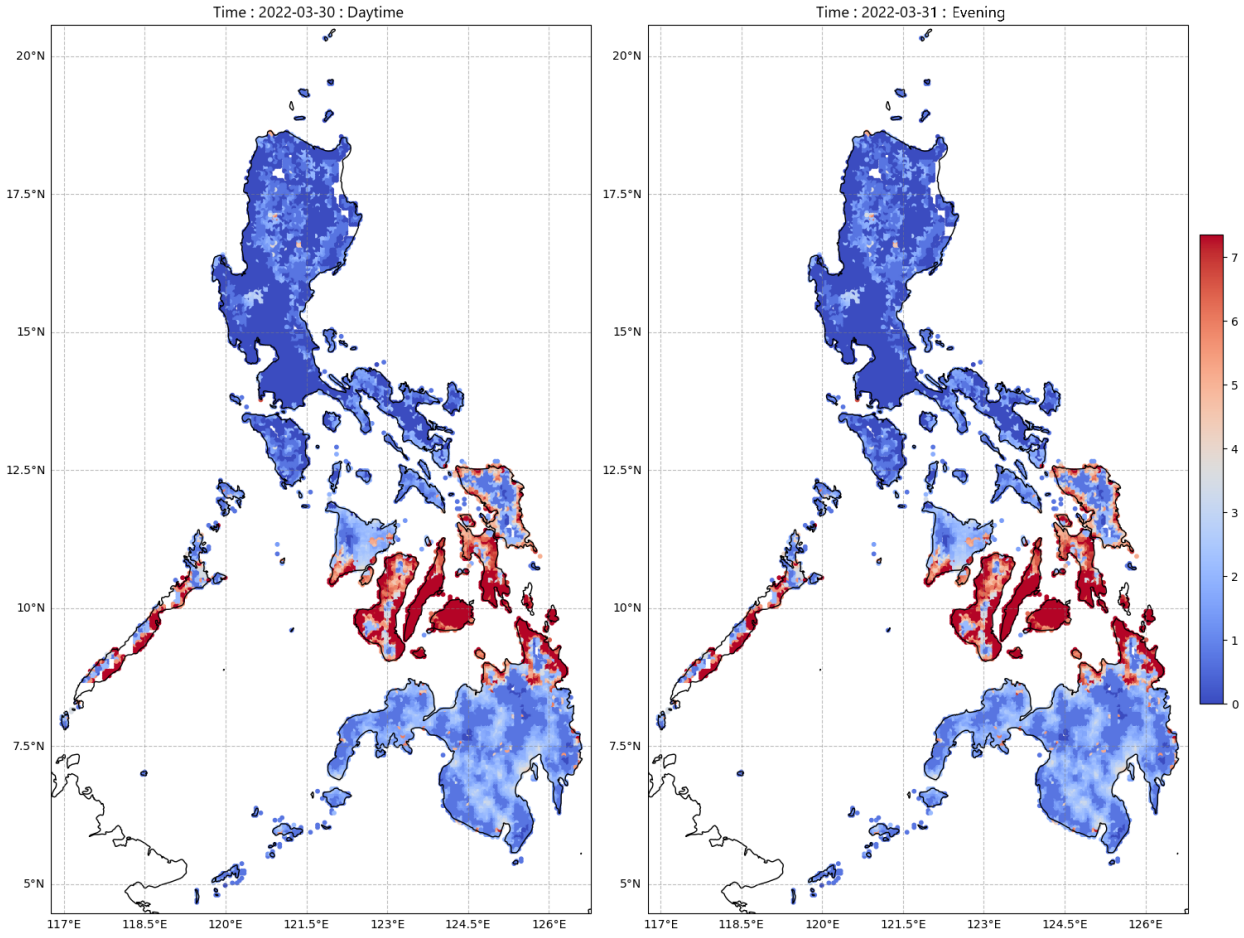


Figure 63: CSI index in space and time for typhoon RAI.

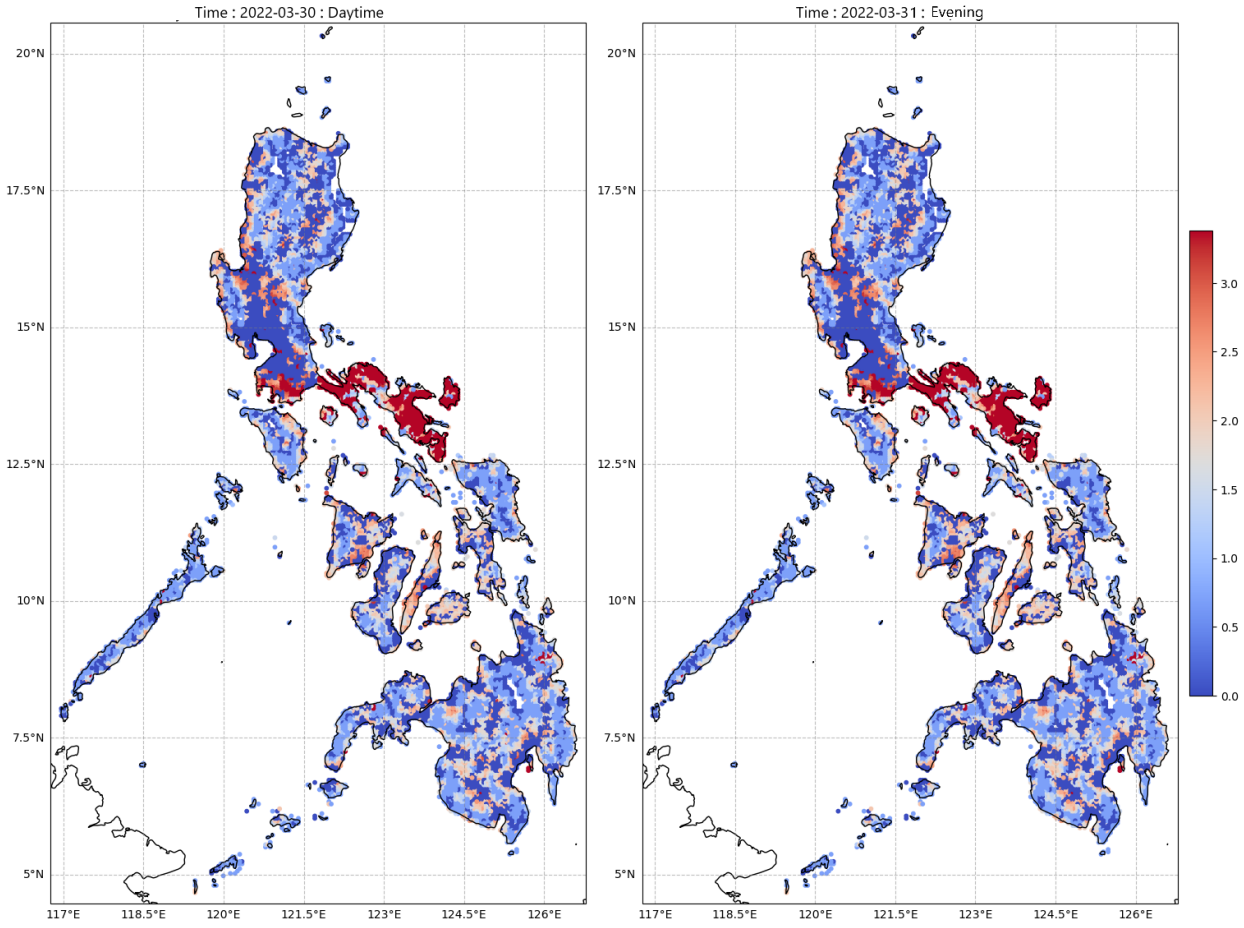


Figure 64: CSI index in space and time for typhoon Goni.

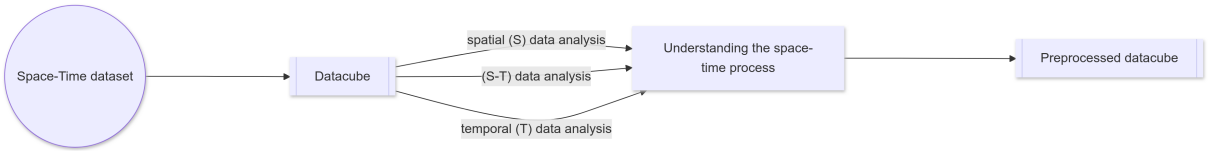


Figure 65: Thesis structure for data anlysis sections.

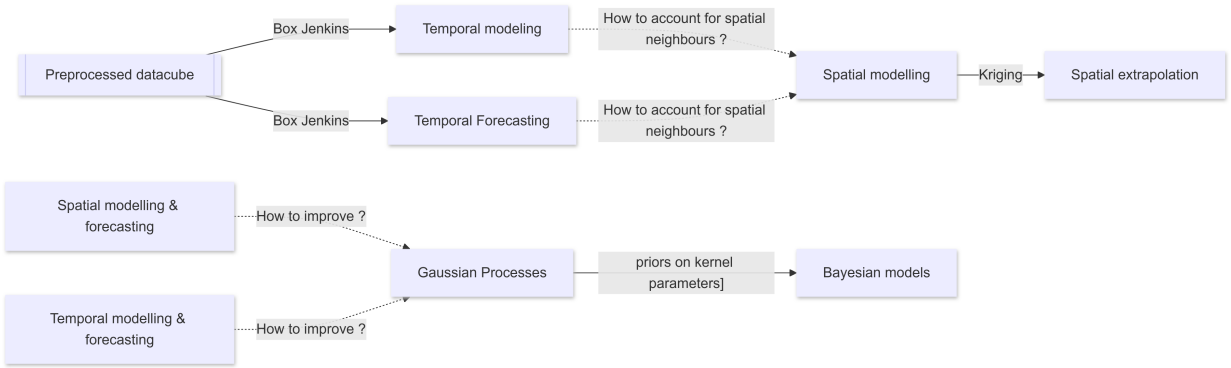


Figure 66: Thesis structure for the Statistical modelling and interpolation sections

have more chance to be misclassified). The parameters a and b can be adjusted to fine-tune the sensitivity of the index according to the specific context and objectives of the analysis. Practitioners have the flexibility to choose the most appropriate mapping, recognizing that different mappings may lead to varying conclusions.

Once we have mapped the impact classes, we build the CSI index using this simple definition

$$CSI = log(z(\mathbf{s}_i, t)) \times I(\mathbf{s}_i) \tag{7.9}$$

effectively building a space-time Crisis sensibility index.

We observe this space-time CSI in practice in figures 63 and 64 presented for Typhoons Rai and Goni. These results highlight the potential for extensive research opportunities. One promising direction is to learn sensitivity patterns across a variety of distinct typhoon situations and ultimately build a static sensitivity likelihhod map across the Philippines. Unfortunately, the current quality of our data does not yet permit such a detailed analysis. Another potential avenue is to perform a space-time interpolation - as explored in Sections 5 and 6 - of crisis sensitivity, particularly after practitioners have refined the impact classes and mapping functions.

8 Conclusion

In this thesis, we explored various approaches to space-time modeling, space-time interpolation, and space-time classification. Our research was conducted using a dataset that was divided into two distinct periods: a relatively stable, non-crisis period and a crisis period characterized by dynamic spatio-temporal events, specifically typhoons. We showed how preemptive analysis could be performed on space-time data regardless of the crisis state and introduced datacubes, a data architecture well fitted for those analysis.

For the non-crisis section, we examined different modeling frameworks, ranging from econometric theories like Box-Jenkins to methods rooted in natural sciences such as kriging. Despite the apparent differences, we

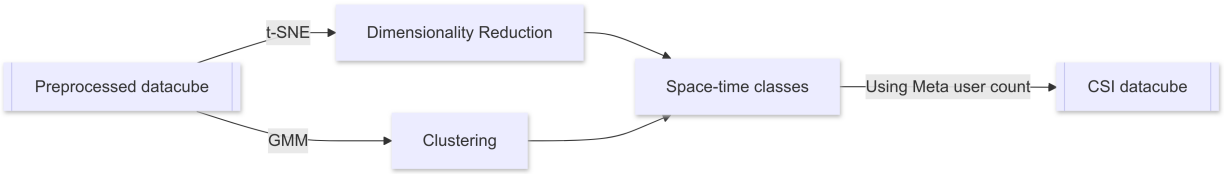


Figure 67: Thesis structure for the machine learning sections

found that these approaches share significant commonalities, as both deal with auto-correlated space and time data, though with different input spaces. Given that our dataset, tracking Meta user counts, exhibited considerable spatial variation but low temporal variation, full joint space-time models were less necessary. To address this and simplify the complex process of parameter selection, we introduced Bayesian modeling for Gaussian processes, which, based on kernel theory, effectively solved many challenges associated with the other models. This method provided greater flexibility and relaxed assumptions, proving particularly useful in our analysis.

During the crisis part of our study, we capitalized on the dynamic changes in space and time to develop typhoon impact classes. A high-impact class was defined by a long recovery time and the total absence of users at specific times. Using methods from kernel theory and Bayesian statistics, we created robust classes and performed dimensionality reduction. Our reduced embedding space allowed us to define these classes and subsequently use them with our space-time Meta user count data to develop a dynamic Crisis Sensitivity Index (CSI), measuring the impact of specific typhoons.

Overall, we highlighted a set of space-time modeling techniques and demonstrated the effectiveness of kernel methods and kernel theory in handling both stationary and highly dynamic space-time processes. These techniques proved applicable to real-world datasets, even those from unconventional sources, providing valuable insights into their strengths and limitations.

8.1 Implications and Future Research

As a final consideration, it is important to acknowledge that many of the challenges addressed in this thesis could potentially be approached using neural networks and other deep learning architectures. There is a thematic connection between space-time models, whether within a frequentist or Bayesian framework, and the principles underlying deep learning. This connection is highlighted in numerous references, such as Cressie (2019) and Bogaert (2024). The application of neural networks to predict multivariate (spatial) time series is currently a highly active area of research, driven by the intrinsic ability of models like LSTMs, transformers, and CNNs to account for both temporal and spatial autocorrelation (Wen et al., 2022; Xia et al., 2020). While our dataset was not well-suited for such an analysis, we encourage further exploration of this promising avenue, as it presents a tremendous opportunity for discovery and innovation.

8.2 Areas for Improvement

A potential area for improvement in this thesis lies in the dataset itself. The data we utilized is limited and contains various constraints, such as the lack of covariates and minimal temporal variation in Meta user count, which is correlated with population density. These factors rendered spatio-temporal models somewhat redundant. An alternative model, perhaps of environmental or econometric nature, might have been more suitable for a comprehensive space-time analysis. However, this choice did not align with our

objective to integrate different areas of research, as the dataset was specifically chosen for its relevance to Dujardin's research in geography (Dujardin, n.d.). Despite these limitations, we are satisfied with our results and believe that we have achieved much of what was possible given the data at hand. Our primary goal was to establish relationships between different areas of research for space-time analysis, and we feel that we have, at least partially, succeeded in doing so.

8.3 Visual summary

For further reference, a visual summary of this thesis structure is provided in Figures 65 and 66 and 67 . This summary encapsulates the key components and findings of our research, offering a concise overview of the methodologies and results discussed throughout the thesis.

A Appendix

A.1 Definition of Distance Metrics

In this section, we define various distance metrics used in the analysis. Each distance metric offers a different way to measure the distance between two points in a space.

A.1.1 Euclidean Distance

The Euclidean distance between two points $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ and $\mathbf{q} = (q_1, q_2, \ldots, q_n)$ in an n-dimensional space is given by:

$$d_{\text{Euclidean}}(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

This metric represents the shortest straight-line distance between the points.

A.1.2 Manhattan Distance

The Manhattan distance or L1 norm between two points $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ and $\mathbf{q} = (q_1, q_2, \ldots, q_n)$ is defined as:

$$d_{\text{Manhattan}}(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^{n} |p_i - q_i|$$

This metric represents the sum of the absolute differences between the coordinates, akin to the distance one would travel along grid lines.

A.1.3 Geodesic Distance

The Geodesic distance is the shortest distance between two points on a curved surface, such as the surface of a sphere. For two points $\mathbf{p}$ and $\mathbf{q}$ on a sphere with radius R, the Geodesic distance is given by:

$$d_{\text{Geodesic}}(\mathbf{p}, \mathbf{q}) = R \cdot \arccos\left(\frac{\mathbf{p} \cdot \mathbf{q}}{R^2}\right)$$

where $\mathbf{p} \cdot \mathbf{q}$ represents the dot product of the vectors $\mathbf{p}$ and $\mathbf{q}$, and R is the radius of the sphere.

A.1.4 Chordal Distance

The Chordal distance between two points $\mathbf{p}$ and $\mathbf{q}$ on a sphere is defined as the straight-line distance through the sphere, connecting the two points. It is given by:

$$d_{\text{Chordal}}(\mathbf{p}, \mathbf{q}) = 2R \sin\left(\frac{\theta}{2}\right)$$

where θ is the central angle subtended by the points $\mathbf{p}$ and $\mathbf{q}$, and R is the radius of the sphere.

This distance is useful when analyzing points on a spherical surface from the perspective of a straight-line connection through the sphere.

A.2 Relationship Between Autocovariance and Semivariance

The relationship between the autocovariance function $C(h)$ and the semivariance function $\gamma(h)$ can be established as follows:

The autocovariance function at lag h is defined as:

$$C(h) = E[Z(x)Z(x+h)],$$

On the other hand, the semivariance function $\gamma(h)$ is defined as:

$$\gamma(h) = \frac{1}{2}E[(Z(x) - Z(x+h))^2].$$

Expanding the square in the semivariance definition:

$$\gamma(h) = \frac{1}{2}E[Z(x)^2 + Z(x+h)^2 - 2Z(x)Z(x+h)].$$

Recognizing that $E[Z(x)^2] = E[Z(x+h)^2] = C(0)$, we have:

$$\gamma(h) = \frac{1}{2}[2C(0) - 2C(h)] = C(0) - C(h).$$

Thus, the semivariance can be expressed as:

$$\gamma(h) = C(0) - C(h),$$

which shows that the semivariance function is the difference between the autocovariance at lag 0 and the autocovariance at lag h.

A.3 ARIMA models

An Autoregressive model of order p, denoted as AR(p), expresses the current value of a time series as a linear function of its past p values plus a random innovation. The An AR(p) model can be written as:

$$X_t = c + \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + \varepsilon_t, \tag{A.1}$$

where:

- X_t is the value of the time series at time t,
- c is a constant term,
- $\phi_1, \phi_2, \ldots, \phi_p$ are the parameters of the model,
- ε_t is the white noise error term at time t.

In backshift notation, the AR(p) model is expressed using the backshift operator B, defined as $BX_t = X_{t-1}$. The model becomes:

$$(1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p)X_t = c + \varepsilon_t, \tag{A.2}$$

or equivalently,

$$\phi(B)X_t = c + \varepsilon_t, \tag{A.3}$$

where $\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p$ is the autoregressive operator.

A Moving Average model of order q, denoted as MA(q), models the current value of a time series as a linear function of current and past q white noise error terms. The MA(q) model can be written as:

$$X_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}, \tag{A.4}$$

where:

- X_t is the value of the time series at time t,
- μ is the mean of the series,
- $\theta_1, \theta_2, \ldots, \theta_q$ are the parameters of the model,
- ε_t is the white noise error term at time t.

In backshift notation, the MA(q) model is expressed as:

$$X_t = \mu + (1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q)\varepsilon_t, \tag{A.5}$$

or equivalently,

$$X_t = \mu + \theta(B)\varepsilon_t, \tag{A.6}$$

where $\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q$ is the moving average operator.

An ARIMA model combines the AR and MA components and includes differencing to make the time series stationary. An ARIMA(p, d, q) model applies d differences to the series and then fits an AR(p) and MA(q) model to the differenced data.

The general form of the ARIMA(p, d, q) model is:

$$\Delta^d X_t = c + \phi_1 \Delta^d X_{t-1} + \cdots + \phi_p \Delta^d X_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q}, \tag{A.7}$$

where $\Delta^d X_t$ denotes the d-th differenced series.

In backshift notation, the differencing operator is represented as $(1 - B)^d$. Therefore, the ARIMA(p, d, q) model can be expressed as:

$$\phi(B)(1 - B)^d X_t = c + \theta(B)\varepsilon_t, \tag{A.8}$$

where:

- $\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p$ is the autoregressive operator,
- $\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q$ is the moving average operator,
- B is the backshift operator,
- c is a constant term,

- ε_t is the white noise error term at time t.

To summarize:

- **AR(p)**: Autoregressive model of order p.
- **MA(q)**: Moving Average model of order q.
- **ARIMA(p, d, q)**: Autoregressive Integrated Moving Average model with order p for the AR part, d for differencing, and q for the MA part.

These models are foundational in time series analysis and forecasting.

A.4 Kriging variance equation proof

With $Z_{\mathbf{s}}(\hat{s_0}) = Z_{\mathbf{s}}(s_0)^*$ and $Z_{\mathbf{s}}(s_0)^* = \sum_{i=1}^n \lambda_i Z_{\mathbf{s}}(\mathbf{s}_i)$, Kriging aims to minimize the variance

$$\text{Var}\left(Z_{\mathbf{s}}(s_0) - Z_{\mathbf{s}}(s_0)^*\right) = E\left[\left(Z_{\mathbf{s}}(s_0) - Z_{\mathbf{s}}(s_0)^*\right)^2\right].$$

This can be expressed as

$$E\left[\left(Z_{\mathbf{s}}(s_0) - Z_{\mathbf{s}}(s_0)^*\right)^2\right] = E\left[Z_{\mathbf{s}}(s_0)^2\right] - 2E\left[Z_{\mathbf{s}}(s_0)^* Z_{\mathbf{s}}(s_0)\right] + E\left[\left(Z_{\mathbf{s}}(s_0)^*\right)^2\right]. \tag{A.9}$$

Substituting $Z_{\mathbf{s}}(s_0)^* = \sum_{i=1}^n \lambda_i Z_{\mathbf{s}}(\mathbf{s}_i)$ into the equation, we get:

$$E\left[Z_{\mathbf{s}}(s_0)^* Z_{\mathbf{s}}(s_0)\right] = E\left[\left(\sum_{i=1}^n \lambda_i Z_{\mathbf{s}}(\mathbf{s}_i)\right) Z_{\mathbf{s}}(s_0)\right] = \sum_{i=1}^n \lambda_i E\left[Z_{\mathbf{s}}(\mathbf{s}_i) Z_{\mathbf{s}}(s_0)\right].$$

And,

$$E\left[\left(Z_{\mathbf{s}}(s_0)^*\right)^2\right] = E\left[\left(\sum_{i=1}^n \lambda_i Z_{\mathbf{s}}(\mathbf{s}_i)\right)\left(\sum_{j=1}^n \lambda_j Z_{\mathbf{s}}(\mathbf{s}_j)\right)\right] = \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j E\left[Z_{\mathbf{s}}(\mathbf{s}_i) Z_{\mathbf{s}}(\mathbf{s}_j)\right].$$

Thus, the expression for the variance becomes:

$$\text{Var}\left(Z_{\mathbf{s}}(s_0) - Z_{\mathbf{s}}(s_0)^*\right) = E\left[Z_{\mathbf{s}}(s_0)^2\right] - 2\sum_{i=1}^n \lambda_i E\left[Z_{\mathbf{s}}(\mathbf{s}_i) Z_{\mathbf{s}}(s_0)\right] + \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j E\left[Z_{\mathbf{s}}(\mathbf{s}_i) Z_{\mathbf{s}}(\mathbf{s}_j)\right].$$

Recognizing that $E\left[Z_{\mathbf{s}}(\mathbf{s}_i) Z_{\mathbf{s}}(\mathbf{s}_j)\right]$ is the covariance function, denoted by $C(\mathbf{s}_i, \mathbf{s}_j)$, and similarly $E\left[Z_{\mathbf{s}}(\mathbf{s}_i) Z_{\mathbf{s}}(s_0)\right] = C(\mathbf{s}_i, s_0)$, and $E\left[Z_{\mathbf{s}}(s_0)^2\right] = C(s_0, s_0)$, the variance simplifies to:

$$\text{Var}\left(Z_{\mathbf{s}}(s_0) - Z_{\mathbf{s}}(s_0)^*\right) = C(s_0, s_0) - 2\sum_{i=1}^n \lambda_i C(\mathbf{s}_i, s_0) + \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j C(\mathbf{s}_i, \mathbf{s}_j).$$

This is the well-known Kriging variance equation in vector matrix form:

$$\text{Var}\left(Z_{\mathbf{s}}(s_0) - Z_{\mathbf{s}}(s_0)^*\right) = \sigma_K^2(s_0) = C(\mathbf{0}) - 2\boldsymbol{\lambda}^T \boldsymbol{\sigma} + \boldsymbol{\lambda}^T \boldsymbol{\Sigma} \boldsymbol{\lambda}$$

A.5 Main Python Packages

In the course of this work, several Python packages have been extensively used for data analysis, geostatistical modeling, machine learning, and visualization. Below is a list of the principal packages used, along with links to their respective documentation:

- **Xarray**: A powerful toolkit for working with labeled multi-dimensional arrays, widely used in atmospheric and climate science.
  - Documentation: http://xarray.pydata.org/en/stable/
- **PyKrige**: A geostatistical toolkit that provides implementations of various kriging methods.
  - Documentation: https://pykrige.readthedocs.io/en/stable/
- **GSTools**: A library for geostatistical modeling, including variogram estimation, kriging, and simulation.
  - Documentation: https://geostat-framework.readthedocs.io/projects/gstools/en/stable/
- **Cartopy**: A library for cartographic projections and geospatial data visualization.
  - Documentation: https://scitools.org.uk/cartopy/docs/latest/
- **PyMC**: A probabilistic programming library for Bayesian statistical modeling and inference.
  - Documentation: https://www.pymc.io/welcome.html
- **scikit-learn**: A comprehensive library for machine learning and data mining.
  - Documentation: https://scikit-learn.org/stable/
- **statsmodels (stattools)**: A library that provides tools for statistical modeling and hypothesis testing.
  - Documentation: https://www.statsmodels.org/stable/index.html
- **Pandas**: A highly efficient library for data manipulation and analysis, particularly with structured data.
  - Documentation: https://pandas.pydata.org/
- **GeoPandas**: Extends Pandas to allow spatial operations on geometric types, facilitating the handling of geospatial data.
  - Documentation: https://geopandas.org/en/stable/
- **Shapely**: A library for manipulation and analysis of planar geometric objects.
  - Documentation: https://shapely.readthedocs.io/en/stable/

B Bibliography

Andresen, M. A. (2021). Modifiable areal unit problem. *CrimRxiv*. https://doi.org/10.21428/cb6ab371.5c28c076

Beenstock, M., & Felsenstein, D. (2007). Spatial Vector Autoregressions. *Spatial Economic Analysis*, 2(2), 167–196. https://doi.org/10.1080/17421770701346689

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Brockwell, J. P. (2016). *Introduction to time series and forecasting*. Springer Science+Business Media.

Chavalier, M., El Malki, M., Kopliku, A., Teste, O., & Tournier, R. (2016). Document-oriented data warehouses: Models and extended cuboids, extended cuboids in oriented document. *2016 IEEE Tenth International Conference on Research Challenges in Information Science (RCIS)*, 1–11. https://doi.org/10.1109/RCIS.2016.7549351

Corani, G., Benavoli, A., & Zaffalon, M. (2021). Time Series Forecasting with Gaussian Processes Needs Priors. In Y. Dong, N. Kourtellis, B. Hammer, & J. A. Lozano (Eds.), *Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track* (Vol. 12978, pp. 103–117). Springer International Publishing. https://doi.org/10.1007/978-3-030-86514-6_7

Cressie, N. A. C. (1993). *Statistics for Spatial Data* (1st ed.). Wiley. https://doi.org/10.1002/9781119115151

Dujardin, S., Jacques, D., Steele, J., & Linard, C. (2020). Mobile Phone Data for Urban Climate Change Adaptation: Reviewing Applications, Opportunities and Key Challenges. *Sustainability*, 12(4), 1501. https://doi.org/10.3390/su12041501

Elhorst, J. P. (2003). Specification and Estimation of Spatial Panel Data Models. *International Regional Science Review*, 26(3), 244–268. https://doi.org/10.1177/0160017603253791

Fischer, M. M., & Getis, A. (Eds.). (2010). *Handbook of Applied Spatial Analysis: Software Tools, Methods and Applications*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-03647-7

Gaussian Processes—PyMC 5.16.2 documentation. (n.d.). Retrieved August 13, 2024, from https://www.pymc.io/projects/docs/en/stable/learn/core_notebooks/Gaussian_Processes.html

Gneiting, T., Genton, M. G., & Guttorp, P. (n.d.). Geostatistical Space-Time Models, Stationarity, Separability and Full Symmetry.

Hafner, C. M. (2020). The Spread of the Covid-19 Pandemic in Time and Space. *International Journal of Environmental Research and Public Health*, 17(11), 3827. https://doi.org/10.3390/ijerph17113827

Jia, S., Kim, S. H., Nghiem, S. V., Doherty, P., & Kafatos, M. C. (2020). Patterns of population displacement during mega-fires in California detected using Facebook Disaster Maps. *Environmental Research Letters*, 15(7), 074029. https://doi.org/10.1088/1748-9326/ab8847

Lee, J. A., & Verleysen, M. (2007). *Nonlinear dimensionality reduction*. Springer.

Maas, P. (2019). Facebook Disaster Maps: Aggregate Insights for Crisis Response & Recovery. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 3173–3173. https://doi.org/10.1145/3292500.3340412

Marinescu, M. (2024). Explaining and Connecting Kriging with Gaussian Process Regression (Version 1). *arXiv*. https://doi.org/10.48550/ARXIV.2408.02331

McElreath, R. (2020). *Statistical rethinking: A Bayesian course with examples in R and Stan* (Second edition). CRC Press, Taylor & Francis Group.

Pebesma, E. J., & Bivand, R. (2023). *Spatial data science: With applications in R* (First edition). CRC Press.

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT Press.

Rauber, P. E., Falcão, A. X., & Telea, A. C. (2016). Visualizing Time-Dependent Data Using Dynamic t-SNE. In *EuroVis 2016—Short Papers* (p. 5 pages). The Eurographics Association. https://doi.org/10.2312/EUROVISSHORT.20161164

Schulz, E., Speekenbrink, M., & Krause, A. (2018). A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions. *Journal of Mathematical Psychology*, 85, 1–16. https://doi.org/10.1016/j.jmp.2018.03.001

Shumway, R. H. (2017). *Time series analysis and its applications: With r examples*. Springer Science+Business Media.

Shumway, R. H., & Stoffer, D. S. (2011). *Time Series Analysis and Its Applications*. Springer New York. https://doi.org/10.1007/978-1-4419-7865-3

Tobler, W. R. (1970). A Computer Movie Simulating Urban Growth in the Detroit Region. *Economic Geography*, 46, 234. https://doi.org/10.2307/143141

Uses of the logarithm transformation in regression and forecasting. (n.d.). Retrieved August 10, 2024, from https://people.duke.edu/~rnau/411log.htm

Wackerly, D. D., Mendenhall, W., & Scheaffer, R. L. (2008). *Mathematical statistics with applications* (7th ed). Thomson Brooks/Cole.

Waters, N. (2017). Tobler's First Law of Geography. In D. Richardson, N. Castree, M. F. Goodchild, A. Kobayashi, W. Liu, & R. A. Marston (Eds.), *International Encyclopedia of Geography* (1st ed., pp. 1–13). Wiley. https://doi.org/10.1002/9781118786352.wbieg1011

Wen, Q., Zhou, T., Zhang, C., Chen, W., Ma, Z., Yan, J., & Sun, L. (2022). Transformers in Time Series: A Survey. https://doi.org/10.48550/ARXIV.2202.07125

Wikle, C. K., Zammit-Mangion, A., & Cressie, N. A. C. (2019). *Spatio-temporal statistics with R*. CRC Press, Taylor & Francis Group.

Wong, D. W. S. (2004). The Modifiable Areal Unit Problem (MAUP). In D. G. Janelle, B. Warf, & K. Hansen (Eds.), *WorldMinds: Geographical Perspectives on 100 Problems* (pp. 571–575). Springer Netherlands. https://doi.org/10

Hafner, C. M. (2024). LDATS2470: Statistical Machine Learning and High Dimensional Data Analysis. Université catholique de Louvain.

Bogaert, P. (2024). LBRTI 2101A: Analyse statistique de données spatiales & temporelles. Université catholique de Louvain.

von Sachs, R. (2024). LSTAT2170: Time Series Analysis. Université catholique de Louvain.

Flaxman, S. (2015). Machine learning in space and time https://www.ml.cmu.edu/research/Flaxman_Thesis_2015.pdf