UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Predictive models for crime in the metropolitan area of São Paulo

Wellington Yuanhe Zhao

Dissertação de Mestrado do Programa Interinstitucional de Pós-Graduação em Estatística (PIPGEs)

ICMC USP
SÃO CARLOS

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: ______________________

Wellington Yuanhe Zhao

Predictive models for crime in the metropolitan area of São Paulo

Master dissertation submitted to the Institute of Mathematics and Computer Sciences – ICMC-USP and to the Department of Statistics – DEs-UFSCar, in partial fulfillment of the requirements for the degree of the Master Interagency Program Graduate in Statistics. *FINAL VERSION*

Concentration Area: Statistics

Advisor: Profa. Dra. Cibele Maria Russo Novelli

USP – São Carlos
October 2023

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

Z63p

Zhao, Wellington Yuanhe
Predictive models for crime in the metropolitan area of São Paulo / Wellington Yuanhe Zhao; orientadora Cibele Maria Russo Novelli. -- São Carlos, 2023.
120 p.

Dissertação (Mestrado - Programa Interinstitucional de Pós-graduação em Estatística) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2023.

1. Modelagem de crime. 2. Modelagem geoespacial. 3. Dados de segurança. I. Novelli, Cibele Maria Russo, orient. II. Título.

Bibliotecários responsáveis pela estrutura de catalogação da publicação de acordo com a AACR2:
Gláucia Maria Saia Cristianini - CRB - 8/4938
Juliana de Souza Moraes - CRB - 8/6176

Wellington Yuanhe Zhao

Modelagem de predição de crimes na região metropolitana de São Paulo

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP e ao Departamento de Estatística – DEs-UFSCar, como parte dos requisitos para obtenção do título de Mestre em Estatística – Programa Interinstitucional de Pós-Graduação em Estatística. *VERSÃO REVISADA*

Área de Concentração: Estatística

Orientadora: Profa. Dra. Cibele Maria Russo Novelli

USP – São Carlos
Outubro de 2023

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA INTERINSTITUCIONAL DE PÓS-GRADUAÇÃO EM ESTATÍSTICA UFSCar-USP

Wellington Yuanhe Zhao

Modelagem de predição de crime na região metropolitana de São Paulo

Tese apresentada ao Departamento de Estatística – Des/UFSCar e ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre ou Doutor em Estatística - Programa Interinstitucional de Pós-Graduação em Estatística UFSCar-USP.

Orientador: Prof. Dr. Cibele Maria Russo Novelli

São Carlos

Janeiro de 2017

ACKNOWLEDGEMENTS

No primeiro lugar, agradeço a minha inteira família em me apoiar de todas as formas possíveis, especialmente meus pais e a minha tia. Eu tenho orgulho de ser um membro da minha família. Saudade dos meus vôs, que depois de 2022, todos foram embora de forma gradativa.

Agradeço a compreensão e a dedicação da minha orientadora, Profa. Cibele M. Russo, pela sua excelente capacidade de me orientar nessa época difícil. Prof. Luis Gustavo Nonato foi um incentivador deste trabalho.

Muitos dos meus amigos também me acompanharam nessa jornada desafiosa. Especialmente, na fase do meu mestrado, eles são: Estevão Marana, Prof. Alexandre G. Casassola, Adriano de Jesus Holanda, Gao XuBo, Zhang Bin, Yan JiangLong, Flávia Castro Motta, Angélica Ribeiro, Lucas Horas, Bianca Souza, Eduardo(Dudu), Tiago Brizolari. Eles são de agradecimentos especiais (I have special thanks to these persons).

Temos mil motivos para ficar triste; mas para ser feliz, talvez uma razão já basta. Para mim, a existência dessas pessoas importantes, listada aqui ou não, é o motivo para eu ficar satisfeito.

Agradeço a banca da qualificação(Prof. Rafael Izbicki e Prof. Cristian Marcelo Villegas Lobos) e da defesa(Profa. Gecynalda Soares da Silva Gomes e Prof. Cristian Marcelo Villegas Lobos) pelas importantes sugestões para essa dissertação de mestrado.

No final, o presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - (Processo 88887.689379/2022-00).

RESUMO

WELLINGTON Y. ZHAO. **Modelagem de predição de crimes na região metropolitana de São Paulo**. 2023. 124 p. Dissertação (Mestrado em Estatística – Programa Interinstitucional de Pós-Graduação em Estatística) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2023.

A questão da segurança pública é um desafio para a sociedade brasileira, e a criminalidade é uma grande preocupação para o estado mais populoso do país, São Paulo. Sempre é desejável para a administração pública modelar e predizer as tendências criminais, levando em consideração as datas históricas disponíveis e a georreferenciação de cada município, ou seja, a latitude e a longitude. Nesse contexto, o uso de modelos espaço-temporais para explicar a relação entre preditores e crimes, bem como a consideração da localização, pode ser de grande importância. Um modelo possível é a modelagem SAR (Espacial Autoregressiva), que leva em conta as covariáveis e as dependências espaciais implícitas. Neste trabalho, com base no número de crimes, a modelagem SAR é utilizada para descrever e modelar as cidades da região metropolitana de São Paulo, no Brasil, incluindo a sazonalidade anual observada nos dados. Para visualizar os dados e desenvolver a modelagem com a matriz de vizinhança espacial, utilizamos pacotes em R como o spatialreg. O método de Lasso é utilizado para fazer uma pré-seleção de variáveis com maior significância, como o número de habitantes por domicílio, a taxa de reprovação e a taxa de abandono do ensino fundamental público nos anos iniciais. Em seguida, o modelo SAR é aplicado para incluir a informação espacial e enriquecer a modelagem de crimes. De forma geral, esse trabalho tem o foco de desenvolver modelagem espaço-temporal para crimes no estado de São Paulo, identificando variáveis preditoras que influenciam na quantidade de crimes em um determinado município. Além do modelo SAR, redes neurais artificiais, como modelos multicamadas e *Long Short-Term Memory* (LSTM), também são utilizadas neste trabalho, e são comparados com o modelo SAR. A proposta desta dissertação é desenvolver uma modelagem preditiva considerando dados espaço-temporais para crimes na região metropolitana de São Paulo, utilizando variáveis preditoras que influenciam na ocorrência e quantidade de crimes em um determinado município. Espera-se que os resultados obtidos sejam úteis para tomada de decisão pela administração pública, pois o trabalho cria um método para analisar os padrões de crimes em um determinado município, e também ajudar a cidade melhorar a questão de segurança pelos diversos fatores sociais.

Palavras-chave: Modelagem de crime; Modelagem geoespacial; Dados de segurança.

ABSTRACT

WELLINGTON Y. ZHAO. **Predictive models for crime in the metropolitan area of São Paulo**. 2023. 124 p. Dissertação (Mestrado em Estatística – Programa Interinstitucional de Pós-Graduação em Estatística) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2023.

The issue of public security is a challenge for Brazilian society, and crime is a major concern for the most populous state in the country, São Paulo. It is always desirable for the public administration to model and predict criminal trends, taking into account historical dates and the georeferencing of each municipality, meaning latitude and longitude. In this context, the use of spatiotemporal models to explain the relationship between predictors and crimes, as well as considering location, can be of great importance. One possible model is Spatial Autoregressive (SAR) modeling, which takes into account covariates and implicit spatial dependencies. In this work, based on the number of crimes, SAR modeling is used to describe and model the cities in the metropolitan region of São Paulo, Brazil, including the annual seasonality observed in the data. To visualize the data and develop modeling with the spatial neighborhood matrix, R packages such as spatialreg are used. The Lasso method is used to pre-select variables with greater significance, such as the number of inhabitants per household, the dropout rate, and the public elementary school dropout rate in the early years. Then, the SAR model is applied to include spatial information and enhance crime modeling. In general, this work focuses on developing spatiotemporal modeling for crimes in the state of São Paulo, identifying predictor variables that influence the quantity of crimes in a given municipality. In addition to the SAR model, artificial neural networks, such as multilayer models and Long Short-Term Memory (LSTM), are also used in the research, and compared with the SAR model. The goal of this dissertation is to develop predictive modeling considering spatiotemporal data for crimes in the metropolitan region of São Paulo, using predictor variables that influence the occurrence and quantity of crimes in a particular municipality. It is expected that the obtained results are useful for decision making by public administration, since the work creates a method to analyze crime patterns in a specific municipality, and also helps the city improve security issues through various social factors.

Keywords: Crimes modelling; Geo-Spatial modelling; Security data.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

CHAPTER

1

INTRODUCTION

Crime problems pose a significant challenge for societies, and they are particularly worrisome in Brazil due to the high incidence of criminal activities. According to the United Nations Office on Drugs and Crime (*UNODC*, 2021), Brazil has elevated rates of violent crimes, including murders and robberies, with a homicide rate of 27.4 per 100,000 inhabitants. This places Brazil among the top 20 countries in terms of intentional homicide rates. Therefore, it is desirable for public administration to model and predict the crime trend. Such a serious social problem is also the main motivation of this research work.

The predominant methodology employed in crime analysis involves a micro-spatial approach, focusing on discrete geographical units. Specifically, this entails the examination of crime patterns within small geographic segments of a city. During the investigation of these crime patterns, it is customary to gather social data that extends beyond criminal occurrences. This supplementary data is subsequently utilized to discern correlations between various social factors and instances of crime. For instance, (*SHERMAN*; *GARTIN*; *BUERGER*, 1989) employed the hot spot method, identifying that a minor fraction of addresses within a city accounted for a substantial proportion of documented criminal activity. In a similar vein, (*CAPLAN*; *KENNEDY*; *MILLER*, 2011) harnessed infrastructural data from Dallas to construct a risk terrain model. Notably,
(*WHEELER*; *STEENBEEK*, 2021) recognized limitations in the risk terrain model and opted to employ a random forest algorithm for crime prediction. In Brazil,
(*MELO*; *MATIAS*; *ANDRESEN*, 2015) conducted a study, utilizing crime data from Campinas to explore concentrations of criminal activity and resemblances among diverse crime categories across spatial dimensions. The findings revealed significant disparities in the distribution of crime within the city.

An alternative approach to understand crime patterns involves the analysis of larger spatial units. In the case of São Paulo state, one can analyze the state's crime data based on the level of municipality. São Paulo state is the most wealthy state in Brazil, each city has its unique features.

The different situation of each city leads to the investigation about the determinant geographical or condition that create the environment for different crimes.

After acquiring the crime pattern, it is possible to use the available data to make the crime model. In this study, one of the primary focal points for this research revolves around the Spatial Autoregressive Model, denoted as the SAR model, which is a spatial-temporal model suitable for analyzing datasets that contain observations on geographical areas or on any units with spatial representation.

In particular, (BROWN, 1982) utilized SAR model to discover some factors as poverty and black ethnics has the higher explanation in crime. (*BEZERRA*, 2021) applied the autoregressive to forecast online food delivery.

Another focus of this research is the utilization of Artificial Neural Networks (ANN) for crime prediction. This method has gained popularity in the academic field, with each ANN approach emphasizing different aspects of the topic. For instance, (*CHUNet al.*, 2019) utilized data from individuals with a history of criminal behavior to predict the likelihood of those individuals to commit crimes in the future, and achieved high precision results. Similarly, (*FORRADELLASet al.*, 2020) divided the Autonomous Area of Buenos Aires into different regions and predicted the number of registered crimes based on various contextual explanatory variables. (*YIet al.*, 2019) presented a similar work as this dissertation: using Neural Network based on Continuous Conditional Random Field (NN-CCRF) model for fine-grained crime prediction, specifically, they applied Long Short-Term Memory (LSTM) as the unary potential and leverages Stacked Denoising AutoEncoder (SDAE) to learn spatial correlations across regions; the analysis was conducted using crime data collected from Chicago and New York, with each crime record containing timestamp, location, and crime type information. The division were made by the city landscape into disjointed regions or grids for analysis. Finally, an interesting work is made by (*NAKIBet al.*, 2018) using convolutional neural network to present a new approach to detect crime scene object like the blood, knife and gun. This method aimed to predict the occurrence of crime scenes. These studies collectively underscore the diverse array of applications for neural networks in crime prediction, encompassing predictions at the individual level, city-wide analyses, and the detection of crime scene elements.

1.1 Objectives and Contributions of This Work

With the purpose to understand the crime pattern in the São Paulo state, and especially in the metropolitan area of São Paulo, this work has the objective to analyze data from different sources, process and to use different models to predict the crime occurrence in the geography areas.

The main contribution of this work are:

- Analyze the criminal data of large area with long history, which permits to get a whole picture of the evolution of crime of São Paulo state. Specifically, it is used the data covers many cities of the metropolitan of São Paulo from 2002 to 2017. On the other hand, previous works such as (*MELO*;*MATIAS*;*ANDRESEN*, 2015) just consider the data of Campinas.

- Perform extensive analysis on the data using the state-of-the-art techniques, specifically SAR model and neural networks models applied for crime prediction. Moreover, there's a check of SAR model results by extensive simulations by varying parameters values.

1.2 Organization of the Document

With the inspiration of above works, this dissertation focuses on several parts: In chapter 2, an exploratory data analysis is conducted. The chapter commences by presenting every variable of the data derived from different sources. Then, after organising the data, the data analysis concentrates on two parts: the correlogram between different type of crimes, and the choropleth's plot of crime per capita. There are two choropleth's plot: one is for the São Paulo state, another is for the metropolitan area of São Paulo.

Chapter 3 the different types of models which are utilized in this research work: This includes the utilization of Linear Regression as a preliminary step for introducing the Spatial Autoregressive (SAR) model. Furthermore, Lasso model for variable selection is also discussed. Central to this chapter are the SAR model, Artificial Neural Networks (ANN) and Long Short Term Memory(LSTM), each serving as critical analytical tools for this investigation.

In Chapter 4, there is the application of the Spatial Autoregressive (SAR) model to the data of the greater São Paulo's area. Initially, it is used the Lasso method to select nine significant variables for the SAR model. The results reveal limitations due to the bipolar nature of the data and a low ρ value. To address these limitations, a modification is created to apply a logarithmic transformation to the data to mitigate bipolar characteristics and replace "total crimes" with "crime per capita" to improve the model's representation of relationships. Additionally, there's an approach using Artificial Neural Networks (ANN) and Long Short-Term Memory (LSTM) models, maintaining the spatial-temporal focus. Comparative analysis showed that the ANN and LSTM models outperform the SAR model in terms of predictive accuracy, using metrics like RMSE and MAE, which is the metrics that tell how accurate the predictions are and, what is the amount of deviation from the actual values. It is also generated a choropleth plot to illustrate predictions from these models. Overall, this chapter presents the application of SAR, ANN, and LSTM models, highlighting the enhanced predictive capabilities of the latter two models.

Chapter 5 brings the discussion of this work. The propose and the significance of this work will be presented in this chapter, followed by the discussion of the process and the methodology during this dissertation. Still in this chapter, It is also stated a future work, and a possible good direction is the use of P-spline's method.

At final, there's also an appendix which the code in R of this work is presented.

CHAPTER 2

EXPLORATORY DATA ANALYSIS

The exploratory data analysis outlines the origins of the dataset and the variables employed in the modeling process. While the dataset originates from São Paulo state, the analysis will specifically emphasize the São Paulo metropolitan area. The analysis will employ two distinct types of visualizations: correlogram plots and choropleth maps. These visualizations will cover both the São Paulo state and its metropolitan area, providing a comprehensive overview of the spatial and statistical relationships within the data.

2.1 Data and Variables

For the purpose of the modeling, data was collected from various sources. The crime data were obtained from the *S*ecretaria de Segurança pública((*SSP*), 2021). Given the crime occurences of all types, the total number of crimes by month were obtained and denoted by the new variable 'total of crime' as the response variable.

The demographic, educational and economic data were obtained from the SEADE (Sistema Estadual de Análise de Dados) and the geolocation information was obtained from IBGE (Instituto Brasileiro de Geografia e Estatística). It is worth noting that the transformed industrial value (VTI) represents the sum of the product and industrial service sales; that is, the difference between the gross value of industrial production(sum of sales of industrial products and services) and the cost of industrial operations(costs directly related to industrial production).

Table 1 – Description of variables related to crimes, in Portuguese and in English. The types of crimes include: murder, rape, assault, robbery, etc.

Variable’s Name	Full Name of Variable
homicídio doloso	intentional murder
número de vítimas em homicídio doloso	intentional murder’s victim number
homicídio doloso por acidente de trânsito	intentional murder by traffic accident
número de vítimas em homicídio doloso por acidente de trânsito	intentional murder’s victim number by traffic accident
homicídio culposo por acidente de trânsito	manslaughter by traffic accident
homicídio culposo outros	other reason for manslaughter
tentativa de homicídio	attempt for murder
lesão corporal seguida de morte	body injury followed by death
lesão corporal dolosa	intentional body injury
lesão corporal culposa por acidente de trânsito	intentional body injury by traffic accident
lesão corporal culposa outras	other reason for intentional body injury
latrocínio	assault with intention of murder
número de vítimas em latrocínio	assault with intention of murder’s victim number
total de estupro	total of rape
total de roubo outros	total of assault for other reason
roubo de veículo	assault of car
roubo a banco	assault of bank
roubo de carga	assault of truck
furto de veículo	robbery of car

Source: SEADE (Sistema Estadual de Análise de Dados)

Table 2 – Description of variables related to demography, in Portuguese and in English.

Variable’s Name	Full Name of Variable
Área	area
Densidade Demográfica	demographic density
Grau	degree of urbanization
Habdom	inhabitant per household
Total H	total of man
Total M	total of woman
idade de zero a catorze	age from zero to fourteen
idade de quinze a vinte e nove	age from fifteen to twenty nine
idade de trinta a cinquenta e nove	age from thirty to fifty nine
idade de sessenta e mais	age above sixty

Source: SEADE (Sistema Estadual de Análise de Dados)

2.2 Data Analysis

The data utilized in this study extends beyond the domain of public security and encompasses various facets such as economy, education, and geographic information from each city, with a total of 102 available variables. Data was divided into training and test sets for predicting-focused analysis. The training data consisted of crimes per capita between 2002 and 2017 with 123,000 observations, while the test data the crimes in 2018 with 7,728 observations.

It is possible to sum up every type of crimes in each city in the year of 2017 and then divide by

Table 3 – Description of variables related to education, in Portuguese and in English. The education data mainly focus on the enrollment number of different degrees of school.

Term	Term meaning
tx aprovacao	approval rate
tx reprovacao	disapproval rate
tx abandono	abandon rate
fai publica	public elementary school early years
fai privado	private elementary school early years
faf publica	public elementary school final years
faf privado	private elementary school final years
medio publica	public high school
medio privada	private high school
matrícula creche	daycare enrollment
matriculas pre escola	pre-school enrollment
matriculas ESAI	elementary school early years enrollment
matriculas ESAF	elementary school final years enrollment
matriculas EM	high school enrollment
municipal	municipal
estadual	state
particular	private

Source: SEADE (Sistema Estadual de Análise de Dados)

the city's population to generate the crime per capita. Due to the low value of crime per capita in the data, the crime per capita has been multiplied by 1 million. The choropleth's plot in Figure 1 shows that the metropolitan and coastal areas have higher values of crimes per capita, indicating that crimes are more likely to occur in developed and touristic cities. The digits on the abcissa represent the latitude, and the digits on the ordenate represent the longitude.

For the choropleth's plot, the information of the latitude and longitude polygons is extracted from (*BRUGNARA*, 2021), which is a Python package with the data in the format of geo-json. So, based the polygon data, the geopandas package in Python and the ggplot in R could draw each city's boundaries and create the map of São Paulo state.

On the other hand, based on the crime data, one can suppose that each different type of crime could be considered as distinct variables and then create the correlogram between these variables from 2002 to 2021. A correlogram is a plot that shows the correlation between the type of crimes in the São Paulo state. The correlogram plot in Figure 2 shows a strong correlation between different type of rape and murder, as well as a relative strong correlation between murder and assault, rape. The correlogram is made by Kendall rank correlation coefficient method, which evaluates the degree of similarity between two sets of ranks given to a same set of objects.

To validate the findings from the correlogram, two scatter plots were constructed. Figure 3 depicts the scatter plot between the variables other robbery and motor vehicle theft, with both variables transformed using a logarithmic function. Conversely, Figure 4 presents the scatter plot between

Table 4 – Description of variables related to industry, in Portuguese and in English. The data mainly describe the total value of a specific sector in a city.

Variable's Name	Full Name of Variable
Valor PIB	GDP value
Valor servico ex	external service value
Valor Agropecuaria	value of agriculture and livestock
Valor Industria	value of industry
Valor Impostos líquido	value of liquid tax
Valor Serviços adm	value of administrative service
Valor Serviços	value of service
Valor Adi	additional value
Valor PIB per capita	value of GDP per capita
VTI alimento	transformed value of food industry
VTI bebidas	transformed value of beverage industry
VTI texteis	transformed value of textile industry
VTI vestuario	transformed value of wearing industry
VTI couro	transformed value of leather industry
VTI madeira	transformed value of wood industry
VTI celulose	transformed value of cellulose industry
VTI impressao	transformed value of print industry
VTI petroleo	transformed value of petroleum industry
VTI biocombustíveis	transformed value of biofuel industry
VTI quimica	transformed value of chemical industry
VTI farmaceutica	transformed value of pharmaceutical industry
VTI borracha	transformed value of rubber industry
VTI min n met	transformed value of mineral and metal industry
VTI metal	transformed value of metal industry
VTI metalurgia	transformed value of metallurgy industry
VTI eletroinformatico	transformed value of electroinformatic industry
VTI maquina	transformed value of machinery industry
VTI outrosequi	transformed value of others equipment industry
VTI moveis	transformed value of furniture industry
VTI prod div	transformed value of diverse products industry
VTI manutencao	transformed value of maintenance industry

Source: SEADE (Sistema Estadual de Análise de Dados)

the logarithm of bank robbery and the logarithm of the number of victims in intentional homicide by traffic accident. The results from both scatter plots align with the correlogram's findings. Specifically, Figure 3 demonstrates a more linear relationship, indicating a strong correlation, whereas Figure 4 shows the opposite result, suggesting a weaker correlation.

Also, there's the choice of the metropolitan region of São Paulo at 2017 as the test data. Therefore, it is necessary to do the choropleth plot (Figure 5) and correlogram plot (Figure 6) for this region.

Preliminary results of this work have been presented in the 24th conference SINAPE (Simpósio Nacional de Prababilidade e Estatística, in Portuguese), at Gramado, Rio Grande do Sul.

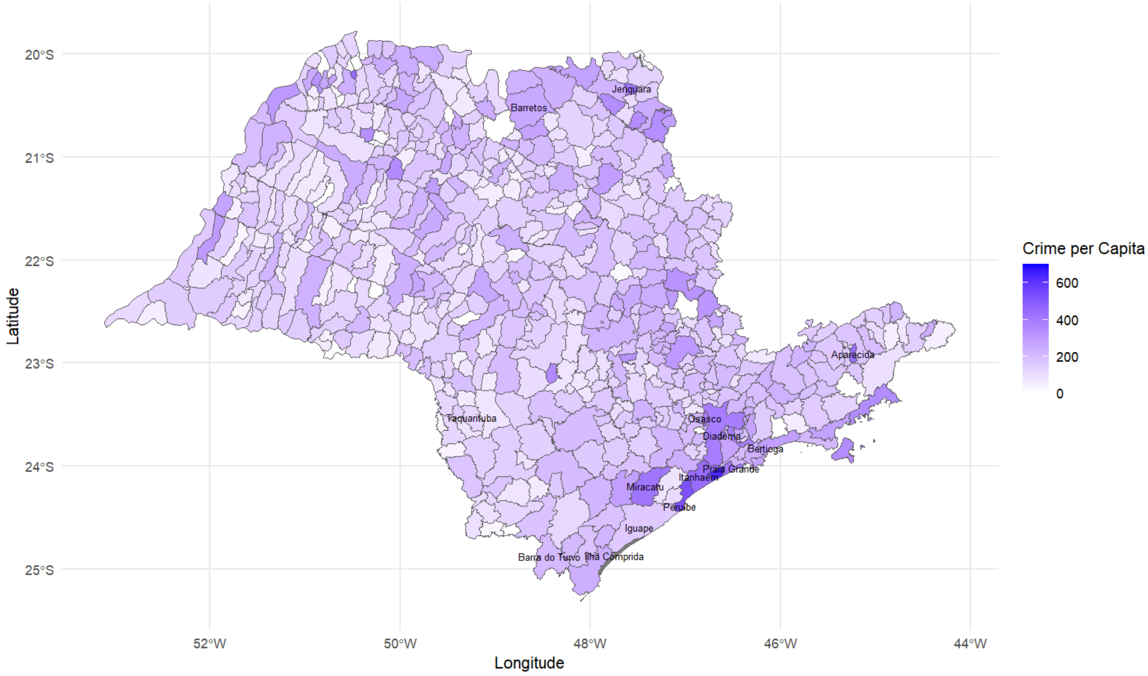

Figure 1 – Choropleth's plot of the São Paulo state crime per capita in the year of 2017.
Source: figure elaborated by the author.

2.3 Conclusion

The correlogram's plot between São Paulo state and the metropolitan area of São Paulo are similar: the correlation between different types of rape and murder are strong, as well as a relative strong correlation between murder and assault, rape. This is the same in the choropleth's plot: the city of São Paulo concentrates higher crime per capita.

The next chapter will discuss the different statistical models, and in Chapter 4, on the application part, those models will be applied to the data that is introduced in this chapter.

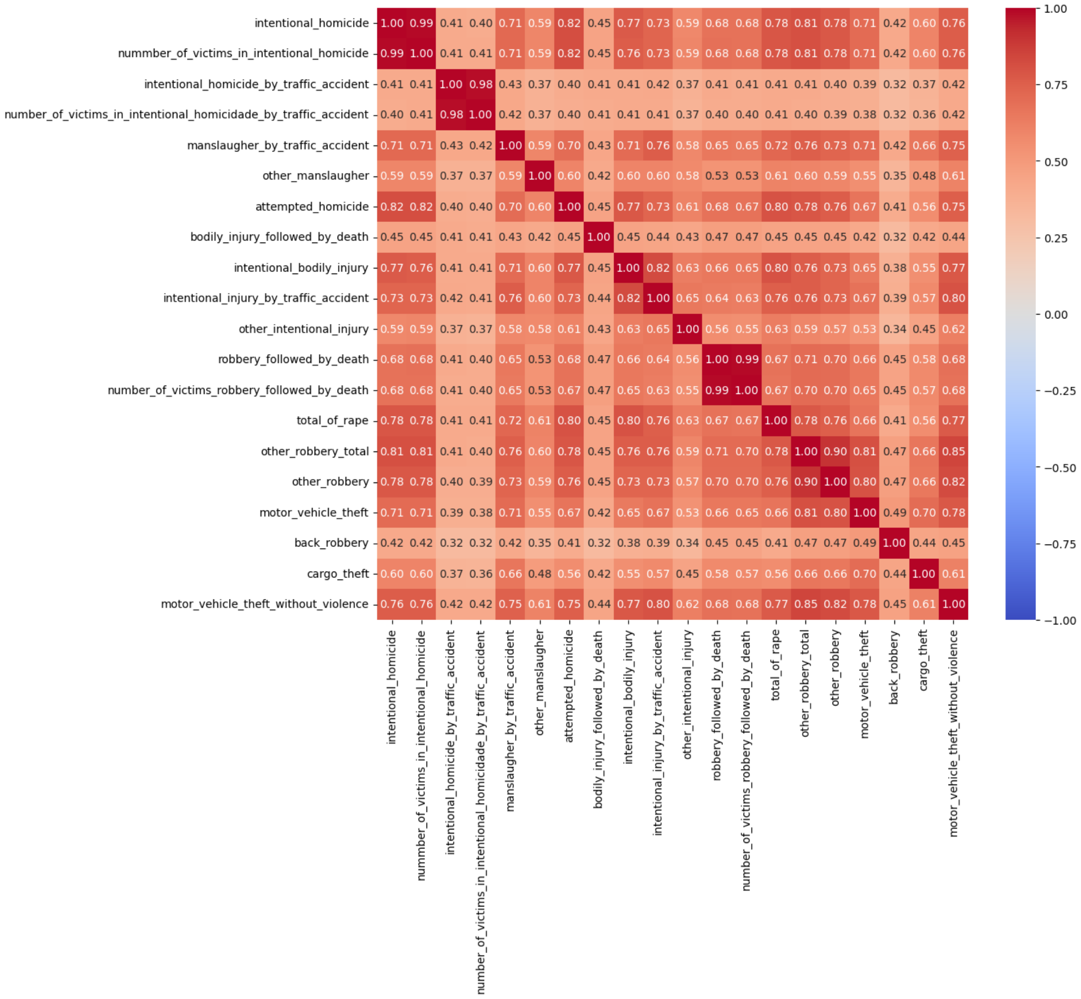

Figure 2 – Correlogram’s plot of different type of crimes in the São Paulo state.
Source: Figure elaborated by the author

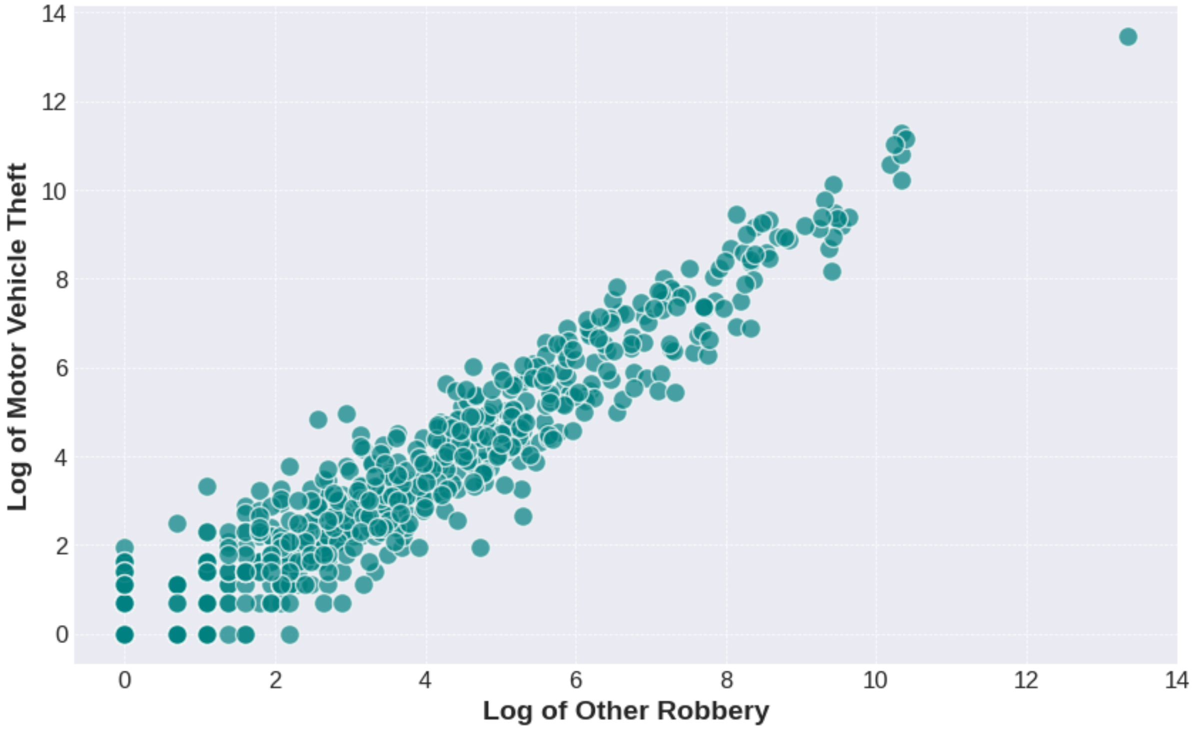

Figure 3 – Scatterplot between log of other robbery and log of motor vehicle theft
Source: figure elaborated by the author

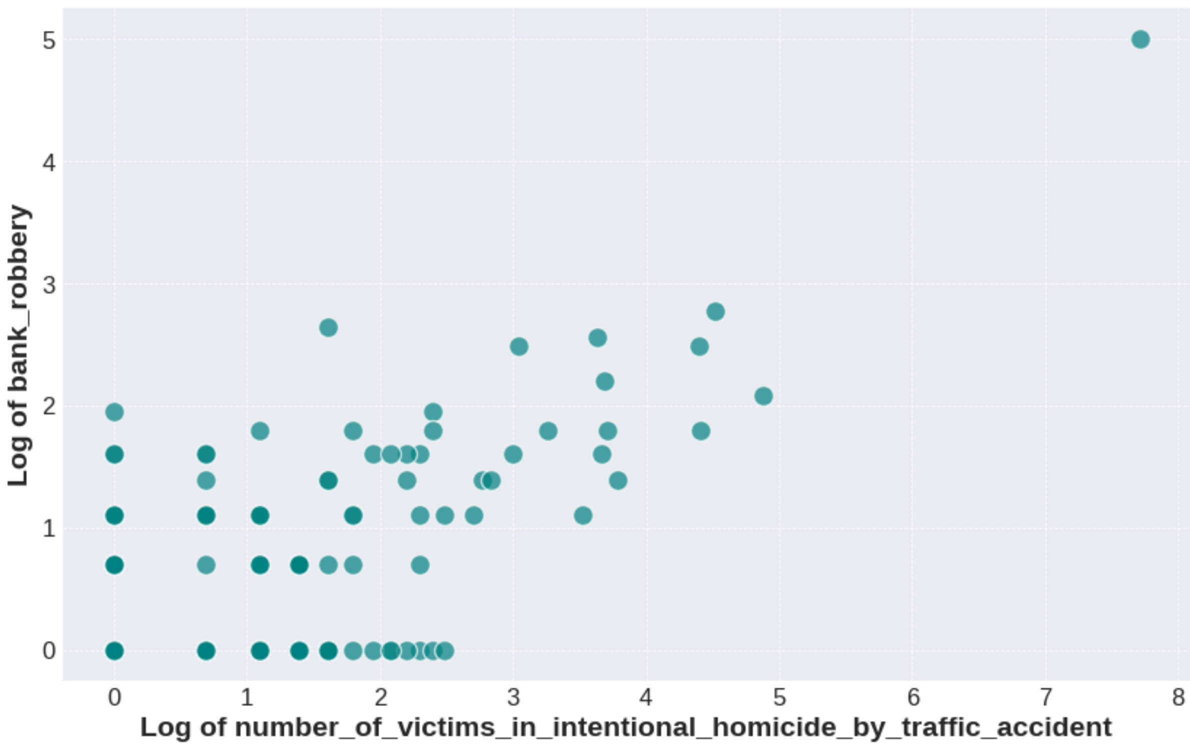

Figure 4 – Scatterplot between log of bank robbery and log of number of victims in intentional homicide by traffic accident
Source: figure elaborated by the author

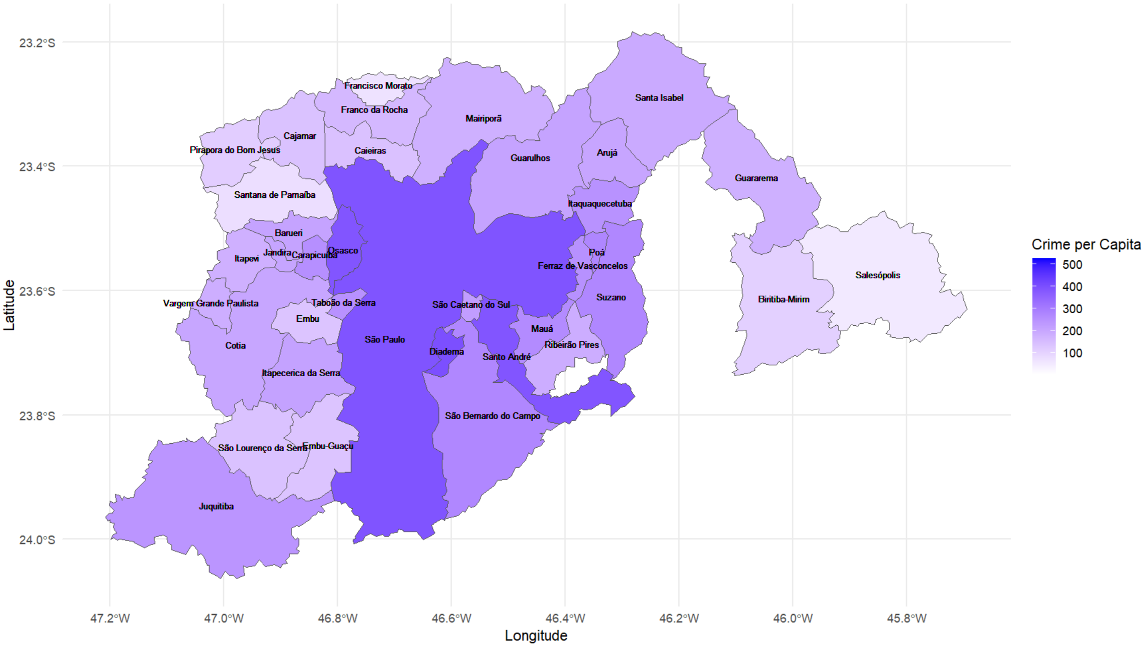

Figure 5 – Choropleth plot of the metropolitan area of São Paulo's crime per capita in the year of 2017.
Source: Figure elaborated by the author

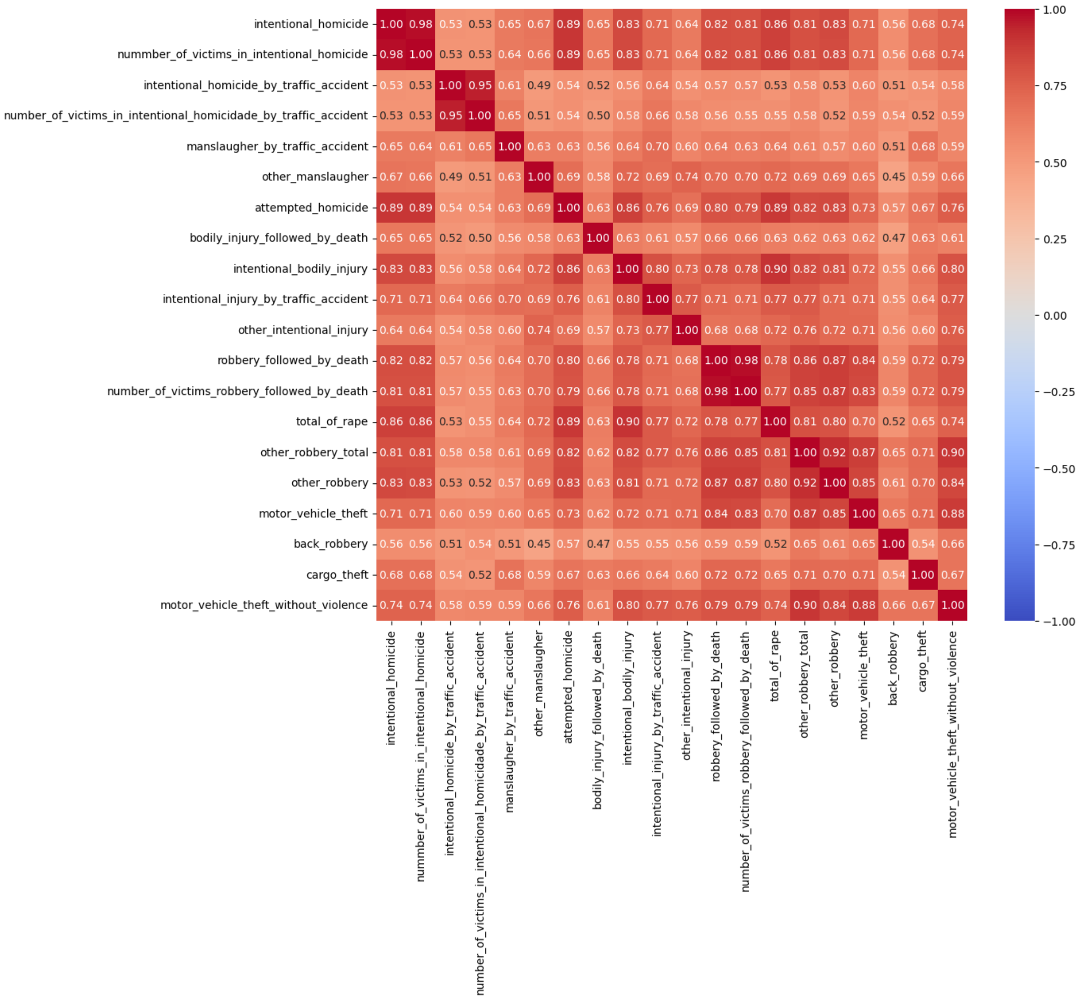

Figure 6 – Correlogram's plot of different type of crimes in the metropolitan area of São Paulo.
Source: Figure elaborated by the author

CHAPTER

3

STATISTICAL MODELLING

This chapter will introduce various model for different purpose: Firstly, is the presentation of the Lasso model as a tool for variable selection, then, is the review of the linear regression theory and SAR model. Finally, a presentation on Artificial Neural Networks (ANN), Recurrent Neural Networks (RNN) and Long Short Term Memory (LSTM) is given. In this work, both the SAR model and Artificial Neural Networks are used for crime prediction and the results are compared.

3.1 Linear Regression Model

Linear regression is widely used in prediction in the areas of Economics and Finance, Medicine and Engineering and so on. It is a statistical technique for modeling the relationship between a dependent variable (often called the target or response) and one or more independent variables (also known as predictors or features). The multiple linear regression model is stated in the following form

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_k x_{ki} + \varepsilon_i \tag{3.1}$$

where y_i is a dependent variable (YAN; SU, 2009), β_0, β_1, β_2,..., β_k are the regression coefficients, and ε_i is the random errors assuming $\mathbb{E}[\varepsilon_i] = 0$ and Var(ε_i) = σ^2 for $i = 1,2,\ldots,n$, and terms are assumed to be uncorrelated and usually supposed to follow a normal distribution, with a constant variance σ^2. The model can also be expressed in the matrix format

$$y = X\beta + \varepsilon \tag{3.2}$$

where

$$y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} X = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1k} \\ 1 & x_{21} & x_{22} & \dots & x_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{nk} \end{bmatrix}, \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix}, \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix} \tag{3.3}$$

Even with no assumptions made for the distribution of the random errros, the least squares estimation of β can be solved as:

$$\hat{b} = \arg\min_{\beta}[(y - X\beta)^T(y - X\beta)] \tag{3.4}$$

where $\hat{b} = (b_0, b_1, ..., b_k)^T$ is a k-dimensional vector of the estimators of the regression coefficients. To obtain the least squares estimation of β, it is necessary to minimize the residual of sum squares by solving the following equation:

$$\frac{\partial}{\partial b}[(y - Xb)^T(y - Xb)] = 0 \tag{3.5}$$

or equivalently

$$\frac{\partial}{\partial b}[(y^T y - 2y^T Xb + b^T X^T Xb)] = 0 \tag{3.6}$$

By taking partial derivative with respect to each component of b we obtain the following normal equation of the multiple linear regression model:

$$X^T Xb = X^T y \tag{3.7}$$

Since $X^T X$ is non-singular it follows that $b = (X^T X)^{-1} X^T y$.

3.2 Lasso

In statistics, shrinkage is the reduction in the effects of sampling variation, and one of the shrinkage method is Lasso. Lasso, short for "Least Absolute Shrinkage and Selection Operator," is a regularization technique used in linear regression. It is an extension of linear regression that aims to improve the model's performance by introducing a form of feature selection. According to (HASTIE *et al.*, 2009) is defined by

$$\hat{\beta}^{Lasso} = \arg\min_{\beta} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{k} x_{ij}\beta_j)^2, \tag{3.8}$$

$$\text{subject to} \sum_{j=1}^{k} |\beta_j| \leqslant t \tag{3.9}$$

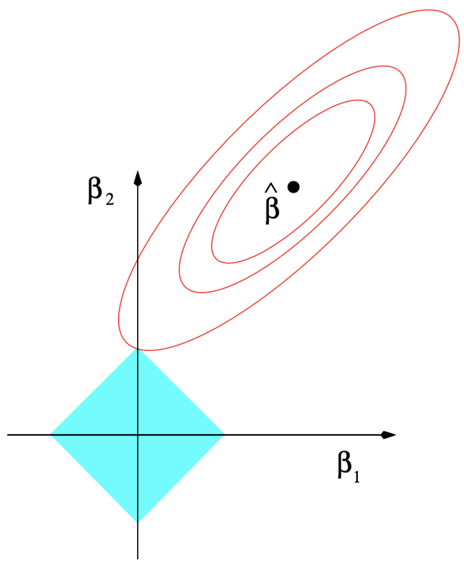

Figure 7 – Estimation picture for the Lasso. The figure shows the contours of the error and constraint functions. The solid blue areas are the constraint regions $|\beta_1|+|\beta_2| \leqslant t$, while the red ellipses are the contours of the least squares error function.
Source: Figure elaborated by (*HASTIE et al.*, 2009)

It is possible to write the Lasso problem in the equivalent Lagrangian form

$$\hat{\beta}^{lasso} = \arg\min_{\beta}\{\frac{1}{2}\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{k} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{k}|\beta_j|\} \tag{3.10}$$

where the parameter $\lambda \geqslant 0$ is a complexity parameter that controls the amount of shrinkage: the larger the value of λ, the greater the amount of shrinkage (which means to limiting the value of the βs), as showed in Figure 7.

3.3 SAR model

The SAR (Spatial Autoregressive) model (*KAZAR; CELIK*, 2012), also known as spatial lag model or mixed regressive model, is an extension of the linear regression model and is represented by Equation (3.11)

$$y_i = \rho \sum_{j=1}^{n} W_{ij}y_j + \sum_{q=0}^{k} x_{iq}\beta_q + \varepsilon_i \tag{3.11}$$

where ρ is the spatial autoregression (autocorrelation) parameter, W is n-by-n neighborhood matrix that accounts for the spatial relationships (dependencies) among the spatial data. ε_i is the iid error term, y_i is the dependent variable, x_{iq} is the independent variable and β_q is the coefficient of x_{iq}. Particularly, when $q = 0$, it is defined that $x_{i0} = 1$ and the is the β_0 is the intercept of the model. The main difference between linear regression and SAR model is the inclusion of the spatial autocorrelation term, represented as $\rho \sum_{j=1}^{n} W_{ij} y_j$. This term is introduced to account for the spatial dependencies among the elements of the dependent variable y_i. It is important to note that when ρ is zero, the SAR model reduces to a standard linear regression model.

Incorporating spatial dependencies into models has been demonstrated to enhance overall classification or prediction accuracy in comparison to using linear regression ($KAZAR; CELIK$, 2012). Spatial dependency is defined as the connections between adjacent pixels within a spatial framework, typically represented by a regular grid (in this work, a grid could be considered as the municipality). When considering a four-neighborhood, the neighboring pixels of the (i, j)th pixel on the regular grid are depicted in equation (3.12).

$$neighbors(i,j) = \begin{cases} (i-1,j), 2 \geqslant i \geqslant \phi, 1 \geqslant j \geqslant q, NORTH, \\ (i,j+1), 1 \geqslant i \geqslant \phi, 1 \geqslant j \geqslant q-1, EAST, \\ (i+1,j), 2 \geqslant i \geqslant \phi-1, 1 \geqslant j \geqslant q, SOUTH, \\ (i,j-1), 1 \geqslant i \geqslant \phi, 2 \geqslant j \geqslant q, WEST, \end{cases} \tag{3.12}$$

where ϕ is the row dimension of the spatial framework (image) and q is the column dimension of spatial framework (image). This (i, j)th pixel of the surface will be the $(\phi \times (i-1)+j)$th row of the matrix C, which is the non-row-standardized (non-normalized) neighborhood matrix of dimension $\phi \times q$ by $\phi \times q$ or n by n.

The authors in ($KAZAR; CELIK$, 2012) state the following entries of matrix C

- $\{(\phi \times (i-1)+j), (\phi \times (i-2)+j)\}$,
- $\{(\phi \times (i-1)+j), (\phi \times (i-1)+j+1)\}$,
- $\{(\phi \times (i-1)+j), (\phi \times (i)+j)\}$
- $\{(\phi \times (i-1)+j), (\phi \times (i-1)+j-1)\}$

will have "1" as its value, while the other elements in the row will be zero. This is illustrated in Figure 8. On the other side, the row-standardized neighborhood matrix W is calculated in two steps: first, calculate each row sum, in which ($KAZAR; CELIK$, 2012) stated that W is $\phi \times q$ by $\phi \times q$ or n-by-n. Second, each element in a row is divided by its corresponding row sum. In other words, $W = D^{-1}C$ and the elements of the diagonal matrix D are defined as $d_{ii} = \sum_{i=1}^{n} c_{ij}$ and $d_{ij} = 0$ ($KAZAR; CELIK$, 2012), in which d_{ii} and d_{ij} is the element inside of the matrix D, and c_{ij} is the element inside of the matrix C; specially, d_{ii} is the diagonal element of the matrix D.

For the spatial weight matrix of São Paulo's state, the latitude and longitude polygon information was extracted from ($BRUGNARA$, 2021), containing the data in geo-json format. For a given city x in the data, if city y shares the same geographic border with x, then the element of xth line, yth column and the element of the yth line, xth column in the spatial weight matrix will

$$\begin{bmatrix}
0&1&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
1&0&1&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&1&0&1&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&1&0&0&0&0&0&0&0&0\\
1&0&0&0&0&1&0&0&1&0&0&0&0&0&0&0\\
0&1&0&0&1&0&1&0&0&1&0&0&0&0&0&0\\
0&0&1&0&0&1&0&1&0&0&1&0&0&0&0&0\\
0&0&0&1&0&0&1&0&0&0&0&1&0&0&0&0\\
0&0&0&0&1&0&0&0&0&1&0&0&1&0&0&0\\
0&0&0&0&0&1&0&0&1&0&1&0&0&1&0&0\\
0&0&0&0&0&0&1&0&0&1&0&1&0&0&1&0\\
0&0&0&0&0&0&0&1&0&0&1&0&0&0&0&1\\
0&0&0&0&0&0&0&0&1&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&1&0&1&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&1\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0
\end{bmatrix}$$

(a) The non-normalized neighborhood matrix **C**

$$\begin{bmatrix}
0&\frac{1}{2}&0&0&\frac{1}{2}&0&0&0&0&0&0&0&0&0&0&0\\
\frac{1}{3}&0&\frac{1}{3}&0&0&\frac{1}{3}&0&0&0&0&0&0&0&0&0&0\\
0&\frac{1}{3}&0&\frac{1}{3}&0&0&\frac{1}{3}&0&0&0&0&0&0&0&0&0\\
0&0&\frac{1}{2}&0&0&0&0&\frac{1}{2}&0&0&0&0&0&0&0&0\\
\frac{1}{3}&0&0&0&0&\frac{1}{3}&0&0&\frac{1}{3}&0&0&0&0&0&0&0\\
0&\frac{1}{4}&0&0&\frac{1}{4}&0&\frac{1}{4}&0&0&\frac{1}{4}&0&0&0&0&0&0\\
0&0&\frac{1}{4}&0&0&\frac{1}{4}&0&\frac{1}{4}&0&0&\frac{1}{4}&0&0&0&0&0\\
0&0&0&\frac{1}{3}&0&0&\frac{1}{3}&0&0&0&0&\frac{1}{3}&0&0&0&0\\
0&0&0&0&\frac{1}{3}&0&0&0&0&\frac{1}{3}&0&0&\frac{1}{3}&0&0&0\\
0&0&0&0&0&\frac{1}{4}&0&0&\frac{1}{4}&0&\frac{1}{4}&0&0&\frac{1}{4}&0&0\\
0&0&0&0&0&0&\frac{1}{4}&0&0&\frac{1}{4}&0&\frac{1}{4}&0&0&\frac{1}{4}&0\\
0&0&0&0&0&0&0&\frac{1}{3}&0&0&\frac{1}{3}&0&0&0&0&\frac{1}{3}\\
0&0&0&0&0&0&0&0&\frac{1}{2}&0&0&0&0&\frac{1}{2}&0&0\\
0&0&0&0&0&0&0&0&0&\frac{1}{3}&0&0&\frac{1}{3}&0&\frac{1}{3}&0\\
0&0&0&0&0&0&0&0&0&0&\frac{1}{3}&0&0&\frac{1}{3}&0&\frac{1}{3}\\
0&0&0&0&0&0&0&0&0&0&0&\frac{1}{2}&0&0&\frac{1}{2}&0
\end{bmatrix}$$

(b) The normalized version i.e., **W**

Figure 8 – The normalized and non-normalized matrix.
Source: Figure elaborated by (KAZAR; CELIK, 2012)

be 1; Otherwise, it will be zero. The spatial weight matrix was created following the approach described in (*BIVAND*; *MILLO*; *PIRAS*, 2021), which utilized the (*BIVAND*, 2022*b*) package.

3.3.1 Derivation of the Likelihood (Concentrated Log-likelihood) Function

According to (KAZAR; CELIK, 2012), it is not adequate to use ordinary least squares to solve the models described by equation (3.11). So another choice is to use the ML theory procedure. The end-result will be the concentrated log-likelihood function that is used in the optimization of SAR model parameter estimate ρ. Equation (3.11) can be explicitly written as follows

$$y_i = (I-\rho W)^{-1}(\beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \ldots + x_{ik}\beta_k + \varepsilon_i) \tag{3.13}$$

where i = 1,...,n is the index for n succesive observations. The error ε_i is a normally, independently and identically distributed with mean $E(\varepsilon_i) = 0$ and variance $V(\varepsilon_i) = \sigma^2$.

Assuming that the errors ε_i, which are the elements of the vector $\varepsilon = [\varepsilon_1, \ldots, \varepsilon_i, \ldots, \varepsilon_n]^T$, also are independently and identically distributed according to a normal distribution defined in equation (3.14). So the matrix $(I-\rho W)$ will be called matrix A. It is worth to note that $\varepsilon_i = (Ay_i - x_i\beta)$.

$$N(\varepsilon_i; 0, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(\frac{-1}{2\sigma^2}(Ay_i - x_i\beta)^2) \tag{3.14}$$

if the vector ε is a multi-variate normal distribution, the normal distribution is then defined in equation (3.15) with a covariance matrix defined $\Sigma = \sigma^2 I$. Also, it is worth to note that $|\Sigma|^{-\frac{1}{2}} = \sigma^{-n}$, $\Sigma^{-1} = \sigma^{-2}I$ and $|\Sigma| = |\sigma^2 I| = \sigma^{2n}$.

$$\begin{aligned} N(\varepsilon_i;0,\Sigma^2) &= (2\pi)^{\frac{-n}{2}}|\Sigma|^{-1}exp(\frac{-1}{2}\varepsilon_i^T\Sigma^{-1}\varepsilon_i) \\ &= (2\pi)^{\frac{-n}{2}}|\Sigma|^{-1}exp(-\frac{1}{2}(Ay_i - x_i\beta)^T\Sigma^{-1}(Ay_i - x_i\beta)) \end{aligned} \tag{3.15}$$

Considering the data matrix x of size n-by-k composed of vectors x_i, and the corresponding observations y_i (where $i = 1,\ldots,n$), it is important to note that these observations follow density function of the form $N(Ay_i - x_i\beta, \sigma^2)$. These density functions exhibit a similar structure to that of the disturbances. The likelihood function for estimating the parameters β and σ^2 based on the sample is defined in Equation (3.16).

$$\begin{aligned} L(\theta|y) &= L((\beta,\sigma^2)|(y_i,x_i,W)) \\ &= \prod_{i=1}^{n} N((Ay - x\beta),\sigma^2) \\ &= (2\pi\sigma^2)^{\frac{-n}{2}}exp(\frac{-1}{2}(Ay - x\beta)^T\Sigma^{-1}(Ay - x\beta))|d\varepsilon/dy| \end{aligned} \tag{3.16}$$

As $|d\varepsilon/dy| = |A|$, which is demonstrated by (*FREUND*;*WALPOLE*, 1980) on the theorem 7.1 on pages 232-233. So the formula could be expressed as

$$L(\theta|y) = (2\pi\sigma^2)^{\frac{-n}{2}}exp(\frac{-1}{2\sigma^2}(Ay - x\beta)^T(Ay - x\beta))|A| \tag{3.17}$$

Considering $\hat{\beta}$ and $\hat{\sigma^2}$

$$\hat{\beta} = (x^Tx)^{-1}x^TAy \tag{3.18}$$

$$\hat{\sigma^2} = \frac{(Ay)^T(I - x(x^Tx)^{-1}x^T)^T(I - x(x^Tx)^{-1}x^T)(Ay)}{n} \tag{3.19}$$

Replacing $\hat{\beta}$ with β in equation (3.17) and $\hat{\sigma^2}$ with σ^2 in equation (3.17) lead to equation below for the log-likelihood function (i.e. the logarithm of the ML function) to be optimized for ρ.

$$\begin{aligned} lnL(y) = ln|I - \rho W| - \frac{n}{2}ln(2\pi) - \frac{2}{n}ln\{\frac{1}{n}\} \\ - \frac{n}{2}ln\{y^T(I - \rho W)^T[I - x(x^Tx)^{-1}x^T]^T[I - x(x^Tx)^{-1}x^T](I - \rho W)y\} - \frac{1}{2n} \end{aligned}$$

In the context of solving a SAR(Spatial Autoregressive) model, one effective derivative-based optimization algorithm that can be employed is the Newton-Raphson (root-finding) method. In this algorithm, the goal is to determine the optimal value for the parameter ρ. To achieve this, it is necessary to calculate the first derivative of the log-likelihood function. The first derivative of the log-likelihood function can be obtained by expanding the Sum of Squared Errors (SSE) term. This derivative provides valuable information about the location of the optimal solution for the ρ parameter. The equation for the first derivative of the log-likelihood function is as follows

$$\frac{\partial lnL(y)}{\partial \rho} = tr((I-\rho W)^{-1}\frac{\partial (I-\rho W)}{\partial \rho})) - \frac{n}{2}(\frac{-y^T M^T MWy - y^T W^T M^T My + 2y^T W^T M^T MWy}{y^T(I-\rho W)^T M^T M(I-\rho W)y}) \tag{3.20}$$

The term M corresponds to $[I - x(x^T x)^{-1}x^T]$, where x is the design matrix that contains the predictive variables.

Inside the spatialreg package in R, the ρ is generated by the optimize function, which is used in a combination of golden section search and successive parabolic interpolation, and was designed for use with continuous functions.

For comparison with the SAR modelling in a predictive approach, the next section the use of the artificial neural network modelling is proposed. There is a significant difference between Artificial neural networks (ANN) and the SAR model since it is not a linear regression model, depending on the circumstances, could be advantageous or not. On the one hand, since ANN is a non-linear model, it can capture a more complex relationship that SAR model cannot reach. However, linear regression has facility to interpret data, which is the shortcoming of the ANN.

3.4 Artificial Neural Network

The design of a neural network is motivated by analogy with the brain, which is living proof that fault-tolerant parallel processing. The brain is a highly complex, nonlinear, and parallel computer (information-processing system). It has the capability to organize its structural constituents, known as neurons, so as to perform certain computations (e.g., pattern recognition, perception, and motor control) many times faster than the fastest digital computer in existence today(*HAYKIN*;*HAYKIN*, 1999).

A neural network is a method of artificial intelligence that teaches computers to process data in a way inspired by the human brain. It is a type of machine learning process, that uses interconnected nodes or neurons in a layered structure, similar to the human brain. The neural network creates an adaptive system that computers use to learn from mistakes and continually improve. There are a vast applications of neural networks, such as medical diagnosis through image classification, financial predictions, energy demand and electrical load forecasting, process and quality control, computer vision and pattern recognition.

A typical neural network structure consists of the following components: the structue of each individual neuron (processing unit), the structure of the network (how neurons are connected), and the training algorithm.

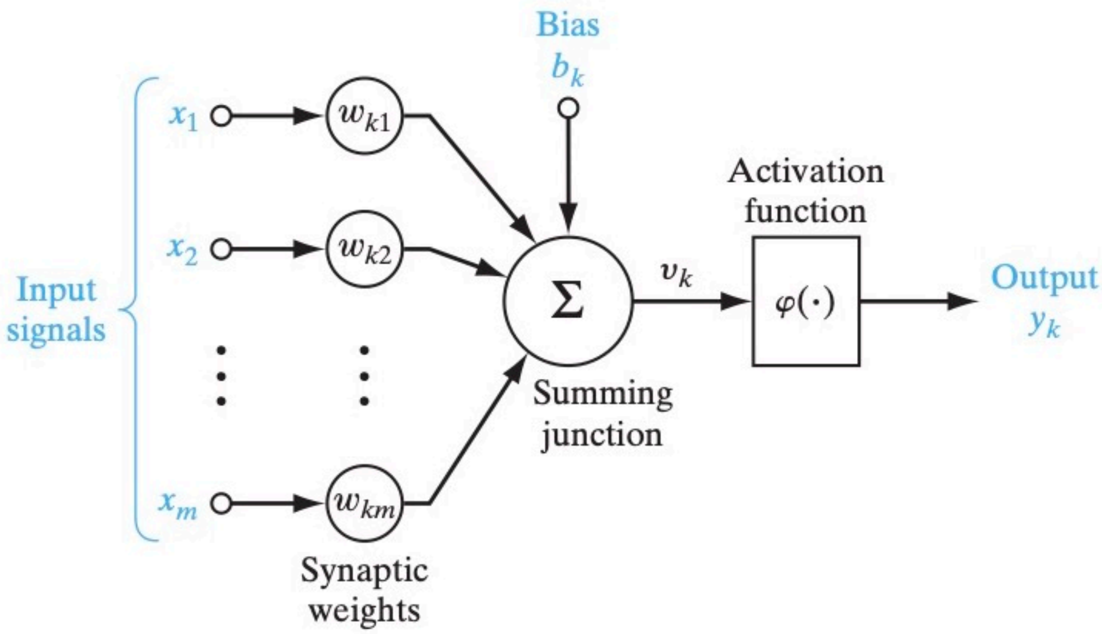

Figure 9 – Nonlinear model of a neuron
Source: Figure elaborated by (*HAYKIN*; *HAYKIN*, 1999)

3.4.1 *Artificial neuron structure*

Here we present the basic of the neuron model:

- A group of connections between neurons is called synapses, and each synapse has its own weight. When a signal x_j is fed into a synapse connected to neuron k, it is multiplied by the synaptic weight w_{kj}.
- All the weighted input signals to neuron k is summed. This process is known as a linear combination.
- An activation function is also used to restrict the amplitude of the output of a neuron. It serves to prevent the output from becoming too large or too small.

The neural model of Figure 9 also includes an externally applied bias, denoted by b_k. The bias b_k has the effect of increasing or lowering the net input of the activation function, depending on whether it is positive or negative, respectively.

In mathematical terms, it is possible to describe the neuron k depicted in Figure 9 by writing the pair of equations:

$$v_k = \sum_{j=1}^{m} w_{kj} x_j \tag{3.21}$$

and

$$y_k = \phi(v_k + b_k) \tag{3.22}$$

where x_1, x_2, ..., x_m are the input signals; w_{k1}, w_{k2},..., w_{km} are the respective synaptic weights of neuron k; v_k (shown in Figure 9) is the linear combiner output due to the input signals; b_k is the bias; $\phi(\cdot)$ is the activation function; and y_k is the output signal of the neuron. The use of bias b_k has the effect of applying an affine transformation to the output v_k of the linear combiner in the model of Figure 9

The activation function, denoted by $\phi(v)$, defines the output of a neuron in terms of the induced local field v. In such manner, there are the following activation functions:

- Threshold function.

  For this type of activation function, described in Figure 10a, there is

$$\phi(v) = \begin{cases} 1, \text{if } v \geq 0 \\ 0, \text{if } v < 0 \end{cases} \tag{3.23}$$

  Correspondingly, the output of neuron k employing such a threshold function is expressed as

$$y_k = \begin{cases} 1, \text{if } v_k \geq 0 \\ 0, \text{if } v_k < 0 \end{cases} \tag{3.24}$$

  where v_k is the induced local field of the neuron; that is,

$$v_k = \sum_{j=1}^{m} w_{kj} x_j + b_k \tag{3.25}$$

- Sigmoid function

  The sigmoid function is a widely used activation function in neural networks, characterized by its "S"-shaped curve. As in Eq. (3.26), it strikes a balance between linearity and nonlinearity and is strictly increasing.

$$\phi(v) = \frac{1}{1 + exp(-av)} \tag{3.26}$$

  The logistic function is a common example of a sigmoid function, defined by a slope parameter $a \in \mathbb{R}$ and a local field v. By adjusting the value of "a", it can generate sigmoid functions with varying slopes, as shown in Figure 10b.

  Activation functions, such as those defined in Equations (3.23) and (3.26), produce output within the range of 0 to +1. Nevertheless, there are scenarios that necessitate activation functions with an output range of -1 to +1. In these instances, the activation function must exhibit an odd behavior in response to the local field. Consequently, the threshold function described in Equation (3.23) would require redefinition to accommodate this requirement.

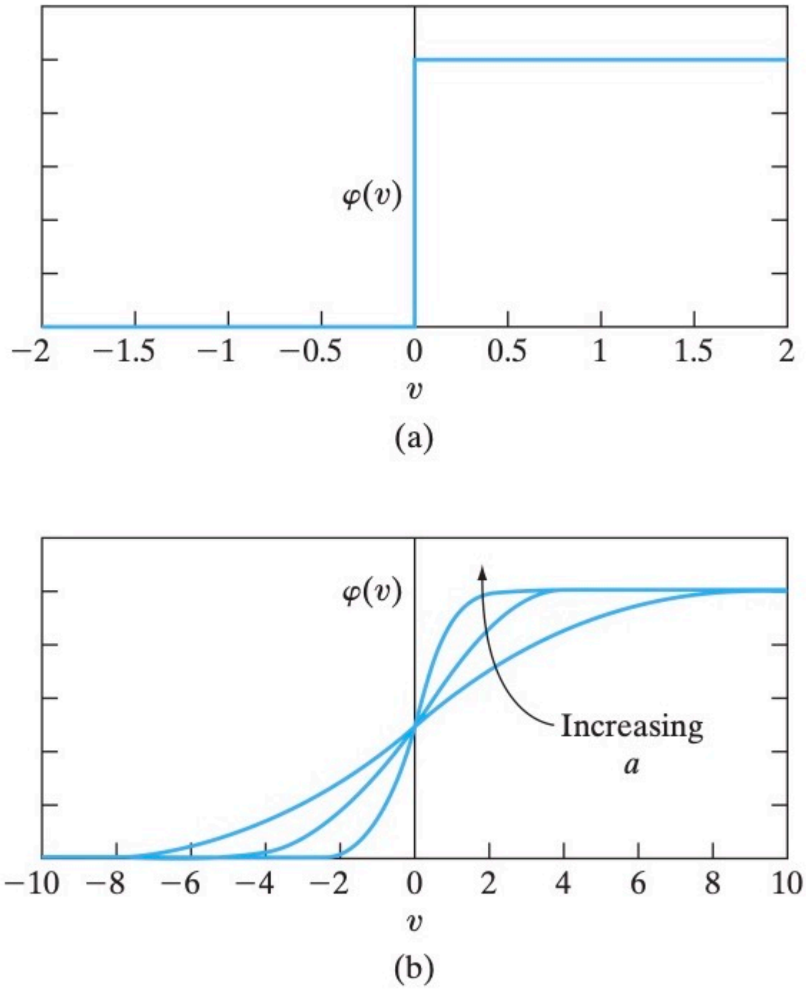

Figure 10 – (a) Threshold function.(b) Sigmoid function for varying slope parameter a.
Source: Figure elaborated by (*HAYKIN*; *HAYKIN*, 1999)

$$\phi_k = \begin{cases} 1, \text{if } v_k > 0 \\ 0, \text{if } v_k = 0 \\ -1, \text{if } v_k < 0 \end{cases} \tag{3.27}$$

which is commonly referred to as the signum function. For the corresponding form of a sigmoid function, it is possible to use the hyperbolic tangent function, defined by

$$\phi = tanh(v) \tag{3.28}$$

- ReLu function

The ReLU function is defined by ($GOODFELLOW; BENGIO; COURVILLE$, 2016) as follow:

$$\phi(v) = max(0, v) \quad (3.29)$$

Rectified Linear Units (ReLU) are relatively easy to optimize because of their similarity to linear units. The only difference between a linear unit and a ReLU is that the latter yields zero for all negative inputs in its domain. As a result, the derivatives through a ReLU maintain a significant magnitude whenever the unit is activated, thereby enhancing its optimization efficiency.

3.4.2 Network structure

Each individual neuron doesn't represent much things. A network of connected neurons can represent many kinds of data patterns. Basically, there are two types of network structures($HAYKIN; HAYKIN$, 1999):

- Feed-Forward Networks: A network with one or more layers of neurons (processors), where data flows in a single direction, meaning there is no feedback loop.
- Recurrent Networks: Networks with connections between processors in the same layer and/or with processors from previous layers (feedback).

3.4.3 Learning in Neural Networks - Training Algorithm

Here is briefly present the training algorithm of Multi-Layer Perceptron (MLP), which is called Back-Propagation. The Back-Propagation algorithm has two phases for each presented data sample($HAYKIN; HAYKIN$, 1999)($BISHOP$, 1995)(BRAGA; CARVALHO; LUDERMIR, 2000):

Feedforward: Inputs propagate through the network from the input layer to the output layer.

Feedback: Errors propagate in the opposite direction to the data flow, going from the output layer to the first hidden layer.

Suppose that there have a set of training data samples $x = x_1, x_2, \ldots, x_N$ and each sample x_i has a label t_i. When the data samples are fed to the neural network, then there is the following error function (cost function):

$$E = \frac{1}{2} \sum_{j=1}^{m} (t_j - y_j)^2 \quad (3.30)$$

where t_j is the desired output value of pattern p for processor j in the output layer and the y_j is the activation state of processor j in the output layer when a data sample is presented, m is the total number of neurons in the output layer

The training algorithm has objective is to minimize the error function by changing the synaptic weights between pairs of neurons. This can be done by gradient descent:

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}} \tag{3.31}$$

where η is the learning rate, E is the cost function to be minimized, and the w_{ij} is the synaptic weight from neural i to j. After some algebraic manipulations, we have

$$\Delta w_{ij} = \eta e_j x_i \tag{3.32}$$

where x_i is the input value received by connection i and e_j is the calculated value of the error for processor j.

If neural j is in the output layer, the error term is defined by the following equation

$$e_j = (t_j - x_j)\phi^{'}(y_j) \tag{3.33}$$

if neural j is an hidden layer, the error term is defined by the following equation

$$e_j = \phi^{'}(y_j)\sum(e_k w_{jk}) \tag{3.34}$$

where neural k is located in the next layer of neural j

Therefore, the Back-Propagation training algorithm can be summarized as follows:

Initialization:

Weights initialized with random and small values ($|w_{ij}| < 0.1$).

Training:

1) Loop until the error of each output processor is <= *tolerance* for all data samples in the training set.

2) Apply an input data sample x_i and its respective desired output vector t_i.

3) Calculate the outputs of processors, starting from the first hidden layer to the output layer.

4) Calculate the error for each neuron in the output layer. If the error <= *tolerance* for all processors, go back to 1).

5) Update the weights for each neuron using Equations (3.32), (3.33), (3.34), starting from the output layer to the input layer.

6) Return to step 1).

3.4.4 ANN applying in the security problem

Artificial Neural Networks can be used to predict the occurrence of crime in specific geographic areas. By analyzing historical crime data, it is possible to identify patterns that may indicate high-risk areas related to security problem. This project will use artificial neural networks, or LSTM (as it will show later), to spatially analyze the crime per capita in the metropolitan area of São Paulo, and then give the prediction to compare with true value of the crime per capita.

As the discussion earlier, since SAR model is a linear regression, which means that it only captures the linear relationship between social factor and crimes, it is useful to have ANN to complement the other part of this project, which can capture more general association between response and covariates. Those social factors are considered as input signals, and the ANN will have synaptic weights and activation functions in the neurons. Depending on the model, there are different number of layers and neurons, where each layer has its own activation function. Finally, the output is the number of the crimes per capita occurred in each city of the metropolitan area of São Paulo.

3.5 Recurrent Neural Networks

A recurrent neural network is a neural network that is specialized for processing a sequence of values $x^{(1)},...,x^{(\tau)}$, where τ is the quantity of time steps, and t is a time point between 1 and τ. ($GOODFELLOW; BENGIO; COURVILLE, 2016$).

For example, consider the classical form of a dynamical system:

$$s^{(t)} = f(s^{(t-1)}; \theta), \tag{3.35}$$

where $s^{(t)}$ is called the state of the system, function f maps the state t from t to $t+1$, and θ is the parameter of the function.

Equation (3.35) is recurrent because the state at time t can return back to the same state at time $t-1$.

For a finite number of time steps τ, this recurrence process can be unfolded by applying the dynamical system definition $\tau - 1$ times. For example, if we unfold equation (3.35) for $\tau = 4$ time steps, we obtain

$$s^{(4)} = f(s^{(3)}; \theta) \tag{3.36}$$

$$= f(f(s^{(2)}; \theta); \theta) \tag{3.37}$$

$$= f(f(f(s^{(1)}; \theta); \theta); \theta) \tag{3.38}$$

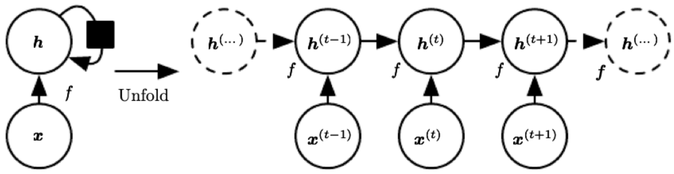

Figure 11 – A recurrent network. Left side is the circuit diagram, and the right side is the unfolded one.
Source: Figure elaborated by (GOODFELLOW; BENGIO; COURVILLE, 2016)

On the other hand, considering the case of a dynamical system with an input signal $x^{(t)}$, to define the values of their hidden units and to indicate the state in the hidden units of the network, we now write the equation of the state variable:

$$h^{(t)} = f(h^{(t-1)}, x^{(t)}; \theta), \tag{3.39}$$

in which $h^{(t)}$ is the state, the same as the $s^{(t)}$.

We can represent the unfolded recurrence after t steps with a function $g(t)$:

$$\begin{aligned} h^{(t)} &= g^{(t)}(x^{(t)}, x^{(t-1)}, x^{(t-2)}, \ldots, x^{(2)}, x^{(1)}) \\ &= f(h^{(t-1)}, x^{(t)}; \theta) \end{aligned} \tag{3.40}$$

The function $g(t)$ takes the whole past sequence $(x^{(t)}, x^{(t-1)}, x^{(t-2)}, \ldots, x^{(2)}, x^{(1)})$ as input and produces the current state, but the unfolded recurrent structure allows us to factorize $g(t)$ into repeated application of a function f.

3.5.1 LSTM

LSTM (Long Short Term Memory) is a recurrent neural network where cells are connected recurrently to each other, replacing the usual hidden units of ordinary recurrent networks. (GOODFELLOW; BENGIO; COURVILLE, 2016). Instead of using a simple unit that just processes input data and past information in a traditional way, LSTM (Long Short-Term Memory) networks use special "LSTM cells". These cells have a built-in memory loop, like a self-reminder. The most important component is the state unit $s_i^{(t)}$, which has a linear self-loop similar to the leaky units, and an example of the leaky unit can be represented as such

$$\mu^{(t)} \longleftarrow \alpha\mu^{(t-1)} + (1-\alpha)\nu^{(t)} \tag{3.41}$$

α is a parameter, $\mu^{(t)}$ is the running average and $\nu^{(t)}$ is some value per time step. When α is near one, the past is remembered; on the contrary, when α is near zero, the past is forgotten.

Inspired by the leaky unit, in the case of the LSTM model, the self-loop weight (or the associated time constant) is controlled by a forget gate unit $f_i^{(t)}$(for time step t and cell i), which sets this weight to a value between 0 and 1 via a sigmoid unit:

$$f_i^{(t)} = \sigma\left(b_i^f + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)}\right) \tag{3.42}$$

In particular, this sigmoid unit $\sigma()$ is the logistic sigmoid function, which is the same as the Equation (3.26). $x^{(t)}$ is the current input vector and $h^{(t)}$ is the current hidden layer vector, containing the outputs of all the LSTM cells, and b^f, U^f, W^f are respectively biases, input weights, and recurrent weights for the forget gates. In comparison with the ANN neuron's model, b^f in LSTM and b_k in ANN are both bias, U^f in LSTM and w_{ki} in ANN are both input weights. The main difference between the LSTM model and ANN is the existence of the recurrent weights, W^f, for the recurrent state in one time step before, $h^{(t-1)}$.

The LSTM cell internal state is thus updated as follows, but with a conditional self-loop weight $f_i^{(t)}$:

$$s_i^{(t)} = f_i^{(t)} s_i^{(t-1)} + g_i^{(t)} \sigma\left(b_i + \sum_j U_{i,j} x_j^{(t)} + \sum_j W_{i,j} h_j^{(t-1)}\right) \tag{3.43}$$

Comparing the Equation (3.41) with the Equation (3.43), the forget gate unit $f_i^{(t)}$ has the utility similar to the parameter α in the running average $\mu^{(t-1)}$. So as mentioned in the name, the forget gate unit controls the past information inside of the model. When $f_i^{(t)}$ is near to 1, the past information $s_i^{(t-1)}$ will be remembered; and if $f_i^{(t)}$ is near 0, the past information $s_i^{(t-1)}$ will be forgot.

The external input gate unit $g_i^{(t)}$ is computed similarly to the forget gate (with a sigmoid unit to obtain a gating value between 0 and 1, but with its own parameters

$$g_i^{(t)} = \sigma\left(b_i^g + \sum_j U_{i,j}^g x_j^{(t)} + \sum_j W_{i,j}^g h_j^{(t-1)}\right) \tag{3.44}$$

The output $h_i^{(t)}$ of the LSTM cell can also be shut off, via the output gate $q_i^{(t)}$, which also uses a sigmoid unit for gating.

$$h_i^{(t)} = \tanh(s_i^{(t)}) q_i^{(t)} \tag{3.45}$$

$$q_i^{(t)} = \sigma\left(b_i^o + \sum_j U_{i,j}^o x_j^{(t)} + \sum_j W_{i,j}^o h_j^{(t-1)}\right) \tag{3.46}$$

It is necessary to consider that the output gate unit $q_i^{(t)}$, external input gate unit $g_i^{(t)}$ and forget gate unit $f_i^{(t)}$ are actually neuron cells, but they are connected together recurrently with the internal state $s_i^{(t)}$ to form the LSTM "cell", which is shown in the Figure 12.

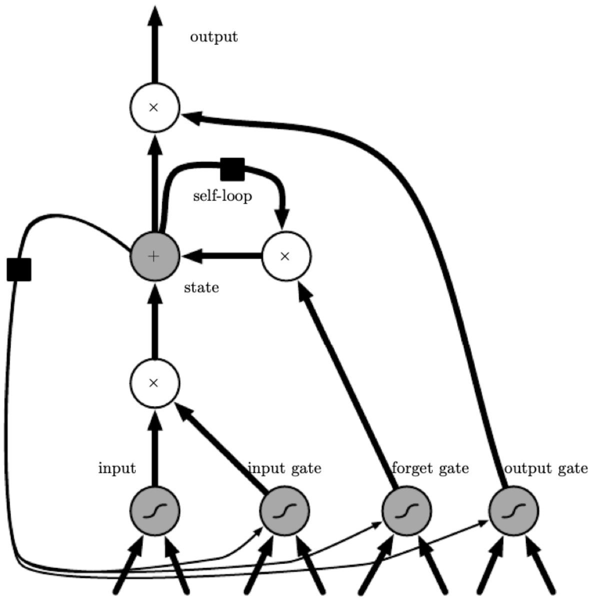

Figure 12 – Block diagram of the LSTM recurrent network "cell". Cells are connected recurrently to each other.
Source: Figure elaborated by (*GOODFELLOW*; *BENGIO*; *COURVILLE*, 2016)

The training algorithm is similar to the one in Multi-Layer Perceptron: 1) Choose an appropriate loss function that matches the nature of the task. For instance, mean squared error (MSE) for regression or categorical cross-entropy for classification. 2) Select an optimization algorithm, such as stochastic gradient descent. The algorithm updates the network's weights during training to minimize the loss function.

3.6 Conclusion

This chapter's main purpose was to discuss the theoretical aspect of different models. In this sense, based on the different approaches of this project, Lasso is used as a model to select variables, SAR is a spatial-temporal model; and on the other hand, although ANN, RNN and LSTM are general models, they will be used as spatial-temporal models for comparison to SAR model.

In addition, the independent variables, such as the education, economy, demography, industry variables, which will be shown in the next chapter, are the inputs of ANN and LSTM, the dependent variable of ANN and LSTM is the crime per capita of each city.

Based on Equation (3.21) and Equation (3.22) and in the context of this project involving a multi-layer ANN model, the initial layer of the model takes input signals denoted as x_j in the Equation (3.21). These input signals represent various social factors, including education, demographics, economy, and industry-related variables. The final layer of the ANN model produces an output, represented as y_k in the Equation (3.22), which serves as the model's prediction for the crime per capita in each city after training..

On the other hand, based on Equations (3.43) and (3.46) and in the context of this project involving a multi-layer LSTM model, the initial layer takes input signal denoted as $x^{(0)}$. The output in LSTM model is represented as $h^{(t)}$, in which t is the number of time steps.

In the next chapter, it will discuss the application of the data presented in Chapter 2. With the exception of the linear regression, all of the models presented in this chpater will be applied in this work.

CHAPTER

4

MODELING AND APPLICATION

In this chapter, the data will be applied into the models to generate crime predictions. The first part concentrates on the SAR model, fitting the SAR model, and making simulations. Due to the great variability in population in cities such as São Paulo and Guarulhos compared to other municipalities of the metropolitan region of São Paulo, a logarithmic function is applied to the explanatory variables, and the variable "total of crimes" is substituted with "crime per capita". After that, a new fitting of the SAR model is made; moreover, a large amount of simulations are performed to verify the parameter precision in the SAR regression modeling. Furthermore, Artificial Neural Networks and LSTM are applied to obtain crime predictions. Finally, the prediction results generated by the SAR model and neural network models are compared.

4.1 SAR Model at São Paulo's State and at the Region of Greater São Paulo

Based on the collected data, cities with populations exceeding 50,000 peoples are selected for analysis. Subsequently, the Lasso method is employed to select the following variables: total crime with 1 month of lag, total crime with 2 months of lag, total crime with 11 months of lag, total of population, population with age of more than 60 years old, quantity of student registered at the municipal pre-school, registration at the state elementary school (beginning), registration at the municipal elementary school (final), registration of municipal high school.

Then, according to (*BIVAND*; *HAUKE*; *KOSSOWSKI*, 2013), which created the (*BIVAND*, 2022*a*) in R, the data will be applied by this package to fit the spatial autogressive(SAR) model. The resulting outcomes are presented in the table below.

The parameter ρ holds a value of 0.01237. Apart from the lag variables, a notable correlation becomes evident between total crime and population. This connection can be attributed to the phenomenon that larger cities tend to exhibit higher crime rates.

Table 5 – Fitted SAR model for crimes at São Paulo state

Variables and Intercept	Estimate	Std. Error	z value	p-value
Intercept	−13.5	5.5	−2.40	0.01
total crime 1m of lag	0.32	0.02	11.03	< 0.01
total crime 2m of lag	0.3	0.02	12.30	< 0.01
total crime 11m of lag	0.2	0.02	8.80	< 0.01
Population	0.00019	0.00005	3.20	< 0.01
population of age 60+	0.0002	0.0004	0.53	0.58
registration at the municipal pre school	0.00047	0.0012	0.30	0.70
registration at the state elementary school(beginning)	0.0003	0.0005	0.61	0.53
registration at the municipal elementary school(final)	0.0008	0.0007	1.10	0.27
registration of municipal high school	−0.0009	0.002	−0.30	0.73

Source: Elaborated by the author.

Again, the SAR model is applied exclusively at the region of greater São Paulo. The region contains the following cities: Caieiras, Cajamar, Francisco Morato, Franco da Rocha, Mairiporã, Arujá, Biritiba-Mirim, Ferraz de Vasconcelos, Guararema, Itaquaquecetuba, Guarulhos, Mogi das Cruzes, Poá, Salesópolis, Santa Isabel, Suzano, Diadema, Mauá, Santo André, São Bernardo do Campo, São Caetano do Sul, Ribeirão Pires, Rio Grande da Serra, Cotia, Embu, Embu-Guaçu, Itapecerica da Serra, Juquitiba, São Lourenço da Serra, Taboão da Serra, Vargem Grande Paulista, Barueri, Carapicuíba, Itapevi, Jandira, Osasco, Pirapora do Bom Jesus, Santana de Parnaíba. Some cities do not participate to the regression due to the lack of the data.

The outcome is presented in Table 6.

Table 6 – Fitted SAR model for crimes at the region of greater São Paulo

Variables and Intercept	Estimate	Std. Error	z value	p-value
ρ	0.0022			0.6156
(Intercept)	−47.8745	29.2237	−1.6382	0.1014
total crime 1m of lag	0.0211	0.0441	0.4784	0.6324
total crime 2m of lag	0.0268	0.0432	0.6212	0.5344
total crime 11m of lag	0.5864	0.0559	10.4952	<0.001
Population	0.0003	0.0003	1.0246	0.3055
population of age 60+	0.0032	0.0023	1.4104	0.1584
registration at the municipal pre school	0.0107	0.0106	1.0126	0.3112
registration at the state elementary school(beginning)	0.0045	0.0040	1.1377	0.2552
registration at the municipal elementary school(final)	0.0183	0.0067	2.7204	0.0065
registration of municipal high school	−0.0501	0.0258	−1.9417	0.0522

Source: Elaborated by the author.

Taking into consideration the pair plot exhibited in Figure 13, it becomes evident that certain variables possess a bipolar concentration characteristic of data (for instance, São Paulo and Guarulhos as major cities showcasing higher values across all variables). This characteristic has the potential to negatively impact model's performance. To address this point, a logarithmic transformation is applied to each variable. This adjustment is depicted in Figure 14, which illustrates a more favorable data concentration.

Moreover, due to an overwhelming quantity of missing values (NA), the variable of total crime with a 11 months lag is removed. Subsequently, the SAR model is once again employed on the refined dataset, yielding the outcomes presented in Table 7.

Table 7 – Fitted SAR model for crimes at the region of greater São Paulo after applied logarithmic function

Variables and Intercept	Estimate	Std. Error	z value	p-value
ρ	0.07823			<0.001
(Intercept)	−5.5047	0.1006	−54.7019	<0.001
total crime 1m of lag	0.0055	0.0052	1.0572	0.2904
total crime 2m of lag	0.0110	0.0038	2.9261	0.0034
Population	0.0166	0.0176	0.9446	0.3449
population of age 60+	1.0218	0.0207	49.3454	<0.001
registration at the municipal pre school	0.0184	0.0153	1.2076	<0.0001
registration at the state elementary school(beginning)	0.0022	0.0019	1.1757	<0.0001
registration at the municipal elementary school(final)	0.0046	0.0016	2.8258	0.0047
registration of municipal high school	−0.0109	0.0026	−4.1751	<0.0001

Source: Elaborated by the author.

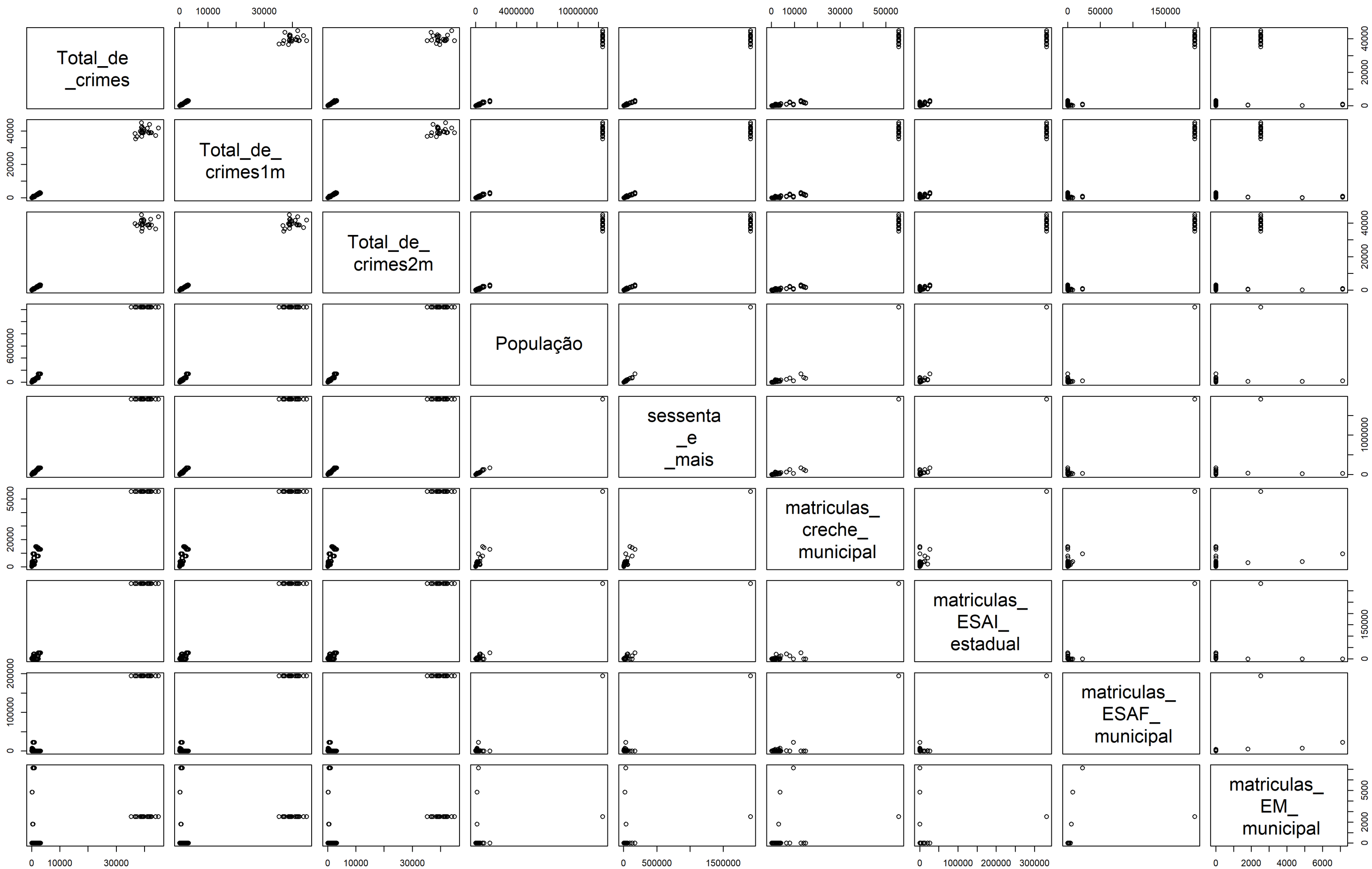

Figure 13 – Pair-plot between the predictor variables before applying the logarithmic function. Source: figure elaborated by the author

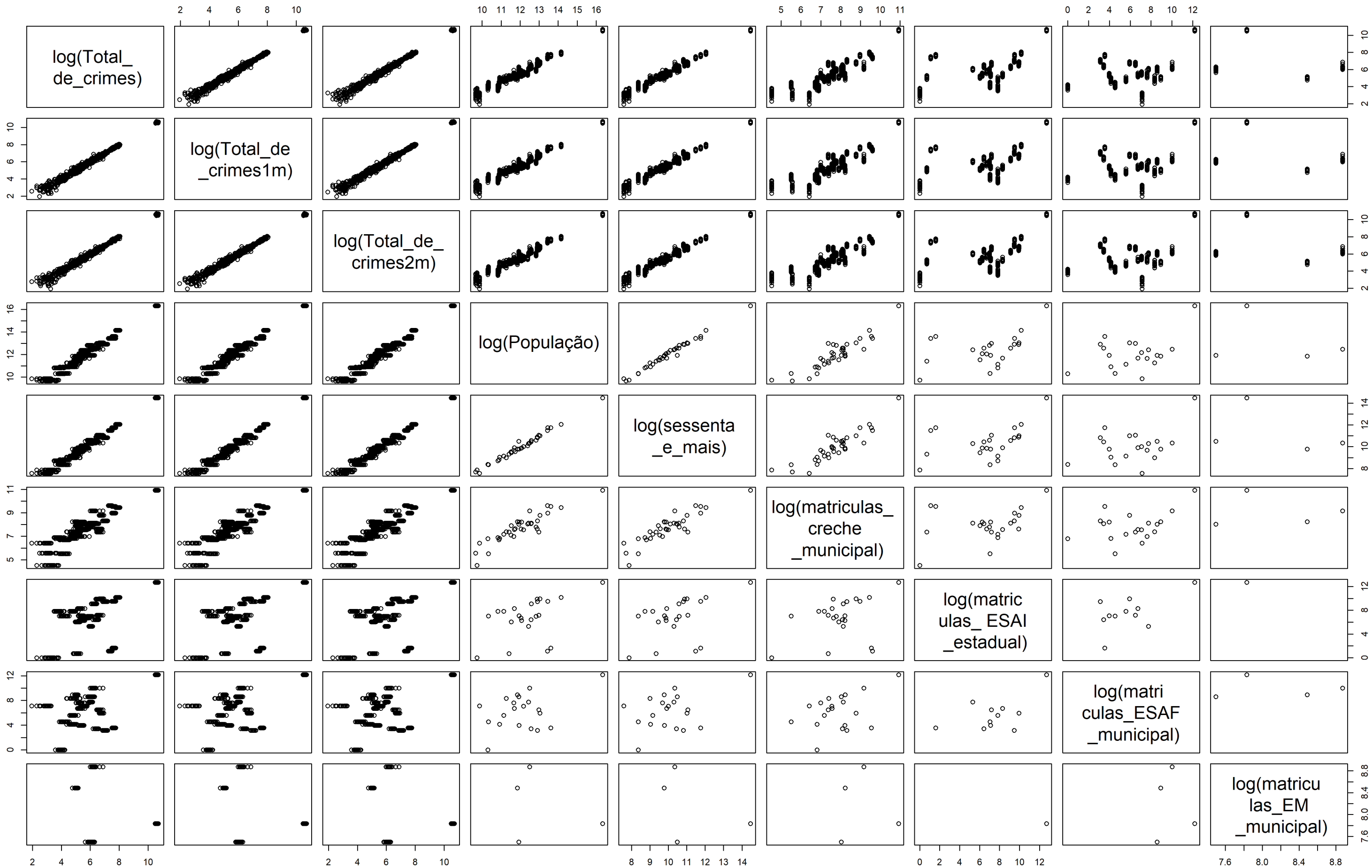

Figure 14 – Pair-plot between the predictor variables after applying the logarithmic function. Source: figure elaborated by the author

The results indicate that the SAR (Spatial Autoregression) models exhibit a low value of the spatial autoregression parameter. In the context of spatial-temporal modeling, a low ρ suggests that the model's behavior closely approximates that of a linear regression, thereby undermining the utility of the SAR model for capturing spatial dependencies. This observation necessitates a thorough examination of the SAR model's functionality, leading to the need for simulation-based investigations to ensure its proper operation.

4.2 Simulation at the Region of Greater São Paulo

The whole dataset includes every explanatory variable from 2002-2017, so the same process was made as before.

From Table 7, we see that the ρ value is very low. In this case, the SAR model is close to the pure linear regression model. Therefore, it is necessary do the following simulations to verify the β and ρ values. The process of the simulation has the following steps:

1. Use the independent variables X and spatial weight matrix W, fix the β(parameters of the independent variables) and ρ as the special values. Note that ρ is between -1 and 1.

2. Generate random values for the elements (i.e., the error term ε with normal distribution of value 0 and constant variance).

3. The variable total crime 11m of lag is deleted because of too much quantity of NA and, due to the bipolar characteristics of the data, logarithmic function is applied at X to adjust the extreme values.

4. Calculate the dependent variable Y.

5. Use the Y and the fixed value of X and W to estimate β and ρ, observing whether it is closed to the fixed β and ρ. Repeat it 1000 times and calculate the mean.

With the theoretical values fixed and the error term defined as the variance $\sigma^2 = 0.01$ and $\sigma^2 = 0.1$ for every element in the vector, the obtained results are described in Tables 8, 9, 10 and 11. Comparing to Table 7, we can see that the results between the model and the simulations match well.

Table 8 – Mean value of the beta and ρ when $\sigma^2 = 0.01$ in SAR model

parameters	theoretical values	estimations
ρ	0.078	0.078
Intercept	−5.505	−5.505
total crime 1m of lag	0.005	0.005
total crime 2m of lag	0.011	0.010
Population	0.016	0.017
population of age 60+	1.022	1.022
registration at the municipal pre school	0.018	0.018
registration at the state elementary school(beginning)	0.002	0.002
registration at the municipal elementary school(final)	0.005	0.005
registration of municipal high school	-0.011	-0.011

Source: Elaborated by the author.

Table 9 – Parameter's bias and RMSE when $\sigma^2 = 0.01$ in SAR model.

parameters	bias	RMSE
ρ	<0.0001	0.0004
Intercept	<0.0001	0.0042
total crime 1m of lag	<0.0001	0.0002
total crime 2m of lag	<0.0001	0.0002
Population	<0.0001	0.0007
population of age 60+	-<0.0001	0.0008
registration at the municipal pre school	-<0.0001	0.0007
registration at the state elementary school(beginning)	<0.0001	0.0001
registration at the municipal elementary school(final)	-<0.0001	0.0001
registration of municipal high school	<0.0001	<0.0001

Source: Elaborated by the author.

Table 10 – Mean value of the beta and ρ when $\sigma^2 = 0.1$ in SAR model.

parameters	theoretical values	estimations
ρ	0.078	0.078
Intercept	-5.505	-5.505
total crime 1m of lag	0.006	0.006
total crime 2m of lag	0.011	0.011
Population	0.017	0.017
population of age 60+	1.021	1.022
registration at the municipal pre school	0.019	0.019
registration at the state elementary school(beginning)	0.002	0.002
registration at the municipal elementary school(final)	0.005	0.005
registration of municipal high school	-0.011	-0.011

Source: Elaborated by the author.

Table 11 – Parameter’s bias and RMSE when $\sigma^2 = 0.1$ in SAR model.

parameters	bias	RMSE
ρ	<0.0001	<0.0001
Intercept	<0.0001	0.006
total crime 1m of lag	<0.0001	<0.0001
total crime 2m of lag	<0.0001	<0.0001
Population	<0.0001	<0.0001
population of age 60+	0	<0.0001
registration at the municipal pre school	<0.0001	<0.0001
registration at the state elementary school(beginning)	<0.0001	<0.0001
registration at the municipal elementary school(final)	<0.0001	<0.0001
registration of municipal high school	<0.0001	<0.0001

Source: Elaborated by the author.

From Tables 8 and 9, 10, 11, it is possible to recover the parameters values, since the bias and RMSE have small values in every parameter.

Additionally, simulations are also performed with ρ is 0.3 or 0.7, since the two values are neither big nor small, and subsequently repeating the procedure with σ^2 = 0.1. The obtained results are described in Tables 12,13,14,15,16. The simulation results show that the model are not sensitive to different parameter setting, with the exception of ρ.

Table 12 – Mean values of betas in different ρ and σ^2 values in SAR model.

parameters	$\rho = 0.3, \sigma^2 = 0.01$	$\rho = 0.7, \sigma^2 = 0.01$
ρ	0.3	0.7
Intercept	-5.505	-5.506
total crime 1m of lag	0.006	0.006
total crime 2m of lag	0.011	0.011
Population	0.017	0.017
population of age 60+	1.022	1.022
registration at the municipal pre school	0.019	0.019
registration at the state elementary school(beginning)	0.002	0.002
registration at the municipal elementary school(final)	0.005	0.005
registration of municipal high school	-0.011	-0.011

Source: Elaborated by the author.

Table 13 – Mean values of betas in different ρ and σ^2 values in SAR model.

parameters	$\rho = 0.3, \sigma^2 = 0.1$	$\rho = 0.7, \sigma^2 = 0.1$
ρ	0.3	0.7
Intercept	-5.506	-5.505
total crime 1m of lag	0.006	0.006
total crime 2m of lag	0.011	0.011
Population	0.017	0.016
population of age 60+	1.022	1.022
registration at the municipal pre school	0.018	0.019
registration at the state elementary school(beginning)	0.002	0.002
registration at the municipal elementary school(final)	0.005	0.005
registration of municipal high school	-0.011	-0.011

Source: Elaborated by the author.

Table 14 – Parameter's bias and RMSE when $\sigma^2 = 0.1$ and $\rho = 0.3$ in SAR model.

parameters	bias	RMSE
ρ	0	0
Intercept	<0.0001	0.006
total crime 1m of lag	-<0.0001	<0.0001
total crime 2m of lag	<0.0001	<0.0001
Population	<0.0001	<0.0001
population of age 60+	<0.0001	<0.0001
registration at the municipal pre school	-<0.0001	<0.0001
registration at the state elementary school(beginning)	-<0.0001	<0.0001
registration at the municipal elementary school(final)	-<0.0001	<0.0001
registration of municipal high school	-<0.0001	<0.0001

Source: Elaborated by the author.

Table 15 – Parameter's bias and RMSE when $\sigma^2 = 0.01$ and $\rho = 0.3$ in SAR model.

parameters	bias	RMSE
ρ	0	0
Intercept	-<0.0001	0.001
total crime 1m of lag	<0.0001	<0.0001
total crime 2m of lag	<0.0001	<0.0001
Population	<0.0001	<0.0001
population of age 60+	<0.0001	<0.0001
registration at the municipal pre school	-<0.0001	<0.0001
registration at the state elementary school(beginning)	<0.0001	<0.0001
registration at the municipal elementary school(final)	-<0.0001	<0.0001
registration of municipal high school	<0.0001	<0.0001

Source: Elaborated by the author.

Table 16 – Parameter's bias and RMSE when $\sigma^2 = 0.1$ and $\rho = 0.7$ in SAR model.

parameters	bias	RMSE
ρ	0	0
Intercept	<0.0001	0.009
total crime 1m of lag	<0.0001	<0.0001
total crime 2m of lag	<0.0001	<0.0001
Population	<0.0001	<0.0001
population of age 60+	<0.0001	<0.0001
registration at the municipal pre school	<0.0001	<0.0001
registration at the state elementary school(beginning)	<0.0001	<0.0001
registration at the municipal elementary school(final)	<0.0001	<0.0001
registration of municipal high school	<0.0001	<0.0001

Source: Elaborated by the author.

For the bias and RMSE suppose Y_i is the value of ith element of a vector, $\hat{Y}$ is the fixed value of the parameter and X_i is the value of ith element of another vector. Then, the bias formula can be shown as

$$bias = \frac{\sum_{i=1}^{N}(Y_i - \hat{Y})}{N}$$

the formula of RMSE is

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Y_i - \hat{Y})^2}{N}}$$

The purpose of these simulations are to verify the authenticity of the SAR model fitting by the R package of (*BIVAND*, 2022*a*). And based on the results of the above simulations, it is possible to conclude that the SAR model fitted in the R code is concrete. Both in the simulations with the original parameter values and in the simulation with the user defined parameter values, the model has shown good performances, both in the perspective of bias and RMSE. In the next subchapter, after fitting SAR model for the data of the metropolitan area of São Paulo, the same simulation will repeat again.

4.3 Fitted SAR Model and Simulation at the Region of Greater São Paulo from 2002-2017

The dataset includes all explanatory variables spanning the period 2002 to 2017. Consequently, I opted to replicate the same procedure as previously outlined. Notably, the variable "total of crimes" is substituted with "crime per capita". The findings from this analysis are presented in Table 17.

Table 17 – Fitted SAR model for crimes at the region of greater São Paulo after applied logarithmic function.

Variables and Intercept	Estimate	Std. Error	z value	p-value
ρ	0.5058			<0.0001
(Intercept)	3.122	0.153	20.396	<0.0001
total crime 1m of lag	0.004	0.002	2.67	0.008
total crime 2m of lag	0.006	0.001	5.18	0
Population	−0.74	0.028	−26.082	<0.0001
population of age 60+	0.786	0.0272	28.918	<0.0001
registration at the municipal pre school	0.0440	0.008	5.7846	0
registration at the state elementary school(beginning)	−0.003	0.001	−3.388	0.001
registration at the municipal elementary school(final)	0.008	0.0007	10.103	<0.0001
registration of municipal high school	−0.014	0.001	−11.437	<0.0001

Source: Elaborated by the author.

Evidently, there is a notable enhancement in the model's performance, particularly in the context of the ρ value. This indicates that the geographical factor has provided a constructive explanation for the crime dynamics within the São Paulo metropolitan region.

Despite the considerable rise in the ρ value, it's important to continuing to conduct simulations. Employing predetermined theoretical values while considering the error term defined as the variance, with σ^2 set at 0.1 for all elements in the vector, the simulation outcomes are presented in 18.

Table 18 – Parameter's bias and RMSE when $\sigma^2 = 0.1$ in SAR model.

parameters	bias	RMSE
ρ	-<0.0001	0.0002
Intercept	-<0.0001	8.768
total crime 1m of lag	-<0.0001	<0.0001
total crime 2m of lag	0	0
Population	-<0.0001	0.0001
population of age 60+	-<0.0001	0.0001
registration at the municipal pre school	<0.0001	<0.0001
registration at the state elementary school(beginning)	<0.0001	<0.0001
registration at the municipal elementary school(final)	<0.0001	<0.0001
registration of municipal high school	<0.0001	<0.0001

Source: Elaborated by the author.

Additionally, the simulations are conducted with $\rho = 0.3$ or 0.7. The obtained results are described in Tables 19, 20, 21, 22, 23, 24.

Table 19 – Parameter's bias and RMSE when $\sigma^2 = 0.1$ and $\rho = 0.3$ in SAR model.

parameters	bias	RMSE
ρ	-<0.0001	0.0113
Intercept	<0.0001	0.0547
total crime 1m of lag	<0.0001	0.0006
total crime 2m of lag	-<0.0001	0.0004
Population	<0.0001	0.0106
population of age 60+	-<0.0001	0.0112
registration at the municipal pre school	-<0.0001	0.0015
registration at the state elementary school(beginning)	-<0.0001	0.0002
registration at the municipal elementary school(final)	-<0.0001	0.0001
registration of municipal high school	<0.0001	0.0004

Source: Elaborated by the author.

Table 20 – Parameter's bias and RMSE when $\sigma^2 = 0.01$ and $\rho = 0.3$ in SAR model.

parameters	bias	RMSE
ρ	<0.0001	0.0010
Intercept	-<0.0001	0.0050
total crime 1m of lag	-<0.0001	0.0001
total crime 2m of lag	-<0.0001	0.0001
Population	<0.0001	0.0009
population of age 60+	<0.0001	0.0009
registration at the municipal pre school	<0.0001	0.0002
registration at the state elementary school(beginning)	-<0.0001	<0.0001
registration at the municipal elementary school(final)	-<0.0001	<0.0001
registration of municipal high school	<0.0001	<0.0001

Source: Elaborated by the author.

Table 21 – Parameter's bias and RMSE when $\sigma^2 = 0.01$ and $\rho = 0.7$ in SAR model.

parameters	bias	RMSE
ρ	-<0.0001	0.0009
Intercept	-<0.0001	0.0096
total crime 1m of lag	-<0.0001	0.0001
total crime 2m of lag	<0.0001	<0.0001
Population	-<0.0001	0.0009
population of age 60+	<0.0001	0.0010
registration at the municipal pre school	-<0.0001	0.0003
registration at the state elementary school(beginning)	<0.0001	<0.0001
registration at the municipal elementary school(final)	-<0.0001	<0.0001
registration of municipal high school	<0.0001	<0.0001

Source: Elaborated by the author.

Table 22 – Parameter's bias and RMSE when $\sigma^2 = 0.1$ and $\rho = 0.7$ in SAR model.

parameters	bias	RMSE
ρ	0	0.0092
Intercept	<0.0001	0.0848
total crime 1m of lag	<0.0001	0.0004
total crime 2m of lag	0	0.0002
Population	0	0.0104
population of age 60+	-<0.0001	0.0104
registration at the municipal pre school	0	0.0030
registration at the state elementary school(beginning)	<0.0001	0.0002
registration at the municipal elementary school(final)	<0.0001	0.0001
registration of municipal high school	<0.0001	0.0004

Source: Elaborated by the author.

Table 23 – Mean values of betas in different ρ and σ^2 in SAR model.

parameters	$\rho = 0.3, \sigma^2 = 0.01$	$\rho = 0.7, \sigma^2 = 0.01$
ρ	0.3	0.6998
Intercept	3.1240	3.126
total crime 1m of lag	0.0041	0.0041
total crime 2m of lag	0.0060	0.0060
Population	−0.7403	−0.7405
population of age 60+	0.7859	0.7859
registration at the municipal pre school	0.0440	0.0441
registration at the state elementary school(beginning)	−0.0026	−0.0026
registration at the municipal elementary school(final)	0.0073	0.0072
registration of municipal high school	−0.0142	−0.0142

Source: Elaborated by the author.

Table 24 – Mean values of betas in different ρ and σ^2 in SAR model.

parameters	$\rho = 0.3, \sigma^2 = 0.1$	$\rho = 0.7, \sigma^2 = 0.1$
ρ	0.3049	0.7
Intercept	3.098	3.122
total crime 1m of lag	0.0040	0.0040
total crime 2m of lag	0.0062	0.0061
Population	−0.7395	−0.7415
population of age 60+	0.7866	0.7885
registration at the municipal pre school	0.0424	0.0425
registration at the state elementary school(beginning)	−0.0026	−0.0026
registration at the municipal elementary school(final)	0.0071	0.0073
registration of municipal high school	−0.0140	−0.0141

Source: Elaborated by the author.

4.4 Artificial Neural Network and LSTM at the Region of Greater São Paulo

Given the intention to establish a spatial-temporal model, I introduce a new variable: population density (calculated by dividing the population by the city's area), aiming at assessing the influence of geographical factors. Subsequently, an Artificial Neural Networks is constructed and designed with the subsequent architecture:

1. Input Layer: The input layer is explicitly defined as two layers: the first layer is the variables such as: crime per capita, population density, total of population, population with age of more than 60 years, quantity of student registered at the municipal pre-school, registration at the state elementary school (beginning), registration at the municipal elementary school (final), registration of municipal high school. The second layer is the spatial weight matrix W used in SAR model. Then I concatenated the two input layers to create a merged layer, which is essential to let the model understand the two sources of the data.

2. Hidden Layers:

- Dense Layer 1:
  - Number of units: 32
  - Activation function: ReLU (Rectified Linear Unit)
  - Dropout Layer 1:
  - Rate: 0.2 (20% of the inputs are randomly set to 0 during training to reduce overfitting)
- Dense Layer 2:
  - Number of units: 16
  - Activation function: ReLU
- Dropout Layer 2:
  - Rate: 0.2

3. Output Layer:

- Dense Layer 3:
  - Number of units: 1
  - No activation function is specified, meaning it will use a linear activation (identity function).

The model follows a sequential architecture, where the output of each layer serves as the input for the next layer. The activation functions help to introduce non-linearity into the network, enabling it to learn complex patterns and relationships within the data.

The model is then compiled with the mean squared error (MSE) loss function, which is commonly used for regression problems.

The model is then trained using the training data and corresponding target values for 50 epochs, with a batch size of 32. A validation splitting of 0.2 is used, meaning 20% of the training data is reserved for validation during the training process.

Here, it is necessary to explain some terms. An *epoch* refers to one complete pass of the entire training dataset through the machine learning algorithm; in this case, the algorithm is the artificial neural networks's model.

The *batch size* is the number of training examples used in one iteration. In the context of neural networks, it's common to train on subsets of the training data at a time rather than the entire dataset. These subsets are called batches. If the batch size is 32, this means that the model update its parameters after it has seen 32 examples.

When training a model, it's a common practice to split the available data into at least two subsets: one for training the model and another for testing its performance. However, to fine-tuning the parameters of the model and prevent overfitting (where the model learns the training data too well and performs poorly on unseen data), a third subset is often created from the original training data, called the validation set. A validation split of 0.2 means that 20% of the total dataset is used as the validation set and the remaining 80% is used for training. It is worth to note that validation set is not a testing data set.

In this neural network, other parameters were also tested. However, those values were discarded because they yielded not ideal results. The parameter value defined in the network was determined through an exhaustive method.

Similar to the ANN is LSTM model. The input layer of the LSTM model is the same as the ANN: explicitly defined in the code as two layers then concatenating the two input layers to create a merged layer. Then is the LSTM layer with 32 LSTM units. And finally, an output layer with one unit and linear activation. The number of training epochs is 50 , with batch size of 32 and validation split of 20% of the data during training.

After the training of the model, it used the test data to see the performance of the ANN and LSTM, and then compared to the SAR model. The result of the comparison is in table 25. Based on the test data, the results in Table 25 demonstrate the superior predicting performance of the ANN and LSTM model when compared to SAR model for predicting crimes in the metropolitan area of São Paulo state. However, it is important to note that, as a regression model, the SAR model can estimate the relationship between the response variable and the explanatory variables, thus providing interpretability to the regression coefficients. Additionally, the SAR model enables the computation of p-values when considering the marginal significance of each parameter.

Finally, based on the results of different models, a choropleth plot illustrates the predicted crime per capita in the metropolitan area of São Paulo (Fig. 15).The first figure shows the choropleth plot of crime per capita on the metropolitan area of São Paulo, which is actually the Figure 8. The result of the plot shows the superior performance of the LSTM model in predicting crime per capita in the metropolitan area of São Paulo.

Table 25 – Predictive performance comparison between SAR model, ANN and LSTM model.

Measure	SAR model	ANN	LSTM
RMSE	77.18	21.5	22.64
MAE	60.77	19.93	19.16

Source: Elaborated by the author.

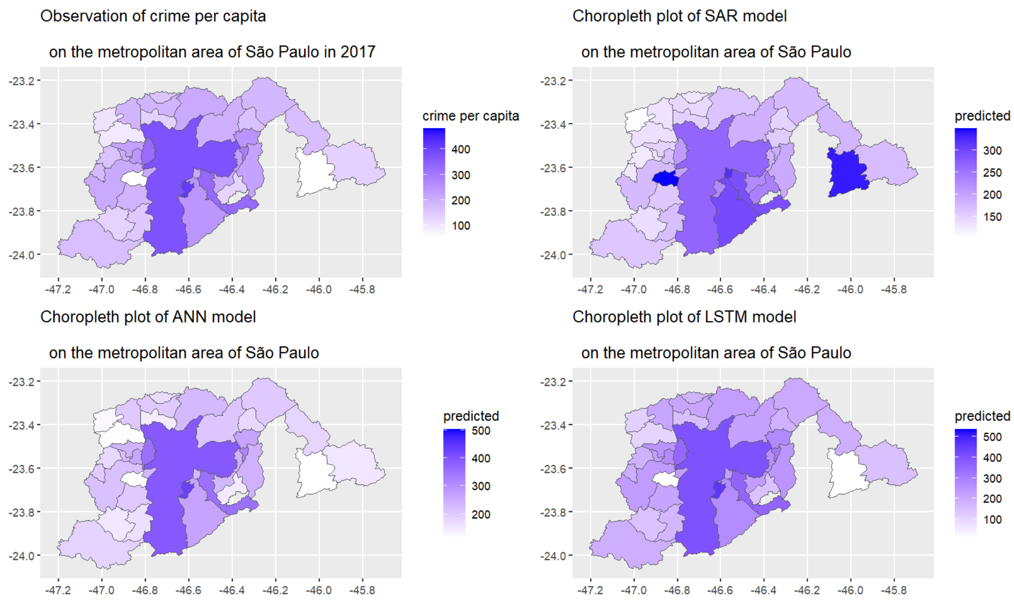

Figure 15 – Choropleth plot of crime per capita in the metropolitan area of São Paulo, based on the predicted data.
Source: figure elaborated by the authors

CHAPTER

5

DISCUSSION AND FURTHER STUDIES

The main objective of this work was to understand the association between the occurrence of crimes and demographic factors using various statistical and computational tools. For this purpose, a predictive approach was proposed to modelling crimes in the metropolitan area of São Paulo in Brazil. The preparation of data was pursued by variable selection with Lasso regression, followed by modelling with spatial-autoregressive (SAR) model, artificial neural network models (ANN) and long short-term memory model (LSTM). As urbanization is a famous phenomenon in Latin America, this study presents importance not only for segurity issue, but also for analyzing various other urban problems.

The work is divided in three major parts: the first part is exploratory data analysis. Based on the source of the data, I make the choropleth's plot and correlogram's plot of São Paulo state and the metropolitan area of São Paulo.

The second part is the theoretical aspect of different models. It briefly reviews linear regression, Lasso, the SAR model, Artificial Neural Networks, RNN, and LSTM. Specifically, in the SAR model section, not only is the model itself introduced, but also the calculation of the derivative-based log-likelihood function. The neural structure and learning algorithm are presented as well. Lastly, the RNN and LSTM are discussed.

The last part is the application part. First of all, the SAR model has been applied to analyze the data of São Paulo state and the region of greater São Paulo. However, the results of the model are not ideal, especially presenting the low value of ρ. For this reason, we perform extensive simulations to verify the authentic of the model. After that, some modifications have been taken and then SAR model has been re-applied to the data of the region of greater São Paulo. The last part is to apply Artificial Neural Network and LSTM for crime prediction and the results are compared with SAR model. Our conclusion is: concerning prediction, Artifical Neural Network, LSTM are better than SAR model, however SAR model permits the data interpretatton.

In the future, one can use P-spline to continue the idea of this work. P-spline is a smoothing technique (MARX; EILERS, 1999) . In the previous phase of our work, we observed that while the SAR model had limitations in terms of precision, it excelled in its ability to interpret the data. We believe that by introducing P-splines into our methodology, it will make a balance between precision and capacity to make interpretation. This will enable to create models that not only make more accurate crime predictions but also provide meaningful insights into the underlying factors driving crime rates.

CHAPTER 6

APPENDIX

The link of my github is:

<https://github.com/weltonyuanhe/Security-modelling-in-the-metropolitan-area-of-Sao-Paulo>

6.1 Lasso

This code in R serves for the LASSO modelling and variable selection

```
install.packages("spdep")
install.packages("spatialreg")
install.packages("rgdal")
install.packages("rgeos")
install.packages("jsonlite")
install.packages("sf")
install.packages("lassopv")
install.packages("covTest")



library(tidyverse)  # Modern data science workflow
library(spdep)
library(spatialreg)
library(rgdal)
library(rgeos)
library(readr)
library(jsonlite)
```

```
library(readr)
library(dplyr)
library(glmnet)
library(lassopv)
library(covTest)



# Change the presentation of decimal numbers to 4 and avoid scientific notation
options(prompt="R> ", digits=4, scipen=7)

agregado_tudo <- read_csv("Desktop/pesquisa 2/agregado_tudo.csv")
View(agregado_tudo)
summary(agregado_tudo)

agregado_tudo <- agregado_tudo %>%
  mutate(across(-1,  ~ as.numeric(replace(., . == '', 0))))

agregado_tudo_lasso <- agregado_tudo[ , -
which(names(agregado_tudo) %in% c("Cod IBGE","name"
,"tempo","Unnamed: 0","Localidades","Área (Em
km2)","Densidade Demográfica (Habitantes/km2)"
,"Grau de Urbanização (Em %)","Hab./Domicílio","Total
H","Total M","00 a 14","15 a 29","30 a 59","60 e mais"
,"Cod_Ibge_x","Localidade_x","cod
municipio","Localidade_y","Cod_Ibge_y","cod_ibge_x","Cod_Ibg
e
","cod_ibge_y" ))]



vc <- "https://raw.githubusercontent.com/tbrugz/geodata-
br/master/geojson/geojs-35-mun.json"
downloader::download(url = vc, destfile = "/tmp/gas.GeoJSON")
municipio <- readOGR(dsn = "/tmp/gas.GeoJSON")

newdata1 <- setdiff(municipio$name, agregado_tudo$name)
newdata2 <- setdiff(agregado_tudo$name, municipio$name)
```

```
municipio <- subset(municipio, name != "Natividade da Serra")
municipio <- subset(municipio, name != "Nova Canaã Paulista")
municipio <- subset(municipio, name != "Mirassolândia")
municipio <- subset(municipio, name != "Nazaré Paulista")
municipio <- subset(municipio, name != "Nova Odessa")
municipio <- subset(municipio, name != "Nipoã")
municipio <- subset(municipio, name != "Nova Castilho")
municipio <- subset(municipio, name != "Nhandeara")
municipio <- subset(municipio, name != "Aparecida d'Oeste")
municipio <- subset(municipio, name != "Estrela d'Oeste")
municipio <- subset(municipio, name != "Guarani d'Oeste")
municipio <- subset(municipio, name != "Moji Mirim")
municipio <- subset(municipio, name != "Nova Aliança")
municipio <- subset(municipio, name != "Palmeira d'Oeste")
municipio <- subset(municipio, name != "Santa Bárbara
d'Oeste")
municipio <- subset(municipio, name != "Santa Rita d'Oeste")
municipio <- subset(municipio, name != "São João do Pau
d'Alho")
municipio <- subset(municipio, name != "São Luís do
Paraitinga")

summary(municipio)
head(municipio)
nrow(municipio)


nmunicipio <- municipio[rep(seq_len(nrow(municipio)), each =
24), ]  # Base R


summary(nmunicipio)
head(nmunicipio)
nrow(nmunicipio)


queens <- poly2nb(municipio, queen = TRUE)
queens[[1]]
str(queens)
```

```
alistw <- nb2listw(queens,style = "W" ,zero.policy = TRUE)
summary(alistw)
names(alistw)
alistw$neighbours[1]


alistw$weights[1]


plot(municipio, borders = 'lightgrey')
plot(queens, coordinates(municipio), pch = 19, cex = 0.6,
add = TRUE, col = "red")



nqueens <- poly2nb(nmunicipio, queen = TRUE)
nqueens[[1]]
nqueens[[2]]
str(nqueens)

a2listw <- nb2listw(nqueens,style = "W" ,zero.policy = TRUE)
a2listw
names(a2listw)
a2listw$neighbours[1]





# VTI_texteis + VTI_vestuario + VTI_couro + VTI_celulose +
VTI_impressao + VTI_petroleo + VTI_biocombustíveis +
VTI_quimica + VTI_farmaceutica + VTI_min_n_met +
VTI_metalurgia +VTI_prod_div +tx_abandono_fai_publica   +
tx_abandono_fai_privada+ tx_aprovacao_faf_publica_x +
tx_reprovacao_faf_publica + tx_abandono_faf_publica  +
tx_reprovacao_faf_privada + tx_abandono_faf_privada
+tx_abandono_medio_publica+ matriculas_creche_estadual
```

```
reg3 <- lagsarlm(formula =

                  Total_de_crimes ~
               #  Total_de_crimes1m + Total_de_crimes2m
               # + Total_de_crimes3m + Total_de_crimes4m +
               Total_de_crimes5m +Total_de_crimes6m
               # + Total_de_crimes7m + Total_de_crimes8m +
               Total_de_crimes9m + Total_de_crimes10m
               # + Total_de_crimes11m +
                  Total_de_crimes12m #+ População + area +
                  Grau + habdom
               # + trinta_a_cinquentanove + sessenta_e_mais
               + Valor_Agropecuaria + Valor_Serviços

               # + Valor_Adi + Valor_PIB_per_capita +
               VTI_bebidas + VTI_metal + VTI_outros_equi
            #     + tx_aprovacao_fai_publica   +
            tx_reprovacao_fai_publica
             #    + tx_aprovacao_fai_privada +
             tx_reprovacao_fai_privada
            #     + tx_aprovacao_medio_publica +
            tx_reprovacao_medio_publica
             #    + tx_aprovacao_medio_privada  +
             tx_reprovacao_medio_privada
              #   + tx_abandono_medio_privada  +
              matriculas_creche_municipal
              #   + matriculas_creche_particular +
              matriculas_pre_escola_estadual
              #   + matriculas_pre_escola_municipal +
              matriculas_pre_escola_particular
              #   + matriculas_ESAI_estadual +
              matriculas_ESAI_municipal
              #   + matriculas_ESAI_particular +
              matriculas_ESAF_estadual
              #   + matriculas_ESAF_municipal +
              matriculas_ESAF_particular
              #   + matriculas_EM_estadual +
              matriculas_EM_municipal
              #   + matriculas_EM_particular,
```

```
                 ,data = agregado_tudo, listw =
                 a2listw,na.action = na.omit)
summary(reg3,Nagelkerke=T)


#-------------------lasso

cols.dont.want <- "Total_de_crimes"

y  <- as.matrix(agregado_tudo_lasso$Total_de_crimes)
X <- as.matrix(data.frame(agregado_tudo_lasso[,!
names(agregado_tudo_lasso) %in% cols.dont.want, drop = F]))
X[is.na(X)] <- 0
y[is.na(y)] <- 0


lambdas_to_try <- 10^seq(-3, 5, length.out = 100)

lasso_cv <- cv.glmnet(X, y, alpha = 1, lambda =
lambdas_to_try,
                      standardize = TRUE, nfolds = 3)

plot(lasso_cv)
lasso_cv$lambda.min

coef(lasso_cv, s = "lambda.min")

summary(lasso_cv)
W <- as.matrix(coef(lasso_cv, s = "lambda.min"))
W

keep_X <- rownames(W)[W!=0]
keep_X <- keep_X[!keep_X == "(Intercept)"]
X <- X[,keep_X]
summary(lm(y~X))
```

6.2 SAR simulation Code

This code in R serves for the SAR modelling and the simulation of the SAR model

```
install.packages("tidyverse")
install.packages("MASS")
install.packages("matlib")
install.packages("stringr")
install.packages("graphics")
install.packages("zoo")
install.packages("Metrics")
install.packages("qtl2")

library(tidyverse)  # Modern data science workflow
library(spdep)
library(spatialreg)
library(rgdal)
library(dplyr)
library(stats)
library(glmnet)
library(matlib)
library(MASS)
library(stringr)
library(graphics)
library(zoo)
library(Metrics)
library(qtl2)
options(prompt="R> ", digits=4, scipen=7)

agregado_tudo <- read_csv("pesquisa 2/agregado_tudo.csv")
#View(agregado_tudo)
summary(agregado_tudo)

agregado_tudo$População <- str_replace(agregado_tudo$População, '\\.', '')

agregado_tudo$População <- sapply(agregado_tudo$População, as.numeric)

target <- c('São Paulo', 'Guarulhos','Caieiras', 'Cajamar', 'Francisco Morato',
'Franco da Rocha',
'Mairiporã','Arujá', 'Ferraz de Vasconcelos', 'Guararema',
'Itaquaquecetuba','Guarulhos', 'Mogi das Cruzes', 'Poá',
'Salesópolis',
'Santa Isabel', 'Suzano','Diadema', 'Mauá', 'Santo André',
```

```
'São Bernardo do Campo'
,'São Caetano do Sul', 'Ribeirão Pires', 'Rio Grande da
Serra','Cotia'
, 'Embu-Guaçu', 'Itapecerica da Serra', 'Juquitiba',
'São Lourenço da Serra', 'Taboão da Serra', 'Vargem Grande
Paulista'
,'Barueri', 'Carapicuíba', 'Itapevi', 'Jandira', 'Osasco'
,'Pirapora do Bom Jesus', 'Santana de Parnaíba')
#filtrar as cidades que estao proximo de sp
agregado_tudo <- agregado_tudo %>%
  filter(name %in% target)
nrow(agregado_tudo)
ncol(agregado_tudo)
agregado_tudo$População[which(agregado_tudo$name ==
"Guarulhos")] <- 1404694
agregado_tudo$População[which(agregado_tudo$name == "São
Paulo")] <- 12396372
#agregado_tudo$Total_de_crimes12m[agregado_tudo$Total_de_cri
mes12m == 0] <- NA
#agregado_tudo_1  <- na.locf(na.locf(agregado_tudo))
#agregado_tudo_1 <- agregado_tudo_1[agregado_tudo_1$name !=
"Arujá", ]
nrow(agregado_tudo_1)


################################coletar os dados de latitude e
longitude pelo formato de geojson

vc <- "https://raw.githubusercontent.com/tbrugz/geodata-
br/master/geojson/geojs-35-mun.json"
downloader::download(url = vc, destfile = "/tmp/gas.GeoJSON")
municipio <- readOGR(dsn = "/tmp/gas.GeoJSON")
summary(municipio)

municipio_1 <- subset(municipio, name %in% agregado_tudo$name)
summary(municipio_1)
head(municipio_1)
nrow(municipio_1)
```

```
######################################################## criar a
matriz w

nmunicipio_1 <- municipio_1[rep(seq_len(nrow(municipio_1)), each =
24), ]  # Base R

summary(nmunicipio_1)
head(nmunicipio_1)
nrow(nmunicipio_1)

queens_1 <- poly2nb(municipio_1, queen = TRUE)
queens_1[[1]]
str(queens_1)

alistw_1 <- nb2listw(queens_1,style = "W" ,zero.policy = TRUE)

#W <- nb2mat(queens_1, glist=NULL, style="B", zero.policy=TRUE)



nqueens_1 <- poly2nb(nmunicipio_1, queen = TRUE)
nqueens_1[[1]]
nqueens_1[[2]]
str(nqueens_1)

a2listw_1<- nb2listw(nqueens_1,style = "W" ,zero.policy = TRUE)
a2listw_1
names(a2listw_1)

W <- nb2mat(nqueens_1, glist=NULL, style="W", zero.policy=TRUE)
W


##################### Análise exploratória de dados -  Cibele

attach(agregado_tudo)
matriz = cbind(Total_de_crimes, Total_de_crimes1m,
Total_de_crimes2m, População, sessenta_e_mais,
matriculas_creche_municipal, matriculas_ESAI_estadual,
```

```
matriculas_ESAF_municipal, matriculas_EM_municipal)
pairs(matriz)

matriz2 = cbind(log(Total_de_crimes), log(Total_de_crimes1m),
log(Total_de_crimes2m), log(População), log(sessenta_e_mais),
log(matriculas_creche_municipal), log(matriculas_ESAI_estadual),
log(matriculas_ESAF_municipal), log(matriculas_EM_municipal))
colnames(matriz2) <-
c('log(Total_de_crimes)','log(Total_de_crimes1m)',
'log(Total_de_crimes2m)', 'log(População)',
'log(sessenta_e_mais)', 'log(matriculas_creche_municipal)',
'log(matriculas_ESAI_estadual)', 'log(matriculas_ESAF_municipal)',
'log(matriculas_EM_municipal)')
pairs(matriz2)



########################################################## regressao sar
agregado_tudo_3 <- agregado_tudo[,
c('crime_per_capita','Total_de_crimes1m',
'Total_de_crimes2m','População',

                         'sessenta_e_mais','matriculas_creche_munic
                         ipal','matriculas_ESAI_estadual'

                         ,'matriculas_ESAF_municipal','matriculas_E
                         M_municipal')]

#agregado_tudo_3 <- log(agregado_tudo_3 + 0.01)

agregado_tudo_3$Total_de_crimes1m[is.na(agregado_tudo_3$Total_de_cr
imes1m)] <- 0

agregado_tudo_3$Total_de_crimes1m[is.na(agregado_tudo_3$Total_de_cr
imes1m)] <- 0
agregado_tudo_3$Total_de_crimes2m[is.na(agregado_tudo_3$Total_de_cr
imes2m)] <- 0
agregado_tudo_3$População[is.na(agregado_tudo_3$População)] <- 0
agregado_tudo_3$sessenta_e_mais[is.na(agregado_tudo_3$sessenta_e_ma
```

```
is)] <- 0
agregado_tudo_3$matriculas_creche_municipal[is.na(agregado_tudo_3$m
atriculas_creche_municipal)] <-0
agregado_tudo_3$matriculas_ESAI_estadual[is.na(agregado_tudo_3$matr
iculas_ESAI_estadual)] <- 0
agregado_tudo_3$matriculas_ESAF_municipal[is.na(agregado_tudo_3$mat
riculas_ESAF_municipal)] <- 0

agregado_tudo_3$matriculas_EM_municipal[is.na(agregado_tudo_3$matri
culas_EM_municipal)] <- 0
agregado_tudo_3

reg3 <- lagsarlm(formula =
                   Total_de_crimes ~
                   Total_de_crimes1m + Total_de_crimes2m +
                   #+ Total_de_crimes3m + Total_de_crimes4m +
                   Total_de_crimes5m +Total_de_crimes6m
                   #+ Total_de_crimes7m + Total_de_crimes8m +
                   Total_de_crimes9m + Total_de_crimes10m
                   # Total_de_crimes12m + #+X
                   População
                 #   area + Grau + habdom
                 #+ trinta_a_cinquentanove +
                 + sessenta_e_mais
                 #+ Valor_Agropecuaria + Valor_Serviços
                 #+ Valor_Adi + Valor_PIB_per_capita + VTI_bebidas
                 + VTI_metal +
                 # +  VTI_outros_equi
                 #+ VTI_texteis
                 #+ VTI_min_n_met
                 #tx_aprovacao_fai_publica   +
                 tx_reprovacao_fai_publica
                 # + tx_aprovacao_fai_privada +
                 tx_reprovacao_fai_privada
                 #  + tx_aprovacao_medio_publica +
                 tx_reprovacao_medio_publica

                 # + tx_aprovacao_medio_privada  +
                 tx_reprovacao_medio_privada
```

```
                 #  + tx_abandono_medio_privada
                 + matriculas_creche_municipal
                 #  + matriculas_creche_particular +
                 matriculas_pre_escola_estadual
                 #  + matriculas_pre_escola_municipal +
                 matriculas_pre_escola_particular
                 + matriculas_ESAI_estadual #+
                 matriculas_ESAI_municipal
                 #  + matriculas_ESAI_particular +
                 matriculas_ESAF_estadual
                 +matriculas_ESAF_municipal#+
                 matriculas_ESAF_particular
                 #+  matriculas_EM_estadual
                 +  matriculas_EM_municipal
                 #  + matriculas_EM_particular,
                 ,data = agregado_tudo_3, listw =
                 a2listw_1,type="lag",method="MC",zero.policy =
                 TRUE)
summary(reg3,Nagelkerke=T)



############ Ajuste com log da resposta e log das
covariáveis - sem Total_de_crimes11m


reg6 <- lagsarlm(formula =
                   log(Total_de_crimes+.01) ~
                   log(Total_de_crimes1m+.01) + log(Total_de_crimes2m+.01) +
                   log(População+.01)
                 + log(sessenta_e_mais+.01)
                 + log(matriculas_creche_municipal+.01)
                  + log(matriculas_ESAI_estadual+.01) #+
                  matriculas_ESAI_municipal
                 +log(matriculas_ESAF_municipal+.01)#+ m
                 atriculas_ESAF_particular
                 +  log(matriculas_EM_municipal+.01)
                  , data = agregado_tudo_3, listw = a2listw_1,
                  type="lag",method="MC",zero.policy = TRUE)
```

```
summary(reg6,Nagelkerke=T)


beta_teorico = reg6$coefficients
beta_teorico
# Colocar os resultados desse ajuste acima na monografia


################################################################loop




i <- 1
#todas as vezes que roda o while, muda o nome do arquivo(por
exemplo, test_22 para test_23
#senão no arquivo em vez de registrar 200 vezes no novo arquivo
vai registrar 400 vezes
#os parametros no arquivo velho) o lugar que modifica o nome do
arquivo é no if-else
while(i <= 1000){

I <- diag(888)

rho <- 0.7
epsilon <- rnorm(n =888, mean = 0, sd = 0.1)
epsilon

beta = reg6$coefficients
beta
beta <- data.matrix(beta)
nrow(beta)

Xis <- agregado_tudo_3[, c('Total_de_crimes1m', 'Total_de_crimes2m','População',

                                      'sessenta_e_mais','matriculas_

                                      creche_municipal','matriculas_
```

```
                                    ESAI_estadual'

                                    ,'matriculas_ESAF_municipal','
                                    matriculas_EM_municipal')]


new1 <- c(rnorm(n = 888, mean = 1, sd = 0))

Xis <- cbind(new1,Xis)
#transformar na para log(0.01)
#Xis$new1[is.na(Xis$new1)] <- log(0.01)
#Xis$Total_de_crimes1m[is.na(Xis$Total_de_crimes1m)] <- log(0.01)
#Xis$Total_de_crimes2m[is.na(Xis$Total_de_crimes2m)] <- log(0.01)
#Xis$População[is.na(Xis$População)] <- log(0.01)
#Xis$sessenta_e_mais[is.na(Xis$sessenta_e_mais)] <- log(0.01)
#Xis$matriculas_creche_municipal[is.na(Xis$matriculas_creche_munici
pal)] <-log(0.01)
#Xis$matriculas_ESAI_estadual[is.na(Xis$matriculas_ESAI_estadual)]
<- log(0.01)
#Xis$matriculas_ESAF_municipal[is.na(Xis$matriculas_ESAF_municipal)
] <- log(0.01)
#Xis$matriculas_EM_municipal[is.na(Xis$matriculas_EM_municipal)]
<- log(0.01)


X <-data.matrix(Xis)
X
nrow(X)
ncol(X)
E  = (I - (rho*W))
ES <- solve(E)
ES
ESD  <- X %*% beta + epsilon
ESD
W
nrow(ESD)
ncol(ESD)
y  <- ES %*% ESD #simulacao dos y
y
```

```
agregado_tudo_n <- Xis
agregado_tudo_n$Total_de_crimes1m  <- Xis$Total_de_crimes1m
agregado_tudo_n$Total_de_crimes2m  <- Xis$Total_de_crimes2m
agregado_tudo_n$População           <- Xis$População
agregado_tudo_n$sessenta_e_mais  <- Xis$sessenta_e_mais
agregado_tudo_n$matriculas_creche_municipal  <-
Xis$matriculas_creche_municipal
agregado_tudo_n$matriculas_ESAI_estadual  <-
Xis$matriculas_ESAI_estadual
agregado_tudo_n$matriculas_ESAF_municipal  <-
Xis$matriculas_ESAF_municipal
agregado_tudo_n$matriculas_EM_municipal  <-
Xis$matriculas_EM_municipal
agregado_tudo_n$ipslon <- y
summary(agregado_tudo_n)
ncol(agregado_tudo_n)
ncol(Xis)



#estimacao dos parametros
reg4 <- lagsarlm(formula =
                  ipslon ~
                   Total_de_crimes1m + Total_de_crimes2m +
                    (População)
                  +(sessenta_e_mais)
                  + (matriculas_creche_municipal)
                  + (matriculas_ESAI_estadual) #+
                  matriculas_ESAI_municipal
                  + (matriculas_ESAF_municipal)#+
                  matriculas_ESAF_particular
                  +  (matriculas_EM_municipal)
                 ,data = agregado_tudo_n, listw =
                 a2listw_1,type="lag",method="MC",zero.policy =
                 TRUE)
summary(reg4,Nagelkerke=T)

matrix_coef <- summary(reg4,Nagelkerke=T)$coefficients  # Extract
coefficients in matrix
```

```
matrix_rho <- summary(reg4,Nagelkerke=T)$rho  # Extract
coefficients in matrix
as.numeric(matrix_coef)# Return matrix of coefficients
as.numeric(matrix_rho)

c <- bias(beta,matrix_coef)
d <- rmse(beta,matrix_coef )


#coef <- as.vector(append(matrix_coef,matrix_rho))
#coef <- round(c(matrix_coef, matrix_rho), 3)
# Matrix manipulation to extract estimates
coef <- as.vector(round(c(matrix_coef, matrix_rho,c,d), 7))
coef <- t(coef)
coef


if(i == 1){

write.table(coef, file = "test_44.csv" , #modifica aqui o nome do
arquivo
            append = TRUE,col.names = TRUE, row.names = FALSE)
}
else{
write.table(coef, file = "test_44.csv", #modifica aqui o nome do
arquivo)
            append = TRUE,col.names = FALSE, row.names = FALSE)
}
print(i)
i = i + 1
#my_data <- read.csv("test_7.csv")
#my_data
#close(file)
}

test_10 <-read_csv("test_44.csv")


View(test_10)
```

```
test_10[c('V1',
'V2','V3','V4','V5','V6','V7','V8','V9','V10','V11','V12')] <-
str_split_fixed(
  test_10$'V1" "V2" "V3" "V4" "V5" "V6" "V7" "V8" "V9" "V10" "V11"
  "V12', ' ', 12)

test_10$V1
test_10$V1 <- as.numeric(unlist(test_10$V1))
mean1 <- mean(test_10$V1, na.rm=TRUE)
mean1
bias(mean1,test_10$V1)
rmse(-5.5046977,test_10$V1)
boxplot(test_10$V1)

test_10$V2 <- as.numeric(unlist(test_10$V2))
mean2 <- mean(test_10$V2, na.rm=TRUE)
mean2
bias(mean2,test_10$V2)
rmse(mean2,test_10$V2)
boxplot(test_10$V2)

test_10$V3 <- as.numeric(unlist(test_10$V3))
mean3 <- mean(test_10$V3, na.rm=TRUE)
mean3
bias(mean3,test_10$V3)
rmse(mean3,test_10$V3)
boxplot(test_10$V3)

test_10$V4 <- as.numeric(unlist(test_10$V4))
mean4 <- mean(test_10$V4, na.rm=TRUE)
mean4
bias(mean4,test_10$V4)
rmse(mean4,test_10$V4)

test_10$V5 <- as.numeric(unlist(test_10$V5))
mean5 <- mean(test_10$V5, na.rm=TRUE)
mean5
bias(mean5,test_10$V5)
rmse(mean5,test_10$V5)
```

```
test_10$V6 <- as.numeric(unlist(test_10$V6))
mean6 <- mean(test_10$V6, na.rm=TRUE)
mean6
bias(mean6,test_10$V6)
rmse(mean6,test_10$V6)


test_10$V7 <- as.numeric(unlist(test_10$V7))
mean7 <- mean(test_10$V7, na.rm=TRUE)
mean7
bias(mean7,test_10$V7)
rmse(mean7,test_10$V7)


test_10$V8 <- as.numeric(unlist(test_10$V8))
mean8 <- mean(test_10$V8, na.rm=TRUE)
mean8
bias(mean8,test_10$V8)
rmse(mean8,test_10$V8)


test_10$V9 <- as.numeric(unlist(test_10$V9))
mean9 <- mean(test_10$V9, na.rm=TRUE)
mean9
bias(mean9,test_10$V9)
rmse(mean9,test_10$V9)



test_10$V10
test_10$V10 <- as.numeric(unlist(test_10$V10))
mean10 <- mean(test_10$V10, na.rm=TRUE)
mean10
bias(mean10,test_10$V10)
rmse(mean10,test_10$V10)
boxplot(test_10$V10)
```

6.3 prediction SAR

This part of the code serves for the prediction of the SAR model

```
file.edit("~/.Renviron")
#install.packages("tidyverse")
#install.packages("MASS")
#install.packages("matlib")
#install.packages("stringr")
#install.packages("graphics")
#install.packages("zoo")
#install.packages("Metrics")
#install.packages("qtl2")
#install.packages("janitor")
install.packages("spdep")
install.packages("classInt")

installed.packages()
library(sf)
library(tidyverse)  # Modern data science workflow
library(spdep)
library(spatialreg)
library(rgdal)
library(dplyr)
library(stats)
library(glmnet)
#library(matlib)
library(MASS)
library(stringr)
library(graphics)
library(zoo)
library(Metrics)
library(qtl2)
library(janitor)
options(prompt="R> ", digits=4, scipen=7)


atn1 <- read_csv("pesquisa 2/atn1.csv")
atn2 <- read_csv("pesquisa 2/atn2.csv")
```

```
#View(atn1)
agregado_tudo_novo_1 <- atn1
agregado_tudo_novo_2 <- atn2

#agregado_tudo_novo_8 <- read_csv("Desktop/pesquisa
#2/agregado_tudo_novo_8.csv")
#agregado_tudo_novo_9csv <- read_csv("Desktop/pesquisa
#2/agregado_tudo_novo_9csv.csv")

#View(agregado_tudo_novo_1)
#View(agregado_tudo_novo_2)

agregado_tudo_novo_1$População <-
str_replace(agregado_tudo_novo_1$População, '\\.', '')

agregado_tudo_novo_1$População <-
sapply(agregado_tudo_novo_1$População, as.numeric)

target <- c('São Paulo', 'Guarulhos','Caieiras', 'Cajamar',
            'Francisco Morato', 'Franco da Rocha',
            'Mairiporã','Arujá', 'Ferraz de Vasconcelos'
            ,'Guararema',
            'Itaquaquecetuba', 'Mogi das
            Cruzes', 'Poá', 'Salesópolis',
            'Santa Isabel', 'Suzano','Diadema', 'Mauá',
            'Santo André', 'São Bernardo do Campo'
            ,'São Caetano do Sul', 'Ribeirão Pires', 'Rio
            Grande da Serra','Cotia'
            ,'Embu','Biritiba-Mirim', 'Embu-Guaçu', 'Itapecerica da Serra',
            'Juquitiba',
            'São Lourenço da Serra', 'Taboão da Serra',
            'Vargem Grande Paulista'
            ,'Barueri', 'Carapicuíba', 'Itapevi',
            'Jandira',
            'Osasco'
            ,'Pirapora do Bom Jesus', 'Santana de Parnaíba')

#filtrar as cidades que estao proximo de sp
```

```
agregado_tudo_novo_1 <- agregado_tudo_novo_1 %>%
  filter(name %in% target)
nrow(agregado_tudo_novo_1)
ncol(agregado_tudo_novo_1)
agregado_tudo_novo_1$População[which(agregado_tudo_novo_1$name ==
"Guarulhos")] <- 1404694
agregado_tudo_novo_1$População[which(agregado_tudo_novo_1$name ==
"São Paulo")] <- 12396372
#agregado_tudo$Total_de_crimes12m[agregado_tudo$Total_de_crimes12m
== 0] <- NA
#agregado_tudo_1  <- na.locf(na.locf(agregado_tudo))
#agregado_tudo_1 <- agregado_tudo_1[agregado_tudo_1$name !=
"Arujá", ]
nrow(agregado_tudo_novo_1)


vc <- "https://raw.githubusercontent.com/tbrugz/geodata-
br/master/geojson/geojs-35-mun.json"
dir.create("C:/temp")
downloader::download(url = vc, destfile = "C:/temp/gas.GeoJSON")
municipio <- st_read("C:/temp/gas.GeoJSON")
summary(municipio)

municipio_1 <- subset(municipio, name %in%
agregado_tudo_novo_1$name)
summary(municipio_1)
head(municipio_1)
nrow(municipio_1)



#############################################################

agregado_tudo_novo_1$Total_de_crimes <-
as.numeric(agregado_tudo_novo_1$Total_de_crimes)
agregado_tudo_novo_1$População <-
as.numeric(agregado_tudo_novo_1$População)

agregado_tudo_novo_1$crime_per_capita <-
```

```
(agregado_tudo_novo_1$Total_de_crimes /
agregado_tudo_novo_1$População) * 100000

agregado_tudo_novo_1$crime_per_capita

df_1 <- agregado_tudo_novo_1[c("name", "Total_de_crimes",
"tempo", "População", "crime_per_capita")]


write.csv(df_1, "agregado_tudo_novo_6.csv", row.names =
FALSE)
View(agregado_tudo_novo_6)
##########################################################
#criar a matriz w

nmunicipio_1 <- municipio_1[rep(seq_len(nrow(municipio_1)),
each = 192), ]  # Base R

summary(nmunicipio_1)
head(nmunicipio_1)
nrow(nmunicipio_1)

queens_1 <- poly2nb(municipio_1, queen = TRUE)
queens_1[[1]]
str(queens_1)

alistw_1 <- nb2listw(queens_1,style = "W" ,zero.policy = TRUE)

#W <- nb2mat(queens_1, glist=NULL, style="B",
#zero.policy=TRUE)



nqueens_1 <- poly2nb(nmunicipio_1, queen = TRUE)
nqueens_1[[1]]
nqueens_1[[2]]
str(nqueens_1)

a2listw_1<- nb2listw(nqueens_1,style = "W" ,zero.policy = TRUE)
```

```
a2listw_1
names(a2listw_1)

W <- nb2mat(nqueens_1, glist=NULL, style="W", zero.policy=TRUE)
W
dim(W)



attach(agregado_tudo_novo_1)

matriz = cbind(crime_per_capita, Total_de_crimes1m,
Total_de_crimes2m, População,’60 e mais’,
matriculas_creche_municipal, matriculas_ESAI_estadual,
matriculas_ESAF_municipal, matriculas_EM_municipal)
str(matriz)
matriz <- data.frame(crime_per_capita, Total_de_crimes1m,
Total_de_crimes2m, População, ’60 e mais’,
matriculas_creche_municipal, matriculas_ESAI_estadual,
matriculas_ESAF_municipal, matriculas_EM_municipal)

matriz$crime_per_capita <-
as.numeric(matriz$crime_per_capita)
matriz$Total_de_crimes1m <-
as.numeric(matriz$Total_de_crimes1m)
matriz$Total_de_crimes2m <-
as.numeric(matriz$Total_de_crimes2m)
matriz$População <- as.numeric(matriz$População)
unique(matriz$"X60.e.mais")

matriz$"X60.e.mais" <- as.numeric(matriz$"X60.e.mais")
matriz$matriculas_creche_municipal <-
as.numeric(matriz$matriculas_creche_municipal)
matriz$matriculas_ESAI_estadual <-
as.numeric(matriz$matriculas_ESAI_estadual)
matriz$matriculas_ESAF_municipal <-
as.numeric(matriz$matriculas_ESAF_municipal)
matriz$matriculas_EM_municipal <-
as.numeric(matriz$matriculas_EM_municipal)
```

```
matriz_numeric <- matriz[sapply(matriz, is.numeric)]

pairs(matriz_numeric)

matriz2 = cbind(log(Total_de_crimes),
log(Total_de_crimes1m), log(Total_de_crimes2m),
log(População), log('60 e mais'),
log(matriculas_creche_municipal),
log(matriculas_ESAI_estadual),
log(matriculas_ESAF_municipal), log(matriculas_EM_municipal))
colnames(matriz2) <-
c('log(Totaldecrimes)','log(Totaldecrimes1m)',
'log(Totaldecrimes2m)', 'log(População)',
'log(sessenta_e_mais)', 'log(matriculas_creche_municipal)',
'log(matriculas_ESAI_estadual)',
'log(matriculas_ESAF_municipal)',
'log(matriculas_EM_municipal)')

pairs(matriz2)



##########################################################
regressao sar
agregado_tudo_3 <- agregado_tudo_novo_1[,
c('crime_per_capita','Total_de_crimes1m',
'Total_de_crimes2m','População','60 e
mais','matriculas_creche_municipal','matriculas_ESAI_estadual'

                                      ,'matriculas_ESAF_municipal','m
                                      atriculas_EM_municipal')]

#agregado_tudo_3 <- log(agregado_tudo_3 + 0.01)_
names(agregado_tudo_3)[names(agregado_tudo_3) == "60 e mais"] <-
"sessenta_e_mais"
nrow(agregado_tudo_3)

agregado_tudo_3$Total_de_crimes1m[agregado_tudo_3$Total_de_crimes1m
== ""] <- 0
```

```
agregado_tudo_3$Total_de_crimes2m[agregado_tudo_3$Total_de_crimes2m
== ""] <- 0



agregado_tudo_3$Total_de_crimes1m[is.na(agregado_tudo_3$Total_de_cri
mes1m)] <- 0
agregado_tudo_3$Total_de_crimes2m[is.na(agregado_tudo_3$Total_de_cri
mes2m)] <- 0
agregado_tudo_3$População[is.na(agregado_tudo_3$População)] <- 0
agregado_tudo_3$sessenta_e_mais[is.na(agregado_tudo_3$sessenta_e_mai
s)] <- 0
agregado_tudo_3$matriculas_creche_municipal[is.na(agregado_tudo_3$ma
triculas_creche_municipal)] <-0
agregado_tudo_3$matriculas_ESAI_estadual[is.na(agregado_tudo_3$matri
culas_ESAI_estadual)] <- 0
agregado_tudo_3$matriculas_ESAF_municipal[is.na(agregado_tudo_3$matr
iculas_ESAF_municipal)] <- 0
agregado_tudo_3$matriculas_EM_municipal[is.na(agregado_tudo_3$matric
ulas_EM_municipal)] <- 0
#View(agregado_tudo_3)
nrow(agregado_tudo_3)
ncol(agregado_tudo_3)

agregado_tudo_3$crime_per_capita <-
as.numeric(agregado_tudo_3$crime_per_capita)
agregado_tudo_3$Total_de_crimes1m <-
as.numeric(agregado_tudo_3$Total_de_crimes1m)
agregado_tudo_3$Total_de_crimes2m <-
as.numeric(agregado_tudo_3$Total_de_crimes2m)
agregado_tudo_3$População <-
as.numeric(agregado_tudo_3$População)
agregado_tudo_3$sessenta_e_mais <-
as.numeric(agregado_tudo_3$sessenta_e_mais)
agregado_tudo_3$matriculas_creche_municipal <-
as.numeric(agregado_tudo_3$matriculas_creche_municipal)
agregado_tudo_3$matriculas_ESAI_estadual <-
as.numeric(agregado_tudo_3$matriculas_ESAI_estadual)
agregado_tudo_3$matriculas_ESAF_municipal <-
as.numeric(agregado_tudo_3$matriculas_ESAF_municipal)
```

```
agregado_tudo_3$matriculas_EM_municipal <-
as.numeric(agregado_tudo_3$matriculas_EM_municipal)

#reg3 <- lagsarlm(formula =
          #         crime_per_capita ~
           #        Total_de_crimes1m + Total_de_crimes2m +
                  #+ Total_de_crimes3m + Total_de_crimes4m
                 # + Total_de_crimes5m +Total_de_crimes6m
                  #+ Total_de_crimes7m + Total_de_crimes8m
                 # + Total_de_crimes9m + Total_de_crimes10m
                  # Total_de_crimes12m + #+X
            #       População
                #   area + Grau + habdom
                #+ trinta_a_cinquentanove +
             #    + sessenta_e_mais
                #+ Valor_Agropecuaria + Valor_Serviços
                #+ Valor_Adi + Valor_PIB_per_capita + VT
               # I_bebidas + VTI_metal +
                # +  VTI_outros_equi
                #+ VTI_texteis
                #+ VTI_min_n_met
                #tx_aprovacao_fai_publica   +
               # tx_reprovacao_fai_publica
                # + tx_aprovacao_fai_privada +
               # tx_reprovacao_fai_privada
                #  + tx_aprovacao_medio_publica +
               # tx_reprovacao_medio_publica
                # + tx_aprovacao_medio_privada  +
               # tx_reprovacao_medio_privada
                #  + tx_abandono_medio_privada
              #   + matriculas_creche_municipal
               # #  + matriculas_creche_particular +
               # matriculas_pre_escola_estadual
                #  + matriculas_pre_escola_municipal +
               # matriculas_pre_escola_particular
              #  + matriculas_ESAI_estadual #+ ma
             # triculas_ESAI_municipal
                #  + matriculas_ESAI_particular +
              #  matriculas_ESAF_estadual
```

```
                # +matriculas_ESAF_municipal#+ m
              #  atriculas_ESAF_particular
                 #+  matriculas_EM_estadual
              #   +  matriculas_EM_municipal
                 #  + matriculas_EM_particular,
               #  ,data = agregado_tudo_3, listw =
               #a2listw_1,type="lag",method="MC",zero.policy
               #= TRUE)
#summary(reg3,Nagelkerke=T)



############ Ajuste com log da resposta e log das covariáveis - sem
Total_de_crimes11m

#algumas vezes vai ter variacao de valor do rho
reg6 <- lagsarlm(formula =
                   log(crime_per_capita+.01) ~
                   log(Total_de_crimes1m+.01) +
                   log(Total_de_crimes2m+.01) +
                   log(População+.01)
                 + log(sessenta_e_mais+.01)
                 + log(matriculas_creche_municipal+.01)
                 + log(matriculas_ESAI_estadual+.01) #+
                 matriculas_ESAI_municipal
                 +log(matriculas_ESAF_municipal+.01)#+
                 matriculas_ESAF_particular
                 +  log(matriculas_EM_municipal+.01)

                 , data = agregado_tudo_3, listw = a2listw_1,
                 type="lag",method="MC",zero.policy = TRUE)

summary(reg6,Nagelkerke=T)



#summary(reg6,Nagelkerke=T)
```

```
options(prompt="R> ", digits=4, scipen=7)

agregado_tudo_novo_2$População <-
str_replace(agregado_tudo_novo_2$População, '\\.', '')

agregado_tudo_novo_2$População <-
sapply(agregado_tudo_novo_2$População, as.numeric)

target <- c('São Paulo', 'Guarulhos','Caieiras', 'Cajamar',
            'Francisco Morato', 'Franco da Rocha',
            'Mairiporã','Arujá', 'Ferraz de Vasconcelos'
            ,'Guararema',
            'Itaquaquecetuba', 'Mogi das
            Cruzes', 'Poá', 'Salesópolis',
            'Santa Isabel', 'Suzano','Diadema', 'Mauá',
            'Santo André', 'São Bernardo do Campo'
            ,'São Caetano do Sul', 'Ribeirão Pires', 'Rio
            Grande da Serra','Cotia'
            ,'Embu','Biritiba-Mirim', 'Embu-Guaçu', 'Itapecerica da
            Serra',
            'Juquitiba',
            'São Lourenço da Serra', 'Taboão da Serra',
            'Vargem Grande Paulista'
            ,'Barueri', 'Carapicuíba', 'Itapevi',
            'Jandira',
            'Osasco'
            ,'Pirapora do Bom Jesus', 'Santana de Parnaíba')

#filtrar as cidades que estao proximo de sp
agregado_tudo_novo_2 <- agregado_tudo_novo_2 %>%
  filter(name %in% target)
nrow(agregado_tudo_novo_2)
ncol(agregado_tudo_novo_2)
agregado_tudo_novo_2$População[which(agregado_tudo_novo_2$name ==
"Guarulhos")] <- 1404694
agregado_tudo_novo_2$População[which(agregado_tudo_novo_2$name ==
"São Paulo")] <- 12396372
#agregado_tudo$Total_de_crimes12m[agregado_tudo$Total_de_crimes12m
== 0] <- NA
```

```
#agregado_tudo_1  <- na.locf(na.locf(agregado_tudo))
#agregado_tudo_1 <- agregado_tudo_1[agregado_tudo_1$name !=
"Arujá", ]
nrow(agregado_tudo_novo_2)



agregado_tudo_novo_2$Total_de_crimes <-
as.numeric(agregado_tudo_novo_2$Total_de_crimes)
agregado_tudo_novo_2$População <-
as.numeric(agregado_tudo_novo_2$População)

agregado_tudo_novo_2$crime_per_capita <-
(agregado_tudo_novo_2$Total_de_crimes /
agregado_tudo_novo_2$População) * 100000

df_2 <- agregado_tudo_novo_2[c("name", "Total_de_crimes",
"tempo", "População", "crime_per_capita")]


write.csv(df_1, "agregado_tudo_novo_5.csv", row.names =
FALSE)
View(agregado_tudo_novo_5)




nmunicipio_2 <- municipio_1[rep(seq_len(nrow(municipio_1)),
each = 12), ]  # Base R

summary(nmunicipio_2)
head(nmunicipio_2)
nrow(nmunicipio_2)

queens_2 <- poly2nb(municipio_1, queen = TRUE)
queens_2[[1]]
str(queens_2)

alistw_2 <- nb2listw(queens_2,style = "W" ,zero.policy = TRUE)
```

```
alistw_2
#W <- nb2mat(queens_1, glist=NULL, style="B", zero.policy=TRUE)



nqueens_2 <- poly2nb(nmunicipio_2, queen = TRUE)
nqueens_2[[1]]
nqueens_2[[2]]
str(nqueens_2)

a2listw_2<- nb2listw(nqueens_2,style = "W" ,zero.policy =
TRUE)
a2listw_2
names(a2listw_2)

W2 <- nb2mat(nqueens_2, glist=NULL, style="W",
zero.policy=TRUE)
round(W2)
W2
#W2 <- kronecker(diag(12), W)


beta = reg6$coefficients
beta
beta <- data.matrix(beta)
nrow(beta)

attach(agregado_tudo_novo_2)

my_colnames1 <- colnames(agregado_tudo_novo_2)
my_colnames1 # Apply colnames function
name(agregado_tudo_novo_2)
Xis_3 <-  agregado_tudo_novo_2 %>% dplyr::select(
"Total_de_crimes1m", "Total_de_crimes2m","População",
                                            "60 e mais", "matriculas_creche_municipal",
                                            "matriculas_ESAI_estadual",
                                            "matriculas_ESAF_municipal",
                                            "matriculas_EM_municipal")
Xis_5 <-  agregado_tudo_novo_2 %>%
```

```
dplyr::select("crime_per_capita")

nrow(Xis_3)

Xis_5$crime_per_capita <- as.numeric(Xis_5$crime_per_capita)
Xis_5$crime_per_capita <- log(Xis_5$crime_per_capita + 0.01)
Xis_5$crime_per_capita[is.na(Xis_5$crime_per_capita)] <- 0.01

Xis_3$Total_de_crimes1m <-
as.numeric(Xis_3$Total_de_crimes1m)
Xis_3$Total_de_crimes2m <-
as.numeric(Xis_3$Total_de_crimes2m)
Xis_3$População <- as.numeric(Xis_3$População)
Xis_3$"60 e mais" <- as.numeric(Xis_3$"60 e mais")
Xis_3$matriculas_creche_municipal <-
as.numeric(Xis_3$matriculas_creche_municipal)
Xis_3$matriculas_ESAI_estadual <-
as.numeric(Xis_3$matriculas_ESAI_estadual)
Xis_3$matriculas_ESAF_municipal <-
as.numeric(Xis_3$matriculas_ESAF_municipal)

Xis_3$matriculas_EM_municipal <-
as.numeric(Xis_3$matriculas_EM_municipal)


Xis_3$Total_de_crimes1m <- log(Xis_3$Total_de_crimes1m +
0.01)
Xis_3$Total_de_crimes2m <- log(Xis_3$Total_de_crimes2m +
0.01)
Xis_3$População <- log(Xis_3$População + 0.01)
Xis_3$"60 e mais" <- log(Xis_3$"60 e mais" + 0.01)
Xis_3$matriculas_creche_municipal <-
log(Xis_3$matriculas_creche_municipal + 0.01)
Xis_3$matriculas_ESAI_estadual <-
log(Xis_3$matriculas_ESAI_estadual + 0.01)
Xis_3$matriculas_ESAF_municipal <-
log(Xis_3$matriculas_ESAF_municipal + 0.01)
Xis_3$matriculas_EM_municipal <-
log(Xis_3$matriculas_EM_municipal + 0.01)
```

```
#transformar na para log(0.01)


Xis_3$Total_de_crimes1m[is.na(Xis_3$Total_de_crimes1m)] <-
0.01
Xis_3$Total_de_crimes2m[is.na(Xis_3$Total_de_crimes2m)] <-
0.01
Xis_3$População[is.na(Xis_3$População)] <- 0.01
Xis_3$"60 e mais"[is.na(Xis_3$"60 e mais")] <- 0.01
Xis_3$matriculas_creche_municipal[is.na(Xis_3$matriculas_creche_muni
cipal)] <-0.01
Xis_3$matriculas_ESAI_estadual[is.na(Xis_3$matriculas_ESAI_estadual
)] <- 0.01
Xis_3$matriculas_ESAF_municipal[is.na(Xis_3$matriculas_ESAF_municipa
l)] <- 0.01
Xis_3$matriculas_EM_municipal[is.na(Xis_3$matriculas_EM_municipal)]
<- 0.01
nrow(Xis_3)


new1 <- c(rnorm(n =444 , mean = 1, sd = 0))

Xis_4 <- cbind(new1,Xis_3)
#Xis_4 <- Xis_3



I <- diag(444)

rho <- 0.5959

X <-data.matrix(Xis_4)
X
nrow(X)
ncol(X)
nrow(beta)
ncol(beta)
```

```
beta
E  = (I - (rho*W2))
nrow(E)
ncol(E)
ES <- solve(E)
#View(ES)
ESD  <- X %*% beta
#View(ESD)
nrow(ES)
ncol(ES)
nrow(ESD)
ncol(ESD)
e  <- ES %*% ESD
#View(e)
y <- Xis_5
y_chapeu <- e


cbind(y,y_chapeu)
combined_data <- cbind(
  agregado_tudo_novo_2$crime_per_capita,
  exp(y_chapeu),
  test_data_sf$predicted
)
colnames(combined_data) <- c("crime_per_capita", "SAR_model", "ANN")
View(combined_data)

install.packages("patchwork")
library(ggplot2)
library(patchwork)


# Create the first plot for "predicted"
plot_predicted <- ggplot(test_data_sf) +
  geom_sf(aes(fill = test_data_sf$predicted, geometry =
  geometry)) +
  scale_fill_gradient(low="white", high="blue",
  name="predicted") +
  labs(title="Choropleth plot of ANN about crime per capita
```

```
\n on the metropolitan area of São Paulo")

# Create the second plot for "exp(y_chapeu)"
plot_exp_y_chapeu <- ggplot(test_data_sf) +
  geom_sf(aes(fill = exp(y_chapeu), geometry = geometry)) +
  scale_fill_gradient(low="white", high="blue",
  name="predicted") +
  labs(title="Choropleth plot of SAR model about crime per
capita \n on the metropolitan area of São Paulo")

# Set the figure size and resolution
options(repr.plot.width=40, repr.plot.height=5, repr.plot.res=80)

final_plot <- plot_predicted + plot_exp_y_chapeu +
  plot_layout(ncol = 2)

final_plot



esti <- y_chapeu - y
esti

y <- unlist(y)
y <- as.numeric(y)
y_chapeu <- as.numeric(y_chapeu)
y_chapeu
res <- exp(y_chapeu)
res
resy <- exp(y)
resy

# Calculating RMSE using rmse()
result = rmse(resy, res)

# Printing the value
print(result)

#erro absoluto médio.
mae = mae(resy, res)
```

6.4 Artificial Neural Network

This part of the code oin R serves for the Artificial Neural Networks and Long Short Term Networks

```
library(reticulate)
#Change the path to your Python installation directory
py_install("pydot")

install.packages("keras")



library(ggspatial)
library(nnet)
library(jpeg)
library(sf)
library(tidyverse)  # Modern data science workflow
library(spdep)
library(spatialreg)
#library(rgdal)
library(dplyr)
library(stats)
library(glmnet)
#library(matlib)
library(MASS)
library(stringr)
library(graphics)
library(zoo)
library(Metrics)
library(qtl2)
library(janitor)
library(neuralnet)
library(NeuralNetTools)
library(DiagrammeR)
library(keras)
#library(graphviz)
```

```
#library(pydot)
library(tensorflow)
#log_dir <- "logs"  # Specify a directory to store TensorBoard logs
#tensorboard_callback <- callback_tensorboard(log_dir
= log_dir,
#histogram_freq = 1, write_graph = TRUE, write_images = TRUE)

options(prompt="R> ", digits=4, scipen=7)



atn1 <- read_csv("C:\Users\yansh\OneDrive\Documentos\pesquisa 2\atn1.csv")
atn2 <- read_csv("C:\Users\yansh\OneDrive\Documentos\pesquisa 2\atn2.csv")
#agregado_tudo_novo_8 <- read_csv("Desktop/pesquisa 2/agregado_tudo_novo_8.csv")
 #agregado_tudo_novo_9csv <- read_csv("Desktop/pesquisa 2/agregado_tudo_novo_9csv.csv")

agregado_tudo_novo_11 <- atn1
agregado_tudo_novo_2 <- atn2

agregado_tudo_novo_11$População <-
str_replace(agregado_tudo_novo_11$População, '\\.', '')

agregado_tudo_novo_11$População <-
sapply(agregado_tudo_novo_11$População, as.numeric)

target <- c('São Paulo', 'Guarulhos','Caieiras', 'Cajamar',
            'Francisco Morato', 'Franco da Rocha',
            'Mairiporã','Arujá', 'Ferraz de Vasconcelos'
            ,'Guararema',
            'Itaquaquecetuba', 'Mogi das
            Cruzes', 'Poá', 'Salesópolis',
            'Santa Isabel', 'Suzano','Diadema', 'Mauá',
            'Santo André', 'São Bernardo do Campo'
            ,'São Caetano do Sul', 'Ribeirão Pires', 'Rio
            Grande da Serra','Cotia'
            ,'Embu','Biritiba-Mirim', 'Embu-Guaçu', 'Itapecerica da Serra',
            'Juquitiba',
            'São Lourenço da Serra', 'Taboão da Serra',
            'Vargem Grande Paulista'
            ,'Barueri', 'Carapicuíba', 'Itapevi',
```

```
            'Jandira',
            'Osasco'
            ,'Pirapora do Bom Jesus', 'Santana de Parnaíba')
#filtrar as cidades que estao proximo de sp
agregado_tudo_novo_11 <- agregado_tudo_novo_11 %>%
  filter(name %in% target)
nrow(agregado_tudo_novo_11)
ncol(agregado_tudo_novo_11)
agregado_tudo_novo_11$População[which(agregado_tudo_novo_11$name
== "Guarulhos")] <- 1404694
agregado_tudo_novo_11$População[which(agregado_tudo_novo_11$name
== "São Paulo")] <- 12396372
agregado_tudo_novo_11$População[which(agregado_tudo_novo_11$name
                                      == "Embu")] <- 276535
agregado_tudo_novo_11$População[which(agregado_tudo_novo_11$name
                                      == "Biritiba-Mirim")] <- 32936
#agregado_tudo$Total_de_crimes12m[agregado_tudo$Total_de_crimes12m == 0] <- NA
#agregado_tudo_1  <- na.locf(na.locf(agregado_tudo))
#agregado_tudo_1 <- agregado_tudo_1[agregado_tudo_1$name != "Arujá", ]
nrow(agregado_tudo_novo_11)


vc <- "https://raw.githubusercontent.com/tbrugz/geodata-
br/master/geojson/geojs-35-mun.json"
dir.create("C:/temp")
downloader::download(url = vc, destfile = "C:/temp/gas.GeoJSON")
municipio <- st_read("C:/temp/gas.GeoJSON")
summary(municipio)

municipio_1 <- subset(municipio, name %in%
agregado_tudo_novo_11$name)
summary(municipio_1)
head(municipio_1)
nrow(municipio_1)


municipio_1_sf <- st_as_sf(municipio_1)

municipio_1_sf$area_km2 <- st_area(municipio_1_sf) / 10^6
```

```
agregado_tudo_novo_11 <- agregado_tudo_novo_11 %>%
  left_join(municipio_1_sf %>% st_drop_geometry(), by = c("name" = "name")) %>%
  mutate(pop_density = População / area_km2)


head(agregado_tudo_novo_11$pop_density)
#View(agregado_tudo_novo_1)

agregado_tudo_novo_11$pop_density <- gsub("[^0-9.]", "",
agregado_tudo_novo_11$pop_density)
agregado_tudo_novo_11$pop_density <-
as.numeric(agregado_tudo_novo_11$pop_density)
head(agregado_tudo_novo_11$pop_density)


agregado_tudo_novo_11$Total_de_crimes <-
as.numeric(agregado_tudo_novo_11$Total_de_crimes)
agregado_tudo_novo_11$População <-
as.numeric(agregado_tudo_novo_11$População)

agregado_tudo_novo_11$crime_per_capita <-
(agregado_tudo_novo_11$Total_de_crimes /
agregado_tudo_novo_11$População) * 100000
agregado_tudo_novo_11$crime_per_capita




data(meuse)
coordinates(meuse) <- c("x", "y")
proj4string(meuse) <- CRS("+init=epsg:28992")
gridded(meuse) <- TRUE
meuse_polygons <- as(meuse, "SpatialPolygonsDataFrame")

municipio_1_sf <- st_as_sf(municipio_1, coords = c("long",
"lat"), crs = 4326)

column
```

```
test_data_norm_df_3 <- as.data.frame(agregado_tudo_novo_11)

test_data_sf_2 <- left_join(test_data_norm_df_3, municipio_1_sf,
by = c("name" = "name")) %>%
  dplyr::select(name, crime_per_capita, pop_density, geometry)

test_data_sf_2$crime_per_capita <-
as.numeric(test_data_sf_2$crime_per_capita)
test_data_sf_2$crime_per_capita

library(ggplot2)



p <- ggplot(test_data_sf_2) +
  geom_sf(aes(fill = crime_per_capita, geometry = geometry)) +
  scale_fill_gradient(low="white", high="blue",
                      name="crime_per_capita") +
  options(repr.plot.width=10, repr.plot.height=5, repr.plot.res=80)

p


test_data_sf_2 <- st_as_sf(test_data_sf_2, coords = c("lon", "lat"), crs = 4326)

test_data_sf_2 <- st_as_sf(test_data_sf_2, sf_column_name = "geometry")


pdois <- ggplot(data = test_data_sf_2) +
  geom_sf(aes(fill = crime_per_capita)) +
  data with fill based on 'crime_per_capita'
  scale_fill_gradient(low = "white", high = "blue",
  name = "Crime per Capita") +
  geom_sf_text(aes(label = name), size = 2.2,
  check_overlap = FALSE, position = position_nudge(y =
  0.005)) +
  theme_minimal() +
  labs(x = "Longitude", y = "Latitude")
```

```
print(pdois)
```

```
############################################################
agregado_tudo_3 <- agregado_tudo_novo_11[,
c('crime_per_capita','População','pop_density','60 e
mais','matriculas_creche_municipal'
,'matriculas_ESAI_estadual','matriculas_ESAF_municipal'
,'matriculas_EM_municipal')]

#agregado_tudo_3 <- log(agregado_tudo_3 + 0.01)_
names(agregado_tudo_3)[names(agregado_tudo_3) == "60 e
mais"] <-
"sessenta_e_mais"
nrow(agregado_tudo_3)

#View(agregado_tudo_3)

agregado_tudo_3$crime_per_capita[is.na(agregado_tudo_3$
crime_per_capita)] <- 0

agregado_tudo_3$pop_density[is.na(agregado_tudo_3$pop_d
ensity)] <- 0
agregado_tudo_3$População[is.na(agregado_tudo_3$Populaç
ão)] <- 0
agregado_tudo_3$sessenta_e_mais[is.na(agregado_tudo_3$s
essenta_e_mais)] <- 0
```

```
agregado_tudo_3$matriculas_creche_municipal[is.na(agreg
ado_tudo_3$matriculas_creche_municipal)] <-0
agregado_tudo_3$matriculas_ESAI_estadual[is.na(agregado
_tudo_3$matriculas_ESAI_estadual)] <- 0
agregado_tudo_3$matriculas_ESAF_municipal[is.na(agregad
o_tudo_3$matriculas_ESAF_municipal)] <- 0
agregado_tudo_3$matriculas_EM_municipal[is.na(agregado_
tudo_3$matriculas_EM_municipal)] <- 0
#View(agregado_tudo_3)
nrow(agregado_tudo_3)
ncol(agregado_tudo_3)

agregado_tudo_novo_3$crime_per_capita <-
as.numeric(agregado_tudo_novo_3$crime_per_capita)
agregado_tudo_3$pop_density <-
as.numeric(agregado_tudo_3$pop_density)
agregado_tudo_3$População <-
as.numeric(agregado_tudo_3$População)
agregado_tudo_3$sessenta_e_mais <-
as.numeric(agregado_tudo_3$sessenta_e_mais)
agregado_tudo_3$matriculas_creche_municipal <-
as.numeric(agregado_tudo_3$matriculas_creche_municipal)
agregado_tudo_3$matriculas_ESAI_estadual <-
as.numeric(agregado_tudo_3$matriculas_ESAI_estadual)
agregado_tudo_3$matriculas_ESAF_municipal <-
as.numeric(agregado_tudo_3$matriculas_ESAF_municipal)
agregado_tudo_3$matriculas_EM_municipal <-
as.numeric(agregado_tudo_3$matriculas_EM_municipal)
#View(agregado_tudo_3)

# Criando uma matriz de entrada com as variáveis
independentes
x <- as.matrix(agregado_tudo_3)

# Criando uma matriz de saída com a variável dependente
y <- as.matrix(agregado_tudo_3$crime_per_capita)

# Normalizando os dados
x_norm <- apply(x, 2, function(x) (x - min(x)) /
```

```
(max(x) -
min(x)))
y_norm <- apply(y, 2, function(y) (y - min(y)) /
(max(y) -
min(y)))
View(x_norm)
# Dividindo os dados em conjunto de treino e teste
set.seed(123)
```

```
agregado_tudo_novo_2$População <-
str_replace(agregado_tudo_novo_2$População, '\\.', '')

agregado_tudo_novo_2$População <-
sapply(agregado_tudo_novo_2$População, as.numeric)

target <- c('São Paulo', 'Guarulhos','Caieiras', 'Cajamar',
            'Francisco Morato', 'Franco da Rocha',
            'Mairiporã','Arujá', 'Ferraz de Vasconcelos'
            ,'Guararema',
            'Itaquaquecetuba', 'Mogi das
            Cruzes', 'Poá', 'Salesópolis',
            'Santa Isabel', 'Suzano','Diadema', 'Mauá',
            'Santo André', 'São Bernardo do Campo'
            ,'São Caetano do Sul', 'Ribeirão Pires', 'Rio
            Grande da Serra','Cotia'
            ,'Embu','Biritiba-Mirim', 'Embu-Guaçu', 'Itapecerica da Serra',
            'Juquitiba',
            'São Lourenço da Serra', 'Taboão da Serra',
            'Vargem Grande Paulista'
            ,'Barueri', 'Carapicuíba', 'Itapevi',
            'Jandira',
            'Osasco'
```

```
            ,'Pirapora do Bom Jesus', 'Santana de Parnaíba')

#filtrar as cidades que estao proximo de sp
agregado_tudo_novo_2 <- agregado_tudo_novo_2 %>%
  filter(name %in% target)
nrow(agregado_tudo_novo_2)
ncol(agregado_tudo_novo_2)
agregado_tudo_novo_2$População[which(agregado_tudo_novo_2$name
== "Guarulhos")] <- 1404694
agregado_tudo_novo_2$População[which(agregado_tudo_novo_2$name
== "São Paulo")] <- 12396372
#agregado_tudo$Total_de_crimes12m[agregado_tudo$Total_de_crimes12m == 0] <- NA
#agregado_tudo_1  <- na.locf(na.locf(agregado_tudo))
#agregado_tudo_1 <- agregado_tudo_1[agregado_tudo_1$name != "Arujá", ]
nrow(agregado_tudo_novo_2)


vc <- "https://raw.githubusercontent.com/tbrugz/geodata-
br/master/geojson/geojs-35-mun.json"
dir.create("C:/temp")
downloader::download(url = vc, destfile = "C:/temp/gas.GeoJSON")
municipio <- st_read("C:/temp/gas.GeoJSON")
summary(municipio)

municipio_2 <- subset(municipio, name %in%
agregado_tudo_novo_2$name)
summary(municipio_2)
head(municipio_2)
nrow(municipio_2)


municipio_2_sf <- st_as_sf(municipio_2)

municipio_2_sf$area_km2 <- st_area(municipio_2_sf) / 10^6

agregado_tudo_novo_2 <- agregado_tudo_novo_2 %>%
  left_join(municipio_2_sf %>% st_drop_geometry(), by = c("name"
  = "name")) %>%
  mutate(pop_density = População / area_km2)
```

```
agregado_tudo_novo_2
nrow(agregado_tudo_novo_2)

agregado_tudo_novo_2$Total_de_crimes <-
as.numeric(agregado_tudo_novo_2$Total_de_crimes)
agregado_tudo_novo_2$População <-
as.numeric(agregado_tudo_novo_2$População)

agregado_tudo_novo_2$crime_per_capita <-
(agregado_tudo_novo_2$Total_de_crimes /
agregado_tudo_novo_2$População) * 100000




head(agregado_tudo_novo_2$pop_density)
#View(agregado_tudo_novo_1)


head(agregado_tudo_novo_2$pop_density)
#View(agregado_tudo_novo_1)
head(agregado_tudo_novo_2)


agregado_tudo_novo_10 <- agregado_tudo_novo_2[,
c('crime_per_capita', 'População','pop_density','60 e mais',
'matriculas_creche_municipal','matriculas_ESAI_estadual',
'matriculas_ESAF_municipal','matriculas_EM_municipal')]

print(agregado_tudo_novo_10)
#agregado_tudo_3 <- log(agregado_tudo_3 + 0.01)_
names(agregado_tudo_novo_10)
[names(agregado_tudo_novo_10) == "60 e mais"] <-
"sessenta_e_mais"
nrow(agregado_tudo_novo_10)

crimeperca <- agregado_tudo_novo_10$crime_per_capita
agregado_tudo_novo_10$pop_density <- gsub("[^0-9.]", "",
```

```
agregado_tudo_novo_10$pop_density)
agregado_tudo_novo_10$pop_density <-
as.numeric(agregado_tudo_novo_10$pop_density)
head(agregado_tudo_novo_10$pop_density)

agregado_tudo_novo_10$crime_per_capita[is.na(agregado_t
udo_novo_10$crime_per_capita)] <- 0
agregado_tudo_novo_10$pop_density[is.na(agregado_tudo_n
ovo_10$pop_density)] <- 0
agregado_tudo_novo_10$População[is.na(agregado_tudo_nov
o_10$População)] <- 0
agregado_tudo_novo_10$sessenta_e_mais[is.na(agregado_tu
do_novo_10$sessenta_e_mais)] <- 0
agregado_tudo_novo_10$matriculas_creche_municipal[is.na
(agregado_tudo_novo_10$matriculas_creche_municipal)]
<-0
agregado_tudo_novo_10$matriculas_ESAI_estadual[is.na(ag
regado_tudo_novo_10$matriculas_ESAI_estadual)] <- 0
agregado_tudo_novo_10$matriculas_ESAF_municipal[is.na(a
gregado_tudo_novo_10$matriculas_ESAF_municipal)] <- 0
agregado_tudo_novo_10$matriculas_EM_municipal[is.na(agr
egado_tudo_novo_10$matriculas_EM_municipal)] <- 0
#View(agregado_tudo_novo_10)
nrow(agregado_tudo_novo_10)
ncol(agregado_tudo_novo_10)

agregado_tudo_novo_10$crime_per_capita <-
as.numeric(agregado_tudo_novo_10$crime_per_capita)
agregado_tudo_novo_10$pop_density <-
as.numeric(agregado_tudo_novo_10$pop_density)
agregado_tudo_novo_10$População <-
as.numeric(agregado_tudo_novo_10$População)
agregado_tudo_novo_10$sessenta_e_mais <-
as.numeric(agregado_tudo_novo_10$sessenta_e_mais)
agregado_tudo_novo_10$matriculas_creche_municipal <-
as.numeric(agregado_tudo_novo_10$matriculas_creche_municipal)
agregado_tudo_novo_10$matriculas_ESAI_estadual <-
as.numeric(agregado_tudo_novo_10$matriculas_ESAI_estadual)
agregado_tudo_novo_10$matriculas_ESAF_municipal <-
```

```
as.numeric(agregado_tudo_novo_10$matriculas_ESAF_municipal)
agregado_tudo_novo_10$matriculas_EM_municipal <-
as.numeric(agregado_tudo_novo_10$matriculas_EM_municipal)

data(meuse)
coordinates(meuse) <- c("x", "y")
proj4string(meuse) <- CRS("+init=epsg:28992")
gridded(meuse) <- TRUE
meuse_polygons <- as(meuse, "SpatialPolygonsDataFrame")

municipio_1_sf <- st_as_sf(municipio_2, coords = c("long",
"lat"), crs = 4326)

test_data_norm_df_3 <- as.data.frame(agregado_tudo_novo_2)

test_data_sf_2 <- left_join(test_data_norm_df_3, municipio_1_sf,
by = c("name" = "name")) %>%
  dplyr::select(name, crime_per_capita, pop_density, geometry)

test_data_sf_2$crime_per_capita <-
as.numeric(test_data_sf_2$crime_per_capita)
test_data_sf_2$crime_per_capita

p <- ggplot(test_data_sf_2) +
              geom_sf(aes(fill = crime_per_capita, geometry =
              geometry)) +
              scale_fill_gradient(low="white", high="blue",
              name="crime_per_capita") +
              labs(title="Choropleth plot of test data about
crime per capita \n on the metropolitan area of
São Paulo using ANN")

options(repr.plot.width=10, repr.plot.height=5, repr.plot.res=80)

p

ggsave("figura_pedidos_por_distrito.png", plot=p, width=10, height=5, dpi=80)
```

```
vc <- "https://raw.githubusercontent.com/tbrugz/geodata-
  br/master/geojson/geojs-35-mun.json"
downloader::download(url = vc, destfile = "/tmp/gas.GeoJSON")
municipio <- readOGR(dsn = "/tmp/gas.GeoJSON")
summary(municipio)

municipio_1 <- subset(municipio, name %in% agregado_tudo_novo_2$name)
summary(municipio_1)
head(municipio_1)
nrow(municipio_1)


nmunicipio_1 <- municipio_1[rep(seq_len(nrow(municipio_1)), each
= 192), ]  # Base R

summary(nmunicipio_1)
head(nmunicipio_1)
nrow(nmunicipio_1)

queens_1 <- poly2nb(municipio_1, queen = TRUE)
queens_1[[1]]
str(queens_1)

alistw_1 <- nb2listw(queens_1,style = "W" ,zero.policy = TRUE)

#W <- nb2mat(queens_1, glist=NULL, style="B", zero.policy=TRUE)



nqueens_1 <- poly2nb(nmunicipio_1, queen = TRUE)
nqueens_1[[1]]
nqueens_1[[2]]
str(nqueens_1)

a2listw_1<- nb2listw(nqueens_1,style = "W" ,zero.policy = TRUE)
a2listw_1
```

```
names(a2listw_1)

W <- nb2mat(nqueens_1, glist=NULL, style="W", zero.policy=TRUE)
W
dim(W)





nmunicipio_2 <- municipio_1[rep(seq_len(nrow(municipio_1)), each
                                = 12), ]  # Base R

summary(nmunicipio_2)
head(nmunicipio_2)
nrow(nmunicipio_2)

queens_2 <- poly2nb(municipio_1, queen = TRUE)
queens_2[[1]]
str(queens_2)

alistw_2 <- nb2listw(queens_2,style = "W" ,zero.policy = TRUE)

#W <- nb2mat(queens_1, glist=NULL, style="B", zero.policy=TRUE)



nqueens_2 <- poly2nb(nmunicipio_2, queen = TRUE)
nqueens_2[[1]]
nqueens_2[[2]]
str(nqueens_2)

a2listw_2<- nb2listw(nqueens_2,style = "W" ,zero.policy = TRUE)
a2listw_2
names(a2listw_2)

W2 <- nb2mat(nqueens_2, glist=NULL, style="W", zero.policy=TRUE)
W2
dim(W2)
```

```
train_data <- as.matrix(agregado_tudo_3)
test_data <- as.matrix(agregado_tudo_novo_10)


train_target <- train_data[, "crime_per_capita"]
test_target <- test_data[, "crime_per_capita"]

train_data_norm <- apply(train_data, 2, function(x) (x - min(x))
/ (max(x) - min(x)))
test_data_norm <- apply(test_data, 2, function(x) (x - min(x)) /
(max(x) - min(x)))
train_target_norm <- (train_target - min(train_target)) /
(max(train_target) - min(train_target))
test_target_norm <- (test_target - min(test_target)) /
(max(test_target) - min(test_target))



#install.packages("keras")
#library(keras)
#install_keras(tensorflow = "2.9.0")
library(reticulate)
use_python("C:/Users/yansh/anaconda3/python.exe", required = TRUE)



create_model_with_W <- function(input_shape, W_shape) {
  input_data <- layer_input(shape = input_shape, name = "data_input")
  input_W <- layer_input(shape = W_shape, name = "W_input")
```

```
  merged_layer <- layer_concatenate(list(input_data, input_W))

  model <- keras_model_sequential() %>%
    layer_dense(units = 32, activation = "relu") %>%
    layer_dropout(rate = 0.2) %>%
    layer_dense(units = 16, activation = "relu") %>%
    layer_dropout(rate = 0.2) %>%
    layer_dense(units = 1)

  model <- keras_model(inputs = list(input_data,
  input_W), outputs = model(merged_layer))

  return(model)
}

input_shape <- dim(train_data_norm)[2]
input_shape
W_shape <- dim(W)[2]
W_shape

model <- create_model_with_W(input_shape, W_shape)


model %>% compile(
  loss = "mean_squared_error",
  optimizer = optimizer_adam(lr = 0.001)
)

history <- model %>% fit(
  list(train_data_norm, W), train_target_norm,
  epochs = 50,
  batch_size = 32,
  validation_split = 0.2,
  #callbacks = list(tensorboard_callback)
)

n_samples_test <- dim(test_data_norm)[1]
test_data_reshaped <- array(test_data_norm, dim =
c(n_samples_test, dim(test_data_norm)[2]))
```

```
W_test_reshaped <- array(W, dim = c(n_samples_test, dim(W)[2]))

predicted_norm <- predict(model, list(test_data_reshaped, W_test_reshaped))



predicted <- predicted_norm * (max(train_target) -
min(train_target)) + min(train_target)



predicted <- predicted_norm * (max(train_target) -
min(train_target)) + min(train_target)





predic <- predicted_norm * (max(train_target) -
min(train_target)) + min(train_target)
predic


mae <- mean(abs(predic - test_target))
rmse <- sqrt(mean((predic - test_target)^2))


cat("MAE:", mae, "\n")
cat("RMSE:", rmse, "\n")





library(keras)
```

```
create_lstm_model_with_W <- function(input_shape, W_shape, lstm_units = 32) {

  input_data <- layer_input(shape = input_shape, name = "data_input")
  input_W <- layer_input(shape = W_shape, name = "W_input")

  merged_layer <- layer_concatenate(list(input_data, input_W))

  lstm_layer <- layer_lstm(units = lstm_units, return_sequences = FALSE)(merged_layer)

  output_layer <- layer_dense(units = 1, activation = "linear")(lstm_layer)

  model_lstm <- keras_model(inputs = list(input_data, input_W), outputs = output_layer)

  return(model_lstm)
}



time_steps <- 1
input_shape <- list(time_steps, dim(train_data_norm)[2])
W_shape <- list(time_steps, dim(W)[2])



train_data_reshaped <- array(train_data_norm, dim =
c(dim(train_data_norm)[1], time_steps, dim(train_data_norm)[2]))
W_reshaped <- array(W, dim = c(dim(W)[1], time_steps, dim(W)[2]))




lstm_model <- create_lstm_model_with_W(input_shape, W_shape,
lstm_units = 32)


lstm_model %>% compile(
  loss = "mean_squared_error",
  optimizer_adam(lr = 0.001, clipnorm = 1)
)
```

```
history <- lstm_model %>% fit(
  list(train_data_reshaped, W_reshaped), train_target_norm,
  epochs = 50,
  batch_size = 32,
  validation_split = 0.2
)


n_samples_test <- dim(test_data_norm)[1]
test_data_reshaped <- array(test_data_norm, dim =
c(n_samples_test, time_steps, dim(test_data_norm)[2]))
W_test_reshaped <- array(W, dim = c(n_samples_test, time_steps,
dim(W)[2]))


predicted_norm_lstm <- predict(lstm_model,
list(test_data_reshaped, W_test_reshaped))


predicted_lstm <- predicted_norm_lstm * (max(train_target) -
min(train_target)) + min(train_target)
predicted_lstm

mae_lstm <- mean(abs(predicted_lstm - test_target))
rmse_lstm <- sqrt(mean((predicted_lstm - test_target)^2))


cat("LSTM Model Metrics:\n")
cat("MAE:", mae_lstm, "\n")
cat("RMSE:", rmse_lstm, "\n")


combinada <- cbind(
```

```
  test_data_sf_2$crime_per_capita,
  predic,
  predicted_lstm,
  #exp(y_chapeu)
)
colnames(combinada) <- c("crime_per_capita", "ANN", "LSTM")
combinada <- as.data.frame(combinada)
View(combinada)



library(ggplot2)
library(patchwork)





plot_predicted <- ggplot(test_data_sf_2) +
  geom_sf(aes(fill = crime_per_capita, geometry = geometry)) +
  scale_fill_gradient(low = "white", high = "blue", name =
  "crime per capita", na.value = "white") +
  labs(title = "Observation of crime per capita\n
  on the metropolitan area of São Paulo in 2017")




plot_exp_y_chapeu <- ggplot(test_data_sf_2) +
  geom_sf(aes(fill = exp(y_chapeu), geometry = geometry)) +  # Specify 'geometry'
  scale_fill_gradient(low = "white", high = "blue", name =
  "predicted") +
  labs(title = "Choropleth plot of SAR model \n
  on the metropolitan area of São Paulo")




plot_legal <- ggplot(test_data_sf_2) +
  geom_sf(aes(fill = predic, geometry = geometry)) +  # Specify 'geometry'
  scale_fill_gradient(low = "white", high = "blue", name =
  "predicted") +
```

```
  labs(title = "Choropleth plot of ANN model \n
  on the metropolitan area of São Paulo")


plot_lstm <- ggplot(test_data_sf_2) +
  geom_sf(aes(fill = predicted_lstm, geometry =
  geometry)) +  #   Specify 'geometry'
  scale_fill_gradient(low = "white", high = "blue", name =
  "predicted") +
  labs(title = "Choropleth plot of LSTM model \n
  on the metropolitan area of São Paulo")



options(repr.plot.width = 60, repr.plot.height = 5, repr.plot.res = 80)


final_plot <- plot_predicted + plot_exp_y_chapeu + plot_legal +
plot_lstm
  plot_layout(nrow = 2, ncol = 2)


final_plot
```

BIBLIOGRAPHY

BEZERRA, H. M. **Spatial Regression Models applied to Orders Demand Forecasting of Online Food Delivery**. github, 2021. Available: <https://github.com/HerlissonMB/demand-forecast>. Citation on page 18.

BISHOP, C. **Neural Networks for Pattern Recognition**. Clarendon Press, 1995. (Advanced Texts in Econometrics). ISBN 9780198538646. Available: <https://books.google.com.br/books?id=T0S0BgAAQBAJ>. Citation on page 41.

BIVAND, R. **Spatialreg Package**. [S.l.], 2022. R package version 1.2-6. Available: <https://https://cran.r-project.org/web/packages/spatialreg/index.html>. Citations on pages 49 and 58.

____. **Spdep Package**. [S.l.], 2022. R package version 1.2-7. Available: <https://cran.r-project.org/web/packages/spdep/index.html>. Citation on page 35.

BIVAND, R.; HAUKE, J.; KOSSOWSKI, T. Computing the j acobian in g aussian spatial autoregressive models: An illustrated comparison of available methods. **Geographical Analysis**, Wiley Online Library, v. 45, n. 2, p. 150–179, 2013. Citation on page 49.

BIVAND, R.; MILLO, G.; PIRAS, G. A review of software for spatial econometrics in r. **Mathematics**, Multidisciplinary Digital Publishing Institute, v. 9, n. 11, p. 1276, 2021. Citation on page 35.

BRAGA, A. de P.; CARVALHO, A. de L. F.; LUDERMIR, T. **Redes neurais artificiais: teoria e aplicações**. Livros Técnicos e Científicos, 2000. ISBN 9788521612186. Available: <https://books.google.com.br/books?id=cUgEaAEACAAJ>. Citation on page 41.

BROWN, M. A. Modelling the spatial distribution of suburban crime. **Economic geography**, Taylor & Francis, v. 58, n. 3, p. 247–261, 1982. Citation on page 18.

BRUGNARA, T. **Geodata-BR Package**. [S.l.], 2021. Python package. Available: <https://raw.githubusercontent.com/tbrugz/geodata-br/master/geojson/geojs-35-mun.json>. Citations on pages 23 and 34.

CAPLAN, J. M.; KENNEDY, L. W.; MILLER, J. Risk terrain modeling: Brokering criminological theory and gis methods for crime forecasting. **Justice quarterly**, Taylor & Francis, v. 28, n. 2, p. 360–381, 2011. Citation on page 17.

CHUN, S. A.; PATURU, V. A.; YUAN, S.; PATHAK, R.; ATLURI, V.; ADAM, N. R. Crime prediction model using deep neural networks. p. 512–514, 2019. Citation on page 18.

FORRADELLAS, R. F. R.; ALONSO, S. L. N.; JORGE-VAZQUEZ, J.; RODRIGUEZ, M. L. Applied machine learning in social sciences: neural networks and crime prediction. **Social Sciences**, MDPI, v. 10, n. 1, p. 4, 2020. Citation on page 18.

FREUND, J.; WALPOLE, R. **Mathematical Statistics**. Prentice-Hall, 1980. (Prentice-Hall mathematics series). ISBN 9780135620663. Available: <https://books.google.com.br/books?id=mjZzwQEACAAJ>. Citation on page 36.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. MIT Press, 2016. (Adaptive Computation and Machine Learning series). ISBN 9780262035613. Available: <https://books.google.com.br/books?id=-s2MEAAAQBAJ>. Citations on pages 11, 41, 43, 44, and 46.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. H.; FRIEDMAN, J. H. **The elements of statistical learning: data mining, inference, and prediction**. [S.l.]: Springer, 2009. Citations on pages 11, 32, and 33.

HAYKIN, S.; HAYKIN, S. **Neural Networks: A Comprehensive Foundation**. Prentice Hall, 1999. (International edition). ISBN 9780132733502. Available: <https://books.google.com.br/books?id=bX4pAQAAMAAJ>. Citations on pages 11, 37, 38, 40, and 41.

KAZAR, B. M.; CELIK, M. **Spatial autoregression (SAR) model: Parameter estimation techniques**. [S.l.]: Springer Science & Business Media, 2012. Citations on pages 11, 33, 34, and 35.

MARX, B. D.; EILERS, P. H. Generalized linear regression on sampled signals and curves: a p-spline approach. **Technometrics**, Taylor & Francis, v. 41, n. 1, p. 1–13, 1999. Citation on page 66.

MELO, S. N. de; MATIAS, L. F.; ANDRESEN, M. A. Crime concentrations and similarities in spatial crime patterns in a brazilian context. **Applied Geography**, Elsevier, v. 62, p. 314–324, 2015. Citations on pages 17 and 19.

NAKIB, M.; KHAN, R. T.; HASAN, M. S.; UDDIN, J. Crime scene prediction by detecting threatening objects using convolutional neural network. In: IEEE. **2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2)**. [S.l.], 2018. p. 1–4. Citation on page 18.

SHERMAN, L. W.; GARTIN, P. R.; BUERGER, M. E. Hot spots of predatory crime: Routine activities and the criminology of place. **Criminology**, Wiley Online Library, v. 27, n. 1, p. 27–56, 1989. Citation on page 17.

(SSP), S. de Segurança Publica de S. P. **Dados estatísticos da Secretaria de Segurança Pública do Estado de São Paulo.** Base de Dados, 2021. Available: <https://basedosdados.org/dataset/0d1b304b-fc9e-4533-b690-868da46d47a7>. Citation on page 21.

UNODC.
Brazilian Crime. 2021. Available: <https://dataunodc.un.org/#state:0>. Citation on page 17.

WHEELER, A. P.; STEENBEEK, W. Mapping the risk terrain for crime using machine learning. **Journal of Quantitative Criminology**, Springer, v. 37, n. 2, p. 445–480, 2021. Citation on page 17.

YAN, X.; SU, X. **Linear Regression Analysis: Theory And Computing**. World Scientific Publishing Company, 2009. ISBN 9789814470087. Available: <https://books.google.se/books?id=afzFCgAAQBAJ>. Citation on page 31.

YI, F.; YU, Z.; ZHUANG, F.; GUO, B. Neural network based continuous conditional random field for fine-grained crime prediction. In: **IJCAI**. [S.l.: s.n.], 2019. p. 4157–4163. Citation on page 18.

ICMC USP
SÃO CARLOS