ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

SCHOOL OF INFORMATION SCIENCES & TECHNOLOGY

DEPARTMENT OF STATISTICS
POSTGRADUATE PROGRAM

Infectious disease prediction: Modeling dengue incidences in France during 2008-2020

Written by
Chatzopoulos Gerasimos

THESIS

Submitted to the Department of Statistics of Athens University of Economics and Business as part of the requirements for the degree of Postgraduate Studies in Applied Statistics

Athens, August 2023

ΑΦΙΕΡΩΣΗ

Το αφιερώνω στην μητέρα μου την Ουρανία.

ACKHNOWLEDGEMENTS

I would like to personally thank my Professor, Pedeli Xanthi, for granting me complete freedom to write my thesis and for being there to provide any help I may have needed.

ABSTRACT

The systematic use of surveillance systems has become a standard practice adopted by many countries. The main purpose of surveillance systems is the effective and timely detection of a disease outbreak so that control measures for the elimination of disease transmission can be rapidly taken. Over the last two decades, several statistical techniques have been developed to this direction aiming to provide accurate and real time infectious disease predictions. One of the most well-developed statistical frameworks suitable for infectious disease prediction is the "HHH4" model (see, e.g., Held 2005) a specific extension of a generalized linear model developed for infectious diseases. Recently, Bracher and Held (2020) extended the HHH4 model to include weighting schemes for past incidence. Another competitive approach based on kernel conditional density estimation (KCDE) and copulas has been suggested by Ray, (2017). The aim of this thesis is to compare the performance of the three approaches mentioned above on monthly counts of reported dengue cases in France from 1/2008 to 12/2020.

ΠΕΡΙΛΗΨΗ

Η συστηματική χρήση των συστημάτων επιδημιολογικής επιτήρησης έχει καταστεί συνήθης πρακτική που υιοθετείται από πολλές χώρες. Ο κύριος σκοπός των συστημάτων επιτήρησης είναι η αποτελεσματική και έγκαιρη ανίχνευση μιας εστίας ασθένειας, ώστε να μπορούν να ληφθούν ταχέως μέτρα ελέγχου για την εξάλειψη της μετάδοσης της νόσου. Κατά τις τελευταίες δύο δεκαετίες, έχουν αναπτυχθεί διάφορες στατιστικές τεχνικές προς την κατεύθυνση αυτή, με σκοπό την παροχή ακριβών και ρεαλιστικών προβλέψεων για λοιμώδεις ασθένειες. Μία από τις πιο ανεπτυγμένες στατιστικές μεθόδους κατάλληλη για την πρόβλεψη μολυσματικών ασθενειών είναι το μοντέλο "HHH4" (βλέπε, π.χ., Held 2005). Το προβλεπτικό μοντέλο αυτό είναι μια ειδική επέκταση ενός γενικευμένου γραμμικού μοντέλου που αναπτύχθηκε για τα λοιμώδη νοσήματα. Πρόσφατα, οι Bracher and Held (2020) επέκτειναν το μοντέλο HHH4 για να συμπεριλάβουν συστήματα στάθμισης για την προηγούμενη επίπτωση. Μια άλλη ανταγωνιστική προσέγγιση που βασίζεται στην εκτίμηση της πυκνότητας υπό συνθήκη (KCDE) και copulas έχει προταθεί από τους Ray et al. (2017). Σκοπός της παρούσας διατριβής είναι η σύγκριση των επιδόσεων των τριών προσεγγίσεων που αναφέρονται ανωτέρω στις μηνιαίες καταμετρήσεις των καταγραφόμενων κρουσμάτων δάγκειου πυρετού στη Γαλλία από 1/2008 έως 12/2020.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

STRUCTURE OF MY THESIS

This thesis aims to provide a comprehensive understanding of infectious diseases, with a particular focus on dengue fever, and highlight the importance of effective prevention strategies.

The first part of the thesis delves into the theoretical aspects of pandemics and infectious diseases. It explores the underlying causes, transmission dynamics, and global impact of these diseases. Special emphasis is placed on dengue fever, a prevalent and significant public health concern worldwide. The thesis examines the characteristics of dengue fever, its epidemiology, and the challenges associated with its prevention and control.

Moving forward, the thesis explores various predictive modeling techniques to assess their suitability for addressing the challenges posed by infectious diseases. It delves into statistical methodologies specifically designed for analyzing and predicting the spread of infectious diseases. The strengths and limitations of these methods are discussed in detail to identify the most suitable approaches for the given problem.

Next, the thesis introduces and presents three main models that have been selected for the analysis of dengue fever in France from 2008 to 2020. These models have been carefully chosen based on their applicability and potential to provide meaningful insights into the spread and impact of dengue fever.

The core focus of the thesis lies in the comparative analysis of the selected models. The performance, accuracy, and predictive capabilities of each model are thoroughly evaluated using real-world data on dengue fever in France. Various evaluation metrics and statistical measures are employed to assess the strengths and weaknesses of each model. The objective is to identify the model that demonstrates the highest level of accuracy, reliability, and utility in predicting and preventing dengue fever outbreaks.

Ultimately, this thesis aims to contribute to the field of infectious disease research by evaluating and comparing different statistical modeling approaches for dengue fever prediction.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

TABLE OF CONTENTS

1 INTRODUCTION 1

1.1 UNDERSTANDING PANDEMICS: OVERVIEW 1

1.1.1 Definition 1

1.1.2 Pandemic – Epidemic – Endemic 2

1.1.3 Pandemics And WHO 3

1.1.4 Pandemic Intervals 5

1.1.5 Pandemic Phases 7

1.2 WORST PANDEMICS IN HISTORY 9

1.2.1 The Black Death (1347-1351) 9

1.2.2 Spanish Flu (1918-1919) 9

1.2.3 HIV/AIDS Pandemic (1981-Present) 9

1.2.4 Asian Flu (1957-1958) 11

1.2.5 Hong Kong Flu (1968-1969) 11

1.2.6 Russian Flu (1889-1890) 11

1.2.7 Third Cholera Pandemic (1852-1860) 12

1.2.8 Sixth Cholera Pandemic (1899-1923) 12

1.2.9 Swine Flu (2009-2010) 12

1.2.10 COVID-19 Pandemic (2019-Present) 13

1.2.11 Overall 13

1.3 INFLUENZA PANDEMICS: TYPES, IMPACT & PUBLIC HEALTH SIGNIFICANCE 14

1.3.1 Influenzas 14

1.3.2 Seasonal Influenza 14

1.3.3 Avian Influenza 15

1.3.4 Impact On Public Health 15

1.3.5 Pandemic Influenzas 16

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

1.4 DENGUE FEVER 18
1.4.1 Dengue Fever 18
1.4.2 Clinical Features 20
1.4.3 Laboratory Criteria 21
1.4.4 Epidemiological Criteria 23
1.4.5 Case classification and severity grading 25
1.4.6 Severity 26
1.4.7 Dengue Fever In France 28
1.4.8 Dengue Fever 2008-2020 29
1.4.9 Why The Statistical Surveillance Of The Dengue Fever Is Important 30
2 UNDERSTANDING PREDICTIVE MODELLING 32
2.1 PREDICTIVE STATISTICAL MODELLING 32
2.1.1 Definition 32
2.1.2 Steps And Phases 32
2.2 EXPLORING TYPES OF PREDICTIVE MODELS 34
2.2.1 Linear Regression 34
2.2.2 Logistic Regression 38
2.2.3 Time Series Analysis 44
2.2.4 Decision Trees 56
2.2.5 Random Forests 61
2.2.6 Support Vector Machines (Svm) 66
2.2.7 Naive Bayes 71
2.2.8 Neural Networks 75
2.2.9 Gradient Boosting Models 78
2.3 PREDICTING INFECTIOUS DISEASES: CHALLENGES AND SUITABLE MODELS 85
2.3.1 Challenges 85

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.2 Why some of the above methods may not be suitable or optimal. ……87

2.3.3 More suitable and broader categories of models for the forecast of infectious diseases. 91

2.3.4 Compartment Models 95

2.3.5 Agent Based Model 106

2.3.6 Spatial and Spatiotemporal models 112

2.3.7 Bayesian Models 118

2.3.8 Generalized Linear Models (Glms) 124

2.3.9 Machine Learning Models 128

2.3.10 Time Series Count Models 132

2.4 HISTORICAL RESEARCH WITH THESE MODELS FOR DENGUE FEVER 134

2.4.1 Compartment Models 134

2.4.2 Spatial Models 135

2.4.3 Bayesian Models 136

2.4.4 Glms 137

2.4.5 Agent-Based Models 138

2.4.6 Unique Features and capabilities of these predictive models . 139

2.5 STATISTICAL DISTRIBUTIONS FOR SIMULATED INFECTIOUS DISEASE DATA 142

2.5.1 Poisson Distribution 142

2.5.2 Negative Binomial Distribution 144

2.5.3 Gaussian (Normal) Distribution 146

2.5.4 Beta-Binomial Distribution 148

2.5.5 Gamma Distribution 150

2.5.6 Weibull Distribution 152

2.5.7 Log-Normal Distribution 154

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.5.8 Binomial Distribution 156

2.5.9 Multinomial Distribution 158

2.5.10 Dirichlet Distribution 160

3 PREDICTIVE MODELS FOR THE DENGUE FEVER CASES IN FRANCE (2008-2020) 163

3.1 EXPLORE OUR DATA 163

3.2 KEY POINTS ABOUT OUR PREDICTIVE MODELS 164

3.3 HHH4 MODEL (ENDEMIC – EPIDEMIC MODELLING OF A REAL COUNT TIME SERIES) 166

3.3.1 Brief history, methodology & parametriazation (Held, Michael Hohle and Hofmann, 2005, (67)) 166

3.3.2 Model Specification and Fitting 171

3.3.3 Model Selection and Further Analysis 173

3.3.4 Results 174

3.3.5 Conclusion 194

3.4 MODEL 2 (KCDE IMPLEMENTATION) 195

3.4.1 Method description – parametrization 195

3.4.2 R code implementation of KCDE model 202

3.4.3 Results 204

3.4.4 Evaluation metrics 209

3.4.5 Conclusion 223

3.5 HHH4 with weighted schemes for past incidence 224

3.5.1 Methodology – parametrization 224

3.5.2 R code implementation for the hhh4 model with weighted schemes 231

3.5.3 Fit model 233

3.5.4 Obtain forecast fuction 234

3.5.5 Actual forecast 236

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.5.6 Results 238
3.5.7 Naïve model forecast with weighted schemes 247
3.5.8 Conclusion 251
3.6 TIME-SERIES MODEL (Arima – Sarima) 252
3.6.1 R code implementation 252
3.6.2 Results 254
4 MODELS SELECTION COMPARISON METHODS ANALYSIS 262
5 FINAL THOUGHTS 268
6 BIBLIOGRAPHY 270

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

1 INTRODUCTION

1.1 UNDERSTANDING PANDEMICS: OVERVIEW

1.1.1 Definition

A pandemic is an outbreak of a disease that occurs on a global scale, affecting a large number of people across multiple countries or continents (1). It is characterized by the widespread transmission of the infectious agent, resulting in significant illness, death, and social disruption. Pandemics typically involve a new or novel pathogen to which the population has limited immunity, leading to rapid and sustained person-to-person transmission.

In a pandemic, the disease spreads beyond its initial epicenter and affects various regions worldwide, causing a significant impact on public health, healthcare systems, economies, and society as a whole. Pandemics often require international coordination and response efforts to mitigate the spread of the disease, develop vaccines or treatments, and provide necessary healthcare and support to affected populations. Examples of pandemics include the 1918 influenza pandemic (Spanish flu) and the ongoing COVID-19 pandemic.

1.1.2 Pandemic – Epidemic – Endemic

In the realm of public health, several terms are used to describe the spread and prevalence of diseases. Among these, three commonly used terms are pandemic, epidemic, and endemic (2,3,4). Each represents a distinct level of disease occurrence and has unique implications for global, regional, and local health.

A **pandemic** is an event that captures global attention. It refers to the worldwide spread of a new or novel disease, affecting a large number of people across multiple countries or continents. Pandemics are characterized by the widespread transmission of the infectious agent, leading to significant illness, mortality, and profound social disruption. Historical examples include the ongoing COVID-19 pandemic and the devastating 1918 influenza pandemic.

On a more localized scale, an **epidemic** occurs when a disease surpasses its expected level within a specific community or region. It involves a sudden increase in the number of cases, often overwhelming local healthcare systems and resources. Epidemics can affect a significant proportion of the population within the affected area. Notable epidemics include the Ebola outbreak in West Africa, the Zika virus outbreak in South America, and the SARS epidemic.

In contrast, an **endemic** refers to a disease that is constantly present at a baseline level within a specific population or geographic area. Endemic diseases maintain a stable prevalence within their respective regions, occasionally causing periodic outbreaks but remaining within the expected range of infection rates. Examples of endemic diseases include malaria in tropical regions, dengue fever in certain countries, and Lyme disease in specific geographic areas.

Understanding the distinctions between these terms is crucial for assessing the scope, impact, and appropriate response to disease outbreaks. While a pandemic commands global attention and necessitates coordinated international efforts, epidemics demand swift and targeted regional interventions. Endemic diseases require ongoing management and control measures to minimize the impact on the affected populations.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

By differentiating between pandemics, epidemics, and endemic diseases, public health professionals can better evaluate and implement the appropriate strategies to prevent, detect, and mitigate the spread of infectious diseases, safeguarding the well-being of communities and populations worldwide.

1.1.3 Pandemics And WHO

The World Health Organization (WHO) plays a central role in addressing and managing pandemics (5,6,7). As the leading global health organization, the WHO is responsible for coordinating international efforts to prevent, detect, and respond to public health emergencies, including pandemics. Here's an overview of the WHO's relationship with pandemics.

Preparedness and Planning

The WHO works to enhance global preparedness for pandemics by developing guidelines, protocols, and strategies for member states. It conducts risk assessments, provides technical support, and promotes capacity-building initiatives to strengthen countries' abilities to prevent, detect, and respond to pandemics.

Surveillance and Early Warning Systems

The WHO maintains a global surveillance network to monitor the emergence and spread of diseases. This includes gathering and analyzing data from member countries, collaborating with national health agencies, and issuing alerts and warnings when potential pandemics or public health threats are identified. The WHO also assists countries in establishing or enhancing their own surveillance systems.

Coordination and Information Sharing

During a pandemic, the WHO acts as a central hub for coordinating and facilitating international collaboration. It convenes meetings and conferences, facilitates information sharing among member states, provides technical guidance, and promotes the exchange of best practices and lessons learned. The WHO also assists countries in mobilizing resources, including medical supplies, vaccines, and financial support, to respond to the pandemic.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Research and Development

The WHO promotes research and development to address pandemics, including supporting studies on epidemiology, transmission, and clinical management of diseases. It collaborates with partners to accelerate the development and deployment of vaccines, diagnostics, and treatments. The WHO also promotes equitable access to these tools, particularly in resource-limited settings.

Risk Communication and Public Health Guidance

The WHO plays a crucial role in communicating accurate and timely information to the public, healthcare professionals, and policymakers during a pandemic. It provides evidence-based guidelines and recommendations on preventive measures, infection control, and treatment options. The WHO also addresses misinformation, dispels myths, and promotes public awareness and adherence to health guidelines.

Overall, the WHO's relationship with pandemics involves proactive preparedness, surveillance, coordination, research, and communication to support member states in preventing, detecting, and responding effectively to global health emergencies.

1.1.4 Pandemic Intervals

Pre-pandemic Interval

The pre-pandemic interval refers to the period before a pandemic is declared or before a new infectious disease outbreak becomes a global health concern. During this phase (8), there may be indications of a potential threat, such as the emergence of a novel virus or reports of localized outbreaks. Health authorities and researchers closely monitor the situation, conduct investigations, and gather data to assess the nature and severity of the disease. Efforts are focused on surveillance, early detection, risk assessment, and preparedness measures to mitigate the potential impact of the emerging disease.

Pandemic Interval

The pandemic interval is the phase when a disease outbreak has reached a global scale and is declared as a pandemic by relevant health organizations, such as the World Health Organization (WHO). It is characterized by sustained community transmission of the disease across multiple regions or continents. During this period, the virus spreads widely, affecting a significant number of people, and there is a heightened risk of severe illness and mortality. Public health measures, such as testing, contact tracing, isolation, quarantine, travel restrictions, and social distancing, are implemented to mitigate the spread of the virus and reduce its impact on healthcare systems and society. Vaccination campaigns and treatment strategies also play a crucial role in managing the pandemic.

Post-pandemic Interval

The post-pandemic interval refers to the period following the peak of the pandemic, when the disease transmission begins to decline, and the number of cases decreases significantly. During this phase, efforts shift from crisis response to recovery, rebuilding, and long-term planning. Public health authorities evaluate the response to the pandemic, assess the effectiveness of control measures, and identify areas for improvement. The focus may shift towards addressing the socio-economic consequences of the pandemic,

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

restoring healthcare systems, and strengthening public health infrastructure. Lessons learned from the pandemic inform future preparedness and response strategies to better handle potential future outbreaks or pandemics.

It's important to note that the durations of these intervals can vary depending on the specific pandemic and its characteristics. Additionally, transitions between these intervals may not be clearly demarcated and can overlap. Flexibility, adaptability, and collaboration among global health organizations, governments, and communities are essential for effectively navigating each interval and mitigating the impact of a pandemic.

1.1.5 Pandemic Phases

The description of pandemic stages and phases can vary depending on the framework or guidelines being used (10). One commonly referenced framework is the World Health Organization's (WHO) pandemic phases, which provide a general guideline for tracking the progression and severity of a pandemic. Here is an overview of the six phases in the WHO pandemic framework with the first three phase belonging to the pre-pandemic interval and the last three to the pandemic intervals:

Phase 1 - No or few cases in humans

This phase indicates that a novel virus is present in animals but has not been detected in humans or only a few isolated cases have occurred.

Phase 2 - Sporadic cases

In this phase, the virus is transmitted from animals to humans, resulting in sporadic cases or small clusters of infections. There is no sustained human-to-human transmission at this stage.

Phase 3 - Localized outbreaks

Phase 3 signifies that the virus has caused small-scale outbreaks in limited geographic areas. Human-to-human transmission becomes more evident, typically within close contacts or family members.

Phase 4 - Widespread transmission

This phase involves increased and sustained human-to-human transmission of the virus, leading to larger outbreaks in multiple areas or countries. The community-level outbreaks may occur, indicating the virus's ability to cause significant illness.

Phase 5 - Pandemic declaration

Phase 5 indicates that the virus has spread to multiple countries or continents, resulting in sustained community-level outbreaks. This phase represents the declaration of a pandemic and highlights the global impact of the disease.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Phase 6 - Sustained pandemic

Phase 6 is characterized by a sustained pandemic with ongoing community-level outbreaks in multiple parts of the world. At this stage, the focus shifts towards mitigation measures, healthcare management, and reducing the impact of the pandemic on public health, society, and the economy.

It's important to note that the stages and phases of a pandemic can vary depending on the specific disease and the guidelines used by health organizations. These frameworks are designed to provide a common understanding of the progression and severity of a pandemic, aiding in preparedness, response planning, and communication.

1.2 WORST PANDEMICS IN HISTORY

1.2.1 The Black Death (1347-1351)

The Black Death (11,12,13,14,15), also known as the Bubonic Plague, was caused by the bacterium Yersinia pestis and was primarily transmitted through fleas that infested rats. The disease caused severe symptoms such as high fever, swollen lymph nodes (buboes), and respiratory failure. The Black Death had devastating social and economic impacts, leading to labor shortages, social unrest, and changes in medieval society. It resulted in an estimated death toll of 75-200 million people. At the time, there was no known cure for the disease, and it had a mortality rate of approximately 30-60%.

1.2.2 Spanish Flu (1918-1919)

The Spanish flu pandemic was caused by an H1N1 influenza A virus. It spread rapidly, partially due to the movement of soldiers during World War I. The virus had a unique characteristic of causing severe illness and mortality in young adults, unlike most influenza viruses that predominantly affect the very young and elderly. The Spanish flu had a profound impact on global health and played a role in shaping public health systems and influenza research. The Spanish flu pandemic, caused by an H1N1 influenza A virus, claimed the lives of an estimated 50-100 million people worldwide. There was no specific cure for the Spanish flu, as antiviral medications were not yet available, and treatment primarily focused on supportive care and symptom management.

1.2.3 HIV/AIDS Pandemic (1981-Present)

The HIV/AIDS pandemic is caused by the human immunodeficiency virus (HIV). It primarily spreads through sexual contact, blood transfusions, and sharing contaminated needles. HIV weakens the immune system, leading to acquired immunodeficiency syndrome (AIDS) and making individuals susceptible to opportunistic infections and certain types of cancer. Despite significant advancements in treatment and prevention, HIV/AIDS continues to be a major global health concern, particularly in sub-Saharan Africa with the 64% of the estimated 40 million living with HIV living in this region.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

The HIV/AIDS pandemic has resulted in approximately 32 million deaths since it was first identified in the early 1980s. Despite significant advancements in antiretroviral therapy (ART) that help control the virus and manage the disease, there is currently no cure for HIV/AIDS. However, with proper treatment and access to healthcare, individuals living with HIV can lead long and healthy lives.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

1.2.4 Asian Flu (1957-1958)

The Asian flu pandemic originated in East Asia and was caused by an H2N2 influenza A virus. It spread globally, resulting in millions of deaths. The Asian flu led to the development of the first effective influenza vaccine and highlighted the importance of rapid vaccine production and distribution during pandemics. The Asian flu pandemic, caused by an H2N2 influenza A virus, resulted in an estimated death toll of around 1-2 million people worldwide. The development of the first effective influenza vaccine was a significant advancement during this pandemic, although it took several months to produce and distribute the vaccine widely.

Number of deaths: More than 1 million

1.2.5 Hong Kong Flu (1968-1969)

The Hong Kong flu pandemic was caused by the H3N2 strain of the influenza virus, which emerged from a genetic reassortment of avian and human viruses. The pandemic resulted in widespread illness and mortality, particularly among the elderly and those with pre-existing health conditions. Vaccines and improved surveillance played a crucial role in controlling the spread of the virus. The Hong Kong flu pandemic, caused by the H3N2 strain of the influenza virus, led to an estimated death toll of 1-4 million people globally. While antiviral medications were available to manage symptoms, there was no specific cure for the Hong Kong flu. Vaccines and improved surveillance played crucial roles in controlling the spread of the virus.

1.2.6 Russian Flu (1889-1890)

The Russian flu pandemic, caused by an H2N2 influenza A virus, originated in central Asia and quickly spread across Europe, Asia, and North America. The pandemic resulted in high mortality rates, particularly among young adults. The Russian flu highlighted the need for international collaboration in pandemic response and led to advances in laboratory techniques for identifying influenza viruses. The Russian flu pandemic, caused by an H2N2 influenza A virus, resulted in an estimated death toll of 1 million people worldwide. During this

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

pandemic, there was no known cure for the flu, and treatment primarily focused on supportive care to alleviate symptoms and manage complications.

1.2.7 Third Cholera Pandemic (1852-1860)

The third cholera pandemic began in India and spread through trade and travel routes to Europe, Asia, and North America. Cholera, caused by the bacterium Vibrio cholerae, leads to severe diarrhea and dehydration. The third pandemic led to improvements in sanitation practices, water supply systems, and public health infrastructure. The third cholera pandemic originated in India and claimed the lives of millions of people worldwide. Cholera does not have a specific cure, but advances in sanitation practices, improved water supply systems, and public health infrastructure played a vital role in controlling the spread of the disease and reducing mortality rates.

1.2.8 Sixth Cholera Pandemic (1899-1923)

The sixth cholera pandemic originated in India and spread to various parts of the world. It caused hundreds of thousands of deaths and prompted the development of effective oral rehydration therapy to treat cholera. The understanding of cholera transmission and control improved during this pandemic. The sixth cholera pandemic resulted in several hundred thousand deaths worldwide. While there is no cure for cholera, the development of effective oral rehydration therapy (ORT) revolutionized the treatment of the disease. ORT helps restore lost fluids and electrolytes, significantly reducing mortality rates.

1.2.9 Swine Flu (2009-2010)

The swine flu pandemic was caused by an H1N1 influenza A virus that originated from a combination of human, avian, and swine influenza viruses. It spread globally and led to significant public health concerns. While the mortality rate was relatively low, the pandemic highlighted the importance of surveillance, rapid response, and global coordination in addressing emerging influenza viruses. The swine flu pandemic, caused by an H1N1 influenza A virus, resulted in an estimated death toll of 150,000-575,000 people globally. Although most cases were mild, there were severe cases and fatalities. Antiviral

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

medications such as oseltamivir (Tamiflu) were used to treat severe cases, but no specific cure for the swine flu existed.

1.2.10 COVID-19 Pandemic (2019-Present)

The ongoing COVID-19 pandemic is caused by the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). It has had a profound global impact, resulting in millions of deaths and causing widespread health, economic, and social disruptions. The pandemic has emphasized the need for robust healthcare systems, widespread testing, vaccination campaigns, and adherence to public health measures like social distancing and mask-wearing.

1.2.11 Overall

These pandemics have shaped our understanding of infectious diseases, influenced public health policies, and spurred scientific advancements in medicine, virology, and epidemiology. They serve as reminders of the importance of preparedness, surveillance, and international cooperation in addressing global health crises. The ongoing COVID-19 pandemic, caused by the SARS-CoV-2 virus, has resulted in millions of deaths worldwide. As of my knowledge cutoff in September 2021, there is no known cure for COVID-19, although several vaccines have been developed and authorized for emergency use. Treatment primarily involves supportive care to manage symptoms and complications.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

1.3 INFLUENZA PANDEMICS: TYPES, IMPACT & PUBLIC HEALTH SIGNIFICANCE

1.3.1 Influenzas

Influenza viruses pose a significant burden on global public health, causing widespread respiratory infections and annual outbreaks (17,18,19,20). Among the various types of influenza, two notable categories are seasonal influenza and avian influenza. While both share similarities, they have distinct characteristics and implications that warrant careful examination. Understanding the differences between these two types of influenza is crucial for implementing effective prevention, surveillance, and response strategies.

1.3.2 Seasonal Influenza

Seasonal influenza refers to the regular, predictable outbreaks that occur during specific seasons in a given geographic region. It is caused by various subtypes of influenza viruses, including influenza A (H1N1 and H3N2) and influenza B with the incubation period to range between 1 and 4 days. The transmission of seasonal influenza primarily occurs through human-to-human contact, typically via respiratory droplets when an infected individual coughs or sneezes. This high contagiousness contributes to the annual occurrence of seasonal influenza during the winter months in temperate regions.

The impact of seasonal influenza on human health can vary from mild (asymptomatic) to severe illness (heavy pneumonia) or even death, resulting in significant morbidity and mortality worldwide. Vulnerable populations such as the elderly, young children, and individuals with underlying health conditions are particularly at risk. Each year, seasonal influenza strains undergo antigenic drift, necessitating the development of new vaccines to match the circulating strains and provide optimal protection.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

1.3.3 Avian Influenza

Avian influenza, commonly known as bird flu, primarily affects birds, particularly wild waterfowl and poultry. It encompasses various subtypes, including H5N1, H7N9, and H9N2. Avian influenza is primarily a zoonotic disease, meaning it can infect humans through direct contact with infected birds or their environments with a incubation period ranging between 2 and 8 days. While human infections are relatively rare, they can be severe (rapid progression of pneumonia with respiratory failure) and carry a high mortality rate. The initial symptoms usually are diarrhea, vomiting, abdominal pain, chest pain, and bleeding from the nose). Sustained by human-to-human transmission of avian influenza is limited but remains a significant concern.

The potential of avian influenza to cause a global pandemic arises from its ability to undergo genetic changes and acquire the capability for sustained human-to-human transmission. Surveillance and rapid response to outbreaks in avian populations are crucial to prevent potential pandemics and protect human health. The monitoring of avian influenza strains in birds, early detection of novel strains, and understanding the potential for virus reassortment are key challenges in addressing this form of influenza.

1.3.4 Impact On Public Health

Both seasonal and avian influenza require robust surveillance systems to monitor the emergence and spread of new strains. Timely identification of circulating strains allows for the development and distribution of effective vaccines and antiviral medications to reduce the impact of seasonal influenza and mitigate the risk of avian influenza outbreaks in humans.

Global preparedness and response strategies play a crucial role in addressing the challenges posed by seasonal and avian influenzas. International collaboration, information sharing, and coordinated efforts are essential for early detection, rapid response, and containment of outbreaks. However, both forms of influenza present their unique challenges. Seasonal influenza demands annual vaccine updates and the unpredictability of circulating strains, while avian influenza necessitates surveillance in bird populations, early detection of

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

novel strains, and understanding the potential for virus reassortment leading to human-to-human transmission.

Seasonal and avian influenzas represent distinct forms of influenza viruses, each with its own characteristics, transmission patterns, and impact on human health. Seasonal influenza recurs annually, primarily affecting humans, while avian influenza primarily affects birds but carries the risk of sporadic human infections and the potential for pandemics. Understanding the differences between these two types of influenza is crucial for implementing targeted prevention, surveillance, and response measures to mitigate their impact on global public health. Through continuous research, surveillance, and international collaboration, we can better prepare for and respond to the challenges posed by seasonal and avian influenzas.

1.3.5 Pandemic Influenzas

Pandemic influenza viruses are a specific type of influenza virus that can cause global outbreaks, or pandemics, with significant impacts on public health. These viruses are distinct from seasonal influenza viruses, which circulate every year and cause regular outbreaks but do not typically result in global pandemics. Here is some information about pandemic influenza viruses:

Origin and Types

Pandemic influenza viruses can originate from animal populations, particularly birds or pigs, and undergo genetic changes that allow them to infect humans. The most well-known pandemic influenza viruses include the 1918 H1N1 "Spanish flu," the 1957 H2N2 "Asian flu," the 1968 H3N2 "Hong Kong flu," and the 2009 H1N1 "swine flu." Each of these viruses had different origins and characteristics.

Rapid Spread

Pandemic influenza viruses have the potential to spread rapidly and widely among humans due to their ability to easily transmit from person to person. The high transmission rates, combined with a lack of pre-existing immunity in the population, contribute to the rapid global spread of the virus.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Severe Illness and Mortality

Pandemic influenza viruses can cause more severe illness and higher mortality rates compared to seasonal influenza viruses. This is often due to several factors, including the novelty of the virus, which leads to a lack of immunity in the population, and the potential for the virus to affect different age groups, including healthy young adults.

Global Impact

Pandemic influenza viruses have the potential to cause significant disruptions to healthcare systems, economies, and daily life. The widespread transmission and severity of illness can lead to increased hospitalizations, overwhelmed healthcare facilities, and strains on medical resources. Additionally, pandemic outbreaks can result in social and economic disruptions, travel restrictions, school closures, and other measures aimed at reducing the spread of the virus.

Preparedness and Response

Given the potential impact of pandemic influenza viruses, international and national health organizations emphasize preparedness and response planning. This includes surveillance systems to detect and monitor the emergence of novel influenza viruses, the development and stockpiling of antiviral medications and vaccines, and the establishment of communication networks for timely information sharing during outbreaks. Preparedness efforts also focus on public education, promoting personal hygiene practices, and ensuring healthcare systems are equipped to handle increased patient loads during a pandemic.

Ongoing Monitoring

Due to the ever-changing nature of influenza viruses, continuous monitoring of potential pandemic strains is essential. Surveillance systems and research networks actively monitor influenza viruses circulating in animal populations to identify any changes that may pose a threat to human health. This surveillance helps inform vaccine development, antiviral strategies, and public health interventions to mitigate the impact of future pandemic influenza outbreaks.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

1.4 DENGUE FEVER

1.4.1 Dengue Fever

Dengue fever is a significant global health concern, affecting millions of people worldwide each year (21,22,23). It is most prevalent in tropical and subtropical regions, with over 100 countries experiencing endemic dengue transmission. Approximately half of the world's population is at risk of contracting dengue (Figure 1).

The main mode of transmission is through the bite of infected Aedes mosquitoes, primarily Aedes aegypti. These mosquitoes become infected when they bite a person already infected with the dengue virus. The virus can then be transmitted to other individuals when they are bitten by infected mosquitoes. While less common, dengue can also be transmitted vertically from mother to child or through blood transfusion or organ transplantation.

Dengue fever can present in various ways, ranging from mild to severe illness. Symptoms include high fever, severe headache, joint and muscle pain, rash, and mild bleeding manifestations such as nose or gum bleeding. In severe cases, known as dengue hemorrhagic fever (DHF) and dengue shock syndrome (DSS), the disease can lead to plasma leakage, organ impairment, and life-threatening complications.

The impact of dengue fever on public health is significant. It places a substantial economic burden on healthcare systems, due to treatment costs, loss of productivity, and the strain on resources. Severe cases of dengue, especially among children, can result in significant morbidity and mortality.

Efforts to control dengue fever focus on various strategies. Vector control measures play a crucial role in preventing dengue transmission, including eliminating mosquito breeding sites, using insecticides, and promoting personal protective measures like mosquito repellents and bed nets. Additionally, early detection and proper management of dengue cases are essential to prevent complications. Supportive care, such as fluid replacement therapy, is critical in treating severe cases.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

One avenue of prevention is the development and implementation of dengue vaccines. Several vaccines have been developed and are licensed in some countries, offering promising prospects for reducing the burden of dengue.

It's important to note that scientific understanding of dengue fever is continually advancing. Ongoing research aims to improve prevention, diagnosis, and treatment strategies to effectively combat this disease and mitigate its impact on global health.

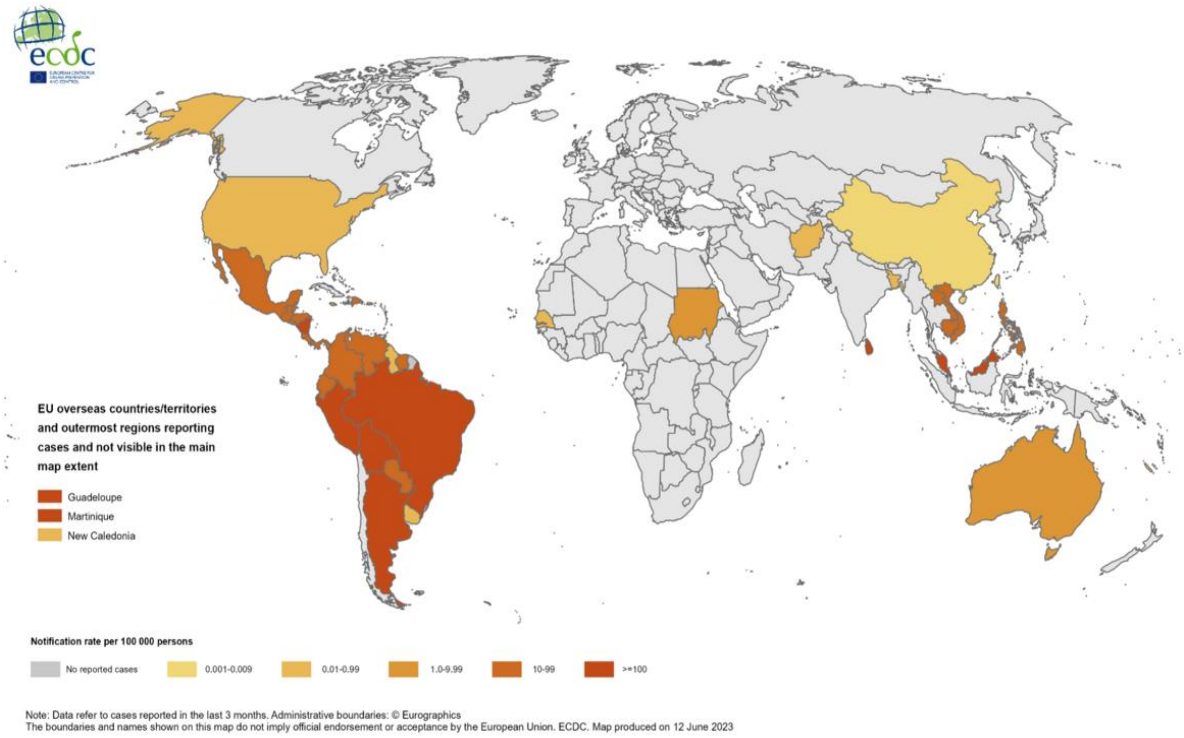

Figure 1. *Dengue fever cases in earth per country (ecdc.europa.eu).*

1.4.2 Clinical Features

Dengue fever presents a range of clinical features that can vary in severity. Here are the common clinical features associated with dengue fever (24).

- High Fever: Dengue fever typically begins with a sudden onset of high fever, often exceeding 104°F (40°C). The fever is usually continuous and lasts for 2-7 days.
- Headache: Headache is a common symptom of dengue fever and is often described as a severe, throbbing pain behind the eyes or in the frontal area of the head.
- Joint and Muscle Pain: Dengue fever is characterized by intense joint and muscle pain, often referred to as "breakbone" pain. It can affect multiple joints and muscles throughout the body, causing discomfort and restricted movement.
- Rash: A rash may develop in individuals with dengue fever, typically appearing 2-5 days after the onset of fever. The rash is usually maculopapular in nature, which means it consists of small raised bumps on the skin.
- Fatigue and Weakness: Fatigue and weakness are common symptoms experienced by individuals with dengue fever. The illness can cause significant tiredness and a general feeling of malaise.
- Eye Pain: Eye pain, particularly pain behind the eyes, is a characteristic symptom of dengue fever. It may worsen with eye movement.
- Mild Bleeding Manifestations: Some individuals with dengue fever may experience mild bleeding manifestations, such as nosebleeds, bleeding gums, or easy bruising. These bleeding symptoms are usually minor and self-limiting.
- Gastrointestinal Symptoms: Dengue fever can cause gastrointestinal symptoms, including nausea, vomiting, abdominal pain, and diarrhea. These symptoms may contribute to dehydration if not properly managed.

It's important to note that the clinical presentation of dengue fever can vary from person to person. Some individuals may experience mild symptoms resembling a common flu-like illness, while others may develop severe

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

manifestations such as dengue hemorrhagic fever (DHF) or dengue shock syndrome (DSS). Severe forms of dengue are characterized by plasma leakage, bleeding, organ impairment, and can be life-threatening if not promptly treated.

1.4.3 Laboratory Criteria

Laboratory criteria play a crucial role in the diagnosis and confirmation of dengue fever. Here are the common laboratory criteria used for diagnosing dengue:

Detection of Dengue Virus

Molecular Tests: Reverse transcription-polymerase chain reaction (RT-PCR) is commonly used to detect the presence of dengue viral RNA in blood samples. This test helps identify the specific serotype of the virus.

Viral Culture: Isolating and growing the dengue virus in cell culture can confirm the presence of the virus. However, this method is time-consuming and less commonly used.

Detection of Dengue-Specific Antibodies

Serology: Serological tests detect the presence of dengue-specific antibodies, including immunoglobulin M (IgM) and immunoglobulin G (IgG).

IgM Antibodies: IgM antibodies are produced in the early stage of infection and can be detected within 3-5 days after symptom onset. Elevated levels of IgM antibodies suggest a recent dengue infection.

IgG Antibodies: IgG antibodies typically appear later in the course of infection and persist for a longer duration. Rising levels of IgG antibodies indicate a secondary dengue infection or previous exposure.

Hematological Findings

Complete Blood Count (CBC): A CBC test helps assess various blood parameters, including platelet count, white blood cell count, and hematocrit (percentage of red blood cells in the blood). In dengue fever, a decrease in platelet count (thrombocytopenia) and an increase in hematocrit (hemoconcentration) may be observed.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

It's important to note that the interpretation of laboratory results should be considered in conjunction with clinical symptoms and epidemiological factors. Laboratory tests should be performed by trained professionals using standardized methods and quality control measures to ensure accurate results.

Additionally, it's worth mentioning that laboratory criteria may vary based on the guidelines and protocols established by different health authorities or countries. Therefore, it is advisable to consult the specific guidelines and recommendations provided by relevant health agencies for accurate and up-to-date information on laboratory criteria for diagnosing dengue fever.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

1.4.4 Epidemiological Criteria

Epidemiological criteria play a crucial role in the diagnosis and assessment of dengue fever. They involve evaluating the individual's exposure history and the presence of dengue transmission in a specific geographic area. Here are the common epidemiological criteria used for diagnosing dengue.

Geographic Location

Dengue fever is typically endemic in specific regions, particularly tropical and subtropical areas. The presence of dengue transmission in a specific geographic location, either locally acquired or through travel history, is an important epidemiological criterion.

Travel History

In non-endemic regions, a history of recent travel to areas where dengue is endemic is significant. It is essential to consider the timing of the travel in relation to the onset of symptoms. Travel to dengue-endemic regions increases the likelihood of dengue infection.

Epidemic Outbreaks

The occurrence of dengue outbreaks in a specific community or region may be an epidemiological criterion. Monitoring the local transmission dynamics and the presence of clusters of dengue cases can help identify potential outbreaks and guide appropriate public health responses.

Vector Exposure

A history of exposure to Aedes mosquitoes, the primary vector for dengue transmission, is relevant. This includes residence or travel to areas with a high prevalence of Aedes mosquito populations or potential breeding sites (e.g., stagnant water sources).

Contact with Dengue Cases

Close contact with individuals diagnosed with dengue fever can contribute to the epidemiological criteria. This includes household contacts, healthcare workers, or individuals involved in the care of dengue patients.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Epidemiological criteria are an integral part of dengue diagnosis, especially in situations where laboratory confirmation may not be readily available or inconclusive. The combination of clinical presentation, laboratory findings, and epidemiological factors enhances the accuracy of dengue diagnosis and helps guide appropriate management and public health interventions.

It's important to note that the interpretation of epidemiological criteria should be considered in conjunction with clinical symptoms and laboratory results to ensure a comprehensive assessment of dengue fever. It is advisable to consult the guidelines and recommendations provided by relevant health authorities or institutions for accurate and up-to-date information on epidemiological criteria for diagnosing dengue fever.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

1.4.5 Case classification and severity grading

Case classification is an important aspect of managing and monitoring dengue fever. It involves categorizing individuals with suspected or confirmed dengue infections based on their clinical presentation, laboratory results, and severity of illness. The World Health Organization (WHO) provides guidelines for case classification, which may vary slightly across different regions and healthcare settings. Here are the common case classifications for dengue fever (see also figure 2).

Suspected Dengue Case

A suspected dengue case is defined as an individual presenting with acute febrile illness (fever) lasting 2-7 days, accompanied by two or more of the following symptoms: headache, pain behind the eyes, joint and muscle pain, rash, or mild bleeding manifestations (e.g., nose or gum bleeding). Suspected cases may or may not have laboratory confirmation.

Probable Dengue Case

A probable dengue case is a suspected case with any of the following:

-Positive laboratory confirmation of dengue-specific IgM antibodies.

-Positive dengue non-structural protein 1 (NS1) antigen test.

-Positive dengue virus isolation or detection by PCR.

-Seroconversion or significant increase in dengue-specific IgG antibody levels in paired serum samples.

Confirmed Dengue Case

A confirmed dengue case is a suspected or probable case with laboratory confirmation. This includes any of the following:

-Isolation of dengue virus.

-Detection of dengue virus RNA by PCR.

-Positive results for both dengue-specific IgM and IgG antibodies in the same or separate samples.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

-Detection of dengue NS1 antigen and positive results for dengue-specific IgM or IgG antibodies.

1.4.6 Severity

In addition to case classification, dengue fever is further categorized based on its severity.

Dengue Fever (DF)

DF refers to the less severe form of dengue, characterized by fever, headache, joint and muscle pain, rash, and mild bleeding manifestations. There is no evidence of severe plasma leakage, organ impairment, or severe bleeding.

Dengue Hemorrhagic Fever (DHF)

DHF is a more severe form of dengue, characterized by the presence of bleeding, plasma leakage, and organ impairment. Individuals with DHF may experience severe abdominal pain, persistent vomiting, rapid breathing, bleeding from mucous membranes, and signs of fluid accumulation (e.g., pleural effusion, ascites).

Dengue Shock Syndrome (DSS)

DSS is the most severe form of dengue, marked by the presence of circulatory failure leading to shock. It is characterized by rapid and weak pulse, cold and clammy skin, restlessness, and altered mental status.

Case classification and severity grading help guide appropriate management and treatment decisions for individuals with dengue fever, ensuring timely interventions to prevent complications and reduce mortality. It is important to follow the guidelines and protocols established by relevant health authorities or institutions for accurate and consistent case classification of dengue fever.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

GLOBAL MAP AND THE RISK OF CONRACTING DENGUE FEVER

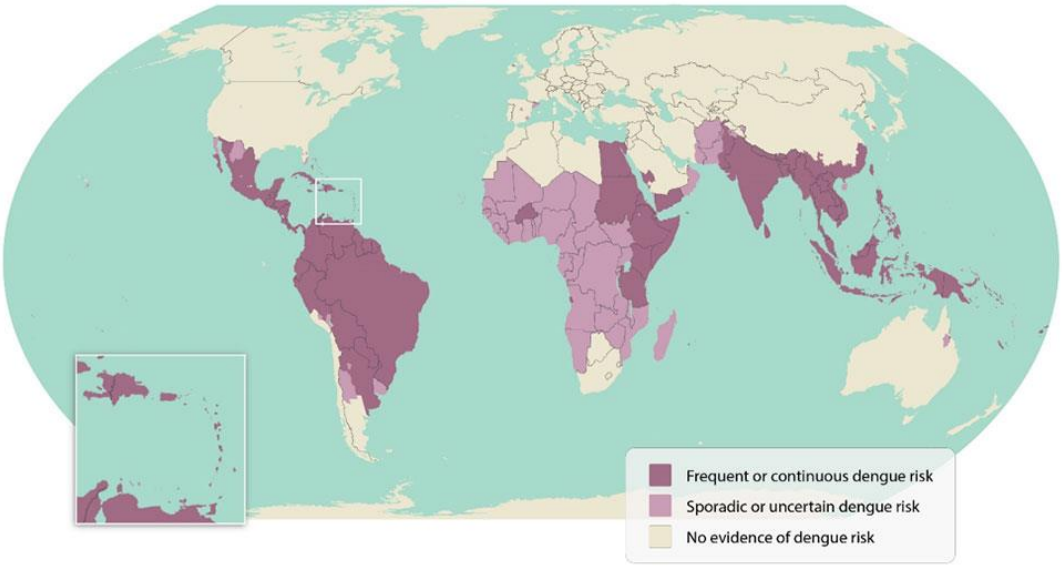

Figure 2. *Dengue fever contraction risk per country (cdc.gov).*

1.4.7 Dengue Fever In France

Dengue fever is not endemic in France, meaning that the virus is not regularly present and transmitted within the country (25,26,27). However, sporadic cases of dengue have been reported in France, primarily in individuals who have traveled to dengue-endemic regions and returned with the infection. These cases are typically considered as imported or travel-associated cases. The first documented cases of dengue fever in France occurred in the 1940s and 1950s when travelers returning from dengue-endemic countries were diagnosed with the disease. These cases were imported, and there was no sustained transmission within France.

In more recent times, localized outbreaks of dengue have been reported in overseas territories of France, such as Reunion Island, Martinique, and French Guiana. These regions have suitable climates for the transmission of dengue, and the presence of the Aedes mosquitoes, which are responsible for transmitting the virus.

These outbreaks have resulted in localized transmission of the virus within these territories. However, it's important to note that these outbreaks do not represent widespread dengue transmission throughout mainland France.

The French health authorities closely monitor and respond to dengue cases to prevent the establishment of local transmission. Measures such as vector control, public awareness campaigns, and surveillance systems are implemented to detect and manage cases effectively.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

1.4.8 Dengue Fever 2008-2020

From 2008 to 2020 (the time period we are going to analyze), France experienced several cases of dengue fever, primarily in individuals who traveled to dengue-endemic regions and returned with the infection. The number of dengue cases reported varied from year to year, with some years experiencing higher case numbers than others. Here is an overview of the history of dengue fever in France during the period from 2008 to 2020.

2008

In 2008, France experienced an outbreak of dengue fever in the overseas territory of Réunion Island. The outbreak resulted in a significant number of cases, with over 30,000 reported cases and several deaths.

2009-2011

During this period, sporadic cases of dengue were reported in mainland France, primarily imported by travelers returning from dengue-endemic regions. The number of cases remained relatively low compared to the outbreak in Réunion Island.

2012-2013

The number of dengue cases in mainland France increased during this period, with a notable increase in imported cases. These cases were primarily associated with travel to Southeast Asia, the Caribbean, and other dengue-endemic areas.

2014-2015

The number of dengue cases remained relatively stable, with sporadic imported cases reported in mainland France. Réunion Island experienced a resurgence of dengue cases during this period.

2016-2020

From 2016 onwards, dengue cases in mainland France continued to be primarily imported, with sporadic cases reported each year. Outbreaks occurred in

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

overseas territories such as Martinique and Guadeloupe, leading to localized transmission and increased case numbers in those areas.

1.4.9 Why The Statistical Surveillance Of The Dengue Fever Is Important

The statistical surveillance of dengue fever in France from 2008 to 2020 is of utmost importance for several reasons.

Monitoring Disease Trends

Statistical surveillance allows health authorities to track the incidence and prevalence of dengue fever cases over time. By analyzing the data collected, trends and patterns can be identified, providing insights into the disease's dynamics and helping to anticipate and respond to potential outbreaks.

Early Detection and Response

Timely detection of dengue cases is crucial for implementing effective public health interventions. Statistical surveillance enables the identification of areas or populations at higher risk of dengue transmission, allowing authorities to allocate resources, implement vector control measures, and provide targeted public health campaigns to prevent further spread.

Assessing Disease Burden

By collecting and analyzing statistical data, health authorities can determine the burden of dengue fever on the healthcare system and society as a whole. This includes estimating the number of cases, severity of illness, hospitalizations, and economic impact. Such information helps in resource allocation, planning healthcare services, and evaluating the effectiveness of prevention and control strategies.

Evaluating Control Measures

Statistical surveillance provides valuable data for assessing the impact of control measures implemented to prevent and control dengue fever. It allows health authorities to evaluate the effectiveness of interventions, such as vector control strategies, public health campaigns, and vaccination programs, by

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

comparing the incidence and distribution of cases before and after their implementation.

Supporting Research and Decision-making and the importance of statistical surveillance

Statistical data on dengue fever in France contributes to the scientific understanding of the disease and its epidemiology. Researchers can analyze the data to identify risk factors, assess the impact of climate change or other environmental factors, study the effectiveness of treatments, and develop predictive models. Decision-makers can rely on this evidence to inform policy decisions related to disease prevention, control, and healthcare planning.

Overall, the statistical surveillance of dengue fever in France from 2008 to 2020 is crucial for understanding the epidemiology of the disease, monitoring trends, guiding public health interventions, and evaluating the impact of control measures. It helps in protecting public health, reducing the burden of dengue fever, and improving the overall management of the disease.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2 UNDERSTANDING PREDICTIVE MODELLING

2.1 PREDICTIVE STATISTICAL MODELLING

2.1.1 Definition

Predictive modeling, also known as predictive analytics or predictive analysis, is a process that involves using historical data and statistical techniques to make predictions or forecasts about future outcomes or events. It is a field within data science that aims to uncover patterns and relationships in data to create models that can predict future outcomes.

2.1.2 Steps And Phases

Predictive modeling typically follows a set of steps, which may include:

Problem definition: Clearly defining the problem or question we want to address and determining what we aim to predict.

Data collection and preparation: Gathering relevant data that is representative of the problem and preparing it for analysis. This step often involves data cleaning, preprocessing, and feature engineering.

Model selection: Choosing the appropriate modeling technique or algorithm based on the characteristics of the data and the problem at hand. This can include regression models, decision trees, random forests, support vector machines, neural networks, or other machine learning algorithms.

Training the model: Using historical data to train the selected model. This involves feeding the model with input data and the corresponding known outcomes or target variables.

Model evaluation and validation: Assessing the performance of the trained model using evaluation metrics and techniques. This step helps determine how well the model generalizes to new, unseen data.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Model deployment and prediction: Applying the trained model to new, incoming data to make predictions or forecasts about future outcomes.

Predictive modeling has various applications across industries, including finance, healthcare, marketing, insurance, and more. It helps organizations make informed decisions, identify trends, anticipate customer behavior, manage risks, and optimize processes based on data-driven insights.

It is important to note that predictive modeling is based on historical data and statistical assumptions, and its accuracy and reliability depend on the quality and representativeness of the data, as well as the assumptions and limitations of the chosen modeling techniques.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.2 EXPLORING TYPES OF PREDICTIVE MODELS

There are various types of predictive models that can be used depending on the nature of the problem and the available data. Here are some commonly used predictive models along with simulated examples in R language:

2.2.1 Linear Regression

Linear Regression

Linear regression models establish a linear relationship between a dependent variable and one or more independent variables (28,29). They aim to find the best-fitting line that minimizes the difference between the predicted and actual values.

This model assumes a linear relationship between the independent variables and the dependent variable.

Example 1

Predicting a person's salary based on their years of education and years of work experience.

Example 2

Predicting Disease Progression Linear regression models can be used to predict the progression of diseases such as cancer or diabetes. By analyzing patient data, including biomarkers, genetic profiles, and treatment history, a linear regression model can estimate disease progression rates and help healthcare providers tailor personalized treatment plans.

Linear Regression: Model

A linear regression model assumes that the expected value of the output variable Y, given the input variables $\boldsymbol{X_1}$, $\boldsymbol{X_2}$, ..., $\boldsymbol{X_p}$, can be represented by a linear combination of these inputs. We have an input vector $\boldsymbol{X^T} = (\boldsymbol{X_1}, \boldsymbol{X_2}, \dots, \boldsymbol{X_p})$, and want to predict a real-valued output Y. The linear regression model has the form.

$$\boldsymbol{f}(\boldsymbol{X}) = \boldsymbol{\beta_0} + \sum_{j=1}^{p} \boldsymbol{X_j \beta_j}.$$

The linear model either assumes that the regression function $\mathrm{E}(Y \mid X)$ is linear, or that the model is very close to linear.

Here the $\boldsymbol{\beta_j}$ **'s** are coefficients (representing the relationship between each independent variable)

The variables $\boldsymbol{X_j}$ can come from different sources:

quantitative input

transformations of quantitative inputs, such as log, square-root or square

basis expansions, such as $\boldsymbol{X_2 = X_1^2, X_3 = X_1^3}$, leading to a polynomial representation.

numeric or "dummy" coding of the levels of qualitative inputs. For example, if G is a five-level factor input, we might create $\boldsymbol{X_j, j = 1, \dots, 5}$, such that $\boldsymbol{X_j = I(G = j)}$**.** Together this group of $\boldsymbol{X_j}$ represents the effect of $\boldsymbol{G}$ by a set of level-dependent constants, since in $\sum_{j=1}^{5} \boldsymbol{X_j \beta_j}$**,** one of the $\boldsymbol{X_j}$ **s** is one, and the others are zero.

interactions between variables, for example, $\boldsymbol{X_3 = X_1 \cdot X_2}$**.**

No matter the source of the X_j, the model is linear in the parameters.

Typically, we have a set of training data $(\boldsymbol{x_1, y_1}) \dots (\boldsymbol{x_N, y_N})$ from which to estimate the parameters $\boldsymbol{\beta}$. Each $\boldsymbol{x_i} = \left(\boldsymbol{x_{i1}, x_{i2}, \dots, x_{ip}}\right)^{\boldsymbol{T}}$ is a vector of feature measurements for the i th case. The most popular estimation method is ordinary least squares, in which we pick the coefficients $\beta = \left(\boldsymbol{\beta_0, \beta_1, \dots, \beta_p}\right)^{\boldsymbol{T}}$ to minimize the residual sum of squares.

$$\begin{aligned} \mathbf{RSS}(\boldsymbol{\beta}) &= \sum_{i=1}^{N} \left(\boldsymbol{y_i - f(x_i)}\right)^2 \\ &= \sum_{i=1}^{N} \left(\boldsymbol{y_i - \beta_0 - \sum_{j=1}^{p} x_{ij} \beta_j}\right)^2 \end{aligned}$$

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

OLS says, to get the parameters estimates for a linear model, find those that minimize the sum of the squared residuals.

R Package: stats

Linear Regression: Package: lm() function in the base R package.

Description: The stats package is a core package in R that provides various statistical functions, including linear regression. It includes the lm () function for fitting linear regression models and obtaining parameter estimates, residuals, and other model diagnostics. The package also offers functions for hypothesis testing, confidence interval estimation, and model comparison.

Visualization in r language

I generated a code in the R programming language to create a simulated dataset with an independent variable 'x' and a dependent variable 'y'. The relationship between 'x' and 'y' is linear, with some random noise added to mimic real-world variability. The code then utilizes **linear regression** to find the optimal linear equation that best fits the data and quantifies the relationship between 'x' and 'y'. By printing a summary of the regression model's key statistics and producing a scatter plot of the data points alongside the fitted regression line, the code visually demonstrates how well the model captures the linear trend in the data (see Appendix 2.1.1 for further details).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

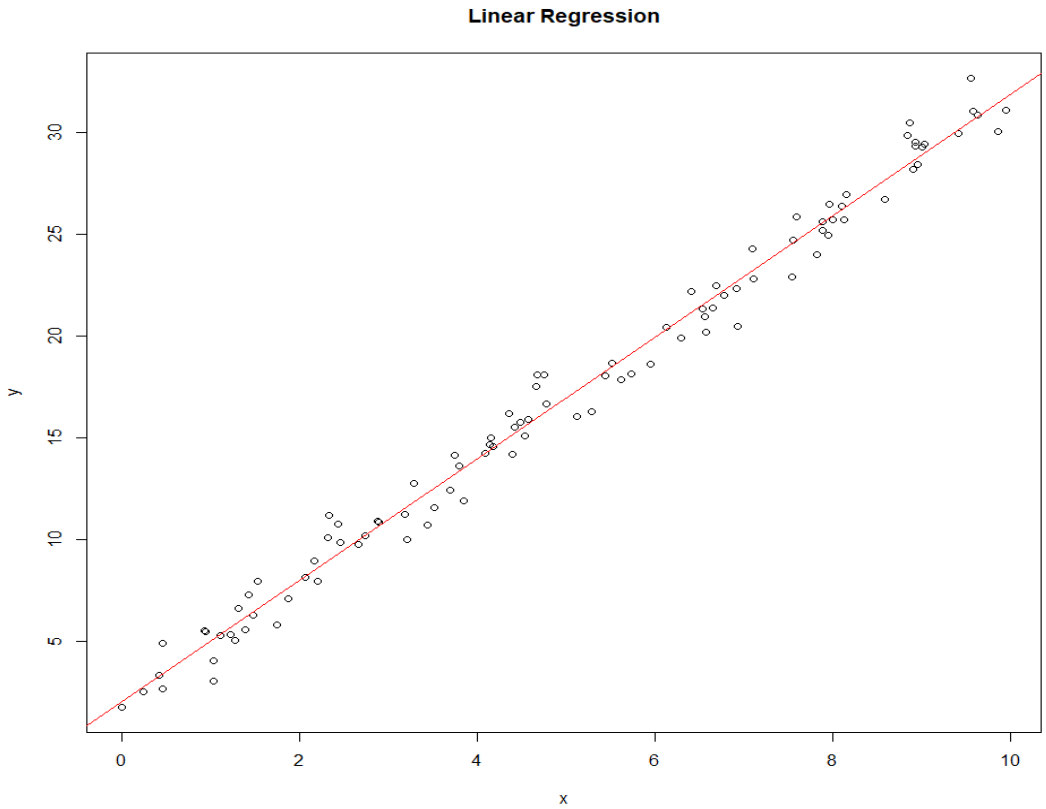

Figure 3. *Linear regression with simulated data graph*

Explanation of the simulated graph

In figure 3, each point represents an observation from our dataset, with its x and y values given by the position along the x-axis and y-axis, respectively. The red line (the line of best fit) represents the relationship between x and y as estimated by the linear regression model. The line is trying to get as close as possible to all points, and it represents the best estimate of y given x according to the model.

2.2.2 Logistic Regression

Logistic Regression

Logistic regression models are used when the dependent variable is categorical or binary. They estimate the probability of an event occurring based on a set of independent variables.

This model is used when the dependent variable is categorical and aims to predict the probability of an event occurring. For instance, predicting the likelihood of a customer churning based on their demographic and behavioral data (30).

Example 1

Credit Risk Assessment Logistic regression can be applied in the financial industry to predict the likelihood of default or credit risk for loan applicants. By analyzing historical data on borrower characteristics, income, credit history, and other factors, a logistic regression model can help lenders assess the probability of a borrower defaulting on a loan.

Example 2

Epidemic Outbreak Prediction Logistic regression can be applied to predict the likelihood of an epidemic outbreak based on various factors such as population density, environmental conditions, and disease transmission patterns. By analyzing historical epidemic data, a logistic regression model can help public health officials anticipate and prepare for potential outbreaks.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Logistic Regression: Model

In logistic regression, we have a binary dependent variable, usually denoted as Y, which can take values 0 or 1. We also have p independent variables or features, denoted as X_1, X_2, ..., X_p. The goal is to model the probability of Y being 1 given the values of the independent variables.

The logistic regression model assumes that the log-odds (also known as the logit) of Y being 1 is a linear combination of the independent variables. Mathematically, we can express this relationship as:

$$\text{logit}(P(Y=1 \mid X_1, X_2, \ldots, X_p)) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p$$

Where:

$P(Y=1 \mid X_1, X_2, \ldots, X_p)$ represents the probability of Y being 1 given the values of the independent variables.

$\beta_0, \beta_1, \beta_2, \ldots, \beta_p$ are the coefficients or weights associated with each independent variable.

$X_1, X_2, \ldots, X_p$ are the values of the independent variables.

To convert the logit into a probability, we use the logistic function or sigmoid function, which is defined as:

$$\sigma(z) = 1 / (1 + e^{(-z)})$$

where **e** is **Euler's number (approximately 2.71828)** and **z** is the linear combination of the independent variables and their corresponding coefficients.

The linear combination of the independent variables $X_1, X_2, \ldots, X_p$ with their coefficients $\beta_1, \beta_2, \ldots, \beta_p$ and the intercept term β_0 can be expressed as:

$$z = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p$$

where β_0 represents the intercept term.

Here, **z** represents the linear combination of the independent variables and their corresponding coefficients. So, substituting z with the equation for logit, we get:

$$P(Y=1 \mid X_1, X_2, \ldots, X_p) = \sigma(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p)$$

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Probability of Y being 1: To obtain the probability of the dependent variable Y being 1, we apply the logistic function to the logit:

$$P(Y=1 \mid X_1, X_2, \ldots, X_p) = \sigma(z) = 1 / (1 + e^{(-z)})$$

This equation gives us the probability of **Y** being **1** given the values of the independent variables.

During the training process, we aim to estimate the coefficients $\beta_0, \beta_1, \beta_2, \ldots, \beta_p$ that best fit the training data. This is usually done by maximizing the likelihood or minimizing the negative log-likelihood of the training data of observing the given data and its usually achieved using optimization algorithms such as gradient descent or Newton's method.

The likelihood function measures the probability of observing the training data given the current values of the coefficients. Maximizing the likelihood is equivalent to minimizing the negative log-likelihood or, equivalently, minimizing the cross-entropy loss function.

Optimization algorithms like gradient descent or Newton's method are commonly used to iteratively update the coefficients until convergence is achieved. The specific algorithm depends on the optimization strategy chosen.

Loss Function: To estimate the coefficients, a common approach is to use the negative log-likelihood as the loss function. For a training set with N samples, the negative log-likelihood (L) is given by:

$$L = -\sum [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)]$$

where y_i represents the true label (0 or 1) of the **i-th** sample, and p_i represents the predicted probability of **Y** being **1** for the **i-th** sample.

Gradient Descent: Gradient descent is a commonly used optimization algorithm to update the coefficients iteratively. The gradient of the negative log-likelihood with respect to the coefficients can be calculated and used to update the coefficients:

$$\beta_j \text{ (new)} = \beta_j \text{ (old)} - \text{learning_rate} * \sum [(p_i - y_i) * X_{ij}]$$

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

where β_j(new) represents the updated value of the **j-th** coefficient, learning_rate is a hyperparameter controlling the step size, and $\mathbf{X_{ij}}$ represents the value of the **j-th** independent variable for the **i-th** sample.

Once the coefficients are estimated, we can make predictions on new data. Given the values of the independent variables $\mathbf{X_1, X_2, ..., X_p}$ for a new observation, we calculate the logit:

$$z = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p$$

Then, we apply the logistic function to obtain the predicted probability of Y being 1:

$$P(Y=1 \mid X_1, X_2, ..., X_p) = \sigma(z) = 1 / (1 + e^{(-z)})$$

If the predicted probability is above a chosen threshold (commonly 0.5), we classify the observation as **Y=1**; otherwise not

R Package: stats

Description: Similar to linear regression, logistic regression is also implemented in the stats package. The **glm ()** function (generalized linear models) within the stats package can be used to fit logistic regression models. By specifying the appropriate family argument as "binomial", the function can estimate logistic regression coefficients and perform inference on the model.

Visualization in r language

I composed a code in the R programming language to simulate a logistic regression model . First, I loaded the "ggplot2" library for data visualization. I established a reproducible environment by setting a seed with **set.seed(123)**. Next, I generated an independent variable 'x' with random values in the range of -10 to 10. Then, I formulated a dependent variable 'z' based on a linear function of 'x' and transformed it using a logistic function to calculate probabilities. Consequently, I derived a binary variable 'y' by performing a random binomial process based on the calculated probabilities.

To model this binary outcome, I employed **logistic regression** through the "glm" function with the "binomial" family. Following that, I created sorted 'x' values and employed the fitted model to predict probabilities for these values.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

I compiled these predictions along with the sorted 'x' values into a dataframe. Additionally, I prepared a dataframe with the original 'x' and 'y' values for comparison.

The core of the visualization was constructed using ggplot2. I depicted the original data points in blue and translucent, while the red line illustrates the predicted probabilities. The title, x-axis label, and y-axis label were assigned using the 'labs' function. The chosen theme was minimalistic.

The finalized plot, showcasing the original data and the logistic regression model's predictions, was generated and printed using the 'ggplot2' package. (for more comprehensive information, please refer to Appendix 2.2.2)

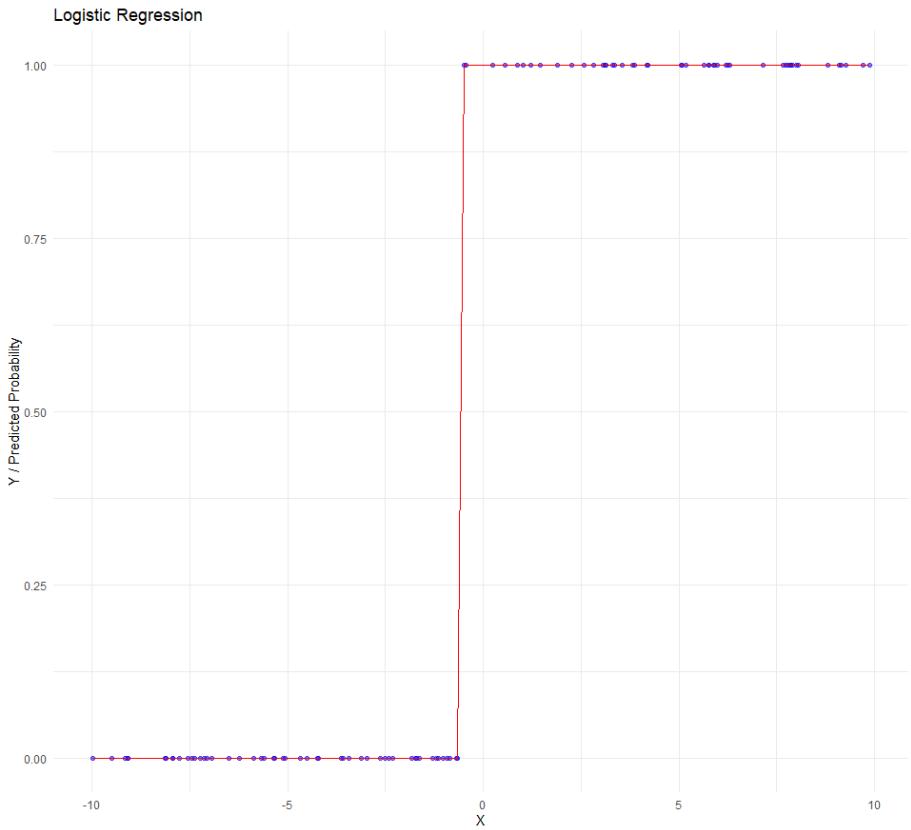

Figure 4. *Logistic regression with simulated data graph*

Explanation of the simulated graph

Figure 4 is a visual representation of a logistic regression analysis. The graph includes both the actual data points and the predictions made by the logistic regression model.

1. Blue Dots: The blue dots represent actual data points. Since the dependent variable y is binary (either 0 or 1), these dots are only seen at **y=0** or **y=1** for different x values.

2. Red Line: The red line represents the logistic regression model's predicted probabilities of y being 1 given the different x values. It is not strictly at **y=0** or **y=1** because it reflects probabilities, which range continuously between 0 and 1. The line shows how the model estimates the probability of y being 1 changes as x varies.

The line is "S"-shaped or sigmoidal, characteristic of the logistic function used in logistic regression. As x becomes large, the predicted probability approaches 1, and as x becomes small, the probability approaches 0.

3. Axes: The x-axis represents the variable x, and the y-axis represents either the binary outcome y for the data points or the predicted probability of y=1 for the logistic regression line.

4. Interpretation: The graph shows that as x increases, the probability of y being 1 (according to the model) also increases. This indicates a positive relationship between x and the likelihood of **y=1**. This relationship was set up when generating the data (**z <- 1 + 3*x**), and the logistic regression model has learned this relationship.

In conclusion, this shows how the probability of y being 1 changes with different x values.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.2.3 Time Series Analysis

Time Series Analysis

Time series models, such as Autoregressive Integrated Moving Average (ARIMA) or Seasonal ARIMA (SARIMA), are used to analyze and forecast data that is collected sequentially over time. These models capture temporal dependencies and patterns in the data (32,33).

This modeling technique is used when the data is collected over time and follows a sequential order. It can be employed to predict future values based on past observations. An example is forecasting stock prices based on historical market data.

Example 1

Demand Forecasting Retail companies often employ time series analysis to forecast future demand for their products. By analyzing historical sales data, including factors such as seasonality, trends, and promotional activities, a time series model can provide estimates of future sales volumes. This information helps companies optimize inventory management, production planning, and resource allocation.

Example 2

Forecasting Influenza Activity Time series analysis can be used to forecast influenza activity levels. By analyzing historical influenza surveillance data, including factors such as seasonality, viral strains, and vaccination rates, a time series model can provide estimates of future influenza activity. This information assists public health authorities in allocating resources and implementing preventive measures.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.2.3.1 *Autoregressive Integrated Moving Average*

ARIMA is a popular time series model used for forecasting. It combines three components: Autoregressive (AR), Integrated (I), and Moving Average (MA).

Autoregressive (AR) Component

The autoregressive component models the relationship between the variable of interest and its past values. It uses the notion of lagged values. The AR(p) model can be represented mathematically as:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \varepsilon_t$$

Where:

$\mathbf{Y_t}$ represents the value of the time series at time t.

c is a constant term.

$\phi_1, \phi_2, \dots, \phi_p$ are the autoregressive parameters representing the weights assigned to the lagged values.

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

Integrated (I) Component

The integrated component of ARIMA is used to make the time series stationary by differencing. It subtracts the series at time t from the series at time **t-1**. The differencing operation can be applied multiple times until stationarity is achieved.

Moving Average (MA) Component

The moving average component models the dependency between the variable of interest and the residuals (errors) from past forecasts. The **MA(q)** model can be represented mathematically as:

$$Y_t = c + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_p \varepsilon_{t-p} + \varepsilon_t$$

Where:

$\mathbf{Y_t}$ represents the value of the time series at time t.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

c is a constant term.

$\theta_1, \theta_2, \ldots, \theta_p$ are the moving average parameters representing the weights assigned to the past residuals or errors.

ε_t represents the white noise or error term at time t

2.2.3.2 *Seasonal Autoregressive Integrated Moving Average*

Seasonal Autoregressive Integrated Moving Average (SARIMA) is an extension of the ARIMA model that incorporates the seasonal component of a time series. It is specifically designed to handle time series data that exhibits seasonal patterns or trends. SARIMA combines the three components of ARIMA (Autoregressive, Integrated, Moving Average) with an additional seasonal component.

Autoregressive (AR) Component

Similar to ARIMA, the AR component of SARIMA models the relationship between the variable of interest and its past values. It considers the lagged values of the seasonal periods. The AR(p) component can be represented as:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \ldots + \phi_p Y_{t-p} + \phi_{1s} Y_{t-s-1} + \phi_{2s} Y_{t-s-2} + \ldots + \phi_{ps} Y_{t-s-p} + \varepsilon_t$$

Where:

Y_t represents the value of the time series at time t.

c is a constant term.

$\phi_1, \phi_2, \ldots, \phi_p$ are the autoregressive parameters representing the weights assigned to the lagged values of the non-seasonal component.

$\phi_{1s}, \phi_{2s}, \ldots, \phi_{ps}$ are the autoregressive parameters representing the weights assigned to the lagged values of the seasonal component.

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

Integrated (I) Component

The integrated component of SARIMA is used to make the time series stationary by differencing, similar to ARIMA. It subtracts the series at time t

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

from the series at time **t-1**. The differencing operation can be applied multiple times until stationarity is achieved.

Moving Average (MA) Component

The MA component of SARIMA models the dependency between the variable of interest and the residuals (errors) from past forecasts, considering both the non-seasonal and seasonal components. The MA(q) component can be represented as:

$$Y_t = c + \theta_1\varepsilon_{t-1} + \theta_2\varepsilon_{t-2} + \dots + \theta_p\varepsilon_{t-p} + \theta_{1s}\varepsilon_{t-s-1} + \theta_{2s}\varepsilon_{t-s-2} + \dots + \theta_{ps}\varepsilon_{t-s-p} + \varepsilon_t$$

Where:

$\mathbf{Y_t}$ represents the value of the time series at time t.

c is a constant term.

$\boldsymbol{\theta_1, \theta_2, \dots, \theta_p}$ are the moving average parameters representing the weights assigned to the past residuals or errors of the non-seasonal component.

$\boldsymbol{\theta_{1s}, \theta_{2s}, \dots, \theta_{ps}}$ are the moving average parameters representing the weights assigned to the past residuals or errors of the seasonal component.

$\boldsymbol{\varepsilon_t}$ represents the white noise or error term at time t.

2.2.3.3 *Differences Between ARIMA and SARIMA*

The main difference between SARIMA and ARIMA is the inclusion of the seasonal component in SARIMA. SARIMA takes into account the seasonal patterns and trends in the data in addition to the non-seasonal components. This allows SARIMA to capture and forecast time series data with both non-seasonal and seasonal variations, making it suitable for datasets that exhibit recurring patterns at specific intervals, such as quarterly or monthly data.

In summary, SARIMA extends the ARIMA model by incorporating a seasonal component, enabling more accurate modeling and forecasting of time series data with both non-seasonal and seasonal variations. The inclusion of the seasonal component allows SARIMA to capture and predict patterns that repeat

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

at regular intervals, improving the accuracy of forecasts for time series data that exhibit seasonal trends or patterns.

Finally the SARIMA model considers the autoregressive, integrated, and moving average components to capture the relationships between the current and past values, differences in the series to achieve stationarity, and the dependency on past residuals. Additionally, the seasonal component accounts for the cyclicality and periodic patterns observed in asthma hospital admissions.

By forecasting future asthma hospital admissions using SARIMA, healthcare institutions can optimize resource allocation, plan for staffing needs, and improve overall patient care by ensuring timely availability of healthcare services and resources tailored to the expected demand. This helps in proactive management of healthcare services, reducing the burden on the healthcare system and enhancing patient outcomes.

Example 1

An example of using SARIMA in medicine could be the forecasting of monthly hospital admissions for a specific medical condition, such as asthma. Asthma is known to have both non-seasonal variations influenced by factors like air pollution and allergens, as well as seasonal variations influenced by weather conditions.

Example 2

By applying SARIMA modeling to historical monthly hospital admission data for asthma, including both the non-seasonal and seasonal components, it is possible to develop a forecasting model that takes into account the underlying trends and seasonal patterns. This can help healthcare providers and policymakers anticipate and plan for future demand, allocate resources effectively, and implement preventive measures or interventions during peak seasons to manage the influx of asthma patients.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.2.3.4 *SAR*

The SAR (Seasonal Autoregressive) model is a time series model that captures the seasonal component in the data and a special case of SARIMA models. It is an extension of the Autoregressive (AR) model and is used for modeling and forecasting time series data with seasonal patterns. The SAR model incorporates lagged values of the time series at seasonal intervals to capture the seasonal behavior.

Seasonal Autoregressive (SAR) Component

The seasonal autoregressive component captures the relationship between the variable of interest and its lagged values at seasonal intervals. It is denoted by SAR (P, s), where P represents the number of seasonal lags and s represents the seasonal period.

Autoregressive (AR) Component

The AR component models the relationship between the variable of interest and its past values at seasonal intervals. It uses the notion of lagged values, similar to the ARIMA model. The AR(p) component can be represented mathematically as:

$$Y_t = c + \phi_1 Y_{t-s} + \phi_2 Y_{t-2s} + \dots + \phi_p Y_{t-ps} + \varepsilon_t$$

Where:

Y_t represents the value of the time series at time t.

c is a constant term.

$\phi_1, \phi_2, \dots, \phi_p$ are the autoregressive parameters representing the weights assigned to the lagged values at seasonal intervals.

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

The seasonal lags, denoted by $Y_{t-s}, Y_{t-2s}, \dots, Y_{t-ps}$, capture the relationship between the current value of the time series and its past values at the same seasonal interval. For example, if the time series has a yearly seasonality, the seasonal lags would represent the values of the time series at the same month of the previous year.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Other Components

The SAR model does not include the Integrated (I) and Moving Average (MA) components of SARIMA. It focuses solely on capturing the seasonal behavior of the time series using the autoregressive component.

IN R

To fit a SAR model in R, we can use the sarima () function from the "forecast" package, similar to fitting a SARIMA model. However, we would only specify the seasonal autoregressive order (p) and the seasonal period (m) in the order argument, without including the non-seasonal autoregressive order (p) and the moving average order (q).

Once the SAR model is fitted, we can use it to make forecasts and analyze the seasonal patterns in the data.

It's important to note that the SAR model assumes that the seasonal patterns are consistent and predictable over time. If the seasonal patterns change or exhibit irregularities, a more flexible model such as SARIMA or other advanced models may be more appropriate.

2.2.3.5 *VAR*

The VAR (Vector Autoregression) model is a multivariate time series model that captures the linear interdependencies between multiple variables. Unlike univariate models like ARIMA or SARIMA, which focus on a single variable, VAR models consider multiple variables and their relationships over time. VAR models are useful for analyzing and forecasting the behavior of multiple related time series simultaneously.

Vector Representation

In VAR, the variables of interest are represented as a vector. Let's say we have k variables of interest, denoted as $\mathbf{Y_1, Y_2, ..., Y_k}$. The variables are arranged in a vector form as $\mathbf{Y = [Y_1, Y_2, ..., Y_k]^T}$.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Autoregressive (AR) Component

The AR component of VAR models the relationship between the variables in the vector and their past values. It uses the notion of lagged values for each variable. The VAR(p) model can be represented mathematically as:

$$\mathbf{Y_t = c + \Phi_1 Y_{t-1} + \Phi_2 Y_{t-2} + \ldots + \Phi_p Y_{t-p} + \varepsilon_t}$$

Where:

$\mathbf{Y_t}$ represents the vector of variables at time t.

c is a constant term.

$\mathbf{\Phi_1, \Phi_2, \ldots, \Phi_p}$ are the autoregressive matrices representing the weights assigned to the lagged values of each variable.

$\boldsymbol{\varepsilon_t}$ is the vector of white noise or error terms at time t.

The autoregressive matrices, denoted by $\mathbf{\Phi_1, \Phi_2, \ldots, \Phi_p}$, capture the relationships between the current values of the variables and their lagged values.

Error Term

The error term ε_t represents the unexplained variation in the VAR model. It is assumed to be a vector of white noise, meaning that the errors are uncorrelated and have constant variances across variables and time.

Use

VAR models are typically estimated using various methods, such as maximum likelihood estimation or least squares estimation, to estimate the autoregressive matrices and the error covariance matrix. Once the VAR model is estimated, it can be used for various purposes, including forecasting, impulse response analysis, and examining dynamic relationships between variables.

In R, we can use the **VAR ()** function from the "vars" package to estimate and analyze VAR models. This package provides various functions to fit VAR models, make forecasts, and conduct diagnostics on the model.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

It's important to note that VAR models assume linearity, stationarity, and other assumptions inherent in autoregressive models. Additionally, VAR models may suffer from the curse of dimensionality when dealing with a large number of variables.

2.2.3.6 *Exponential Smoothing State Space Models*

Exponential Smoothing State Space Models (ETS) is a family of time series models that are widely used for forecasting. ETS models provide a flexible framework for capturing and forecasting the underlying patterns and trends in a time series. They are based on the concept of exponentially weighted averages, where more recent observations are given higher weights.

The components of ETS models

Level (L)

The level component represents the underlying average or baseline of the time series. It is denoted by L(t) and captures the long-term average value of the series.

Trend (T)

The trend component captures the systematic change in the time series over time. It represents the slope or direction of the series. The trend is denoted by T(t) and can be additive or multiplicative, depending on the nature of the time series.

Seasonality (S)

The seasonality component represents the recurring patterns or cycles in the time series that occur at fixed intervals. It captures the regular fluctuations in the data. The seasonality component is denoted by S(t)and can be additive or multiplicative.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Component combinations

The ETS model combines these components using different combinations of additive and multiplicative effects. The most common forms of ETS models include:

ETS(AAA): Additive error, additive trend, and additive seasonality.

ETS(AAM): Additive error, additive trend, and multiplicative seasonality.

ETS(AMA): Additive error, multiplicative trend, and additive seasonality.

ETS(MAM): Multiplicative error, additive trend, and multiplicative seasonality.

ETS(MMM): Multiplicative error, multiplicative trend, and multiplicative seasonality.

Each component in the ETS model can be dampened or smoothed to control the level of volatility or variability in the forecasted values.

To estimate the parameters of the ETS model and make forecasts, various methods can be used, such as maximum likelihood estimation or state space modeling techniques.

In summary, Exponential Smoothing State Space Models (ETS) provide a flexible framework for modeling and forecasting time series data by capturing the level, trend, and seasonality components. They offer different combinations of additive and multiplicative effects to accommodate various types of time series patterns. ETS models are widely used in forecasting applications across different industries.

Package: forecast

Description: As mentioned earlier, the forecast package is commonly used for time series analysis. It provides functions for fitting Autoregressive Integrated Moving Average (ARIMA) models and Seasonal ARIMA (SARIMA) models. The **auto.arima ()** function in this package automatically selects the best ARIMA or SARIMA model based on model selection criteria, while the **Arima()** function allows us to specify the order of the model manually.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Visualization in R language

I generated this following graph in R to demonstrate a **time series forecasting** example using the "forecast" and "ggplot2" libraries. After initializing reproducibility with **set.seed(123)**, I created a synthetic time series using an ARIMA simulation with 100 data points. The generated data was then converted into a time series object. By employing the **auto.arima** function, I automatically selected an appropriate ARIMA model for the data. Subsequently, I made predictions for the next 20 time points using the **forecast** function. To visualize the results, I utilized the **autoplot** function from the "forecast" package, which creates a comprehensive plot showcasing the original data, the fitted ARIMA model, and the forecasted values, all presented with minimalistic styling and appropriately labeled axes (see Appendix 2.2.4 for more details).

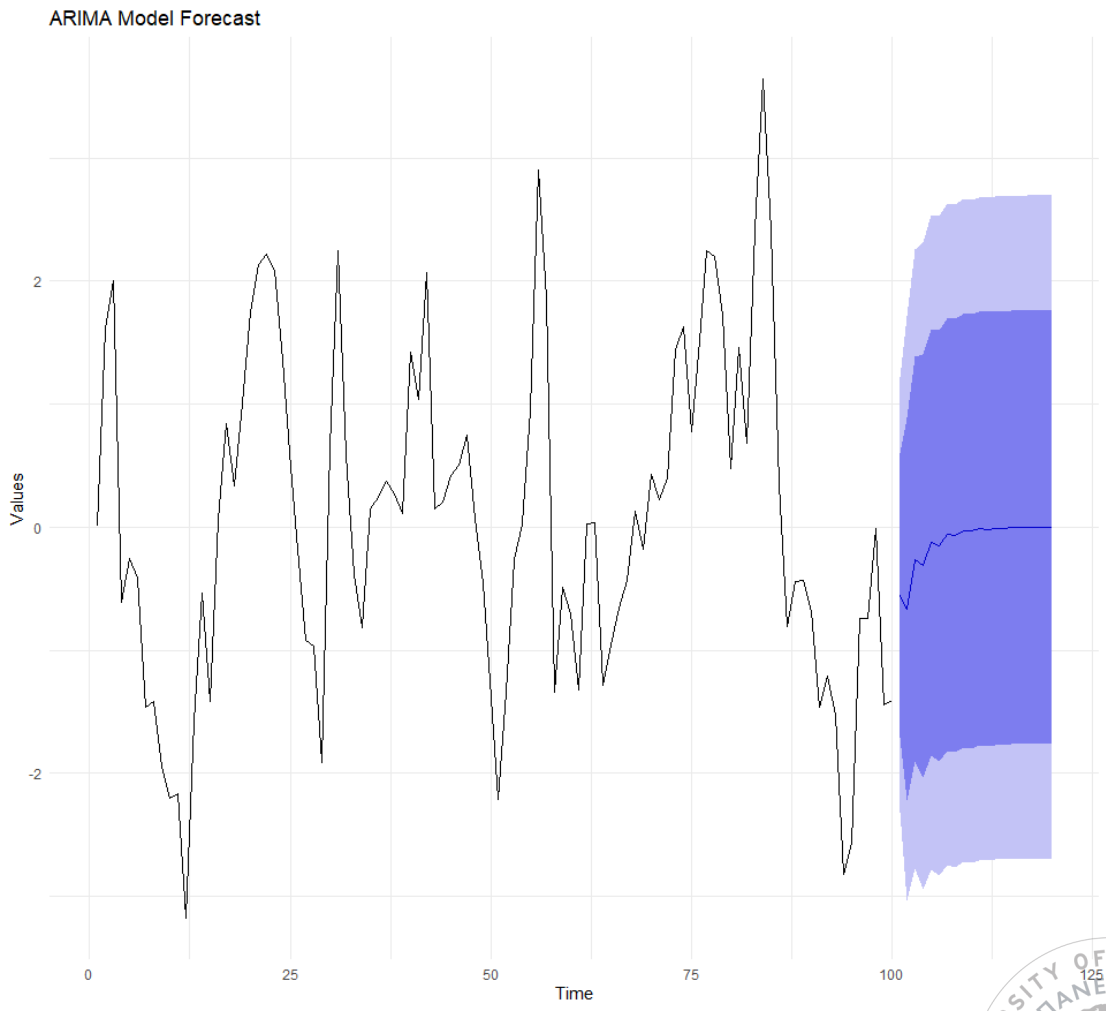

Figure 5. *Arima model forecast with simulated data graph*

Explanation of the simulated graph

Figure 5 is a visual representation of a time series forecast using an ARIMA model. The graph includes both the actual time series data and the predictions made by the ARIMA model.

1. Black Line (Training Data): The black line represents the actual time series data used to train the ARIMA model. The time series is generated using an ARIMA (1, 0, 1) model, with an autoregressive component of 0.6 and a moving average component of 0.3.
2. Blue Line (Forecast): The blue line represents the forecasted values from the fitted ARIMA model for 20 time periods ahead. These are the predicted values based on the pattern that the model has learned from the training data.
3. Shaded Region (Confidence Interval): The shaded region around the blue line represents the 80% and 95% prediction intervals for the forecast. This region quantifies the uncertainty associated with the forecast, indicating the range within which the future values are expected to lie with 80% and 95% confidence.
4. Axes: The x-axis represents time, and the y-axis represents the values of the time series.
5. Interpretation: The plot shows how the model anticipates the time series to evolve in the future, given the patterns it has observed in the data up to now. The width of the shaded region shows how certain (or uncertain) the model is about its forecast—the wider the region, the greater the uncertainty.

In summary, this plot shows the forecasted values and the associated uncertainty. This can be used to understand future patterns or trends based on historical time series data.

To conclude, all of the models that i have described until now are more appropriate for continuous time series data and maybe not suitable for count time series data.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.2.4 Decision Trees

Decision Trees

Decision trees are hierarchical structures that recursively split the data based on different features to make predictions (34). They can handle both categorical and numerical data and are useful for both regression and classification tasks.

They are tree-like structure with decision nodes and leaf nodes. Parameters: The splitting criteria at each decision node, such as feature thresholds and rules, determined by algorithms like Gini index or information gain. The parameters define the branching decisions in the tree structure.

Decision tree models create a tree-like structure of decisions and their possible outcomes. They are used for classification and regression problems. For instance, predicting whether a customer will purchase a product based on their age, gender, and browsing behavior.

Decision Tree

A decision tree is a hierarchical model that recursively partitions the input space into disjoint regions based on a set of decision rules. Each internal node represents a decision based on a specific feature or attribute, and each leaf node represents a predicted outcome or class label.

Splitting Criterion

The splitting criterion determines the attribute and the corresponding threshold value that optimally divides the data at each internal node. Common splitting criteria include Gini impurity and information gain.

Gini Impurity

Gini impurity measures the impurity or diversity of a node in terms of the probability of incorrectly classifying a randomly chosen element from that node. Mathematically, the Gini impurity for a node "m" with "K" classes is calculated as:

$$\mathbf{Gini(m) = 1 - \Sigma\ (p_i)^2}$$

where p_i is the probability of an element belonging to class "i" in node "m".

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Information Gain

Information gain is a measure of the reduction in entropy or uncertainty achieved by splitting a node based on a particular attribute. It quantifies how much information the attribute provides in terms of classifying the data. Mathematically, the information gain for a node "m" when splitting on attribute "A" is calculated as:

$$\text{Gain}(m, A) = H(m) - \Sigma\ (|D_v| / |D|) * H(D_v)$$

where H(m) is the entropy of node "m", D_v is the subset of data at node "m" for which attribute "A" takes on value "v", $|D_v|$ is the number of elements in D_v, |D| is the total number of elements at node "m", and $H(D_v)$ is the entropy of D_v.

Pruning

Decision trees tend to overfit the training data. Pruning is a technique used to prevent overfitting by reducing the complexity of the tree. It involves removing or collapsing nodes that do not contribute significantly to improving the generalization performance of the tree.

Prediction

To make predictions using a decision tree, we traverse the tree from the root node to a leaf node based on the values of the input features. At each internal node, we evaluate the decision rule based on the corresponding attribute and threshold value. We follow the appropriate branch based on the outcome of the decision until we reach a leaf node, which provides the predicted outcome or class label.

Example 1

Customer Churn Prediction Telecom companies can use decision trees to predict customer churn. By considering factors such as call duration, service usage patterns, billing complaints, and customer demographics, a decision tree model can identify key predictors of churn. This helps in devising targeted retention strategies to reduce customer attrition.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Example 2

Identifying High-Risk Patients for Infectious Diseases Decision trees can be used to identify high-risk patients for infectious diseases such as tuberculosis or hepatitis. By considering factors such as demographic information, medical history, and lifestyle choices, a decision tree model can help prioritize screening and preventive interventions for individuals most susceptible to infection

R Package: rpart

The rpart package provides functions to build and visualize decision trees using the Recursive Partitioning and Regression Trees (rpart) algorithm. It includes the rpart () function for fitting decision trees and the plot() function to visualize the tree structure.

rpart package for creating decision trees, and **rpart.plot** package for visualizing decision trees

Visualization in r language

Here is a simulated graph created in R language, which showcases an example of creating and visualizing a decision tree using the "rpart" and "rpart.plot" libraries. After loading the necessary packages, a sample dataset using the well-known Iris dataset is created. Using the **rpart** function, a decision tree model is trained to predict the "Species" variable based on the other features present in the dataset. To enhance the visualization, a color palette is defined for the nodes of the tree. The decision tree is then plotted with customized settings, including specific colors for the nodes, a shadow effect, and other visual parameters. The resulting plot provides a clear representation of the decision tree structure, allowing viewers to comprehend the branching decisions made by the model (see Appendix 2.2.5 for more details).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

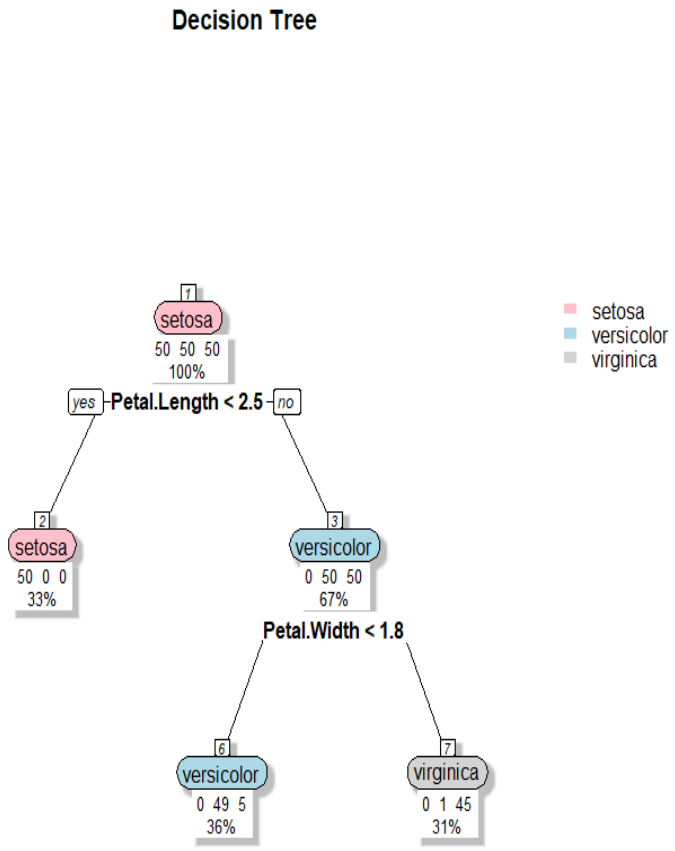

Figure 6. *Decision tree with simulated data graph*

Explanation of the simulated graph

Figure 6 is a visual representation of a decision tree model trained on the Iris dataset, which is used commonly for illustrating multiclass classification problems.

1. Nodes (Rectangles): Each node represents a decision based on one of the features (Sepal.Length, Sepal.Width, Petal.Length, or Petal.Width). Each decision splits the data into two subsets according to the condition specified in the node.

The top node is the root of the tree where the whole dataset starts. The nodes at the bottom of the tree are called leaf nodes and they represent the final decision, i.e., the predicted class (in this case, one of the iris species: setosa, versicolor, or virginica).

Each node contains the following information:

- The splitting rule: For example, "Petal.Length < 2.45".
- The predicted class: The majority class of the instances in that node.
- The percentage of instances from the total dataset in that node.
- The distribution of instances among different classes.

2. Edges (Lines Connecting Nodes): Edges represent the outcome of the test at the starting node. The left branch is the "true" branch and the right one is the "false" branch regarding the condition.
3. Colors: The color of each node indicates the predicted class, with different colors for setosa, versicolor, and virginica. The color palette is defined by the box.palette parameter.
4. Axes: The tree is structured vertically with the root node at the top and leaf nodes at the bottom.
5. Interpretation: The decision tree provides a flowchart-like structure to classify new instances. Starting from the root node, we follow the decision rules down the tree until we reach a leaf node, which give us the final prediction.

In summary, this decision tree plot provides a visualization of the decision rules generated by the model, offering an easily interpretable way to understand the model's predictions. It shows the process the model would follow to predict the species of a new iris sample based on its sepal and petal dimensions.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.2.5 Random Forests

Random Forests

Random forests are an ensemble learning method that combines multiple decision trees to make predictions (35). They improve the accuracy and robustness of predictions by reducing overfitting and incorporating multiple perspectives from different trees.

Random forests are also an ensemble learning technique that combines multiple decision trees to make predictions. They are used for classification and regression tasks and can handle large datasets with complex relationships. It improves upon the individual decision tree's performance by reducing overfitting and increasing robustness.

Bootstrap Aggregation (Bagging)

Random Forest employs a technique called bootstrap aggregation (or bagging) to create diverse subsets of the training data for each decision tree. Each subset is obtained by randomly sampling the data with replacement, resulting in multiple bootstrap samples.

Random Feature Selection

In addition to using bootstrap samples, Random Forest introduces random feature selection. At each node of a decision tree, instead of considering all features, only a random subset of features is considered for splitting. This random selection of features helps to introduce diversity among the decision trees.

Voting for Predictions

For classification tasks, Random Forest combines the predictions of the individual decision trees using majority voting. Each tree "votes" for a class label, and the class with the most votes is selected as the final prediction. For regression tasks, the average prediction of all decision trees is taken as the final prediction.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Out-of-Bag Error Estimation

Random Forest utilizes the out-of-bag (OOB) samples, which are the data points not included in the bootstrap sample of each tree, to estimate the model's performance. These OOB samples are used to calculate the OOB error, which serves as an estimate of the model's generalization error.

Mathematical Formulation

Let X be the input data matrix of size n x p, where n is the number of samples and p is the number of features. The Random Forest algorithm can be summarized as follows:

a. For each decision tree t = 1 to T:

Create a bootstrap sample *Xt* of size n by sampling with replacement from X.

Randomly select a subset of features *Ft* of size m (m << p).

Construct a decision tree using *Xt* and *Ft*.

b. For a new input sample x:

Obtain predictions from all T decision trees.

For classification: Select the class with the majority vote as the final prediction.

For regression: Take the average of all predictions as the final prediction.

Random Forest combines the predictions from multiple decision trees, leveraging the diversity and robustness of the ensemble to make accurate predictions.

Example 1

Disease Diagnosis In the field of healthcare, random forests can be used for disease diagnosis. By analyzing patient data, including medical history, symptoms, lab results, and demographic information, a random forest model can predict the likelihood of a patient having a specific disease. This aids doctors in making accurate diagnoses and prescribing appropriate treatments.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Example 2

Dengue Fever Risk Prediction Random forests can be employed to predict the risk of dengue fever outbreaks in specific regions. By analyzing historical climate data, mosquito population dynamics, and epidemiological factors, a random forest model can assess the likelihood of dengue transmission, helping public health agencies allocate resources for mosquito control and public awareness campaigns.

Random Forests Predictive Model

Ensemble of multiple decision trees. Parameters: The collection of decision trees, each with its own set of parameters (splitting criteria and rules). The random forest model combines the predictions from multiple trees to make the final prediction.

R Package: randomForest

The randomForest package implements the random forest algorithm for building ensemble models. It offers the **randomForest ()** function to train a random forest model and the **predict()** function to make predictions using the trained model. The package also provides options for feature importance assessment and tuning the random forest parameters.

randomForest package for creating and analyzing random forest models.

Visualization in r language

I developed a subsequent code in R, exemplifying the utilization of the "randomForest" and "ggplot2" libraries to construct and visualize a Random Forest regression model. After loading the essential libraries, I established reproducibility using **set.seed(123)**. Independent variables, 'x1' and 'x2', were generated with random values in the range of -10 to 10. A dependent variable 'y' was created as a linear combination of 'x1', 'x2', and some added noise. The data was structured into a dataframe for analysis. By employing the **randomForest** function, a Random Forest regression model was trained to predict 'y' based on 'x1' and 'x2'. Subsequently, predictions were made using

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

the model on the same dataset, and the results were incorporated into the dataframe.

To visualize the outcomes, a scatter plot was crafted using "ggplot2". Actual 'y' values were depicted in blue, while predicted 'y' values were plotted in red. The legend was provided to indicate the correspondence between colors and actual versus predicted data. The plot title, x-axis label, y-axis label, and legend title were configured for clarity. Overall, the plot illustrates the fitted Random Forest model's predictions against the actual data points, facilitating a visual assessment of the model's performance (see Appendix 2.2.7 for further insights).

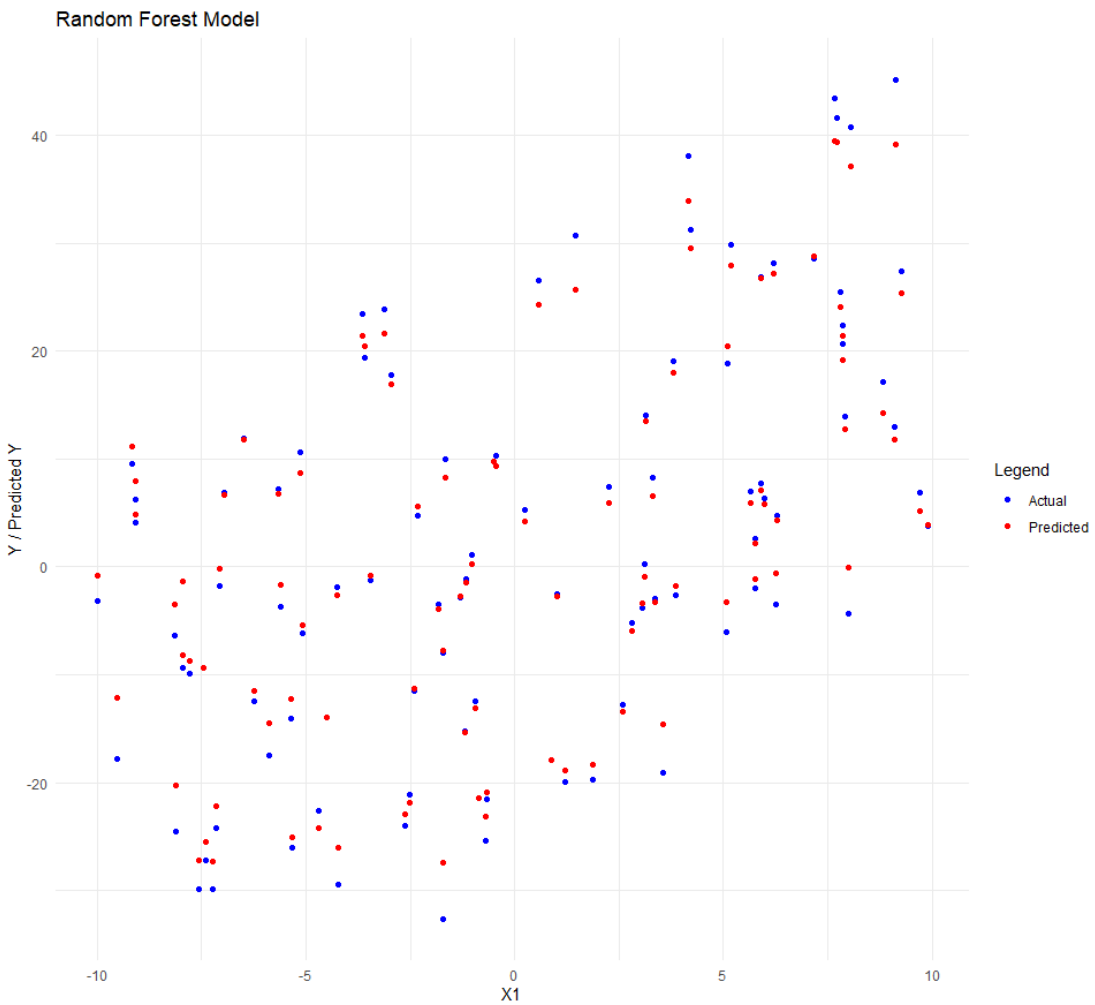

Figure 7. *Random forest with simulated data graph*

Explanation of the simulated graph

For a more analytical explanation, in this script, i have generated two independent variables x1 and x2 and a dependent variable y which is a linear combination of x1 and x2 with some normally distributed noise. Then we fit a Random Forest model to predict y from x1 and x2 and make predictions.

The resulting plot displays actual y values (in blue) and predicted y values (in red) against x1. Note that in the presence of multiple independent variables, we can only visualize the relationship between the dependent variable and one independent variable at a time, or we need to resort to more complex visualization methods suitable for higher dimensions. The color of the points indicates whether the values are actual (blue) or predicted (red).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.2.6 Support Vector Machines (Svm)

SVM is a supervised learning model used for both regression and classification. It separates data points into different classes by finding an optimal hyperplane that maximizes the margin between the classes.

SVM is a machine learning algorithm used for classification and regression tasks. It finds the optimal hyperplane to separate different classes or predict continuous values. For example, classifying emails as spam or non-spam based on various features (37,38).

Example 1

Email Spam Classification SVM can be used for email spam classification. By analyzing email content, metadata, and sender information, an SVM model can learn to distinguish between spam and legitimate emails. This helps in automatically filtering out unwanted spam messages, improving email user experience.

Example 2

Predicting Antibiotic Resistance SVM can be applied to predict antibiotic resistance patterns for specific bacteria strains. By analyzing genomic data, environmental factors, and previous antibiotic resistance trends, an SVM model can help identify which antibiotics are more likely to be effective in treating infections caused by certain bacterial strains.

The Support Vector Machine (SVM) algorithm aims to find a hyperplane with the largest margin between classes in a binary classification problem. The hyperplane can be represented by the equation:

$$\mathbf{w}^T\mathbf{x} + \mathbf{b} = \mathbf{0}$$

where:

w is a vector orthogonal (perpendicular) to the hyperplane, which determines its orientation.

x is the input vector representing the features of a data point.

b is the bias term that shifts the hyperplane away from the origin.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

In order to classify a new data point, the algorithm calculates the sign of the equation $\mathbf{w^T x + b}$. If the result is positive, the point belongs to one class, and if it's negative, it belongs to the other class.

The SVM algorithm can handle both linearly separable and non-linearly separable data by employing the kernel trick. This technique maps the original feature space into a higher-dimensional space, where the data might be linearly separable.

A kernel function, denoted as **K(x, y)**, is used to compute the similarity or distance between two data points in the transformed space. Common kernel functions include:

Linear Kernel: $\mathbf{K(x, y) = x^T y}$ • Polynomial Kernel: $\mathbf{K(x, y) = (x^T y + c)^d}$ • Radial basis function (Gaussian) Kernel: $\mathbf{K(x, y) = \exp(-\gamma\|x - y\|^2)}$

In SVM, the objective is to find the hyperplane that not only separates the data but also maximizes the margin or distance between the support vectors, which are the data points closest to the decision boundary.

To manage scenarios where the data is not perfectly linearly separable, SVM introduces slack variables (ξ) and a regularization parameter (C). The slack variables allow some misclassifications or margin violations, while the regularization parameter controls the trade-off between achieving a larger margin and reducing misclassifications.

The SVM algorithm solves an optimization problem to find the optimal hyperplane by minimizing the objective function:

$$\mathbf{0.5 * \|w\|^2 + C\sum\xi}$$

subject to the constraints:

$\mathbf{y_i(w^T x_i + b) \geq 1 - \xi_i}$ (for all i = 1, 2, ..., N) • $\mathbf{\xi_i \geq 0}$ (non-negativity constraint)

where: • $\mathbf{\|w\|^2}$ represents the squared Euclidean norm of the weight vector w. • **C** is the regularization parameter. • $\mathbf{y_i}$ is the class label of the i-th training sample. • $\mathbf{x_i}$ is the feature vector of the i-th training sample. • $\mathbf{\xi_i}$ are the slack variables.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

In summary, the Support Vector Machine (SVM) is a machine learning algorithm that finds an optimal hyperplane to separate data into two classes or predict continuous values. It utilizes a kernel function to transform the data into a higher-dimensional space and can handle both linearly separable and non-linearly separable data. The algorithm aims to maximize the margin between the classes while controlling the trade-off between maximizing the margin and minimizing misclassifications using the regularization parameters.

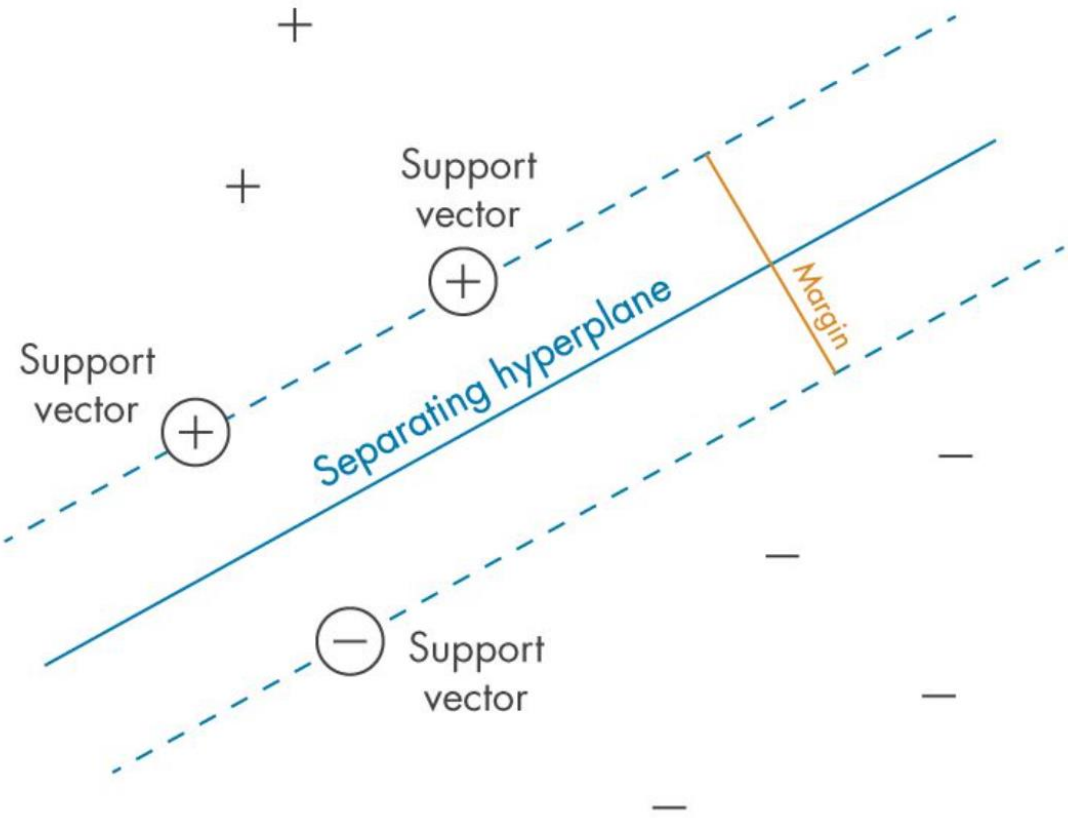

Figure 8. SVM Separating hyperplane, support vectors and margin

Package: e1071

Description: The e1071 package offers an implementation of SVM in R. It provides the **svm ()** function to train an SVM model and the **predict ()** function for making predictions. The package supports various SVM formulations, such as C-classification, nu-classification, and regression. It also includes functions for tuning SVM parameters.

e1071 package provides the **svm ()** function for implementing SVM models, and **caret** package for additional functionalities.

Visualization in r language

Here is a formulation in R to showcase an example of employing the "e1071" and "ggplot2" libraries for constructing and visualizing a **Support Vector Regression (SVR) model**. After loading the necessary libraries, I ensured reproducibility by using **set.seed(123)**. An independent variable 'x' was generated with random values within the range of -10 to 10. The dependent variable 'y' was designed as a linear combination of 'x' with added noise. Subsequently, a dataframe was constructed to encapsulate the data.

An SVR model was fitted using the **svm** function, aiming to predict 'y' based on 'x'. Predictions were generated for the dataset using the trained model. To facilitate proper visualization, the data was reordered based on the 'x' values. Using "ggplot2", a scatter plot was created, where actual 'y' values were denoted in blue data points, and a red line represented the predicted 'y' values. A legend was included to clarify the mapping of colors to actual versus predicted data. The plot title, x-axis label, y-axis label, and legend title were defined to enhance clarity. The resultant plot provides a visual representation of the SVR model's predictions against the actual data points, aiding in evaluating the model's performance (see Appendix 2.2.8 for further insights).

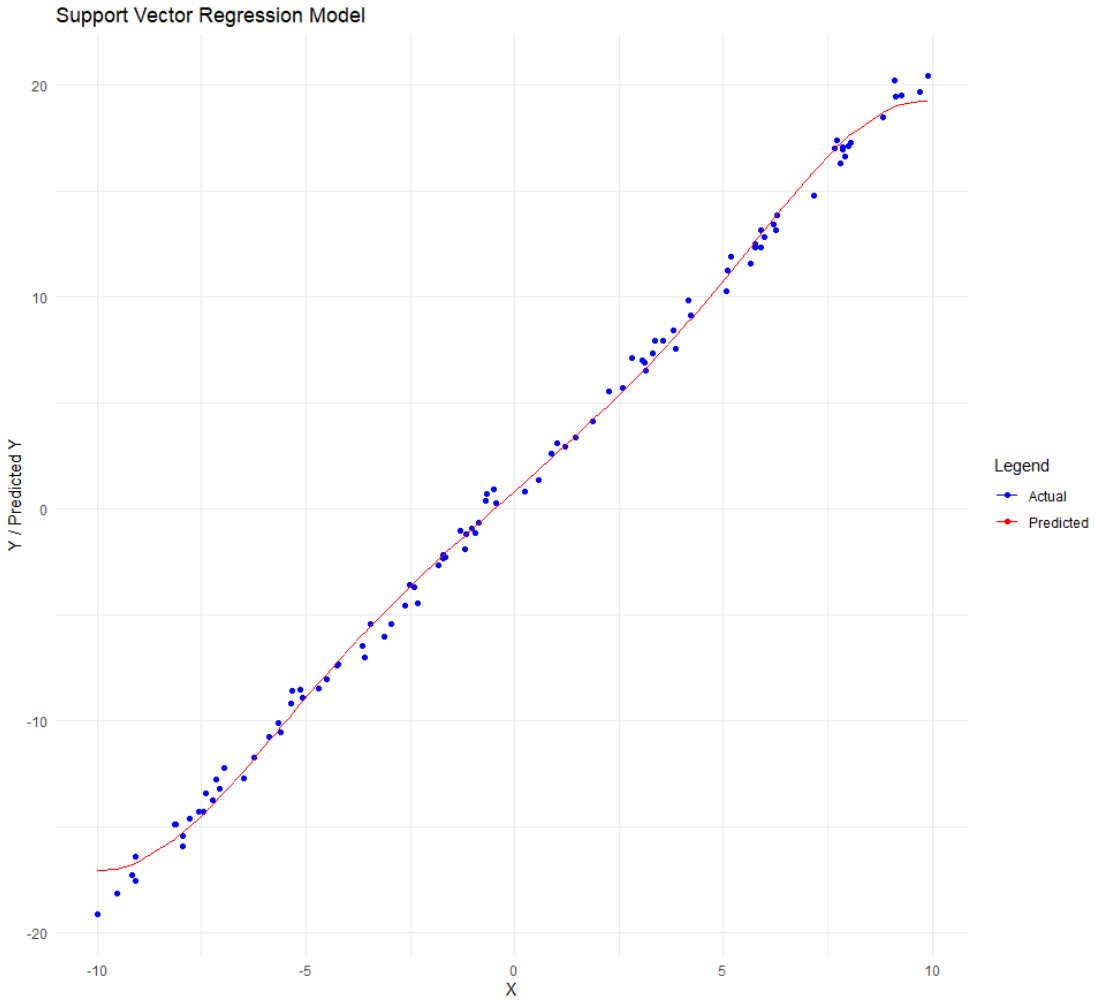

Figure 9. *SVM with simulated data graph*

Explanation of the simulated graph

For a better explanation in figure 9, I have generated an independent variable x and a dependent variable y which is a linear function of x with some normally distributed noise. Then we fit an SVR model to predict y from x and make predictions.

The resulting plot displays actual y values (blue points) and predicted y values (red line) against x. The color indicates whether the values are actual (blue) or predicted (red).

2.2.7 Naive Bayes

Naive Bayes

Naive Bayes models are probabilistic classifiers based on Bayes' theorem. They assume that the features are independent of each other and calculate the conditional probabilities of different classes given the input data (39).

Naive Bayes is a simple yet effective probabilistic classifier based on Bayes' theorem. It assumes that the features are conditionally independent given the class label, which is known as the "naive" assumption.

Bayes' Theorem

Bayes' theorem provides the foundation for Naive Bayes. It states that the posterior probability of a class given the observed features can be calculated using the prior probability of the class and the likelihood of the features given the class. The formula for Bayes' theorem is as follows:

$$P(y \mid x) = (P(x \mid y) * P(y)) / P(x)$$

where $P(y \mid x)$ is the posterior probability of class y given the features x, $P(x \mid y)$ is the likelihood of observing the features x given the class y, $P(y)$ is the prior probability of class y, and $P(x)$ is the probability of observing the features x.

Naive Bayes Assumption: The naive assumption in Naive Bayes assumes that the features are conditionally independent given the class label. This assumption simplifies the calculation of the likelihood term. Mathematically, it can be expressed as:

$$P(x \mid y) = P(x_1 \mid y) * P(x_2 \mid y) * \ldots * P(x_n \mid y)$$

where $x_1, x_2, \ldots, x_n$ represent the individual features.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Maximum a Posteriori (MAP) Decision Rule

Naive Bayes uses the MAP decision rule to assign a class label to a given instance. It selects the class that maximizes the posterior probability. Mathematically, the MAP decision rule can be written as:

$$y = \text{argmax } P(y \mid x) = \text{argmax } (P(x \mid y) * P(y))$$

where $\hat{y}$ is the predicted class label.

Estimation of Class and Feature Probabilities

To apply Naive Bayes, the class probabilities P(y) and the feature probabilities P(x | y) need to be estimated from the training data. Common estimation techniques include the maximum likelihood estimation (MLE) or applying smoothing methods like Laplace smoothing (add-one smoothing).

Naive Bayes is known for its simplicity, scalability, and efficiency. While it makes a strong assumption of feature independence, it can still perform well in practice, especially in text classification and other similar tasks.

Example

Naive Bayes can be employed in medical diagnosis applications, where patient symptoms and test results are used to predict the likelihood of different diseases or conditions. The features could include various medical indicators or test results, and the classes could represent different diseases. Naive Bayes estimates the probabilities of these features given each disease and predicts the most probable disease based on the observed symptoms.

Visualization in r language

Here is a composed example in R that serves as an illustration of utilizing the "e1071" and "ggplot2" libraries to establish and visualize a **Naive Bayes classification model**. After loading the essential libraries, I initiated reproducibility with **set.seed(123)**. An independent variable 'x' was generated with random values spanning from -10 to 10. Subsequently, the dependent variable 'y' was formulated based on a condition: when 'x' is greater than 0, 'y' is labeled as "positive," otherwise, it is labeled as "negative."

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

To organize the data, I created a dataframe that comprises the 'x' values and converted 'y' into a factor for classification purposes. Employing the **naiveBayes** function, I constructed a Naive Bayes classification model with 'x' predicting 'y'. Predictions were generated for the same dataset using the trained model. For visualization, the data was sorted based on 'x'.

Using "ggplot2", a scatter plot was generated where 'y' values were distinguished by colors, and data points were displayed with varying transparency and size. To visualize the predicted classes, I employed "rug" plots, with blue indicating the predicted "positive" class and red representing the predicted "negative" class. The plot title, x-axis label, y-axis label, and color legend title were set to enhance clarity. The resulting plot presents an intuitive visual representation of the Naive Bayes classification model's performance, with the rug plots providing insight into the model's classification decisions based on 'x' values (see Appendix 2.2.9 for further insights).

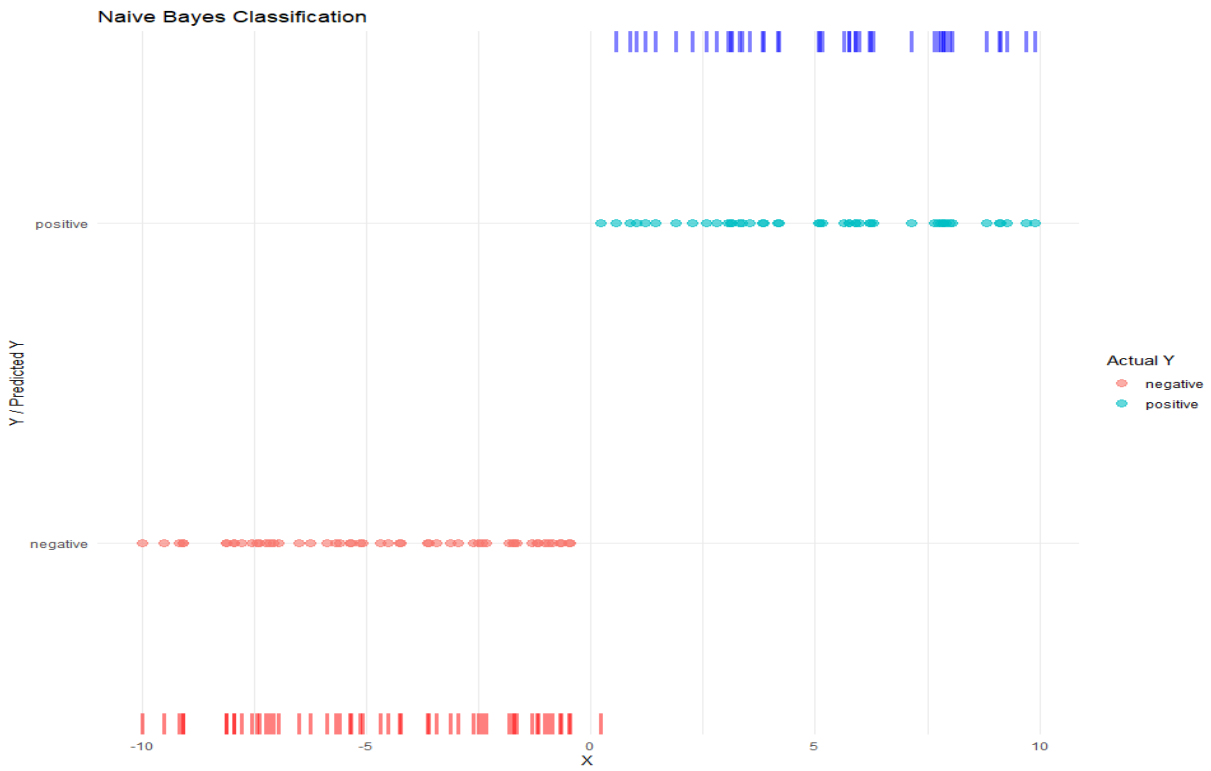

Figure 10. *Naïve bayes classification with simulated data graph*

Explanation of the simulated graph

In this script, i have generated an independent variable x and a dependent variable y which is "positive" if x is greater than 0 and "negative" otherwise. Then i fit a Naive Bayes model to predict y from x and make predictions.

The resulting plot displays actual y values (points) and predicted y values (rugs at the top and bottom of the plot) against x. Blue indicates "positive" and red indicates "negative". Points and rugs of the same color denote correct predictions, while different colors denote misclassifications.

2.2.8 Neural Networks

Neural Networks

Neural networks are a type of machine learning model inspired by the structure of the human brain (40,41). They consist of interconnected layers of nodes (neurons) and are capable of learning complex patterns and relationships in the data.

Neural networks are complex models inspired by the human brain's structure. They can learn intricate patterns and relationships in the data. Examples include image recognition, natural language processing, and speech recognition.

Example 1

Image Recognition Neural networks have been widely applied to image recognition tasks. For instance, in autonomous vehicles, neural networks can analyze camera footage and identify objects such as pedestrians, traffic signs, and vehicles. This enables the vehicle's AI system to make real-time decisions based on the identified objects in the environment.

Example 2

Early Detection of Infectious Disease Outbreaks Neural networks can be utilized to detect early signs of infectious disease outbreaks. By analyzing diverse data sources such as social media feeds, news reports, and clinical data, a neural network model can identify patterns and signals that indicate the emergence of a potential outbreak. This aids in timely response and containment measures.

Model

Complex interconnected network of artificial neurons. Parameters: Weights and biases associated with the connections between neurons in the network. These parameters are learned through training using optimization algorithms like gradient descent or backpropagation.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Neural Network Overview

Neural networks are powerful machine learning models inspired by the structure and functioning of the human brain. They consist of interconnected nodes, called neurons, organized in layers. Neural networks are used for tasks such as classification, regression, and pattern recognition.

Neuron Computation

The computation in a neuron involves two main steps: the linear transformation and the activation function. Given the input vector x, the linear transformation computes the weighted sum of the inputs, which is then passed through an activation function to produce the neuron's output. Mathematically, this can be expressed as:

$$z = w^T x + b \quad a = g(z)$$

where w represents the weight vector, b is the bias term, g is the activation function, z is the weighted sum of inputs, and a is the neuron's output.

Feedforward Propagation

In a neural network, information flows from the input layer through the hidden layers to the output layer. This process is called feedforward propagation. Each layer performs the neuron computation described above, using the outputs from the previous layer as inputs.

Activation Functions

Activation functions introduce non-linearities into the neural network, allowing it to learn complex patterns and relationships. Common activation functions include the sigmoid, tanh, and ReLU (Rectified Linear Unit) functions.

Backpropagation

Neural networks are trained using a technique called backpropagation, which adjusts the weights and biases based on the error between the predicted outputs and the true outputs. Backpropagation calculates the gradient of the error with

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

respect to the network parameters and updates the weights using gradient descent optimization.

Loss Function

A loss function is used to quantify the discrepancy between the predicted outputs and the true outputs. It serves as a measure of how well the network is performing. Common loss functions include the mean squared error (MSE), binary cross-entropy, and categorical cross-entropy.

Optimization Algorithms

Optimization algorithms, such as gradient descent, are used to minimize the loss function and update the network's parameters. Variants of gradient descent, such as stochastic gradient descent (SGD) and Adam, adjust the learning rate and update the parameters iteratively during training.

Training and Testing: Neural networks are typically trained on labeled training data, where the true outputs are known. The network learns to minimize the loss function by adjusting the weights and biases. After training, the network can be used for prediction on new, unseen data.

Package: neuralnet

Description: The neuralnet package allows us to create and train neural network models. It provides the **neuralnet ()** function to build a neural network and the **compute ()** function to make predictions using the trained model. The package supports both feedforward and recurrent neural networks.

Neural Networks: Packages: **nnet** package for implementing neural networks, and **caret** package for additional functionalities and model training.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.2.9 Gradient Boosting Models

Gradient Boosting Models

Gradient boosting models, such as XGBoost and LightGBM, iteratively build an ensemble of weak learners (typically decision trees) to make predictions. They optimize the loss function by iteratively minimizing errors and improving prediction accuracy (42,43).

Gradient Boosting Models, specifically Gradient Boosting Machines (GBMs), are powerful machine learning algorithms used for both regression and classification tasks. GBMs belong to the ensemble learning methods and work by combining multiple weak learners, typically decision trees, to create a strong predictive model.

The main idea behind gradient boosting is to iteratively train new models that are focused on correcting the mistakes of the previous models. This iterative process helps improve the overall predictive performance of the model. Let's explore the mathematical formulation and key concepts of Gradient Boosting Models:

Weak Learners

Gradient Boosting Models use weak learners as base models. In most cases, decision trees are employed as weak learners due to their ability to capture complex interactions and non-linear relationships. Mathematically, if we have T weak learners, our final model $F(x)$ can be represented as a sum of these learners:

$$\boldsymbol{F(x) = \sum_{t=1}^{T} f_t(x)}$$

where $\boldsymbol{f_t(x)}$ is the **t-th** weak learner.

Objective Function

The objective of Gradient Boosting is to minimize a loss function $\boldsymbol{L(y, F(x))}$ by iteratively adding new weak learners to the ensemble. Here, y is the true output and $\boldsymbol{F(x)}$ is the predicted output. The choice of loss function depends on the task at hand. For example, for regression tasks, mean squared error (MSE) or

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

mean absolute error (MAE) can be used, while for classification tasks, log loss or exponential loss may be employed.

Ensemble Creation

Gradient Boosting builds an ensemble of weak learners by adding models sequentially. Each new model is fitted to the residuals or errors of the previous models. This sequential process is guided by gradients, hence the name "gradient" in Gradient Boosting. In each iteration t, a new weak learner $\boldsymbol{f_t(x)}$ is fit to the negative gradient of the loss function:

$$\boldsymbol{f_t(x) = \mathbf{argmin}_f \sum_{i=1}^{N} L\big(y_i, F_{t-1}(x_i) + f(x_i)\big)}$$

Learning Rate

A learning rate (or shrinkage parameter) denoted by η is introduced to control the contribution of each weak learner to the ensemble. A smaller learning rate makes the model more robust by reducing the impact of each individual tree, but it also increases the number of iterations needed for convergence.

Prediction

To make predictions, the ensemble of weak learners is combined. Each weak learner contributes a weighted prediction based on its importance in the ensemble. The final prediction is obtained by summing the individual predictions of all weak learners.

$$\boldsymbol{F(x) = \sum_{t=1}^{T} \eta f_t(x)}$$

Regularization

To prevent overfitting, Gradient Boosting models can incorporate regularization techniques such as subsampling (stochastic gradient boosting), which randomly selects a subset of data at each iteration, or by limiting the depth or complexity of the weak learners.

Summary

In summary, Gradient Boosting Models are ensemble learning methods that combine weak learners, often decision trees, to create a strong predictive model. The models are trained iteratively by minimizing a loss function, with

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

each new model focused on correcting the mistakes of the previous models. The learning rate controls the contribution of each weak learner, and regularization techniques can be employed to prevent overfitting. The final prediction is obtained by combining the weighted predictions of all weak learners in the ensemble.

Example

Gradient Boosting can be used to predict housing prices based on various features such as location, size, number of rooms, and other relevant factors. By training a gradient boosting model on historical housing data with corresponding prices, the model can learn the patterns and relationships between the features and the target variable. This trained model can then be used to make predictions on new, unseen housing data to estimate their prices.

Gradient Boosting Overview

Gradient Boosting is an ensemble learning method that combines multiple weak prediction models, typically decision trees, to create a strong predictive model. It builds the model in an iterative manner, where each subsequent model focuses on correcting the mistakes made by the previous models.

Loss Function: Gradient Boosting starts by defining a loss function that measures the difference between the predicted values and the actual values. The choice of loss function depends on the problem at hand. For example, in regression tasks, the mean squared error (MSE) or the mean absolute error (MAE) can be used.

Base Model Construction: The first base model, often a decision tree, is trained on the input features and their corresponding target values. This model serves as the initial prediction.

Gradient Calculation: The gradient of the loss function with respect to the predictions of the previous model is calculated. This gradient represents the direction and magnitude of the error made by the previous model.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Model Training: A new base model is trained to minimize the loss function by fitting it to the negative gradient of the loss function. The goal is to find the best split points in the feature space to minimize the error.

Ensemble Creation: The newly trained model is added to the ensemble by combining it with the previous models. The predictions from each model are weighted based on their contribution to the overall error reduction.

Iterative Process: Steps 4-6 are repeated for a specified number of iterations, with each new model focusing on correcting the errors made by the previous models. The learning process is guided by the gradient of the loss function.

Final Prediction: The final prediction is obtained by summing the predictions from all the models in the ensemble. The weights assigned to each model's prediction reflect their importance based on their performance in reducing the loss function.

Gradient Boosting is a powerful technique that can be applied to both regression and classification problems. By iteratively improving upon the weaknesses of the previous models, it can create a strong ensemble model that provides highly accurate predictions.

Package: xgboost

The xgboost package is an efficient implementation of gradient boosting. It offers the **xgboost()** function to train a gradient boosting model and the **predict()** function to make predictions. The package supports various customization options, such as controlling boosting parameters, handling missing values, and performing cross-validation.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Visualization in r language

Here i provided an example of using the "gbm" and "ggplot2" libraries to formulate and visualize a **Gradient Boosting Model**. After loading the necessary libraries, I established reproducibility using **set.seed(123)**. An independent variable 'x' was generated with random values within the range of -10 to 10. Additionally, a dependent variable 'y' was structured as a linear combination of 'x' with added noise. The data was organized into a dataframe for analysis.

A Gradient Boosting Model was trained utilizing the **gbm** function. It was designed to predict 'y' based on 'x', with Gaussian distribution assumptions. The model was configured with 100 trees and an interaction depth of 4. Subsequently, predictions were generated for the dataset using the trained model. To ensure compatibility, the predicted values were converted to numeric format.

The data was sorted based on 'x' values for proper visualization. Using "ggplot2", a scatter plot was constructed. Actual 'y' values were represented by blue data points, while the red line illustrated the predicted 'y' values. A color legend was included to indicate the mapping of colors to actual versus predicted data. The plot title, x-axis label, y-axis label, and legend title were specified for clarity. The resulting plot visually illustrates the predictions made by the Gradient Boosting Model against the actual data points, offering insights into the model's performance (see Appendix 2.2.11 for further insights).

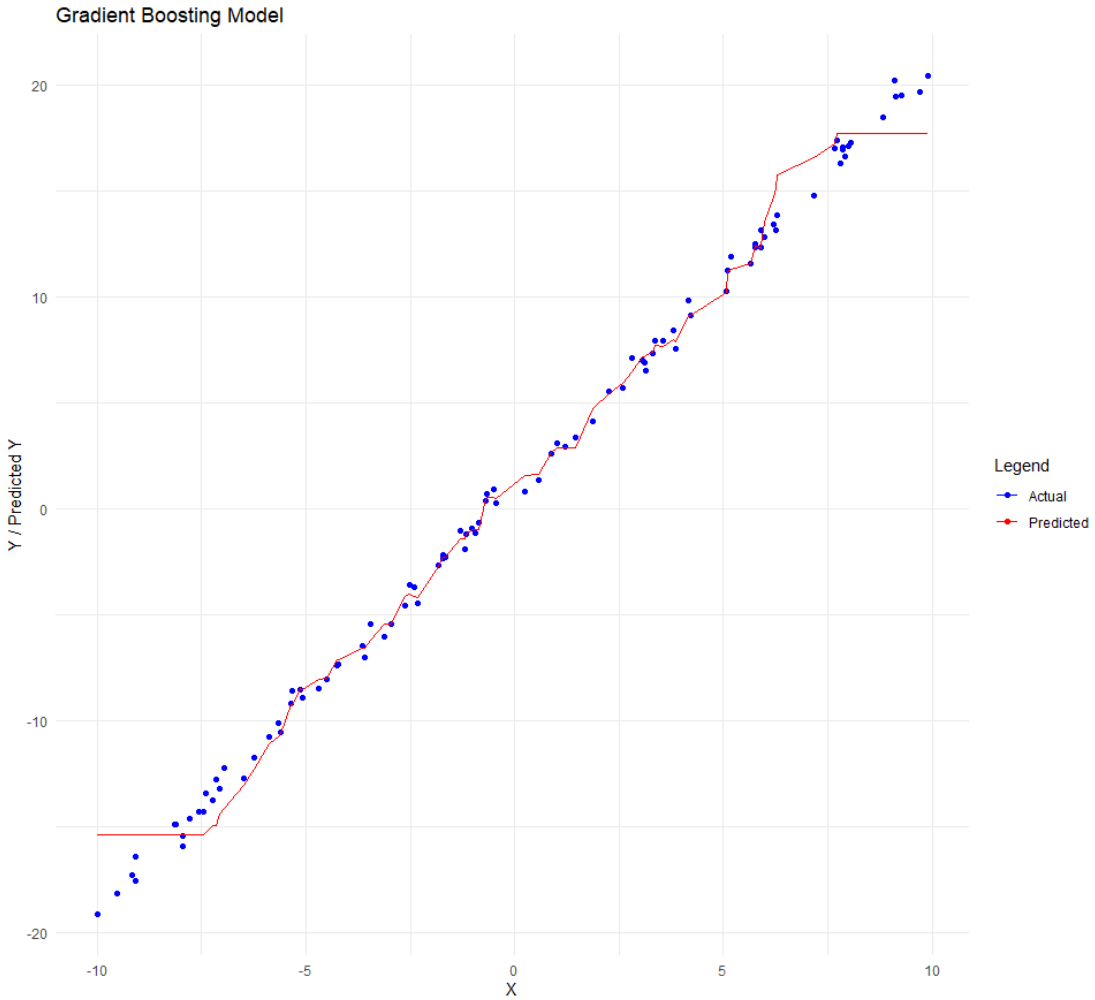

Figure 11. *Gradient boosting model with simulated data graph*

Explanation of the simulated graph

In figure 11, i have generated an independent variable x and a dependent variable y which is a linear function of x with some normally distributed noise. Then we fit a Gradient Boosting Model to predict y from x and make predictions.

The resulting plot displays actual y values (blue points) and predicted y values (red line) against x. The color indicates whether the values are actual (blue) or predicted (red).

These models represent the functional form and the associated parameters used in each predictive statistical modeling technique. The parameter estimation

process involves fitting the model to the available data, allowing the model to make predictions or classifications based on new observations.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3 PREDICTING INFECTIOUS DISEASES: CHALLENGES AND SUITABLE MODELS

2.3.1 Challenges

When predicting and preventing infectious diseases, we may encounter several challenges. Additionally, while the models mentioned earlier can be powerful tools, they may face certain limitations in this context. Here are some challenges and potential problems we may encounter:

Limited or incomplete data

Obtaining comprehensive and high-quality data on infectious diseases can be challenging. Data may be incomplete, contain errors, or be limited in terms of geographic coverage or historical depth. Insufficient data may affect the accuracy and reliability of predictive models.

Rapidly evolving diseases

Infectious diseases can rapidly evolve, making it difficult to capture the full extent of their spread and impact. Models trained on historical data may struggle to adapt to sudden changes, new variants, or emerging diseases without sufficient up-to-date information.

Complex transmission dynamics

Infectious diseases often involve intricate transmission dynamics, influenced by factors such as population density, human behavior, environmental conditions, and intervention measures. Capturing these complexities in predictive models can be challenging, and oversimplification may lead to inaccurate predictions.

Uncertainty and variability

There is inherent uncertainty and variability associated with infectious diseases. Outbreaks can be influenced by numerous unpredictable factors, including the effectiveness of public health measures, vaccination rates, and individual behavior. Models may struggle to account for all these variables accurately.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Limited understanding of new diseases

When dealing with novel or emerging infectious diseases, there may be limited prior knowledge and data available. This can hinder the development of accurate predictive models as historical patterns may not exist or may not fully reflect the behavior of the new disease.

Model assumptions and limitations

Each predictive model has its own set of assumptions and limitations. Linear regression, for example, assumes a linear relationship between variables, which may not hold for complex disease dynamics. Logistic regression assumes the independence of observations, which may not be valid for diseases with clustering effects. Understanding the assumptions and limitations of the chosen models is crucial for accurate interpretation and decision-making. ***Also the most important fact is that infectious diseases data often appear as count time series data. So traditional time series models like ARIMA or SARIMA are not appropriate***.

Ethical and privacy considerations

When working with infectious disease data, privacy concerns and ethical considerations may arise. Data sharing, data anonymization, and maintaining patient confidentiality can present challenges. These considerations may impact the availability and accessibility of data for predictive modeling.

Overall

To address these challenges and potential problems, it is important to incorporate domain expertise, continually update models with the latest data, consider multiple modeling approaches, and validate predictions against real-time data. Collaborating with public health agencies, epidemiologists, and infectious disease experts can help ensure that the models are informed by the latest research and insights.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.2 Why some of the above methods may not be suitable or optimal.

While the mentioned models have been successfully applied in various prediction tasks, including healthcare, it's important to note that the suitability of these algorithms for infectious disease prevention or early detection depends on several factors. Here are some considerations for each mpdel:

Linear Regression

Linear regression is a statistical method that assumes a linear relationship between the input features and the target variable. This assumption may not be suitable for modeling infectious disease dynamics, which often involve non-linear and complex interactions. Factors like transmission rates, population dynamics, and environmental variables can exhibit non-linear relationships. Additionally, linear regression may not adequately address heteroscedasticity, a common issue in infectious disease data with varying levels of variability. Count data in disease modeling often contains zeros, which linear regression may struggle to handle effectively. Moreover, linear regression can extrapolate beyond the feasible range of count data, which may lead to inaccurate predictions in infectious disease dynamics..

Logistic Regression

Logistic regression is commonly used for binary classification problems, such as predicting the presence or absence of a disease, making it well-suited for scenarios with binary outcomes. It estimates the probability of an event occurring based on input features, which is particularly useful when dealing with such cases. However, infectious disease prediction often involves more complex scenarios, including multiple classes, such as different disease stages, and the analysis of time-series data. In these situations, logistic regression may not be the most suitable choice due to its limitations. These limitations include overdispersion, as infectious disease data often exhibit higher variability than logistic regression assumes. Additionally, logistic regression, like linear regression, relies on the assumption of linearity between input features and the log-odds of the outcome, which may not hold in cases of complex and non-linear infectious disease dynamics..

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Time Series Analysis

Time series analysis techniques, such as Autoregressive Integrated Moving Average (ARIMA) or Seasonal ARIMA (SARIMA), are suitable for modeling and forecasting time-dependent data. However, when applied to infectious disease data, it's important to acknowledge their limitations. These limitations include the assumption of linearity and stationarity, which may not fully capture the non-linear and non-stationary nature of infectious disease dynamics. Additionally, ARIMA models may struggle to account for the presence of seasonal patterns in disease data without proper preprocessing. While these techniques are valuable for historical trend analysis, their use in real-time prediction and prevention of infectious diseases may be constrained by these factors.

Decision Trees

Decision trees are versatile and can handle both classification and regression problems. However, they may not be the best choice for infectious disease prediction, as they tend to be prone to overfitting and may not capture complex relationships between features and disease outcomes effectively.

Random Forests

Random forests are an ensemble learning method that combines multiple decision trees to make predictions. While they can improve predictive accuracy compared to a single decision tree, they may still have limitations in capturing the complex dynamics and interdependencies in infectious disease prevention or early detection.

Neural Networks

Neural networks, including deep learning models, have shown promise in various healthcare applications. However, they often require large amounts of labeled data and computationally intensive training. Availability of such data and computational resources can be challenging in the context of infectious disease prevention, especially during the early stages of an outbreak.

Support Vector Machines (SVM)

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Support Vector Machines (SVMs are effective for binary classification tasks, but they may face challenges in handling multi-class classification or time-series-count data. In the context of infectious disease prediction, these challenges include their linearity assumption, the need for adaptation to multi-class scenarios, sensitivity to outliers, and the complexity of parameter tuning, which can pose difficulties in effectively capturing the complexities of infectious disease dynamics and multi-class scenarios.

Gradient Boosting

Gradient Boosting is a powerful ensemble learning method that can handle complex relationships and improve prediction accuracy. While it has been successfully applied in various domains, its application in infectious disease prevention or early detection may require careful consideration of the specific characteristics and dynamics of the disease.

Naive Bayes

Naive Bayes models are probabilistic classifiers that assume independence between features given the class variable. They have been widely used in text classification and spam filtering tasks but may also be applicable to infectious disease prediction. Naive Bayes models can handle multi-class classification and can be particularly useful when there are limited training data available. However, they make the strong assumption of feature independence, which may not hold in all cases. Despite this limitation, Naive Bayes models can provide fast and efficient predictions, making them suitable for certain infectious disease prediction scenarios where the assumption aligns with the data characteristics.

Overall

It's worth noting that infectious disease prediction and prevention involve a range of factors, including epidemiological models, population dynamics, environmental factors, and public health interventions. These aspects may require specialized approaches beyond the algorithms mentioned here. Domain-specific knowledge and collaboration with experts in epidemiology and public

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

health are crucial for effective infectious disease prediction and prevention strategies. ***Finally as noted before, the most important characteristic that makes these methods unsuitable is the fact that we deal with count time series data***.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.3 More suitable and broader categories of models for the forecast of infectious diseases.

When it comes to predicting, forecasting, and preventing infectious diseases, particularly pandemics, several predictive models and approaches can be valuable. Here are some of the most suitable models commonly used in this context:

2.3.3.1 *Compartment Models*

Compartmental models, such as the Susceptible-Infected-Recovered (SIR) model and the Susceptible-Exposed-Infected-Recovered (SEIR) model, as well as Agent-Based Models (ABMs), are valuable tools in understanding disease transmission dynamics, forecasting infection rates, and evaluating the impact of interventions. These models explicitly capture the intricate dynamics of disease spread within populations, enabling us to estimate crucial parameters like the reproduction number (R0), which represents the transmission rate, and assess the effectiveness of interventions. By simulating diverse scenarios and interventions, epidemiological models contribute to predicting the future trajectory of an outbreak, evaluating the efficacy of different control strategies, and informing decision-making in public health.

2.3.3.2 *Machine Learning Models*

Machine learning algorithms offer powerful tools for infectious disease prediction and prevention. Models such as Random Forests, Support Vector Machines (SVM), and Gradient Boosting Machines (GBM) can handle complex data and capture nonlinear relationships. These models can incorporate various features, including demographic data, environmental factors, and mobility patterns, to predict disease spread and identify high-risk areas. Additionally, deep learning models like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) can be employed for image analysis (e.g., detecting pathogens in medical images) or time series analysis (e.g., predicting disease progression).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.3.3 *Spatial Models and Spatiotemporal Models*

Spatial models are an essential approach to comprehending the spatial distribution and dissemination of infectious diseases. They utilize techniques like spatial autocorrelation, geostatistics, and spatial regression to identify hotspots, clustering patterns, and factors that contribute to disease spread. Geographic Information Systems (GIS) are instrumental in visualizing and analyzing spatial data, facilitating targeted interventions and resource allocation. These models specifically aim to unravel the spatial distribution and transmission of infectious diseases, taking into account geographic factors such as proximity, connectivity, and environmental conditions that influence disease transmission. By incorporating spatial dependencies and patterns, spatial models can effectively identify disease hotspots, evaluate the impact of local factors on disease risk, and inform precise prevention strategies. This knowledge is crucial for optimizing resource deployment, implementing localized interventions, and curtailing disease transmission within specific areas. Spatiotemporal models include both space and time in the equation.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.3.4 *Bayesian Models*

Bayesian models offer valuable tools for integrating prior knowledge and updating predictions as new data becomes available. These models are particularly well-suited for infectious disease forecasting, as they enable the estimation of parameters and uncertainties. By incorporating expert knowledge, historical data, and real-time surveillance data, Bayesian models can generate probabilistic forecasts that inform decision-making. Their probabilistic framework makes Bayesian models ideal for predicting, forecasting, and preventing infectious diseases. These models effectively integrate diverse sources of information, such as expert opinions, historical data, and real-time observations, to estimate disease parameters, make predictions, and evaluate the effectiveness of interventions. Furthermore, Bayesian models provide uncertainty estimates, which are crucial for decision-making in situations characterized by uncertainty.

2.3.3.5 *Generalized Linear Models*

Generalized Linear Models (GLMs) encompass a wide range of models that extend linear regression to accommodate various types of dependent variables and error distributions. They are versatile tools for infectious disease prediction and prevention. GLMs can handle different types of response variables, including binary outcomes (e.g., presence or absence of infection), count data (e.g., number of cases), or time-to-event data (e.g., time until recovery or death). By incorporating covariates, GLMs provide valuable insights into the relationship between predictors and disease outcomes. Models such as logistic regression, Poisson regression, or survival analysis (e.g., Cox proportional hazards model) are examples of GLMs commonly used in infectious disease research. GLMs offer a flexible framework to model the associations between predictors and disease outcomes, making them suitable for infectious disease prediction and prevention. They facilitate the analysis of diverse response variables and allow for the estimation of regression coefficients, enabling the identification of significant predictors and the quantification of their effects on disease risk or progression. This knowledge supports risk assessment,

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

intervention planning, and the identification of population subgroups at higher risk.

2.3.3.6 *Agent-Based Models*

Agent-based models (ABMs) are computational models that simulate the behavior and interactions of individual agents within a system. In the context of infectious diseases, ABMs can be used to model the spread and dynamics of the disease at the individual level, considering factors such as agent characteristics, social networks, mobility patterns, and disease transmission mechanisms. ABMs provide a more detailed and granular representation of the population compared to other modeling approaches, allowing for the exploration of complex interactions and emergent behaviors.

Finally

It's important to note that the choice of predictive models depends on the specific characteristics of the infectious disease, the available data, and the research objectives. Combining multiple models and approaches, can enhance the accuracy and robustness of predictions, forecasts, and prevention strategies for infectious diseases.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.4 Compartment Models

2.3.4.1 *SIR*

Commonly used type of epidemiological model is the compartmental model, which divides the population into different compartments based on the disease status of individuals. Here, I will explain analytically the mathematical formulations of the **SIR** (Susceptible-Infectious-Recovered) model, which is a widely used compartmental model (44,45).

In the SIR model, the population is divided into three compartments:

Susceptible (S): Individuals who are susceptible to the disease can become infected.

Infectious (I): Individuals who are infected and can transmit the disease to susceptible individuals.

Recovered (R): Individuals who have recovered from the disease and acquired immunity.

The transition between these compartments is governed by a set of differential equations that describe the rates at which individuals move between compartments.

Let's denote:

N: Total population size.

S(t): Number of individuals susceptible to the disease at time t**.**

I(t): Number of infectious individuals at time t.

R(t): Number of individuals who have recovered from the disease at time t.

The SIR model assumes that the total population size remains constant **(N = S(t) + I(t) + R(t)).** The rate of change of individuals in each compartment is described by the following differential equations:

Rate of change of susceptible individuals

$$dS(t)/dt = -\beta * S(t) * I(t) / N$$

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

β represents the transmission rate, which indicates the probability of a susceptible individual becoming infected upon contact with an infectious individual.

S(t) * I(t) / N represents the number of effective contacts between susceptible and infectious individuals.

Rate of change of infectious individuals:

$$dI(t)/dt = \beta * S(t) * I(t) / N - \gamma * I(t)$$

The first term **β * S(t) * I(t) / N** represents the rate at which susceptible individuals become infected.

The second term **γ * I(t)** represents the rate at which infectious individuals recover from the disease and move into the recovered compartment.

γ is the recovery rate or the reciprocal of the average duration of infectiousness.

Rate of change of recovered individuals: $dR(t)/dt = \gamma * I(t)$

This equation indicates that the rate at which individuals recover from the disease is proportional to the number of infectious individuals.

By solving these differential equations numerically or analytically, we can track the changes in the number of susceptible, infectious, and recovered individuals over time. The SIR model provides insights into the dynamics of disease transmission, the impact of different transmission and recovery rates, and the effectiveness of interventions in reducing the number of infections.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.4.2 *SEIR*

The **SEIR** model introduces an additional compartment, the exposed (E) compartment, compared to the SIR model. This compartment represents individuals who have been exposed to the disease but are not yet infectious. The main difference in the mathematical formulation between the SIR and SEIR models lies in the additional equation that describes the dynamics of the exposed compartment (44,45).

For the SIR model, the differential equations describing the rate of change of each compartment are as follows:

Rate of change of susceptible individuals:

$$dS(t)/dt = -\beta * S(t) * I(t) / N$$

Rate of change of infectious individuals:

$$dI(t)/dt = \beta * S(t) * I(t) / N - \gamma * I(t)$$

Rate of change of recovered individuals:

$$dR(t)/dt = \gamma * I(t)$$

Here, the SIR model only considers two compartments: susceptible (S) and infectious (I), along with the recovered (R) compartment.

In contrast, the SEIR model includes an additional equation to describe the rate of change of the exposed compartment. The differential equations for the SEIR model are as follows:

Rate of change of susceptible individuals:

$$dS(t)/dt = -\beta * S(t) * I(t) / N$$

Rate of change of exposed individuals:

$$dE(t)/dt = \beta * S(t) * I(t) / N - \alpha * E(t)$$

Rate of change of infectious individuals:

$$dI(t)/dt = \alpha * E(t) - \gamma * I(t)$$

Rate of change of recovered individuals:

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

$dR(t)/dt = \gamma * I(t)$

In the SEIR model, the new equation represents the rate at which individuals in the exposed compartment transition to the infectious compartment. This transition is governed by the parameter α, which represents the reciprocal of the average duration of the latent period or the incubation period.

The SEIR model allows for a more realistic representation of disease dynamics by accounting for the time individuals spend in the exposed compartment before becoming infectious. This extension enables the model to capture the latency period observed in many infectious diseases, where individuals are infected but not yet capable of transmitting the disease.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.4.3 *SIRS*

The **SIRS** model is another extension of the basic SIR (Susceptible-Infectious-Recovered) model that allows for the possibility of individuals losing immunity and becoming susceptible again (44,45). The main difference between the SIRS and SIR models lies in the way the recovered individuals are treated.

In the SIR model, individuals who recover from the disease acquire lifelong immunity and do not transition back to the susceptible compartment. The differential equations for the SIR model are as follows:

Rate of change of susceptible individuals:

$$dS(t)/dt = -\beta * S(t) * I(t) / N$$

Rate of change of infectious individuals:

$$dI(t)/dt = \beta * S(t) * I(t) / N - \gamma * I(t)$$

Rate of change of recovered individuals:

$$dR(t)/dt = \gamma * I(t)$$

Here, once individuals recover (I -> R), they remain in the recovered compartment and do not transition back to being susceptible.

In contrast, the SIRS model allows for the possibility of individuals losing immunity and transitioning back to the susceptible compartment. The differential equations for the SIRS model are as follows:

Rate of change of susceptible individuals:

$$dS(t)/dt = -\beta * S(t) * I(t) / N + \omega * R(t)$$

Rate of change of infectious individuals:

$$dI(t)/dt = \beta * S(t) * I(t) / N - \gamma * I(t)$$

Rate of change of recovered individuals:

$$dR(t)/dt = \gamma * I(t) - \omega * R(t)$$

In the SIRS model, the new term $\omega * R(t)$ represents the rate at which recovered individuals lose their immunity and transition back to the susceptible

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

compartment. The parameter ω controls the rate of waning immunity and is typically a small value compared to the infectiousness rate (β) and recovery rate (γ).The SIRS model allows for a more realistic representation of infectious diseases where immunity is not lifelong, and individuals can become susceptible again after a certain period of time. This extension enables the model to capture scenarios where individuals may experience multiple infections or where immunity may wane over time, affecting the long-term dynamics of the disease.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.4.4 *Susceptible-Exposed-Infectious-Quarantined-Recovered*

Like the SIR model, the SEIQR model is a compartmental model used to describe the spread of a disease in a population. However, it adds two more compartments to the three used in the SIR model, namely the 'Exposed' and 'Quarantined' compartments, resulting in a total of five compartments:

Susceptible (S): Individuals who are susceptible to the disease but have not yet been infected.

Exposed (E): Individuals who have been infected but are not yet infectious. This represents the incubation period of the disease.

Infectious (I): Individuals who are infected and can spread the disease to others.

Quarantined (Q): Individuals who are known to be infected and have been isolated from the rest of the population. This compartment is important for modeling the impact of interventions such as quarantine measures.

Recovered (R): Individuals who have recovered from the disease and are now immune.

Assuming the total population size (N) is constant $(N = S(t) + E(t) + I(t) + Q(t) + R(t))$, the transitions between these compartments are represented by the following set of differential equations:

Rate of change of susceptible individuals:

$$dS/dt = -\beta * S(t) * I(t) / N$$

This equation is similar to the SIR model, where β is the transmission rate.

Rate of change of exposed individuals:

$$dE/dt = \beta * S(t) * I(t) / N - \sigma * E(t)$$

The first term $\beta * S(t) * I(t) / N$ represents the rate at which susceptible individuals become exposed.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

The second term **σ * E(t)** represents the rate at which exposed individuals become infectious. Here, σ is the rate at which the disease progresses from the exposed stage to the infectious stage.

Rate of change of infectious individuals:

$$dI/dt = \sigma * E(t) - (\gamma + \psi) * I(t)$$

The first term **σ * E(t)** represents the rate at which exposed individuals become infectious.

The second term $(\gamma + \psi) * I(t)$ represents the rate at which infectious individuals either recover or are quarantined. Here, γ is the recovery rate and ψ is the rate of quarantine.

Rate of change of quarantined individuals:

$$dQ/dt = \psi * I(t) - \gamma * Q(t)$$

The first term **ψ * I(t)** represents the rate at which infectious individuals are quarantined.

The second term $\gamma * Q(t)$ represents the rate at which quarantined individuals recover.

Rate of change of recovered individuals:

$$dR/dt = \gamma * I(t) + \gamma * Q(t)$$

This equation indicates the rate at which infectious and quarantined individuals recover from the disease.

The SEIQR model provides a more detailed and nuanced picture of disease spread compared to the SIR model. It is particularly useful for diseases with a significant incubation period and for evaluating the impact of quarantine measures.

As for the SEI model, it lacks the quarantined and recovered compartments, making it simpler but less detailed. It's typically used for diseases where recovery doesn't confer immunity or where the recovered compartment isn't necessary for the model's purposes.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Modeling Infectious Disease Spread (SIR Model): This R generated example employs the "deSolve" and "ggplot2" libraries to **simulate and visualize the spread of an infectious disease using the Susceptible-Infectious-Recovered (SIR) model**. The SIR model equations are defined in the **sir_model** function, accounting for the rate of infection, recovery, and population dynamics. Initial conditions and parameter values are specified, including the transmission rate (**beta**) and recovery rate (**gamma**). By solving the differential equations over a defined time interval, the progression of the disease is predicted. The results are stored in a data frame and subsequently visualized using "ggplot2." The resulting plot showcases the changes in the susceptible, infected, and recovered populations over time, with distinct color coding for each compartment.

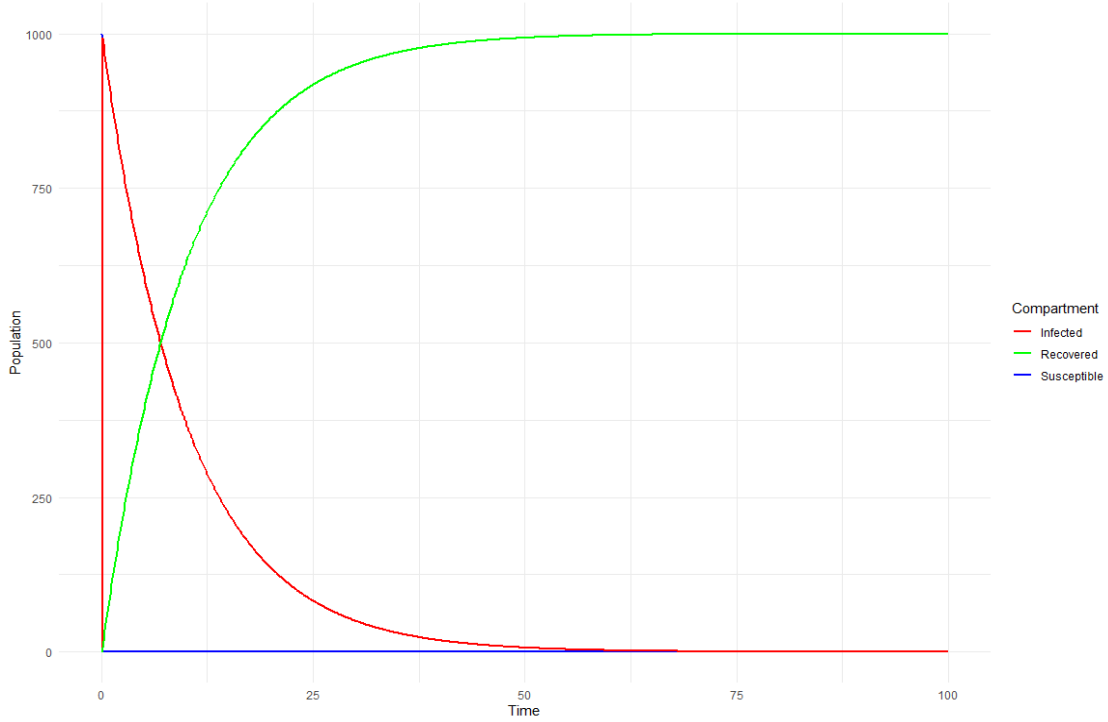

Figure 12. *SIR model with simulated data graph*

Explanation of the simulated graph

The graph in figure 12 was created using the provided above code (see appendix 2.3.4) represents the dynamics of a Susceptible-Infected-Recovered (SIR) model over time.

The x-axis represents time, indicating the progression of the disease outbreak over the specified time period.

The y-axis represents the population size, indicating the number of individuals in each compartment (Susceptible, Infected, and Recovered) at any given time.

The three lines on the graph correspond to the different compartments of the SIR model:

The blue line represents the Susceptible population, which represents individuals who are susceptible to the disease and can potentially become infected.

The red line represents the Infected population, which represents individuals who are currently infected with the disease.

The green line represents the Recovered population, which represents individuals who have recovered from the disease and have acquired immunity.

The initial conditions for the model are set such that there are 999 individuals in the Susceptible compartment, 1 individual in the Infected compartment, and 0 individuals in the Recovered compartment.

The model parameters are set as **beta = 0.2** and **gamma = 0.1**. **beta** represents the transmission rate of the disease, indicating how easily the infection spreads from infected to susceptible individuals. **gamma** represents the recovery rate, indicating how quickly infected individuals recover from the disease.

The lines on the graph illustrate how the population transitions between compartments over time. Initially, the number of infected individuals increases, while the number of susceptible individuals decreases. As the outbreak progresses, the number of infected individuals eventually decreases, while the number of recovered individuals increases.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

The color scheme of the lines corresponds to the different compartments, with blue representing the Susceptible compartment, red representing the Infected compartment, and green representing the Recovered compartment.

By observing the graph, we can gain insights into the dynamics of the disease outbreak. The lines illustrate the changes in population size for each compartment over time, allowing us to analyze the progression of the disease and the impact of parameters such as transmission rate and recovery rate on the overall dynamics of the outbreak.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.5 Agent Based Model

2.3.5.1 *Agent Based Model*

Agent-Based Models (ABMs) are often described using mathematical formulations that capture the behavior of individual agents and their interactions (46,47,48). However, the specific mathematical equations used in ABMs can vary widely depending on the nature of the system being modeled and the rules governing agent behavior. Instead of providing a general mathematical formulation for ABMs, I will explain the key components of an ABM and how they can be represented mathematically:

2.3.5.2 *Agent-Based Models (Abms) Notation*

Agents

An agent's state can be represented by a vector of variables: $\mathbf{X_i = \{x_{1i}, x_{2i}, \ldots, x_{ni}\}}$, where i denotes a specific agent and n is the total number of variables associated with the agent.

Environment

A 2D grid environment could be represented as a matrix $\mathbf{E = [e_{ij}]}$, where i and **j** denote the coordinates on the grid, and e_{ij} represents an attribute of that location in the environment.

Rules and Behaviors

Rules in ABMs are often defined as functions which govern the change in the agents' state variables. For example, the state variables of agent **A** ($X_a = \{x_{1a}, x_{2a}\}$) might evolve over time t according to the functions: $x_{1a}(t+1) = f(x_{1a}(t), x_{2a}(t))$ and $x_{2a}(t+1) = g(x_{1a}(t), x_{2a}(t))$, where f and g represent specific rules or behaviors of the agent.

Interactions

If the state of agent A is affected by the state of agent B, an interaction might be represented as: $\mathbf{x_{1a}(t+1) = h(x_{1a}(t), x_{1\beta}(t))}$, where h is a function describing the interaction between agent A and agent B.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Simulation and Analysis

During the simulation, we would iterate over time steps, updating the state of the agents and the environment according to the defined rules and interactions. For analysis, we might compute statistics over the state variables at each time step, such as their mean $(\mu = \Sigma x_i/N)$ or variance $(\sigma^2 = \Sigma(x_i - \mu)^2/N)$, where Σ denotes the sum over all agents, and **N** is the total number of agents.

Overall

These equations are just a general representation. The specific functions f, g, and h, and the state variables $x_1, x_2, \ldots, x_n$, would be defined based on the specifics of the system being modeled.

But agent-Based Models (ABMs) is a broader category that includes several specific types of models, including individual-based models, Monte Carlo simulations, and cellular automata. Here's how these types of models relate to the framework I described:

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.5.3 *Individual-Based Models*

These are a type of ABM where each agent represents an individual organism or entity. The state variables $(X_i = \{x_{1i}, x_{2i}, ..., x_{n\ i}\})$ would represent characteristics of the individual, and the rules and interactions **(e.g.,** $x_{1a}(t+1) = h(x_{1a}(t), x_{1\beta}(t)))$ would simulate behaviors and interactions at the individual level. The specific mathematical formulation would depend on the nature of the individuals and their interactions.

Monte Carlo simulations

These are a type of stochastic (random) ABM. In these models, the rules and interactions often include random elements. For example, a rule might be $x_{1a}(t+1) = h(x_{1a}(t), x_{1\beta}(t)) + \varepsilon$, where ε is a random variable representing some form of noise or uncertainty. Monte Carlo simulations are typically used to explore the impact of randomness on system dynamics and to estimate probabilities or uncertainty ranges.

Cellular automata models

These are a type of spatial ABM, where the agents are cells in a grid (like the 2D grid environment $\mathbf{E} = [e_{ij}]$ I mentioned), and the state variables represent the state of each cell. The rules and interactions are typically local, meaning that the state of a cell at time **t+1** depends on its own state and the states of its nearby cells at time t. For example, in the famous "Game of Life" cellular automaton, each cell's next state depends on the number of its neighbors that are in the "alive" state.

These are all different ways to structure and analyze ABMs, and they can be combined in various ways depending on the system we're modeling. The mathematical formulation of each type of model would be tailored to the specifics of that model.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Agent-Based Simulation: This R graph demonstration employs the use of the **ggplot2** library to create visualizations that demonstrate the movement of an agent within a two-dimensional space over a series of iterations. We define a function called **simulate_agent_movement** that generates simulated agent positions based on random movement directions at each iteration. We then set the number of iterations to 4 and create individual plots for each iteration using a custom color palette. Each plot displays the agent's movement trajectory up to the current iteration, highlighted by a solid line, and its position at that iteration, represented by a colored dot. Additionally, dashed gray lines connect the agent's positions across iterations, providing a sense of continuity. The titles, axis labels, and overall appearance of the plots are customized using the **ggplot2** library's styling options. Finally, we arrange the plots in a 2-column grid using the **gridExtra** library's **grid.arrange** function to allow for easy comparison of the agent's movement patterns over iterations. This code visually illustrates how an agent's movement evolves over time in a simulated environment, providing a clear representation of dynamic behavior.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

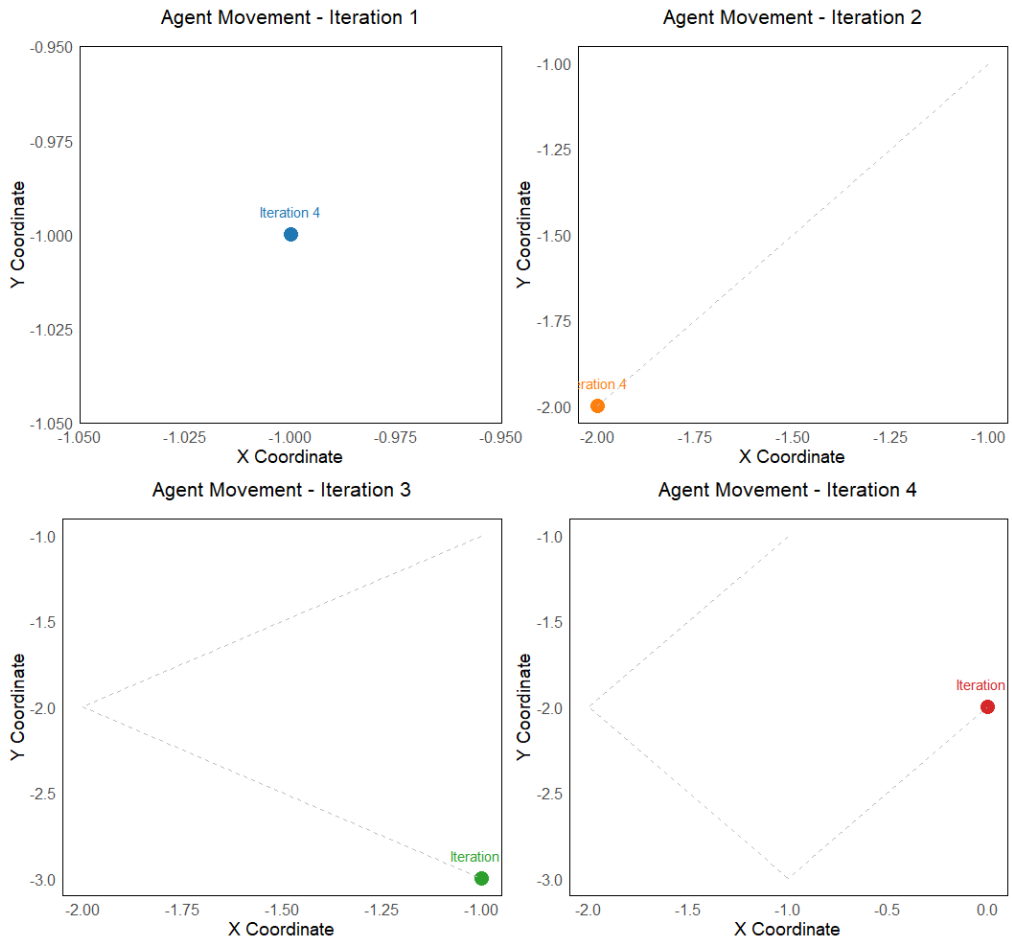

Figure 13. *Agent based model with simulated data graph*

Explanation of the graph

So to explain in a more analytical manner we visualize an agent moving randomly within a two-dimensional space. The agent is equipped with a simple set of rules: at each iteration, it randomly chooses a direction (either left/right along the x-axis or up/down along the y-axis) and moves one unit in that direction. While seemingly straightforward, this simplistic setup allows us to observe how individual actions accumulate to produce dynamic patterns.

The graph showcases the agent's movement over four iterations. Each dot on the graph represents the agent's position after a specific number of iterations.

The gray dashed lines connect the agent's positions, highlighting the trajectory it has taken. During Iteration 1, the agent starts at the origin (0,0). As the simulation progresses to Iteration 2, the agent's position shifts based on its random movement choices. It becomes evident that at each iteration, the agent has an equal likelihood of moving in either the x or y direction.

Moving forward to Iteration 3, the agent's journey becomes more intricate. The green line tracks the agent's movement trajectory up to this point, showcasing the path it has traversed. The green dot represents the agent's position after completing three iterations. Notably, the randomness inherent in the agent's movement choices is evident in the diverse set of positions it occupies.

Continuing into Iteration 4, the agent's cumulative path unfolds. The red line illustrates the entire trajectory the agent has undertaken since the beginning of the simulation. The red dot marks the agent's position after four iterations. By observing the positions of the red dots collectively, we gain insights into the distribution of agent positions within the confined environment. Analyzing the agent's movement over time allows us to uncover patterns that arise from seemingly random decisions.

In this visual representation of ABM, we witness the emergence of complex behaviors from simple rules. This example, though elementary, mirrors the underlying principles that underpin agent-based simulations. The trajectory of the agent offers a glimpse into how individual actions, governed by probabilistic rules, can give rise to patterns that mirror real-world phenomena. Agent-based models empower researchers to explore a plethora of domains, from epidemiology and social dynamics to urban planning and ecology. By analyzing agent trajectories, researchers unlock a valuable tool for unraveling the intricate dynamics of complex systems.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.6 Spatial and Spatiotemporal models

2.3.6.1 *Spatial models*

Spatial models are used to analyze and understand patterns and processes that vary across space. These models incorporate spatial relationships and dependencies into the mathematical formulation. There are various types of spatial models, each with its own mathematical representation (49). Here, I'll explain two common types: spatial autocorrelation and spatial regression.

Spatial Autocorrelation

Spatial autocorrelation measures the degree of similarity or dissimilarity between spatially adjacent locations. It quantifies how the values of a variable at one location are related to the values at neighboring locations. Moran's I is a commonly used measure of spatial autocorrelation.

Mathematical Formulation

Let's consider a variable Y observed at N spatial locations indexed by i = 1, 2, ..., N. The spatial autocorrelation can be computed using the following formula:

$$\text{Moran's I} = (N / S_0) * \Sigma_i\Sigma_j \ (w_{ij} \ * (Y_i - \bar{Y}) * (Y_j \ - \bar{Y})) / \Sigma_i(Y_i - \bar{Y})^2$$

where S_0 is the sum of all spatial weights, w_{ij} is the spatial weight between locations i and j, Y_i is the value of the variable at location i, and $\bar{Y}$ is the mean value of the variable across all locations. The values of Moran's I range from -1 to 1, where positive values indicate positive spatial autocorrelation (similar values tend to be close together) and negative values indicate negative spatial autocorrelation (dissimilar values tend to be close together).

Spatial Regression

Spatial regression models incorporate both spatial relationships and explanatory variables to analyze the relationship between a dependent variable and its predictors, accounting for spatial dependence. One commonly used spatial regression model is the Spatial Lag Model.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Mathematical Formulation

Consider a dependent variable Y and a set of explanatory variables X observed at N spatial locations. The spatial lag model can be represented as:

$$Y = \rho WY + X\beta + \varepsilon$$

where ρ is the spatial autocorrelation parameter, W is the spatial weights matrix, β is the vector of coefficients for the explanatory variables, and ε is the error term. The term ρWY represents the spatial lag of the dependent variable, which captures the influence of neighboring locations on the current location.

The estimation of spatial regression models involves solving for the parameter estimates (ρ and β) using techniques such as maximum likelihood estimation or generalized method of moments. These models allow for the examination of how the spatial structure and explanatory variables jointly influence the dependent variable.

These are just simplified explanations of spatial autocorrelation and spatial regression. Spatial modeling encompasses a wide range of techniques and approaches, each with its own mathematical formulation. The specific choice of model depends on the research question, data characteristics, and the nature of the spatial processes under investigation.

R code example simulation

This generated graph in R, utilizes the "sp," "spdep," and "ggplot2" libraries for spatial autocorrelation analysis and visualization. It generates a 10x10 grid with random data and converts it into a spatial object. The code then defines neighbors and creates a spatial weights matrix for Moran's I test. Subsequently, it calculates lagged values and performs spatial regression, adding residuals to the data. The results are visualized with two separate plots: one displaying the spatial distribution of original data using colored tiles, and the other illustrating the spatial pattern of residuals. These visualizations provide insights into the spatial autocorrelation and model performance (see Appendix 2.3.6 for additional context).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

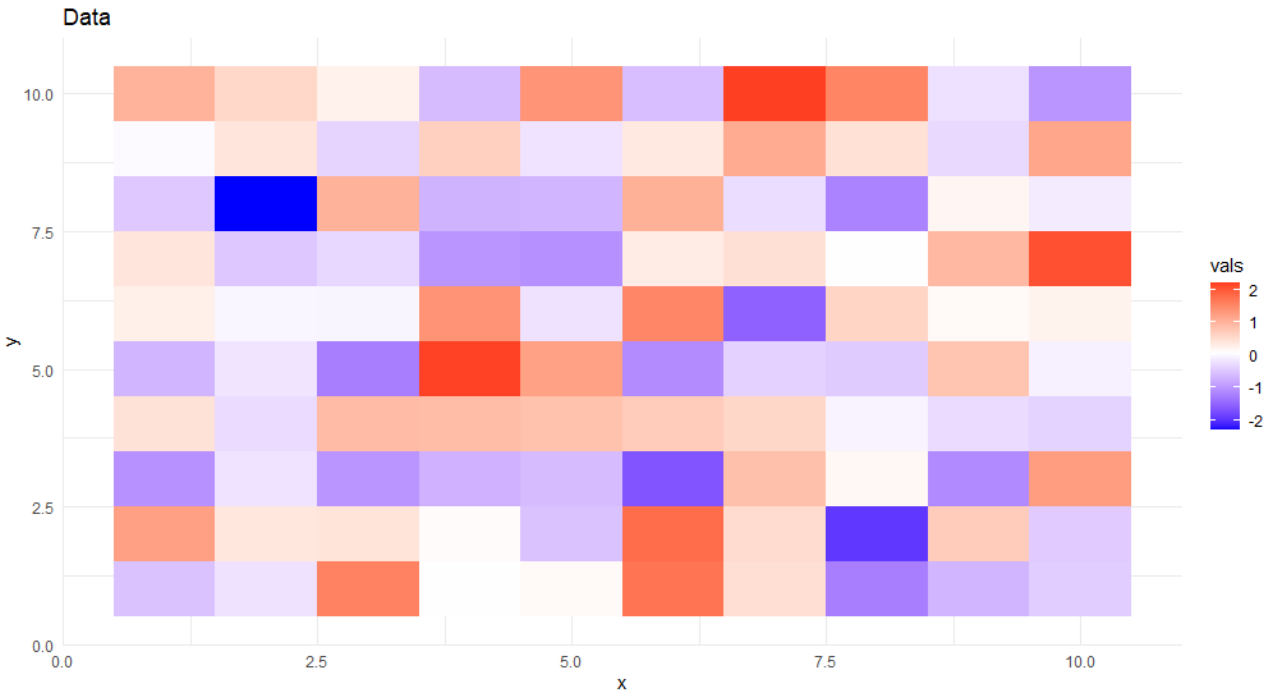

Figure 14. *Spatial distribution of original data*

Explanation of the simulated graph (Spatial distribution of original data)

The first plot in figure 14 visualizes the randomly generated spatial data. The x- and y-axes represent the coordinates of the points in the grid, and the color of each tile represents the value of the 'vals' variable at that point. This 'vals' variable was generated using a standard normal distribution. The color gradient has been set such that low values are blue, high values are red, and values around the median are white. Thus, this plot provides a visual understanding of how these random values are distributed across the grid.

Explanation of the simulated graph (residuals)

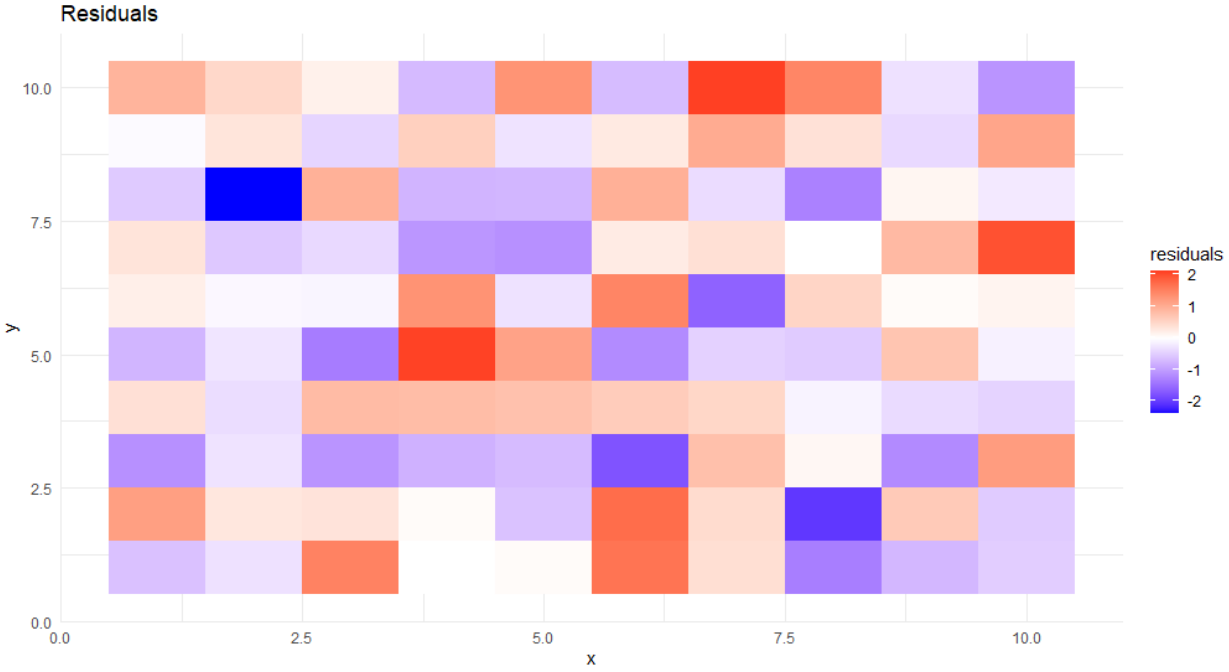

Figure 15. *Spatial pattern of residuals*

Explanation of the simulated graph (spatial pattern of residuals)

The second plot in figure 15 represents the residuals from the spatial regression model that was fit to predict 'vals' using the spatially lagged values of 'vals'. Residuals are the differences between the observed and predicted values, so they give an idea of the accuracy of the model. In this plot, each tile's color shows the residual for the corresponding grid point. Again, the color gradient is set to show low residuals in blue, high residuals in red, and residuals around the median in white. Large residuals (either positive or negative) indicate points where the model's predictions were far off the mark. If the model is well-fitted, we would expect the residuals to be randomly distributed and centered around zero (white color).

So, these two plots together provide a visualization of both the original spatial data and the performance of the spatial regression model on this data.

2.3.6.2 *Spatiotemporal Models*

Spatial-Temporal Autoregressive Models

Spatial-temporal autoregressive models integrate both spatial and temporal dimensions to analyze and predict patterns that evolve over time and space. One common formulation is the space-time autoregressive model.

Mathematical Formulation: Space-Time Autoregressive Model

$$Yit=\rho WYit+\alpha t+\beta Xit+\epsilon it$$

where:

- ρ is the spatial autocorrelation parameter.
- W is the spatial weights matrix.
- Yit is the observed variable at location i and time t.
- αt captures the temporal autocorrelation at time t.
- Xit represents the set of explanatory variables at location i and time t.
- ϵit is the error term.

The term $\rho WYit$ accounts for the spatial lag effect, considering the influence of neighboring locations on the current location, and αt captures the temporal autocorrelation. Estimation involves determining the parameters ρ, αt, and β using appropriate techniques such as maximum likelihood estimation.

Epidemic Models with Spatial Components

Epidemic models with spatial components incorporate spatial factors, such as population density and movement, to understand and predict the transmission dynamics of diseases in a geographic context.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Mathematical Formulation: Spatial Epidemic Model

$$Yit=\lambda iSi(t)Ii(t)+\mu i$$

where:

- λi represents the transmission rate at location i.
- $Si(t)$ is the susceptible population at location i and time t.
- $Ii(t)$ is the infected population at location i and time t.
- μi is the external infection rate.

This simple formulation captures the spatial variations in disease transmission dynamics.
The parameters λi and μi would be estimated based on observed data and the dynamics of the specific disease under investigation.

In summary, spatial-temporal models provide a powerful framework for analyzing and predicting phenomena that evolve over both space and time. The specific formulation depends on the nature of the data and the research question, and parameter estimation often involves sophisticated statistical techniques.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.7 Bayesian Models

2.3.7.1 *Bayesian Models*

Mathematical formulation of Bayesian models for predicting, forecasting, and preventing infectious diseases.

Bayesian Inference: Bayesian inference involves updating our beliefs about a parameter or unknown quantity based on prior knowledge and observed data. The process can be mathematically represented using Bayes' theorem (50,51):

$$P(\theta|D) = (P(D|\theta) * P(\theta)) / P(D)$$

where:

$P(\theta|D)$ represents the posterior distribution of the parameter θ given the data D,

$P(D|\theta)$ is the likelihood of the data given the parameter θ,

$P(\theta)$ is the prior distribution of the parameter θ,

$P(D)$ is the marginal likelihood of the data.

2.3.7.2 *Hierarchical Bayesian Models*

Hierarchical Bayesian models are commonly used in infectious disease modeling. These models involve multiple levels of parameters and allow for the incorporation of uncertainty at each level.

Mathematical Formulation: Consider a hierarchical Bayesian model for disease transmission with parameters $\theta_1, \theta_2, \ldots, \theta_k$. The model can be represented as follows:

Likelihood: $P(D|\theta_1, \theta_2, \ldots, \theta_k) = \prod P(D_i|\theta_1, \theta_2, \ldots, \theta_k)$

where D_i represents the observed data (e.g., number of cases, time series of infections) and $P(D_i|\theta_1, \theta_2, \ldots, \theta_k)$ is the likelihood of the data given the parameters.

Priors: $P(\theta_1, \theta_2, \ldots, \theta_k) = P(\theta_1) * P(\theta_2|\theta_1) * P(\theta_3|\theta_1, \theta_2) * \ldots * P(\theta_k|\theta_1, \theta_2, \ldots, \theta_{k-1})$

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

where $\mathbf{P(\theta_1)}$ represents the prior distribution of the first-level parameter $\boldsymbol{\theta_1}$, $\mathbf{P(\theta_2|\theta_1)}$ represents the conditional prior distribution of the second-level parameter θ_2 given θ_1, and so on.

Posterior Distribution: $P(\theta_1, \theta_2, \ldots, \theta_k \,|D) = (P(D|\theta_1, \theta_2, \ldots, \theta_k) * P(\theta_1) * P(\theta_2|\theta_1) * P(\theta_3|\theta_1, \theta_2) * \ldots * P(\theta_k \,|\theta_1, \theta_2, \ldots, \theta_{k-1})) / P(D)$

The posterior distribution captures the updated beliefs about the parameters given the observed data, incorporating both the prior knowledge and the likelihood of the data.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.7.3 *Model Validation And Prediction*

Bayesian models can be validated and used for prediction by computing posterior predictive distributions. This allows for the generation of future scenarios and estimation of model performance.

Mathematical Formulation: Given the observed data D, the posterior predictive distribution can be calculated as:

$$P(D^*|D) = \int P(D^*|\theta) * P(\theta|D)\, d\theta$$

where **D*** represents the future data or unobserved data, **P(D*|θ)** is the likelihood of the future data given the parameters, and **P(θ|D)** is the posterior distribution obtained from the observed data.

By sampling from the posterior distribution and combining it with the likelihood of future data, predictions and forecasts can be made. This allows for uncertainty quantification and the generation of probabilistic predictions.

2.3.7.4 *Bayesian Models' Applications*

Bayesian Hierarchical Models

These models are a type of Bayesian model that allows for the incorporation of hierarchical structure in the data and parameters.

Mathematical Formulation: In a Bayesian hierarchical model, parameters are considered to have their own distributional assumptions. This is mathematically expressed as follows:

Suppose we have n data points $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ and associated parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$. A Bayesian hierarchical model could be written as:

Level 1 (Data Level): $P(y_i \mid \theta_i)$ for $i = 1, \ldots, n$.

Level 2 (Parameter Level): $\theta_i \sim P(\theta_i \mid \varphi)$ for $i = 1, \ldots, n$.

Level 3 (Hyperparameter Level): $\varphi \sim P(\varphi)$, where φ represents hyperparameters.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Bayesian State-Space Models

These models are a type of Bayesian model that are used when dealing with sequential or temporal data. The goal is to estimate an underlying "state" process given some noisy measurements.

Mathematical Formulation: Let's denote the sequence of states by $\mathbf{x} = (x_1, x_2, \ldots, x_t)$ and the sequence of observations by $\mathbf{y} = (y_1, y_2, \ldots, y_t)$. A Bayesian state-space model could be written as:

State Equation (process model): $x_t = f(x_{t-1}, w_t)$, where $w_t \sim P(w_t)$ describes the process noise.

Observation Equation (measurement model): $y_t = g(x_t, v_t)$, where $v_t \sim P(v_t)$ describes the measurement noise.

Bayesian Network Models

These models, also known as Belief Networks or Graphical Models, represent the probabilistic relationships among a set of variables. Each variable in a Bayesian network corresponds to a node in the network, and the relationships between variables are represented by directed edges.

Mathematical Formulation: Let's denote the variables in the network by $\mathbf{X} = (X_1, X_2, \ldots, X_n)$. The joint distribution of these variables in a Bayesian network can be written as:

$P(X_1, X_2, \ldots, X_n) = \Pi\, P(X_i \mid \text{parents}(X_i))$, where parents($X_i$) denotes the parents of the variable X_i in the network, and the product is over all variables in the network.

Overall

Each of these models has its own strengths and is suited to different types of data and research questions. The choice of model depends on the structure of the data, the nature of the problem, and the available computational resources.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Finally in Bayesian models, the choice of prior distributions and the modeling assumptions play a crucial role in the final results. These choices should be made based on domain knowledge, expert.

Visualization in R language

Bayesian Inference for Case Numbers: This R code simulates Bayesian inference for predicting new cases over time. It begins with a vector of cases and calculates the number of time points. Prior distribution parameters are set as **prior_alpha** and **prior_beta**. Bayesian updating is performed by calculating the posterior alpha and beta using the prior and observed cases. A posterior predictive distribution is generated using the beta distribution and random sampling. Summary statistics of the posterior predictive distribution, including mean, median, and a 95% credible interval, are computed. The distribution is visualized using a histogram overlaid with a density line, and mean marker. The results, including mean, median, and credible interval, are printed. The code provides insights into probabilistic predictions for new case numbers (see Appendix 2.3.7 for further details).

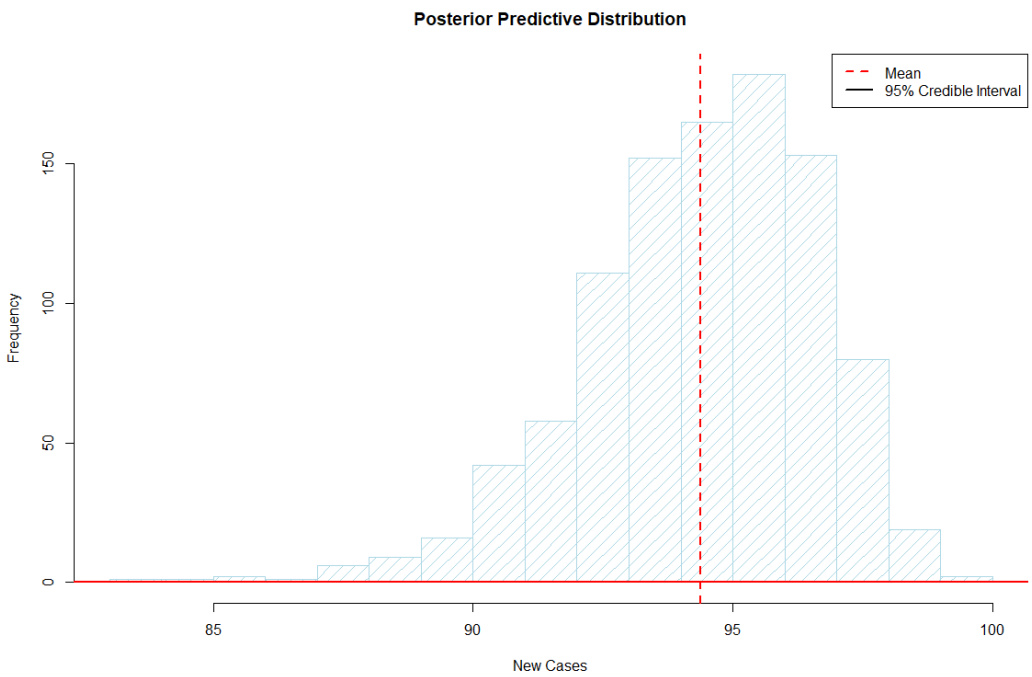

Figure 16. *Bayesian model with simulated data graph*

Explanation of the graph

Distribution Shape: The histogram and density curve provide information about the shape of the distribution of new cases. If the distribution is symmetric and bell-shaped, it indicates that the model predictions are centered around a certain value. If the distribution is skewed, it suggests that there may be a bias or deviation from the expected value.

Central Tendency: The mean (represented by the dashed red line) gives us an estimate of the average number of new cases. It represents the central tendency of the posterior predictive distribution. The closer the mean is to the peak of the distribution, the more likely it is to be a representative value for the new cases.

Variability: The width of the distribution, as indicated by the spread of the histogram and density curve, reflects the variability or uncertainty in the predictions. A wider distribution suggests higher uncertainty, while a narrower distribution implies more confidence in the estimates. The 95% credible interval (represented by the black vertical lines) provides a range within which we can be reasonably confident that the true value of new cases falls.

Outliers: Outliers, if present, are represented as bars that deviate significantly from the main distribution. They indicate unusual or extreme values of new cases that may warrant further investigation or consideration.

By analyzing the shape, central tendency, variability, and presence of outliers in the graph, we can assess the reliability and confidence of the predictions. This information can guide decision-making, resource allocation, and further analysis in the context of managing and understanding infectious diseases.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.8 Generalized Linear Models (Glms)

2.3.8.1 *Generalized Linear Models*

GLMs extend the traditional linear regression model to handle a wider range of response variables by incorporating a link function and a variance function (52,53).

Mathematical Formulation: Let's denote the response variable as Y and the predictor variables as $\mathbf{X_1, X_2, ..., X_p}$.

a. Linear Predictor: The linear predictor is represented as:

$$\eta = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p ,$$

where η is the linear predictor, $\boldsymbol{\beta_0, \beta_1, \beta_2, ..., \beta_p}$ are the regression coefficients, and $\mathbf{X_1, X_2, ..., X_p}$ are the predictor variables.

b. Link Function: The link function g(·) relates the linear predictor to the mean of the response variable. It can be represented as:

$$\mathbf{g(\mu) = \eta},$$

where μ represents the mean of the response variable.

c. Variance Function: The variance function V(μ) models the relationship between the mean of the response variable and its variance. It can be represented as:

$$\mathrm{Var}(Y) = V(\mu),$$

where **Var(Y)** represents the variance of the response variable.

2.3.8.2 *Model Components*

a. Probability Distribution: The choice of probability distribution depends on the nature of the response variable. Common distributions include:

Binomial: Y follows a binomial distribution when the response is binary (e.g., presence or absence of a disease).

Poisson: Y follows a Poisson distribution when the response is a count (e.g., number of cases).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Gaussian: Y follows a Gaussian distribution when the response is continuous (e.g., disease incidence rate).

b. Link Function: The link function $g(\cdot)$ maps the linear predictor to the mean of the response variable based on the chosen probability distribution. Common link functions include:

Logit: $g(\mu) = \log(\mu / (1 - \mu))$ for the binomial distribution.

Log: $g(\mu) = \log(\mu)$ for the Poisson and Gaussian distributions.

Identity: $g(\mu) = \mu$ for the Gaussian distribution.

c. Variance Function: The variance function $V(\mu)$ models the relationship between the mean of the response variable and its variance. Common variance functions include:

Identity: $V(\mu) = \mu$ for the Gaussian distribution.

Log: $V(\mu) = \log(\mu)$ for the Poisson distribution.

2.3.8.3 *Estimation And Inference*

The estimation of GLMs involves fitting the model to the observed data using maximum likelihood estimation or Bayesian methods. The regression coefficients $\beta_0, \beta_1, \beta_2, \ldots, \beta_p$ are estimated, and inference can be made based on these estimates.

To make predictions or forecasts, new values of the predictor variables are plugged into the fitted model equation, and the link function and variance function are applied to estimate the expected response or outcome.

It's important to note that the specific mathematical formulation of a GLM depends on the chosen probability distribution, link function, and variance function, which are selected based on the characteristics of the response variable and the assumptions of the model.

Overall, GLMs provide a flexible framework for modeling infectious diseases by accommodating different types of response variables, capturing the relationships between predictors and outcomes, and considering appropriate distributions and functional forms.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

R Code example of a GLM model

Predictive Modeling for Daily Cases: This R generated graph uses the "ggplot2" and "scales" libraries to illustrate a predictive modeling scenario. It simulates daily case data for the month of January using a Poisson distribution. The data is structured into a dataframe with 'date' and 'cases'. A Generalized Linear Model (GLM) is fitted, modeling 'cases' based on 'date' using the Poisson family. Predictions for February are generated by extending 'new_dates'. The original and predicted data are combined, and a line and point plot is created using "ggplot2". Blue points and line depict actual data, while red points denote predicted cases. The x-axis is labeled with dates using a specific format. The resulting plot offers insights into the predictive modeling of daily cases (see Appendix 2.3.8 for further details).

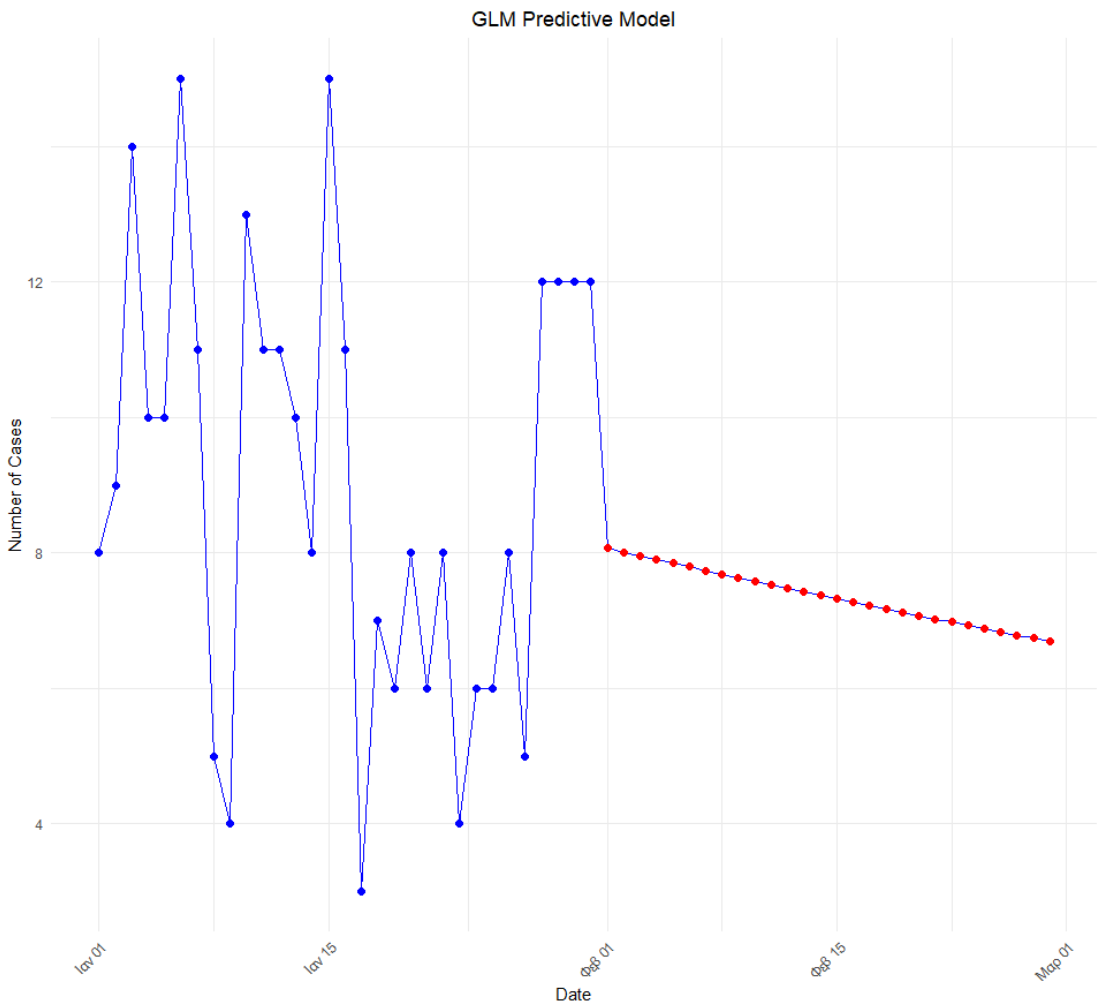

Figure 17. *GLM predictive model with simulated data graph*

Explanation of the graph

As started before the graph in figure 17 shows the observed and predicted number of infectious disease cases over time based on the GLM predictive model. Here's what we can infer from the graph:

Observed Data: The blue line and blue circles represent the actual number of cases of the infectious disease over a specific period, as obtained from the available data. The blue line connects the observed data points, showing the trend of the disease.

Predicted Data: The red circles represent the predicted number of cases of the infectious disease for future dates, based on the GLM predictive model. The model takes into account the relationship between the date and the number of cases observed in the past to make predictions.

Trend Analysis: By observing the trend of the blue line and how it fits with the red circles, we can assess the performance of the GLM predictive model. If the red circles closely follow the trend of the blue line, it suggests that the model captures the underlying patterns in the data and provides reasonable predictions.

Forecasting: The red circles beyond the observed data points represent the model's predictions for future dates. These predictions can be used to anticipate the number of infectious disease cases in the coming days or weeks.

Confidence in Predictions: The width of the red circles can indicate the level of uncertainty or confidence in the predictions. Wider circles imply higher uncertainty, while narrower circles suggest more confidence in the predicted values.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.9 Machine Learning Models

2.3.9.1 *Machine Learning Models*

Machine learning models are data-driven models that learn patterns and relationships from historical data to make predictions or decisions. They are well-suited for infectious disease prediction and prevention as they can handle complex interactions between multiple variables and adapt to changing patterns in the data.

Mathematical Formulation: Machine learning models use various algorithms and techniques to learn from data (54,55). While the mathematical formulations differ across different machine learning algorithms, here are some common components:

a. Input Data: Let's denote the input data as X, which consists of predictor variables such as demographic information, environmental factors, and disease-related indicators.

b. Target Variable: The target variable Y represents the outcome of interest, such as disease occurrence, transmission, or severity.

c. Model Parameters: Machine learning models have parameters that are learned from the data during the training process. These parameters are adjusted to minimize the discrepancy between the predicted outcomes and the observed outcomes.

d. Training Algorithm: The training algorithm is used to estimate the model parameters based on the input data. This involves optimizing an objective function that captures the model's ability to predict the target variable accurately.

e. Prediction Function: Once the model is trained, it can make predictions on new, unseen data. The prediction function takes the learned parameters and the input data to produce the predicted outcome.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.9.2 *Training, Evaluation, And Deployment*

Machine learning models are trained using historical data with known outcomes. The performance of the models is evaluated using metrics such as accuracy, precision, recall, and F1-score. The trained models can then be deployed to make predictions on new data or to forecast future disease trends.

It's important to note that different machine learning models have different mathematical formulations and assumptions. The choice of a specific model depends on the characteristics of the data, the nature of the infectious disease, and the specific prediction or prevention task at hand.

Overall, machine learning models offer powerful tools for predicting, forecasting, and preventing infectious diseases by leveraging historical data and capturing complex patterns and interactions among variables.

Visualization in r

This R code utilizes the "randomForest" and "ggplot2" libraries to perform disease prediction using a random forest model. A synthetic dataset is generated with simulated features, and an 80-20 split is used for training and testing. The random forest model is trained on the training set and used for predictions on the test set. The code calculates the accuracy of predictions and visualizes the relative importance of features using a bar plot. This process offers insights into disease prediction accuracy and the significance of input features (see Appendix 2.3.9 for additional context).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

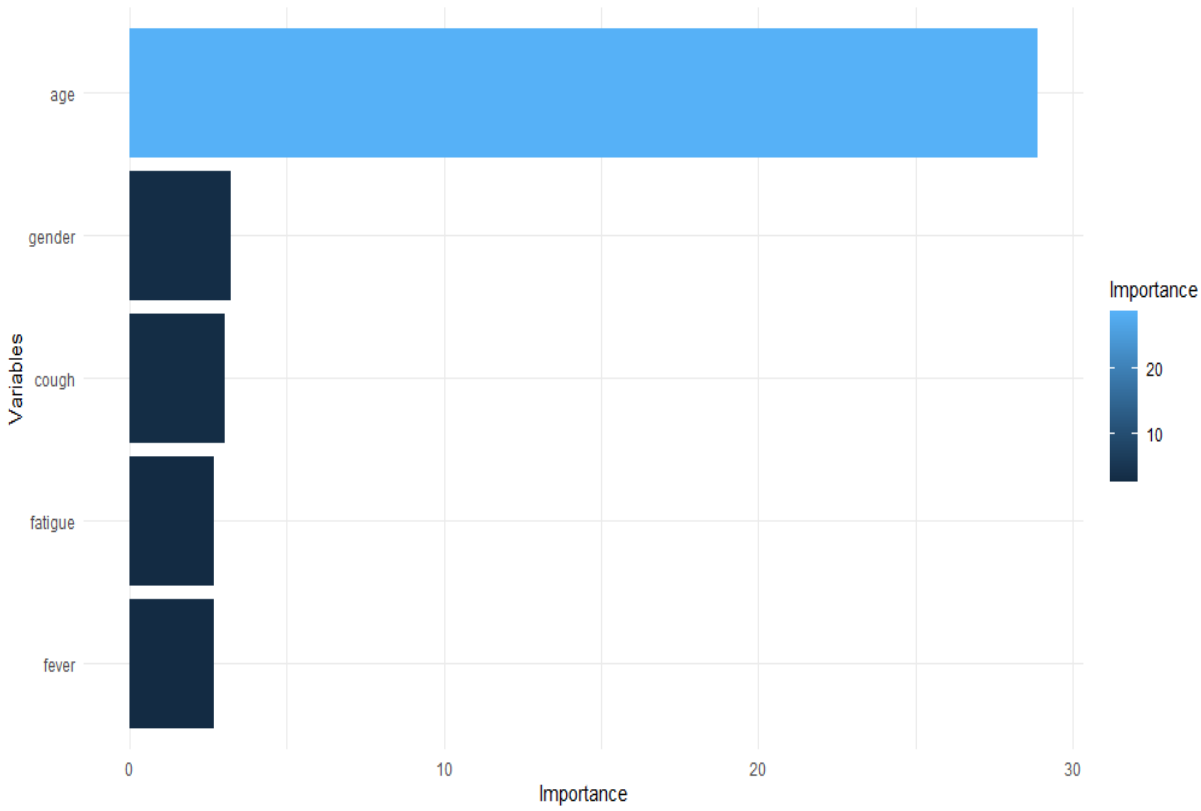

Figure 18. *Machine learning model with simulated data graph*

Explanation of the graph

The plot in figure 18 visualizes the feature importance based on the Random Forest model that we trained.

Feature importance is a way of quantifying which features (variables) contribute the most to the predictions made by a machine learning model. For the Random Forest model, this is typically measured by the Gini importance or mean decrease in impurity.

The x-axis lists the features in our dataset, which in this case are age, gender, fever, cough, and fatigue. The y-axis represents the importance score for each feature. The higher the score, the more important the feature is in making predictions. This score is calculated by the Random Forest algorithm and it reflects how much the prediction error increases when the information from that feature is permuted, hence breaking the relationship between the feature and the target.

The color in the bars also corresponds to the importance of the variable, with darker colors indicating higher importance.

In summary, the graph helps to understand which features are most influential in predicting the disease status according to our Random Forest model. For instance, if 'age' has the highest bar, it means that age is the most significant predictor of the disease status in our model.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.3.10 Time Series Count Models

Time series count models are specifically tailored to analyze count data over time. Two common models for this purpose are Poisson regression models and Negative Binomial models.

Poisson Regression Models

Poisson regression models are designed for count data, making them particularly useful for scenarios like daily infection counts. These models capture temporal dependencies and overdispersion.

For a response variable Yt representing counts at time t, the Poisson regression model is expressed as:

$$Yt \sim \text{Poisson}(\lambda t)$$

where:

- λt is the average rate of events (e.g., infections) at time t.

Temporal dependencies can be incorporated by including lagged counts or other time-related covariates in the model.

Overdispersion Handling: While Poisson models assume constant variance equal to the mean (no overdispersion), they can be extended to handle overdispersed count data by using Poisson-gamma or Poisson-lognormal models.

Negative Binomial Models

Negative Binomial models are suitable for count data with overdispersion, allowing for varying infection rates over time.

For a response variable Yt, the Negative Binomial model is expressed as:

$$Yt \sim \text{NegBin}(\mu t, \phi)$$

where:

- μt is the mean of the distribution at time t.
- ϕ is the dispersion parameter controlling overdispersion.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

These models can account for varying infection rates over time by incorporating time-related covariates or random effects.

In summary, Poisson regression models and Negative Binomial models offer effective tools for analyzing time series count data, considering temporal dependencies and overdispersion. The choice between these models depends on the nature of the data and the specific characteristics of the count process under investigation.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.4 HISTORICAL RESEARCH WITH THESE MODELS FOR DENGUE FEVER

Some research papers that have explored the use of different models for forecasting infectious diseases, specifically focusing on dengue fever:

2.4.1 Compartment Models

Hladish, T. J., (2018). Forecasting dengue vaccine demand in disease-endemic and non-endemic countries. Human Vaccines & Immunotherapeutic, 14(7), 1617-1627.

The goal of this research was to forecast the demand for a dengue vaccine in both disease-endemic and non-endemic countries. The problem addressed was the need for accurate predictions of vaccine demand to inform vaccination strategies and ensure sufficient vaccine supply.

The researchers collected data on dengue prevalence, population size, and vaccine coverage rates from disease-endemic and non-endemic countries. They developed compartmental models to estimate the demand for a dengue vaccine in different scenarios, considering factors such as disease burden and population characteristics.

Luo, L., (2017). Forecasting dengue fever in China using ARIMA model. Public Health, 147, 17-22.

The goal of this research was to forecast dengue fever in China using an Autoregressive Integrated Moving Average (ARIMA) model. The problem addressed was the need for accurate and timely predictions of dengue fever outbreaks to support public health planning and resource allocation.

The researchers collected historical dengue fever data from China and applied an ARIMA model to forecast future occurrences. They analyzed the data to identify trends and patterns and used the ARIMA model to generate predictions for future dengue fever cases in China.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.4.2 Spatial Models

Ramadona, A. L., (2019). Spatial modeling of dengue fever considering socioeconomic and environmental factors in Bandung City, Indonesia. Spatial and Spatio-temporal Epidemiology, 30, 100301. The goal of this research was to develop a spatial model for dengue fever considering socioeconomic and environmental factors in Bandung City, Indonesia. The problem addressed was the need for a comprehensive understanding of the spatial patterns of dengue transmission and the influence of local factors to guide targeted interventions and control strategies.

The researchers collected data on dengue fever cases, population density, land use, and climate variables in Bandung City, Indonesia. They developed a spatial model that integrated these factors to identify high-risk areas for dengue transmission. The model aimed to provide insights for targeted interventions and control strategies.

Gao, Z., (2018). Predicting dengue fever outbreaks in China using a Bayesian hierarchical model. PLoS Neglected Tropical Diseases, 12(6), e0006616. The goal of this research was to predict dengue fever outbreaks in China using a Bayesian hierarchical model. The problem addressed was the need for accurate and timely predictions of dengue outbreaks at a regional level to facilitate preparedness and response measures. The researchers gathered climate data, mosquito population data, and historical dengue cases from different regions of China. They developed a Bayesian hierarchical model that incorporated these data sources to predict dengue outbreaks. The model aimed to estimate the risk of dengue outbreaks in different regions and provide early warning for preventive measures.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.4.3 Bayesian Models

Thompson, R. N., (2019). Improved inference of time-varying reproduction numbers during infectious disease outbreaks. Epidemics, 29, 100356. The goal of this research was to improve the inference of time-varying reproduction numbers during infectious disease outbreaks. The problem addressed was the need for more accurate estimation of reproduction numbers, a key epidemiological parameter, to understand the transmission dynamics and guide control strategies. The researchers collected case count data and other relevant information from infectious disease outbreaks. They developed a Bayesian framework to estimate time-varying reproduction numbers, considering factors such as transmission rates and intervention measures. The goal was to improve the accuracy of reproduction number estimates for better outbreak control.

Reiner, R. C., (2013). A systematic review of mathematical models of mosquito-borne pathogen transmission: 1970-2010. Journal of The Royal Society Interface, 10(81), 20120921. The goal of this research was to conduct a systematic review of mathematical models of mosquito-borne pathogen transmission. The problem addressed was the need to synthesize existing models to gain insights into the transmission dynamics, study design, and limitations of these models for guiding control strategies. The researchers conducted a systematic review of mathematical models of mosquito-borne pathogen transmission. They analyzed various Bayesian models used in the literature to understand the transmission dynamics of diseases like malaria and dengue. The review aimed to provide insights into the applications and limitations of Bayesian modeling approaches.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.4.4 Glms

Apinonkul, B., (2016). The use of generalised linear mixed models to predict incidence of dengue infection in Southern Thailand. PLoS Neglected Tropical Diseases, 10(9), e0004973. The goal of this research was to use generalized linear mixed models to predict the incidence of dengue infection in Southern Thailand. The problem addressed was the need for accurate prediction models to identify high-risk areas and populations, guide targeted interventions, and allocate resources effectively. The researchers collected data on meteorological variables, mosquito abundance, and socioeconomic indicators in Southern Thailand. They applied Generalized Linear Mixed Models (GLMMs) to predict dengue incidence in the region. The models incorporated these factors to identify the key predictors of dengue transmission and develop a predictive model for targeted interventions.

Poovorawan, Y., (2011). Dynamic changes in viral load patterns in dengue patients exhibiting different viral clearance profiles. Intervirology, 54(6), 310-318. The goal of this research was to analyze the dynamic changes in viral load patterns in dengue patients with different viral clearance profiles. The problem addressed was the need for a better understanding of the viral dynamics and clearance patterns in dengue infection to improve clinical management and treatment strategies. The researchers collected viral load data from dengue patients and used GLMs to analyze the dynamic changes in viral load over time. They investigated the relationship between viral clearance profiles and disease outcomes. The study aimed to provide insights into the viral kinetics of dengue and its implications for disease management.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.4.5 Agent-Based Models

Lopes, L. Q., (2016). Spatio-temporal analysis of dengue: Insights from agent-based modeling and simulation. PLoS Neglected. The goal of this research was to perform spatio-temporal analysis of dengue using agent-based modeling and simulation. The problem addressed was the need for a detailed understanding of the complex interactions between human behavior, mosquito populations, and environmental factors to predict and control dengue transmission more effectively. The researchers developed an agent-based model to analyze the spatio-temporal dynamics of dengue transmission. They created a simulation framework where individual agents represented humans and mosquitoes, and their interactions and movement were modeled based on real-world data. The model incorporated factors such as mosquito breeding sites, human mobility, and interventions (e.g., vector control measures). The goal was to gain insights into the impact of different control strategies and interventions on dengue transmission dynamics and evaluate their effectiveness in reducing disease burden.

In this research, the researchers used an agent-based modeling approach to simulate the spread of dengue fever. The model considered individual agents representing humans and mosquitoes, their interactions, and relevant environmental factors. By running simulations based on real-world data, the researchers aimed to understand how the disease spreads in different scenarios and assess the potential impact of various interventions and control measures. The ultimate objective was to provide insights and recommendations for effective strategies to prevent and control dengue transmission.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.4.6 Unique Features and capabilities of these predictive models

Let's explore the unique features and uses of different modeling approaches for infectious disease prediction.

Compartment Models

Unique Features: Compartment models, such as the SIR model, simplify complex infectious disease dynamics by dividing the population into distinct compartments. They allow for the exploration of disease transmission and the impact of interventions.

Uses: Compartment models are widely used for studying disease spread, estimating key epidemiological parameters (e.g., transmission rates), predicting disease outcomes, and evaluating intervention strategies.

Spatial Models

Unique Features: Spatial models consider the geographic distribution of disease occurrence and the influence of spatial factors (e.g., distance, connectivity) on disease spread. They account for spatial autocorrelation and heterogeneity.

Uses: Spatial models help identify high-risk areas, understand spatial patterns of disease transmission, guide resource allocation, and inform targeted interventions and control measures.

Agent-Based Models

Unique Features: Agent-based models (ABMs) simulate individual entities (agents) with specific characteristics and behaviors. They allow for the representation of complex interactions and heterogeneity within a population.

Uses: ABMs are valuable for studying individual-level behaviors, contact patterns, and the impact of social dynamics on disease spread. They can simulate interventions and evaluate their effectiveness in different scenarios.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Generalized Linear Models (GLMs)

Unique Features: GLMs provide a flexible framework for modeling the relationship between predictors and infectious disease outcomes. They can handle various types of response variables and accommodate non-normal error distributions.

Uses: GLMs are commonly used for analyzing infectious disease data, such as count data (Poisson regression), binary outcomes (logistic regression), or time-to-event data (survival analysis). They help identify risk factors, estimate associations, and make predictions.

Bayesian Models

Unique Features: Bayesian models use Bayes' theorem to update prior beliefs based on observed data. They allow for the incorporation of prior knowledge and uncertainty quantification through probability distributions.

Uses: Bayesian models are useful for estimating model parameters, making predictions, and performing model comparisons. They can integrate diverse data sources, handle missing data, and provide posterior distributions for inference.

Machine Learning Models

Unique Features: Machine learning models leverage algorithms to automatically learn patterns and relationships from data. They can handle complex and high-dimensional datasets, capture non-linear relationships, and adapt to changing conditions.

Uses: Machine learning models are employed for various tasks in infectious disease prediction, including outbreak detection, disease surveillance, risk prediction, and treatment response assessment. They can incorporate diverse data sources (e.g., clinical, genomic, social media) and provide accurate predictions.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

uncertainties and provide a framework for incorporating expert opinions and updating forecasts as new data becomes available.

These models are tailored to the unique characteristics of infectious diseases and aim to provide more accurate and specialized predictions compared to general-purpose forecasting models. They consider disease-specific factors and dynamics, enabling researchers and public health practitioners to make informed decisions and take appropriate actions to prevent, control, and manage infectious diseases.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.5 STATISTICAL DISTRIBUTIONS FOR SIMULATED INFECTIOUS DISEASE DATA

In the realm of statistical tools for analyzing infectious disease data, several distributions play pivotal roles (65,66).

2.5.1 Poisson Distribution

The Poisson distribution is a discrete probability distribution that expresses the probability of a given number of events occurring in a fixed interval of time or space. Formally, the probability mass function of the Poisson distribution is given by:

$$P(X=k) = \lambda^k * e^{-\lambda} / k!$$

where:

P(X=k) is the probability of **k occurrences** (events)

λ is the average rate of occurrence per interval

e is Euler's number, a mathematical constant approximately equal to 2.71828

k! is the factorial of k, which is the product of all positive integers up to k

The term $e^{-\lambda}$ represents **e** raised to the power of **-λ**. The negative exponent means that as **λ** (the average rate of occurrence) increases, the term e^{-it} decreases, modeling the decreasing probability of higher counts.

Example

Modeling the number of new cases of malaria per week in a specific region. The Poisson distribution is commonly used to analyze count data for diseases like malaria.

Visualization in R language

Here, a scenario is illustrated using R to model the number of new malaria cases per week. The mean number of cases per week is set as λ = 5. Synthetic data is generated with a Poisson distribution representing the occurrence of cases. The distribution's properties are depicted through visualization: a histogram showcases the data distribution, while the Probability Mass Function (PMF) is

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

plotted, revealing the probability of observing different numbers of new cases. This example demonstrates how the Poisson distribution can be employed to model disease occurrences and provides insights into their probabilistic nature (see Appendix 2.5.1 for additional context).

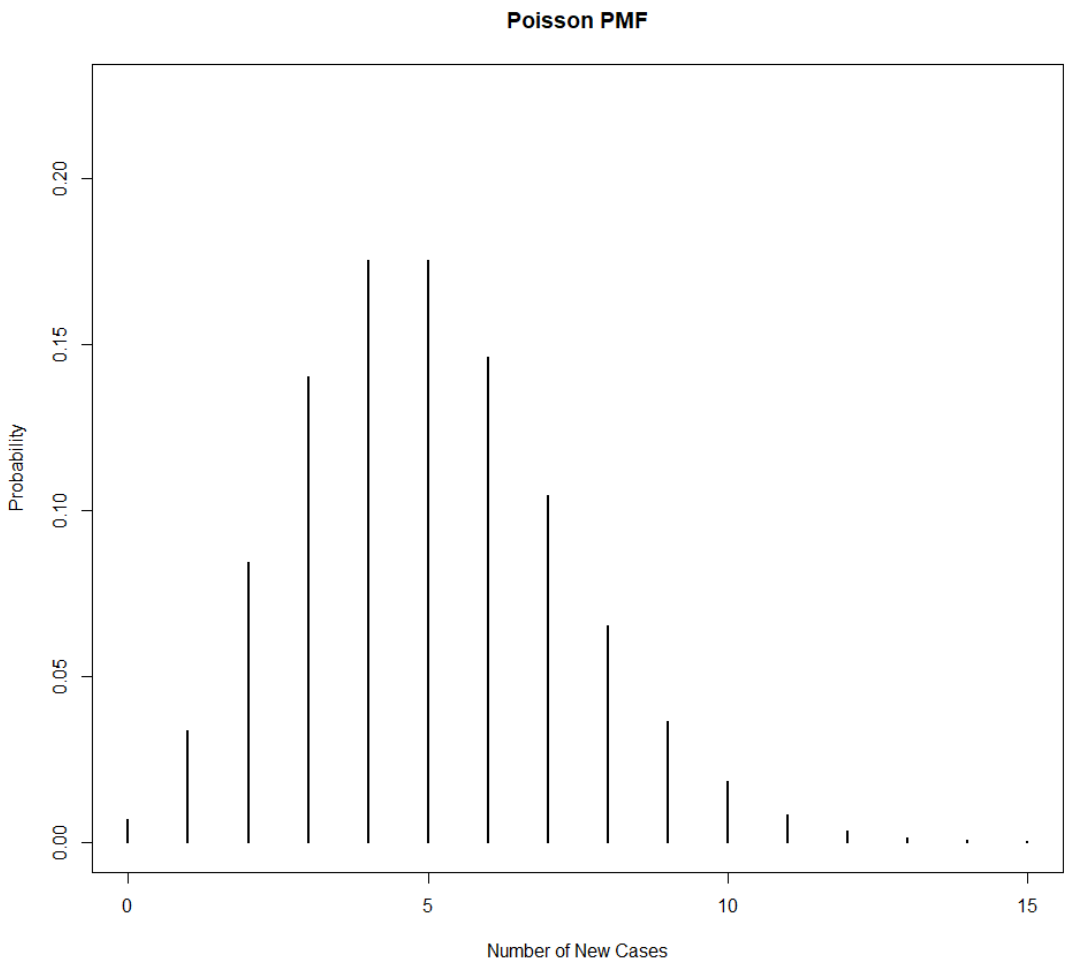

Figure 19. *Poisson distribution with simulated data graph*

Explanation of the graph

Figure 19 represents the observed data distribution of the number of new cases of malaria per week. The probability mass function (PMF) plot shows the probability of observing a specific number of cases based on the Poisson distribution.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.5.2 Negative Binomial Distribution

Negative Binomial Distribution: The Negative Binomial distribution is a discrete probability distribution used to model the number of failures (r - 1) that occur before r successes in a sequence of independent Bernoulli trials, each with a constant probability of success, p.

The probability mass function of the Negative Binomial distribution is given by:

$$P(X=k) = (k+r-1)C(r-1) * p^r * (1-p)^k$$

where:

P(X=k) represents the probability of observing k failures before r successes.

(k+r-1)C(r-1) denotes the number of combinations of **k+r-1** objects taken **r-1** at a time.

p is the probability of success in each Bernoulli trial.

Example

Modeling the number of reported cases of dengue fever in different neighborhoods within a city. The negative binomial distribution accounts for overdispersion commonly observed in disease count data.

Negative Binomial Distribution Illustration:

The R code demonstrates modeling the number of reported cases of dengue fever using the negative binomial distribution. A synthetic dataset is generated with a specified size parameter and probability of success. Visualizations are employed to represent the distribution: a histogram displays the data distribution, while a Probability Mass Function (PMF) plot showcases the probability of observing various numbers of reported cases. This example illustrates the application of the negative binomial distribution to model disease reports and visualize their probabilistic characteristics (see Appendix 2.5.2 for additional insights).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

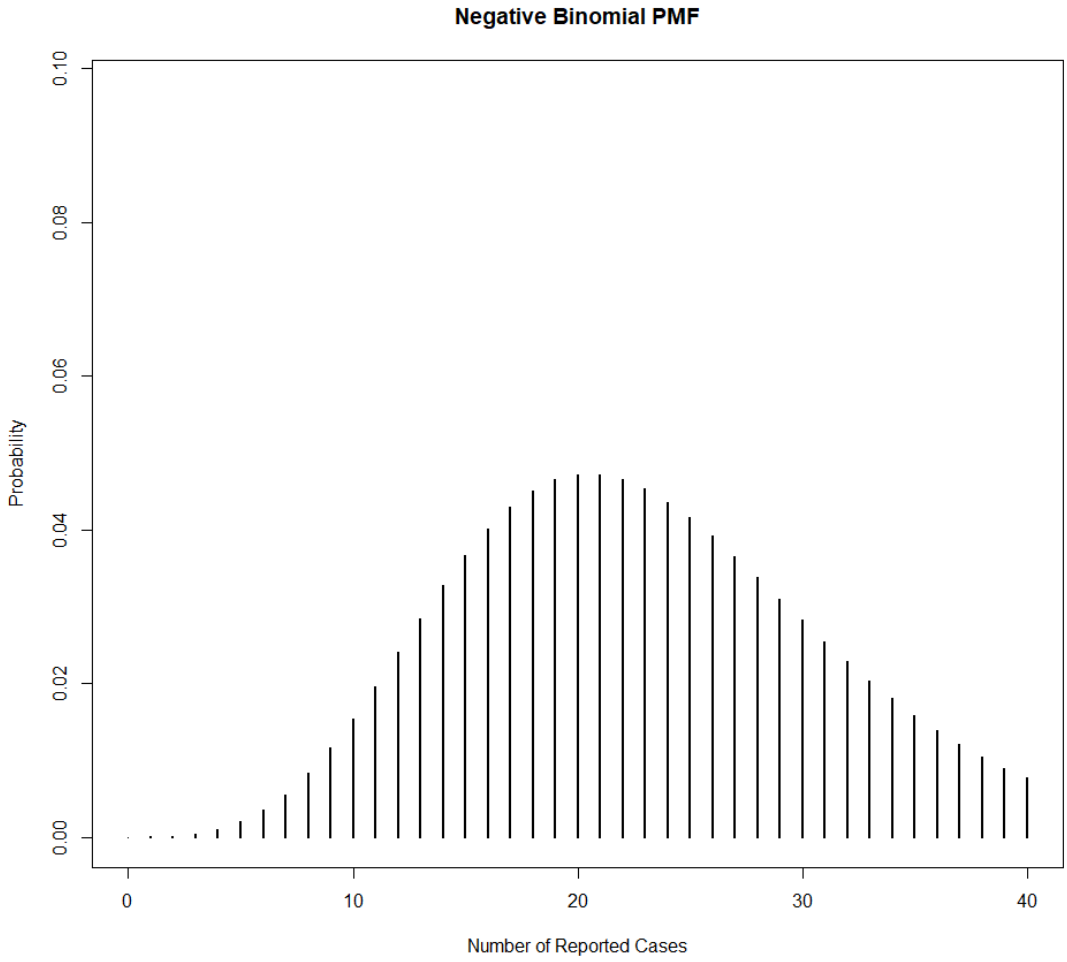

Figure 20. *Negative binomial distribution with simulated data graph*

Explanation of the graph

Figure 20 represents the observed data distribution of the number of reported cases of dengue fever. The probability mass function (PMF) plot shows the probability of observing a specific number of cases based on the negative binomial distribution.

2.5.3 Gaussian (Normal) Distribution

Gaussian (Normal) Distribution: The Gaussian distribution, also known as the Normal distribution, is a continuous probability distribution that is symmetric and bell-shaped. It is widely used to model continuous data in various fields.

The probability density function of the Gaussian distribution is given by:

$$f(x) = (1 / \sqrt{(2\pi\sigma^2)}) * e^{(-(x-\mu)^2 / (2\sigma^2))}$$

where:

f(x) represents the probability density at the point x.

μ is the mean of the distribution.

σ^2 is the variance of the distribution.

e is Euler's number (approximately 2.71828).

π is a mathematical constant (approximately 3.14159).

Example

Modeling the body temperature of individuals infected with influenza in a clinical study. The normal distribution is frequently used to represent continuous variables in epidemiological analyses.

Normal Distribution Illustration:

The provided R code exemplifies modeling the body temperature of individuals infected with influenza using the normal distribution. Synthetic data is generated with a specified mean and standard deviation. Visualizations are employed to represent the distribution: a histogram showcases the data distribution, and a Probability Density Function (PDF) plot demonstrates the probability density of various body temperatures. This example underscores the application of the normal distribution to model characteristics like body temperature and visualize their probabilistic properties (see Appendix 2.5.4 for additional context).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

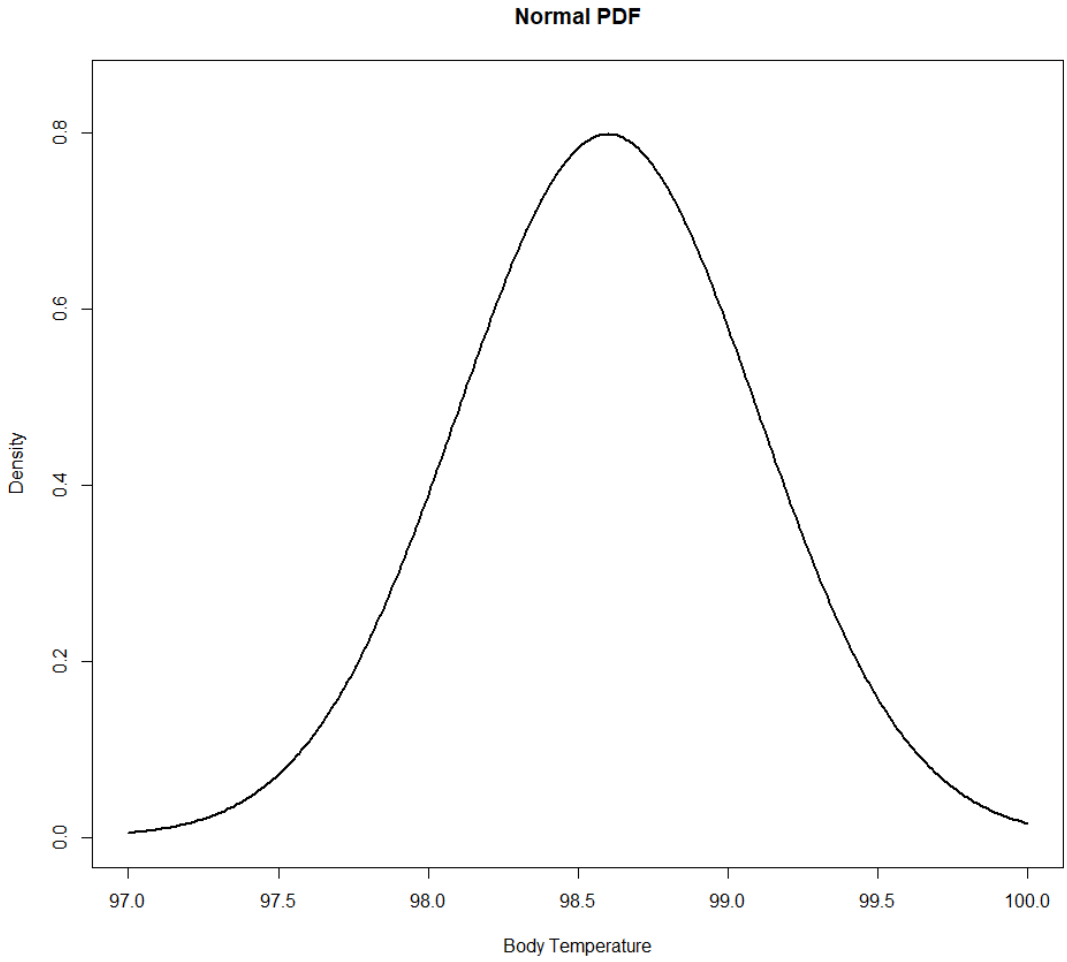

Figure 21. *Gaussian (Normal) distribution with simulated data graph*

Explanation of the graph

Figure 21 represents the observed data distribution of body temperature among individuals infected with influenza. The probability density function (PDF) plot shows the density of observing a specific body temperature based on the normal distribution.

2.5.4 Beta-Binomial Distribution

Beta-Binomial Distribution: The Beta-Binomial distribution is a discrete probability distribution that combines the Beta distribution (which models the probability of success) with the Binomial distribution (which models the number of successes in a fixed number of trials).

The probability mass function of the Beta-Binomial distribution is given by:

$$P(X=k) = (nCk) * (B(k + \alpha, n - k + \beta) / (B(\alpha, \beta)))$$

where:

P(X=k) represents the probability of observing k successes in n trials.

(nCk) denotes the number of combinations of n objects taken k at a time.

B(a, b) represents the Beta function.

α and β are the shape parameters of the Beta distribution.

Example

Modeling the proportion of individuals vaccinated against measles in different age groups in a population-based survey. The beta-binomial distribution captures the heterogeneity in vaccination rates.

Beta-Binomial Distribution Illustration:

The code demonstrates modeling the proportion of individuals vaccinated against measles using the beta-binomial distribution. Synthetic data is generated with specified shape parameters and the number of trials. A visualization is utilized to depict the distribution: a bar plot illustrates the count of individuals based on their vaccination status. This example underscores the application of the beta-binomial distribution to model vaccination proportions and visualize their distribution (see Appendix 2.5.4 for additional insights).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

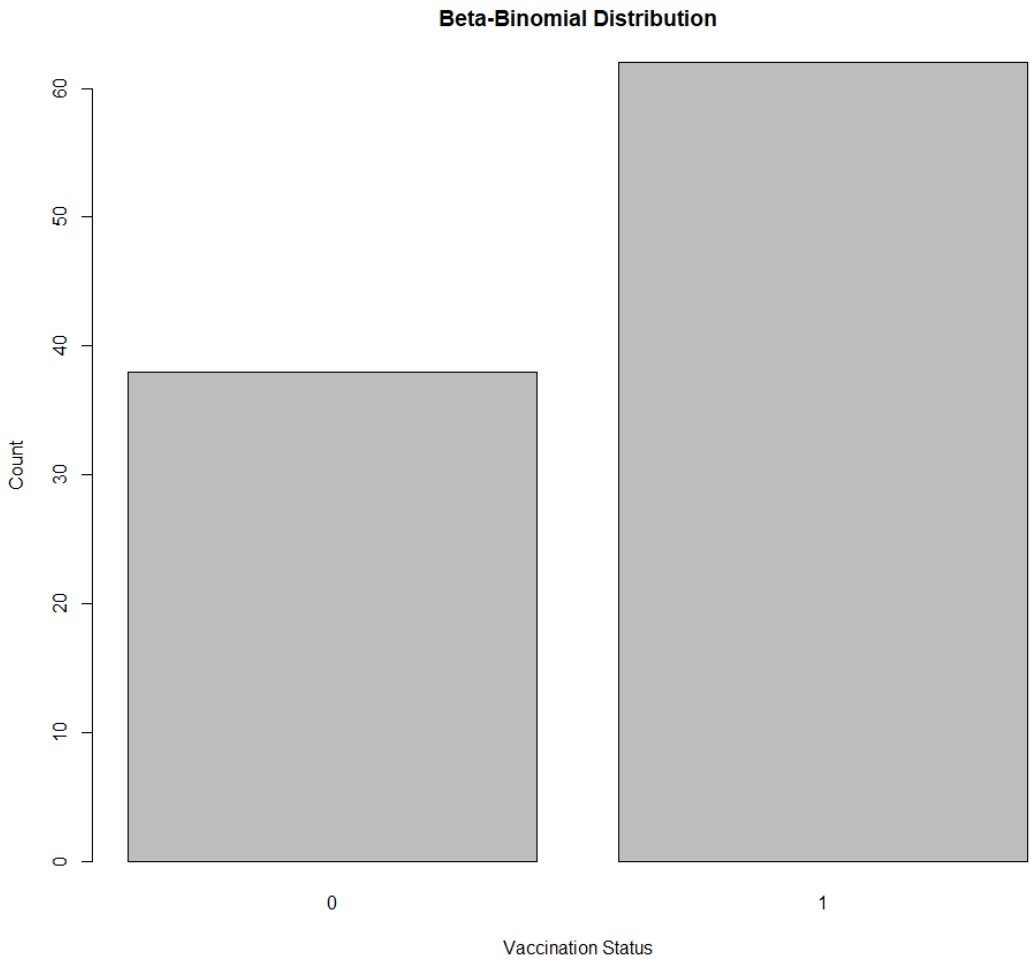

Figure 22. *Beta-Binomial distribution with simulated data graph*

Explanation of the graph

Figure 22 represents the observed data distribution of the proportion of individuals vaccinated against measles. It shows the count of individuals categorized as vaccinated or not vaccinated.

2.5.5 Gamma Distribution

Gamma Distribution: The Gamma distribution is a continuous probability distribution that is often used to model positive-valued data or durations. It has a shape parameter (α) and a rate parameter (β).

The probability density function of the Gamma distribution is given by:

$$f(x) = (\beta^{\alpha} / \Gamma(\alpha)) * x^{(\alpha-1)} * e^{(-\beta x)}$$

where:

f(x) represents the probability density at the point x.

β^{α} / $\Gamma(\alpha)$ is a normalization constant.

$\Gamma(\alpha)$ represents the Gamma function.

e is Euler's number (approximately 2.71828).

Example

Modeling the duration of symptoms for patients with typhoid fever. The gamma distribution is commonly used to model positive-valued continuous variables, such as disease duration.

Gamma Distribution Illustration:

The provided R code exemplifies modeling the duration of symptoms for patients with typhoid fever using the gamma distribution. Synthetic data is generated with specified shape and rate parameters. Visualizations are employed to represent the distribution: a histogram displays the data distribution, and a Probability Density Function (PDF) plot demonstrates the probability density of various symptom durations. This example highlights the application of the gamma distribution to model durations of symptoms and visualize their probabilistic characteristics (see Appendix 2.5.5 for further context).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

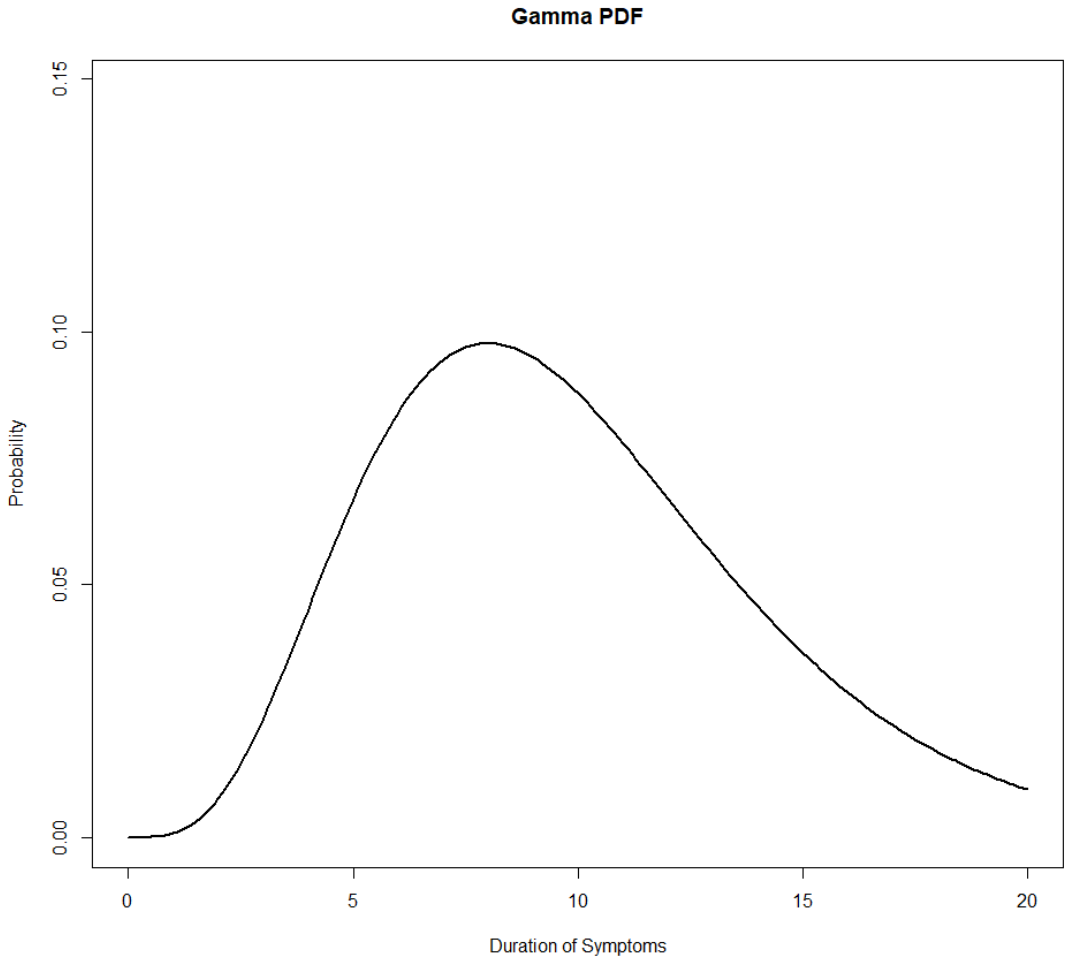

Figure 23. *Gamma distribution with simulated data graph*

Explanation of the graph

Figure 23 represents the observed data distribution of the duration of symptoms for patients with typhoid fever. The probability density function (PDF) plot shows the probability of observing a specific duration of symptoms based on the gamma distribution.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.5.6 Weibull Distribution

The Weibull distribution is a continuous probability distribution used to model lifetimes or durations of events. It has two parameters: the shape parameter (k) and the scale parameter (λ).

The probability density function of the Weibull distribution is given by:

$$f(x) = (k / \lambda) * (x / \lambda)^{(k-1)} * e^{(-(x / \lambda)^k)}$$

where:

f(x) represents the probability density at the point x.

k is the shape parameter.

λ is the scale parameter.

e is Euler's number (approximately 2.71828).

Example

Modeling the time to recovery for patients with cholera in a clinical trial. The Weibull distribution is often employed to analyze time-to-event data in epidemiology.

Weibull Distribution Illustration:

The code illustrates modeling the time to recovery for patients with cholera using the Weibull distribution. Synthetic data is generated with specified shape and scale parameters. Visualizations are utilized to depict the distribution: a histogram showcases the data distribution, while a Probability Density Function (PDF) plot demonstrates the probability density of various recovery times. This example underscores the application of the Weibull distribution to model recovery times and visualize their probabilistic characteristics (see Appendix 2.5.6 for additional insights).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

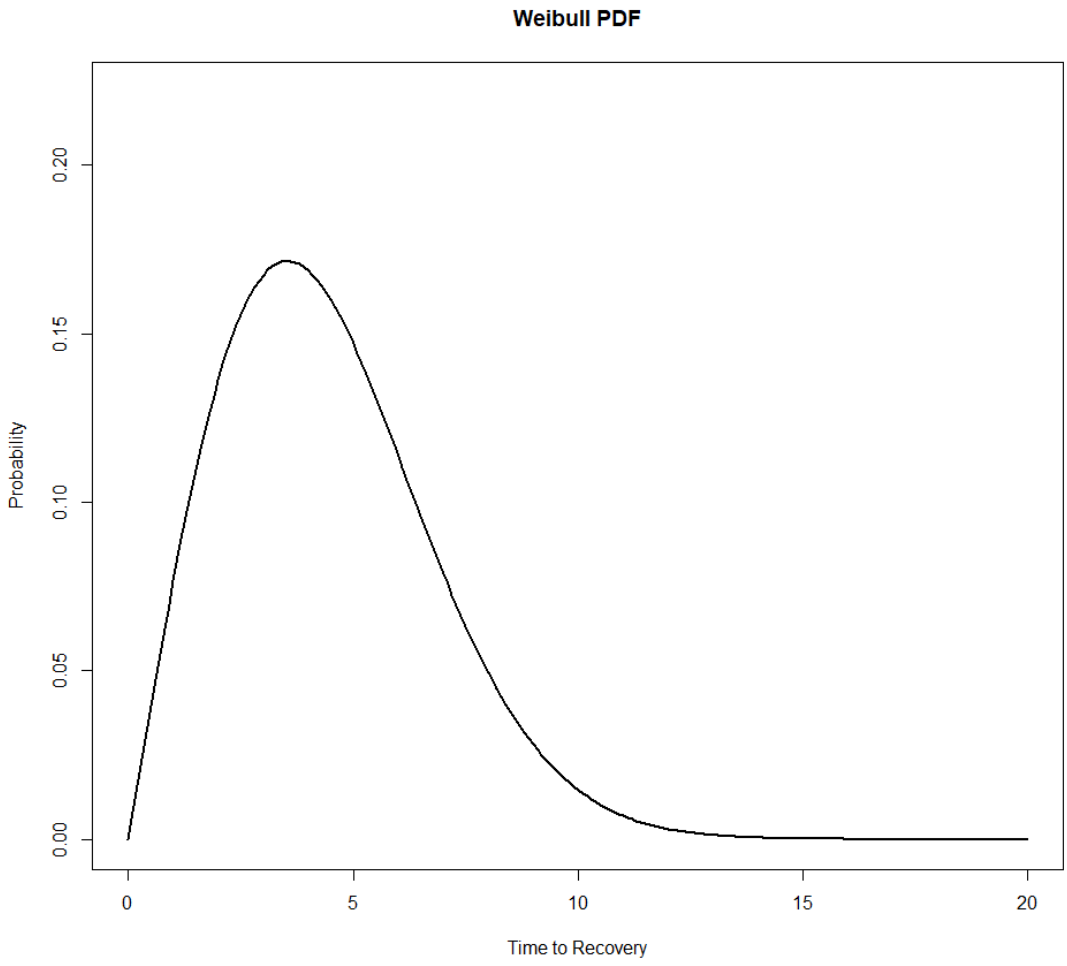

Figure 24. *Weibull distribution with simulated data graph*

Explanation of the graph

Figure 24 represents the observed data distribution of the time to recovery for patients with cholera. The probability density function (PDF) plot shows the probability of observing a specific time to recovery based on the Weibull distribution

2.5.7 Log-Normal Distribution

The Log-Normal distribution is a continuous probability distribution commonly used to model variables that are the exponential of a normally distributed random variable.

The probability density function of the Log-Normal distribution is given by:

$$f(x) = (1 / (x\sigma\sqrt{(2\pi)})) * e^{(-(\ln(x)-\mu)^2 / (2\sigma^2))}$$

where:

f(x) represents the probability density at the point x.

μ is the mean of the natural logarithm of the variable.

σ^2 is the variance of the natural logarithm of the variable.

e is Euler's number (approximately 2.71828).

π is a mathematical constant (approximately 3.14159).

Example

Modeling the distribution of incubation periods for norovirus infection in a community outbreak. The log-normal distribution accommodates positively skewed variables commonly observed in infectious disease studies.

Log-Normal Distribution Illustration:

This code exemplifies modeling the distribution of incubation periods for norovirus infection using the log-normal distribution. Synthetic data is generated based on a specified mean and standard deviation of the logarithm of incubation periods. Visualizations are employed to represent the distribution: a histogram showcases the data distribution, and a Probability Density Function (PDF) plot illustrates the density of various incubation periods. This example emphasizes the application of the log-normal distribution to model incubation periods and visualize their probabilistic characteristics (see Appendix 2.5.7 for further context).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

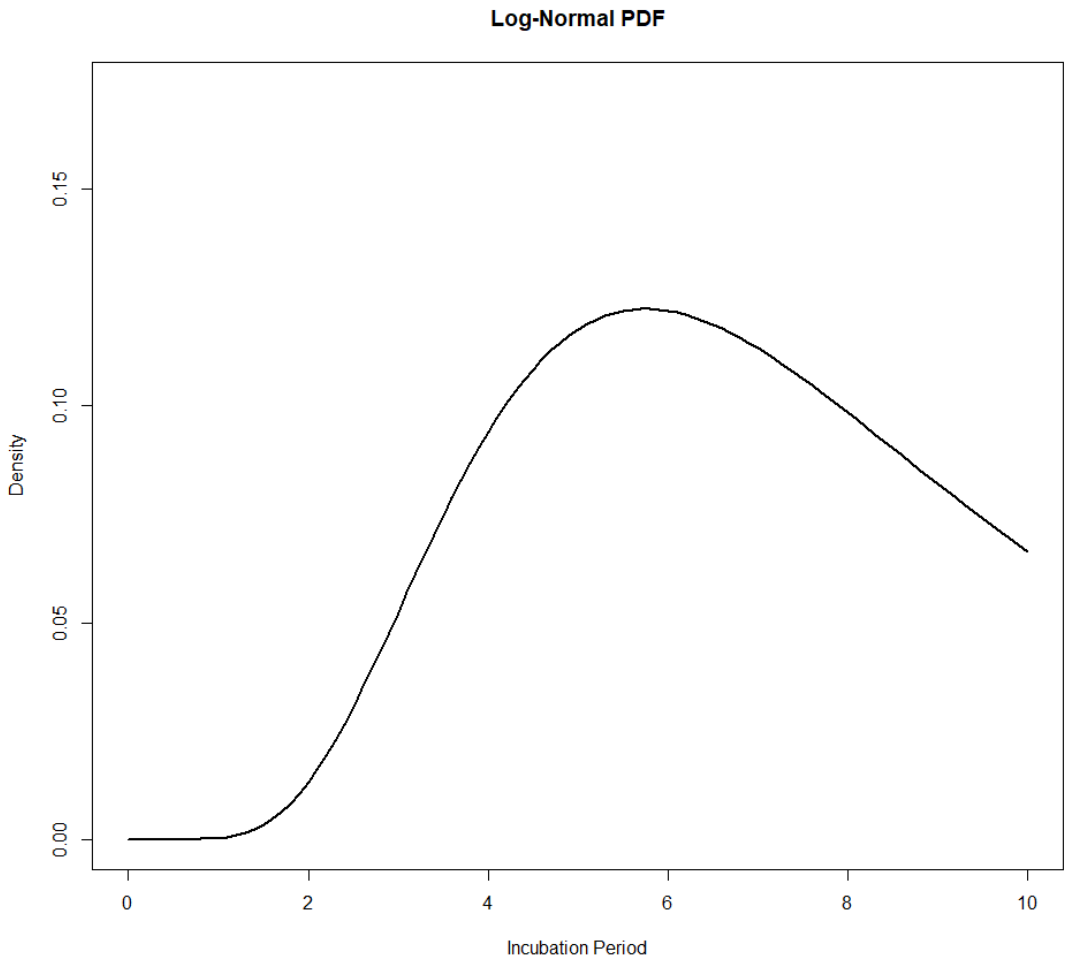

Figure 25. *Log-Normal distribution with simulated data graph*

Explanation of the graph

Figure 25 represents the observed data distribution of the incubation periods for norovirus infection. The probability density function (PDF) plot shows the density of observing a specific incubation period based on the log-normal distribution.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2.5.8 Binomial Distribution

The Binomial distribution is a discrete probability distribution used to model the number of successes in a fixed number of independent Bernoulli trials, each with a constant probability of success, p.

The probability mass function of the Binomial distribution is given by:

P(X=k) = (nCk) * p^k * (1-p)^(n-k)

where:

P(X=k) represents the probability of observing k successes in n trials.

(nCk) denotes the number of combinations of n objects taken k at a time.

p is the probability of success in each Bernoulli trial.

Example

Modeling the probability of HIV infection among intravenous drug users. The binomial distribution is used to analyze binary outcomes, such as infection status.

Binomial Distribution Illustration:

The provided code demonstrates modeling the probability of HIV infection among intravenous drug users using the binomial distribution. Synthetic data is generated based on a specified number of trials and the probability of success. A visualization in the form of a bar plot is employed to depict the distribution, displaying the count of infections based on the number of occurrences. This example highlights the application of the binomial distribution to model infection probabilities and visualize their distribution (see Appendix 2.5.8 for additional insights)

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

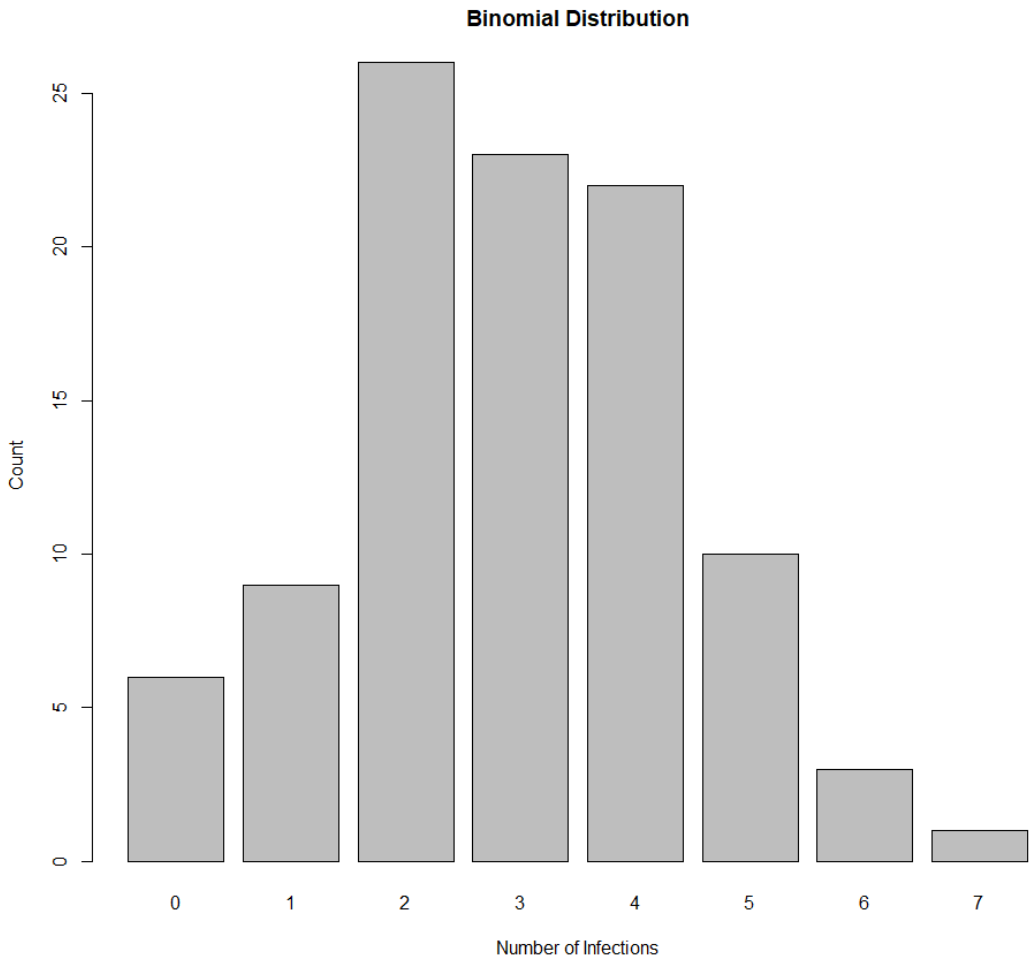

Figure 26. *Binomial distribution with simulated data graph*

Explanation of the graph

The bar plot in figure 26 represents the observed data distribution of the number of HIV infections among intravenous drug users. It shows the count of individuals categorized by the number of infections.

2.5.9 Multinomial Distribution

Multinomial Distribution: The Multinomial distribution is a discrete probability distribution used to model the probability of observing different outcomes in a categorical variable with multiple categories. It generalizes the Binomial distribution to more than two categories.

The probability mass function of the Multinomial distribution is given by:

$P(X_1=k_1, X_2=k_2, ..., X_k=k_k) = (nCk_1,k_2,...,k_k) * p_1^k_1 * p_2^k_2 * ... * p_k^k_k$

where:

$P(X_1=k_1, X_2=k_2, ..., X_k=k_k)$ represents the probability of observing k_1 occurrences of outcome 1, k_2 occurrences of outcome 2, and so on.

$(nCk_1,k_2,...,k_k)$ denotes the multinomial coefficient, which is the number of combinations of n objects with k_1 objects of outcome 1, k_2 objects of outcome 2, and so on.

$p_1, p_2, ..., p_k$ represent the probabilities of each outcome.

Example

Modeling the distribution of disease severity (mild, moderate, severe) among patients with dengue fever. The multinomial distribution is used when analyzing categorical outcomes with more than two categories.

Multinomial Distribution Illustration:

The R code showcases modeling the distribution of disease severity among patients with dengue fever using the multinomial distribution. Synthetic data is generated based on specified probabilities for mild, moderate, and severe disease categories. A bar plot visualizes the distribution, presenting the count of patients with different levels of disease severity. This example underlines the application of the multinomial distribution to model disease severity proportions and visualize their distribution (see Appendix 2.5.9 for additional context).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

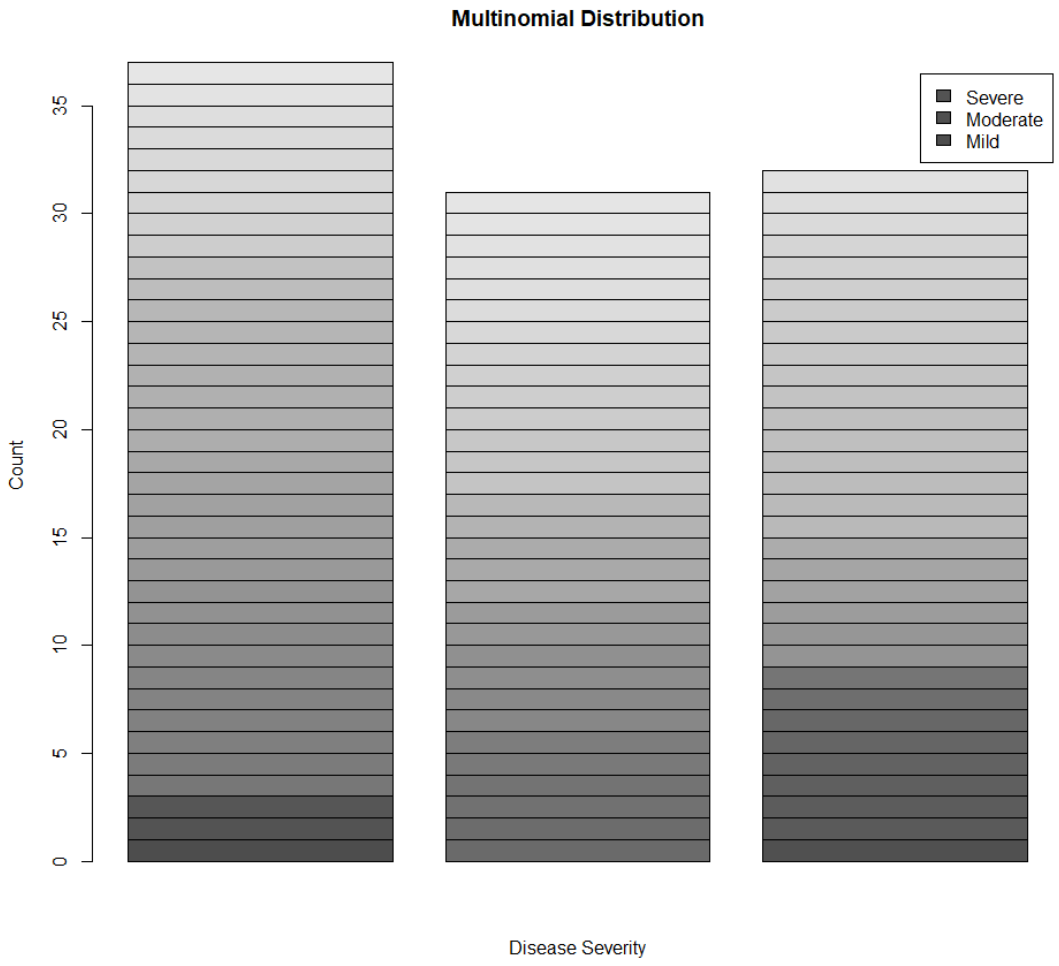

Figure 27. *Multinomial distribution with simulated data graph*

Explanation of the graph

The bar plot in figure 27 represents the observed data distribution of disease severity among patients with dengue fever. It shows the count of individuals categorized by the severity of the disease.

2.5.10 Dirichlet Distribution

The Dirichlet distribution is a continuous probability distribution used to model random variables that represent probabilities or proportions over a fixed number of categories. It is often used as a prior distribution in Bayesian analysis.

The probability density function of the Dirichlet distribution is given by:

$f(x_1, x_2, ..., x_k) = (\Gamma(\alpha_0) / (\Gamma(\alpha_1) * \Gamma(\alpha_2) * ... * \Gamma(\alpha_k))) * x_1^{(\alpha_1-1)} * x_2^{(\alpha_2-1)} * ... * x_k^{(\alpha_k-1)}$

where:

$\mathbf{f(x_1, x_2, ..., x_k)}$ represents the probability density at the point $\mathbf{(x_1, x_2, ..., x_k)}$.

$\mathbf{\Gamma(\alpha)}$ represents the Gamma function.

$\mathbf{\alpha_0 = \alpha_1 + \alpha_2 + ... + \alpha_k}$ is the sum of the shape parameters.

$\mathbf{\alpha_1, \alpha_2, ..., \alpha_k}$ are the shape parameters that determine the concentration of probabilities for each category.

$\mathbf{x_1, x_2, ..., x_k}$ represent the probabilities or proportions for each category.

Example

Modeling the distribution of different serotypes of poliovirus in a population over time. The Dirichlet distribution is useful for analyzing proportions when multiple categories sum to a constant.

These distributions, when applied to real-life examples, enable epidemiologists to gain insights into various aspects of epidemic diseases, such as incidence, severity, duration, and risk factors.

Dirichlet Distribution Illustration:

The code demonstrates modeling the distribution of different serotypes of poliovirus using the Dirichlet distribution. The 'ggtern' package is used to visualize the distribution in a ternary plot. Synthetic data is generated based on specified proportions of serotypes and normalized to ensure they sum up to 1. The ternary plot showcases the distribution of the three serotypes. This example

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

underscores the application of the Dirichlet distribution to model proportions in a ternary space and visually represent their distribution (see Appendix 2.5.10 for additional insights).

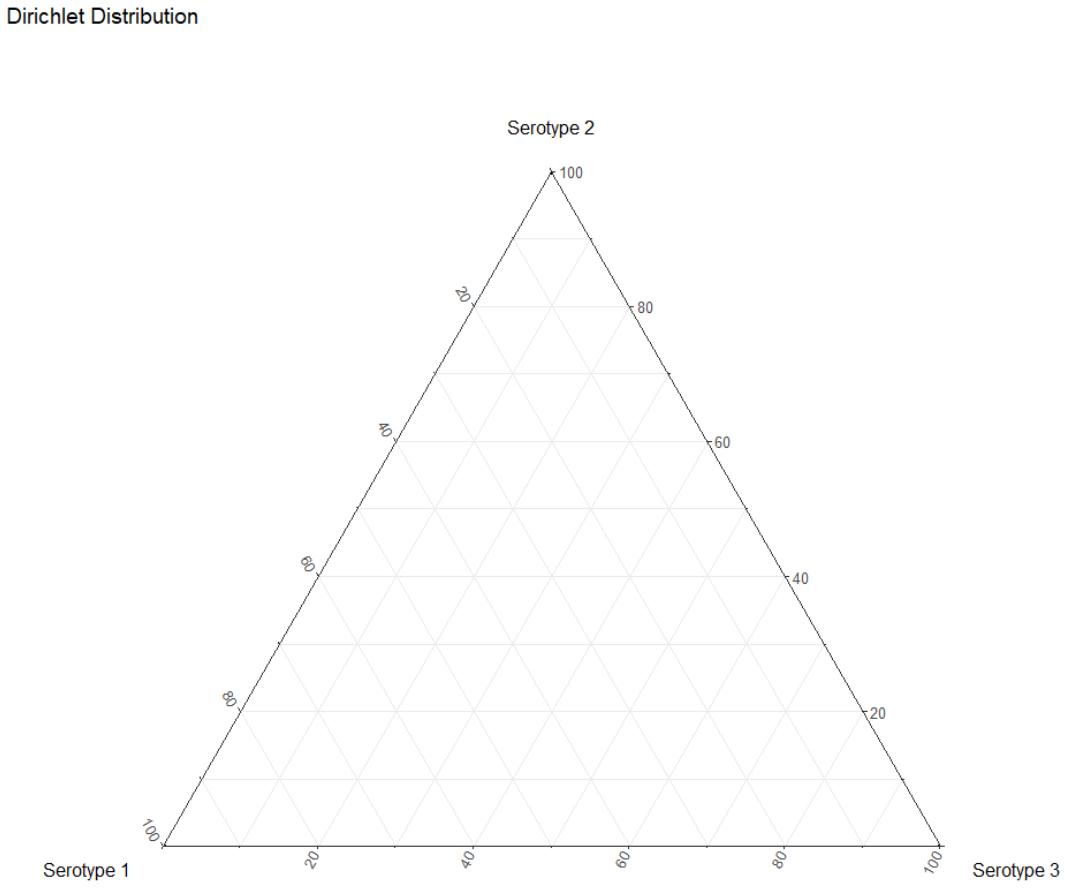

Figure 28. *Dirichlet distribution with simulated data graph*

Explanation of the graph

In this code, i generated a set of random samples from a Dirichlet distribution to approximate the distribution of serotypes. Each sample represents a set of proportions for the three serotypes (Serotype 1, Serotype 2, and Serotype 3). The positions of the points within the triangle indicate the relative proportions of each serotype in a given sample.

Interpreting figure 28: Each point within the triangle represents a specific combination of proportions for the three serotypes. The vertices of the triangle correspond to the pure components, where one serotype is present at 100% and the others are absent. The interior of the triangle represents mixtures of the serotypes, where all three serotypes are present in varying proportions. Points along the edges of the triangle represent binary mixtures of the serotypes. The plot allows us to visualize the variability and distribution of the serotypes within the population. We can observe the range of proportions and the clustering of points, which indicates the patterns and heterogeneity in the distribution of the serotypes.

It's important to note that in the given example, we approximated the Dirichlet distribution using a multinomial distribution. The accuracy of the approximation depends on the sample size and the underlying assumptions of the multinomial distribution.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3 PREDICTIVE MODELS FOR THE DENGUE FEVER CASES IN FRANCE (2008-2020)

3.1 EXPLORE OUR DATA

This brief chapter presents a graph illustrating of our Dengue fever data in France.

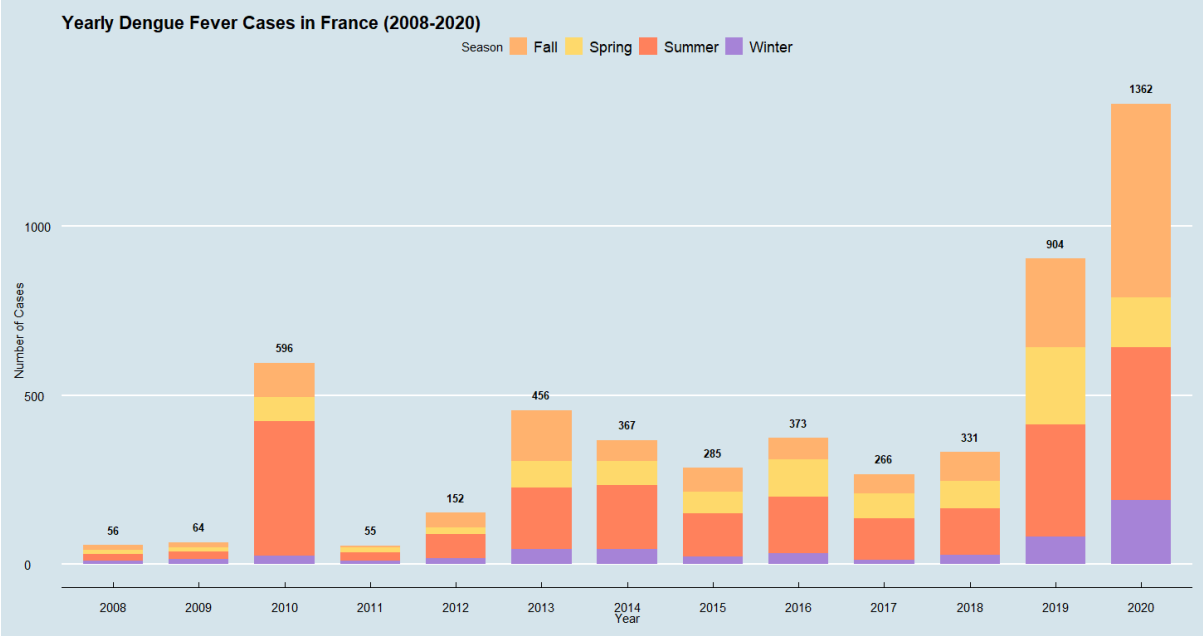

***Figure 29**. Dengue fever French data basic illustration*

- Time series count data: This refers to the type of data we have, which consists of counts recorded at regular time intervals (in our case, monthly intervals)
- Monthly data: This specifies the frequency of our data collection. In our dataset, we have recorded counts on a monthly basis.
- Seasonality by observation: Each observation (in this case, each month) show different counts, influenced by seasonal factors. For example, infectious disease cases increase during certain months (summer or fall) due to weather conditions or other seasonal factors.

3.2 KEY POINTS ABOUT OUR PREDICTIVE MODELS

Before we embark on a detailed exploration of each individual model, let's begin by highlighting some key points about our predictive models. These insights serve as a foundational understanding that will guide our journey into the intricacies of each approach. By starting with this overview, we can better appreciate the nuances and significance of the models we'll delve into.

HHH4:

- Focuses on using only the incidence from the directly preceding month to predict the next month.
- Potential limitation: The serial interval (time between successive cases in a chain of transmission) varies across diseases.
- Endemic-epidemic model.
- Simplicity: HHH4 is relatively simple, with fewer parameters and reduced computational time compared to some other models

HHH4 with Weighted Schemes:

- A variant of HHH4 that incorporates weighted schemes for past incidences.
- The weighting is inspired by different shapes of the serial interval distribution, providing more flexibility like poisson, geometrical e.t.c.
- This can help in accounting for and potentially mitigating certain dependencies in the data.

KCDE (Kernel Conditional Density Estimation):

- Goal is to obtain predictions distributions related to incidences after time t, using time series incidence and time series of covariates up to time t.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

- The density estimate for an observation at a future time, conditional on covariates, is a weighted combination of contributions from previous observations with similar covariate values.
- Used to obtain separate predictive distributions for incidence in each remaining week of the season.
- KCDE advantage: Offers the capability to condition forecasts on several preceding observations mitigating certain dependencies in the data.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.3 HHH4 MODEL (ENDEMIC – EPIDEMIC MODELLING OF A REAL COUNT TIME SERIES)

3.3.1 Brief history, methodology & parametriazation (Held, Michael Hohle and Hofmann, 2005, (67))

Because of the infectious diseases there's been a significant surge in interest concerning the statistical analysis of multivariate time series data.

At first the statistical analysis of these data was mainly done with

- Computer-intensive Markov chain Monte Carlo (MCMC) methods (e.g., Mugglin., 2002; Svensson and Lindbäck, 2002; Knorr-Held and Richardson, 2003).
- Empirical Bayes techniques for time-space disease surveillance by Böhning (2003) were introduced later.

Approaches for analyzing this kind of data often rely on mathematical modeling, with the log-linear Poisson regression model serving as a common tool. This model effectively manages the inherent overdispersion in such data sets, enabling researchers to interpret the occurrence of disease with the aid of latent variables, which remain unknown but can be inferred from the observed data. These latent variables may exhibit both time-dependent and serial dependence characteristics.

Adjustments for seasonal correlation - that is, predictable fluctuations that recur every calendar year - can be executed with quasi-likelihood methods (Zeger in 1988 and later developed further by Zeger and Qaqish in the same year). To explain briefly, we might presume that the count (y_t) at a given time (t=1,…,n) is Poisson-distributed, where the mean is given by the exponential of the poisson mean (η_t).

$$\eta_t = \alpha + \beta t + \sum_{s=1}^{S} \left(\gamma_s \sin(\omega_s t) + \delta_s \cos(\omega_s t)\right) \quad (1.1)$$

- S denotes the number of harmonics incorporated in the analysis, with ω_s representing the Fourier frequencies. For monthly data, these frequencies could be defined as $\boldsymbol{\omega_s = 2s\pi/12}$, as suggested by Diggle (1990)

According to the nomenclature introduced by Cox (1981), these models are categorized as parameter-driven, meaning that they are influenced by a set of independent variables.

Similar parameter-driven models are used in state-space and other non-stationary models, with suitable prior distributions on the latent parameters. In these models, latent parameters are allowed to change over time, following the theories outlined by Jørgensen., 1999 and Fahrmeir and Knorr-Held, 2000.

The problem with these solo parameter-driven methods is that they:

Often fail to accurately capture localized epidemics, which is crucial for infectious disease surveillance data.

As a result, model expansions incorporating additional factors were seen as necessary. A beneficial approach involves introducing the count of prior cases y_{t-1} as extra explanatory variables within the model. According to Cox's (1981) terminology, this addition transforms the model into an observation-driven one.

Therefore, the model can now be considered both parameter-driven and observation-driven when combined with the primary equation. However, this adds a degree of complexity. If the observed counts y_{t-1} are integrated into the linear predictor of (1.1), it suggests that y_t follows a Poisson distribution with mean μ_t, given by:

$$\mu_t = \exp(\eta_t + \lambda y_{t-1})$$

- Here, η_t is the linear predictor from the original model, λ is a new parameter to be estimated, and y_{t-1} is the observed count at time t-1.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

This mathematical formalization provides a comprehensive framework for analyzing multivariate time series of counts.

This model though only describes a negative association and is incapable of demonstrating a positive association without growing exponentially over time, as per Diggle., 2002 (Section 10.4). So Zeger and Qaqish (1988) proposed a modification where the logarithm of the observed counts, with $\log 0$ is replaced by $\log d, 0 < d < 1$, minus the linear predictor η_{t-1} of y_{t-1}, enters the model as an explanatory variable, that is:

$$\mu_t = \exp\big(\eta_t + \lambda(\max(\log y_{t-1}, \log d) - \eta_{t-1})\big).$$

This model can be seen as a size-dependent branching process (Diggle., 2002) and has better properties; in particular, it allows for positive association between successive counts. However, the introduction of the parameter d and the regularization of past counts y_{t-1} through η_{t-1} are complicated and seem to be slightly unnatural. Interpretation of the autoregressive parameter λ is not straightforward in this formulation. Alternatively, Knorr-Held and Richardson (2003) let the logarithm of $1 + y_{t-1}$ enter as an explanatory variable. They avoid regularization by modulating the dependence on the previous counts by latent $0 - 1$ indicators, which are assumed to follow a two-stage hidden Markov model.

So Held., (2005) suggest the following model for the analysis of univariate time series of infectious disease counts $\{y_t; t = 1, \dots, T\}$. The counts are assumed to be Poisson distributed with conditional mean

$$\mu_t = \lambda y_{t-1} + \nu_t, \ (\lambda, \nu_t > 0)$$

where λ and ν_t are unknown quantities. The mean incidence is decomposed additively into two components: an epidemic or autoregressive component λy_{t-1}, and an endemic component ν_t. The former should be able to capture occasional outbreaks whereas the latter explains a baseline rate of cases with stable temporal pattern. Held., (2005) suggest the following parametric model for the endemic component:

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

$$\log(\nu_t) = \alpha + \beta t + \left\{\sum_{s=1}^{S} \gamma_s \sin(\omega_s t) + \delta_s \cos(\omega_s t)\right\},$$

where α is an intercept, β is a trend parameter, and the terms in curly brackets are used to model seasonal variation. Here, γ_s and δ_s are unknown parameters, S denotes the number of harmonics to include, and $\omega_s = 2\pi s/$ freq are Fourier frequencies (e.g. freq = 12 for monthly data). For ease of interpretation, the seasonal terms can be written equivalently as

$$\gamma_s \sin(\omega_s t) + \delta_s \cos(\omega_s t) = A_s \sin(\omega_s t + \varphi_s)$$

with amplitude $A_s = \sqrt{\gamma_s^2 + \delta_s^2}$ describing the magnitude, and phase difference $\tan(\varphi_s) = \delta_s/\gamma_s$ describing the onset of the sine wave. To account for overdispersion, the Poisson model may be replaced by a negative binomial model. Then, the conditional mean μ_t remains the same but the conditional variance increases to $\mu_t(1 + \mu_t\psi)$ with additional unknown overdispersion parameter $\psi > 0$.

The model is extended to multivariate time series $\{y_{it}\}$ in Held., (2005) and Paul., (2008) by including an additional neighbor-driven component, where past cases in other (neighboring) units also enter as covariates. This model also assumes that Yit has a negative binomial distribution and his conditional mean μ_{it} is then given by

$$\mu_{it} = \lambda y_{i,t-1} + \phi \sum_{j \neq i} w_{ji} y_{j,t-1} + e_{it}\nu_t$$

where the unknown parameter ϕ quantifies the influence of other units j on unit i, w_{ji} are weights reflecting between-unit transmission and e_{it} corresponds to an offset (such as population fractions at time t in region i). A simple choice for the weights is $w_{ji} = 1$ if units j and i are adjacent and 0 otherwise.

When analyzing a specific disease observed in, say, multiple regions or several pathogens (such as influenza and meningococcal disease), the assumption of equal incidence levels or disease transmission across units is questionable. To address such heterogeneity, the unknown quantities λ, ϕ, and ν_t in (2) may also depend on unit i. This can be done via

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

unit-specific fixed parameters, e.g. $\log(\lambda_i) = \alpha_i$ (Paul., 2008);

unit-specific random effects, e.g $\log(\lambda_i) = \alpha_0 + a_i, a_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_\lambda^2)$ Paul and Held, 2011);

linking parameters with known (possibly time-varying) explanatory variables, e.g. $\log(\lambda_i) = \alpha_0 + x_i\alpha_1$ with region-specific vaccination coverage x_i (Herzog., 2011).

In general, the parameters of all three model components may depend on both time and unit. A call to hhh4 in R language fits a Poisson or negative binomial model with conditional mean:

$$\mu_{it} = \lambda_{it} y_{i,t-1} + \phi_{it} \sum_{j \neq i} w_{ji} y_{j,t-1} + e_{it} \nu_{it}$$

to a (multivariate) time series of counts. Here, the three unknown quantities are modelled as log-linear predictors:

$$\begin{aligned} \log(\lambda_{it}) &= \alpha_0 + a_i + u_{it}^\top \alpha \\ \log(\phi_{it}) &= \beta_0 + b_i + x_{it}^\top \beta \\ \log(\nu_{it}) &= \gamma_0 + c_i + z_{it}^\top \gamma \end{aligned}$$

where $\alpha_0, \beta_0, \gamma_0$ are intercepts, α, β, γ are vectors of unknown parameters corresponding to covariate vectors u_{it}, x_{it}, z_{it}, and a_i, b_i, c_i are random effects. For instance, the model with $S = 1$ seasonal terms may be represented as $\boldsymbol{z}_{it} = (t, \sin(2\pi/\text{ freq } t), \cos(2\pi/\text{ freq } t))^\top$. The stacked vector of all random effects is assumed to follow a normal distribution with mean 0 and covariance matrix Σ.

If the model does not contain random effects, standard likelihood inference can be performed. Otherwise, inference is based on penalized quasilikelihood as described in detail in Paul and Held (2011).

Note for R language: For a single time series of counts Y_t , hhh4 can be regarded as an extension of glm.nb from package MASS (Ripley 2023) to account for autoregression. With multiple regions, spatio-temporal dependence is adopted (note that our data are of a single region so it doesn't affect us).

All this theory is implemented is the R package called "surveillance" and it will help us create our R code for implementation to our dengue French real data.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.3.2 Model Specification and Fitting

Because of the complexity of the R code implementation, below I'm going to explain in brief summary the process for better understanding. This is the first part of the code:

1. **Loading Required Libraries**: The code starts by loading the necessary R packages, including the "surveillance" and "lubridate" libraries.
2. **Preparing Data**: A dataset named "data" is created, containing two columns: "Time" representing time intervals and "NumValue" representing the corresponding numerical values. The "lubridate" package is used to generate a sequence of dates from 2008 to 2020.
3. **Data Transformation**: The "Year" column is extracted from the "Time" column using the "lubridate" package.
4. **Selecting Training Data**: Indices for training data are selected based on the years from 2008 to 2018.
5. **Transforming Data**: The data is transformed into the sts (space-time surveillance data) format, which is suitable for use with the "surveillance" package functions.
6. **Defining Model Specifications**: Different model specifications are defined, including various options for the type of statistical distribution ("family"), autoregressive terms ("S_ar"), endogenous terms ("S_end"), and lag of the autoregressive term ("lag_ar").
7. **Model Fitting Loop**: A loop iterates over each combination of model specifications. Inside the loop:
    - A "hhh4" model is fitted using the "surveillance" package's functions, based on the current specification.
    - One-step-ahead predictions are generated for each fitted model.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

- Prediction scores (log scores, ranked probability scores, Dawid-Sebastiani scores, spherical error scores) are calculated and their means are stored in the model_specifications dataframe.

8. **Storing Results**: The results, including the model specifications and fitted models, are stored in the "surveillance_fits" list.
9. **Saving Results**: The "surveillance_fits" list is saved as an RDS file in a specified location on the filesystem.

In summary, this code performs time series analysis using the "surveillance" package in R. It fits different statistical models with various specifications to the given time series data, generates predictions, calculates prediction scores, and stores the results for further analysis. The main goal is to evaluate different model specifications and understand their performance on the provided dataset (see appendix 3.3.2 for the full code implementation).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.3.3 Model Selection and Further Analysis

This next part of the code involves selecting the best-fitting model based on specified selection criteria (AIC, BIC, or log score) and then performing further analysis. Here's a recap of the additional code:

1. **Setting Prediction Horizons and Statistics**: We define the prediction horizons (number of time steps ahead to predict) and the prediction statistics we want to calculate for evaluating the model's performance.
2. **Identifying Prediction Time Indices**: We identify the indices of the data that correspond to the years specified for prediction.
3. **Converting Dates**: We convert the "Time" column in the data to the "time" column using the "ymd" function from the "lubridate" package.
4. **Adding Time Index Column**: We calculate a "time_index" column representing the number of days since the origin date (1970-01-01). This column will be used for calculating the periodic kernel.
5. **Loading Surveillance Fits**: We read the previously saved "surveillance_fits" object from the RDS file.
6. **Calculating AIC and BIC**: We calculate the AIC and BIC values for each fitted model in the "surveillance_fits" using the **summary(sfit)$AIC** and **summary(sfit)$BIC** commands.
7. **Selecting Best Model**: We determine the best-fitting model based on the specified selection criteria (AIC, BIC, or log score). Depending on the selected criteria, we identify the index of the best model specification and assign the corresponding fitted model to the "surveillance_fit" variable.

In summary, this continuation of the code focuses on evaluating the fitted models based on different selection criteria (AIC, BIC, or log score). It calculates the AIC and BIC for each model and selects the best-fitting model accordingly. The selected model will be used for further analysis and prediction tasks using the specified prediction horizons and statistics.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.3.4 Results

MEAN LOG SCORE

Bringing our model selection process to a conclusive point, we have a key performance indicator that highlights the chosen model's performance. The Mean Log Score, a measure of how well our model performs.

Mean Log Score: 4.92

But what is the mean log score and why it's important and what this result means for our model?

- The log mean score is calculated by taking the logarithm of the predicted probabilities or outcomes and then computing the mean of those values. It provides a measure of the average accuracy or quality of the model's predictions.
- A lower log mean score indicates better performance because it reflects a smaller average distance between the predicted probabilities or outcomes and the true values. In other words, a more accurate and precise model will have a lower log mean score.
- This 4.92 score tells us that our model is pretty good at capturing patterns and making accurate predictions within our dataset.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

These metrics offer a dual perspective that guides model selection by considering both goodness-of-fit and model complexity. AIC and BIC provide a comprehensive framework for weighing the effectiveness of various model options, enabling us to make well-informed decisions and refine our model selection process.

AIC (Akaike Information Criterion):

AIC: 910.08

AIC = -2 * log(L) + 2 * k

Where:

- **log(L)** is the log-likelihood of the data given the model.
- **k** is the number of parameters in the model.

The AIC seeks to strike a balance between how well a model fits the data and how complex the model is. It rewards models that fit the data well (larger log-likelihood values) while also considering the complexity of the model. The second term, 2 * k, represents a penalty for model complexity.

Mathematically, the AIC incorporates both the goodness-of-fit (measured by the log-likelihood) and the model's complexity. As the number of parameters k increases, the penalty term 2 * k becomes larger, favoring simpler models. This ensures that while a model might fit the data well, it must do so while also considering the trade-off with model complexity.

In summary, the AIC mathematically encapsulates the concept of balancing fit and simplicity. By considering both model fit and the number of parameters, the AIC guides model selection towards options that effectively capture the data's patterns without being overly complex.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

BIC (Bayesian Information Criterion):

BIC: 953.10

BIC = -2 * log(L) + k * log(n)

Where:

- **log(L)** is the log-likelihood of the data given the model.
- **k** is the number of parameters in the model.
- **n** is the number of data points.

The BIC rewards models that fit the data well (larger log-likelihood values) while also penalizing models with more parameters. The second term, **k * log(n)**, represents the penalty for model complexity. As **k** increases (more parameters), the penalty term becomes larger, favoring simpler models.

Mathematically, the BIC balances between goodness-of-fit and model complexity. A model with a better fit (higher log-likelihood) will have a smaller value for **-2 * log(L)**. However, if the model becomes too complex (higher **k**), the penalty term **k * log(n)** will increase, potentially offsetting the gain in fit. This encourages the selection of simpler models that can still provide a reasonable fit to the data.

In summary, the BIC mathematically embodies the principle of Occam's razor by penalizing model complexity. It finds a compromise between fit and simplicity, helping to avoid overfitting and selecting models that are likely to generalize better to new data.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

DAWID - SEBASTIANI SCORE: DSS

DSS: 7.12

Dawid-Sebastiani Score (DSS):

- The DSS measures the distance between the forecasted probability distribution and the observed outcome.
- It is a proper score, meaning it is optimized when the forecasted distribution matches the true distribution.
- Lower values of DSS indicate better forecast performance, as it reflects a smaller distance between the forecast and the true outcome.

SPHERICAL ERROR SCORE (SES)

SES: 199.86

Spherical Error Score (SES):

- The SES measures the distance between the forecasted point estimate and the observed outcome.
- It is a scale-dependent measure, meaning it is sensitive to the scale of the data.
- Lower values of SES indicate better forecast performance, as it reflects a smaller distance between the forecasted point estimate and the true outcome.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

RPS: 9.82

- The RPS assesses the quality of probabilistic forecasts by comparing the cumulative distribution function (CDF) of the forecast with the observed cumulative distribution.
- It quantifies the average discrepancy between the forecasted probabilities and the observed outcomes.
- Lower values of RPS indicate better forecast performance, as it reflects a smaller discrepancy between the forecasted probabilities and the observed outcomes.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

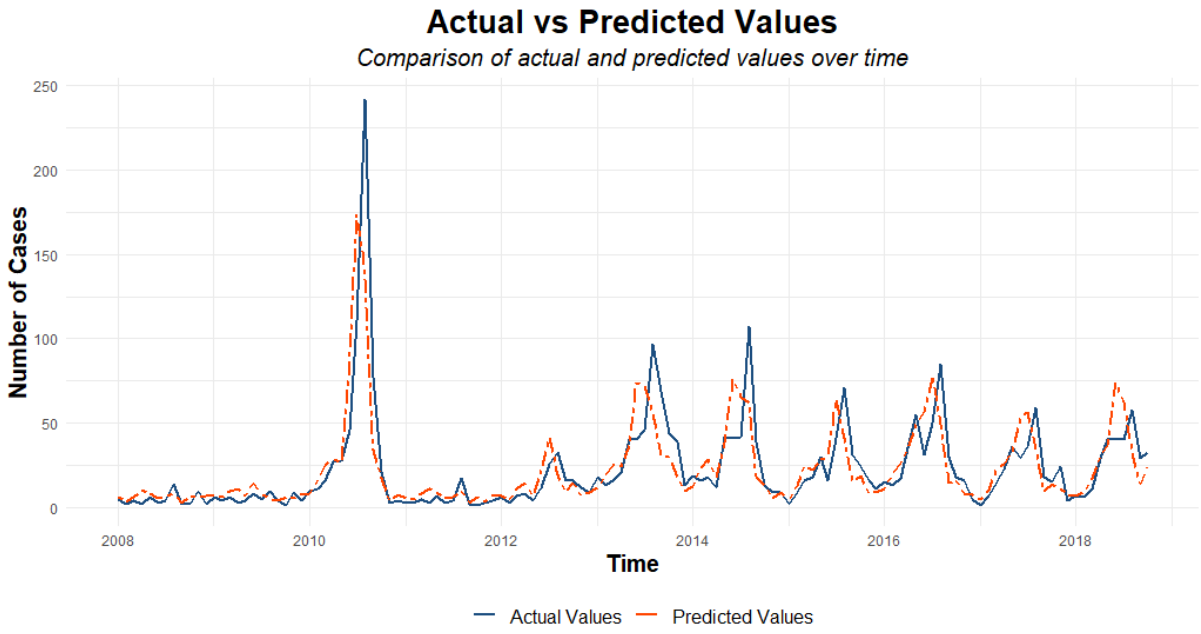

Figure 30. *Actual vs Predicted values for our first HHH4 model*

The line plot in figure 30 visualizes both the actual values and the predicted values of the number of cases over time.

1. **Actual values** are represented by the blue line, and it corresponds to the real data from the dataset.
2. **Predicted values** are represented by the orange dashed line, which shows the output of the fitted model.

Looking at the plot, we can assess how well the model's predictions align with the actual values. If the model is performing well, the blue and orange lines should be close to each other, indicating that the predicted values are close to the actual ones. However, any significant divergence between these two lines would suggest areas where the model's predictions are inaccurate. The plot provides a powerful way to visually assess the performance of the model over time. Note that the strong performance of the model in this plot can be attributed to its ability to incorporate all the previous forecasts when making one-step-ahead predictions. This feature allows the model to capture and leverage the temporal dependencies and patterns present in the data, resulting in accurate predictions that closely align with the actual values.

RMSE: 17.41

The RMSE is a popular metric to measure the differences between the predicted and actual values in a model. It squares the residuals before taking the mean to remove any negative values. This can make it more sensitive to large errors because they will contribute more to the total, due to squaring. The lower the RMSE, the better the model's performance. However, its value depends on the scale of our target variable, so it's most useful when comparing models predicting the same outcome.

This indicates that, on average, the model's predictions are about 17.41 units away from the actual values. Given that this metric is sensitive to large errors, it implies that the model may have made a few predictions that were significantly different from the actual values. The higher the RMSE, the larger the amount of error the model has made.

The squaring means larger errors have exponentially more impact, making RMSE more sensitive to outliers than MAE. Like MAE, a smaller RMSE is generally better.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

MEAN ABSOLUTE ERROR: MAE

MAE: 10.27

MAE is similar to RMSE, but it uses absolute values instead of squares, which is less sensitive to outliers than RMSE. This measures the average magnitude of the errors in a set of predictions, without considering their direction, It's simply the average absolute difference between predicted and actual values. The MAE is a linear score, meaning all individual differences are weighted equally in the average. Like RMSE, a lower MAE indicates a better model performance as it means predictions are closer to actual values, but what constitutes a "good" MAE depends on the context, such as the scale of our target variable. If our target variable ranges from 0 to 100, an MAE of 10.27 could be considered decent, but if it ranges from 0 to 1, then that MAE is quite high.

This tells us that on average, the model's predictions are about 10.27 units away from the actual values. The MAE is not as sensitive to large errors as the RMSE, so this is generally a more reliable measure of average prediction error.

MEAN ABSOLUTE PERCENTAGE ERROR: MAPE

MAPE: 0.65

Mean Absolute Percentage Error (MAPE): MAPE is another metric for measuring the accuracy of a model, but it expresses errors as a percentage, which can be easier to understand and interpret than absolute error quantities. It's particularly useful when we want to understand the error rate in terms of the actual values themselves, since it gives a simple, interpretable metric in the form of an average percentage error. A lower MAPE indicates a better model performance.

The MAPE indicates that the average error of the model's predictions is 65% of the actual values. This is a large percentage and suggests that the model's predictions may not be very accurate considering.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

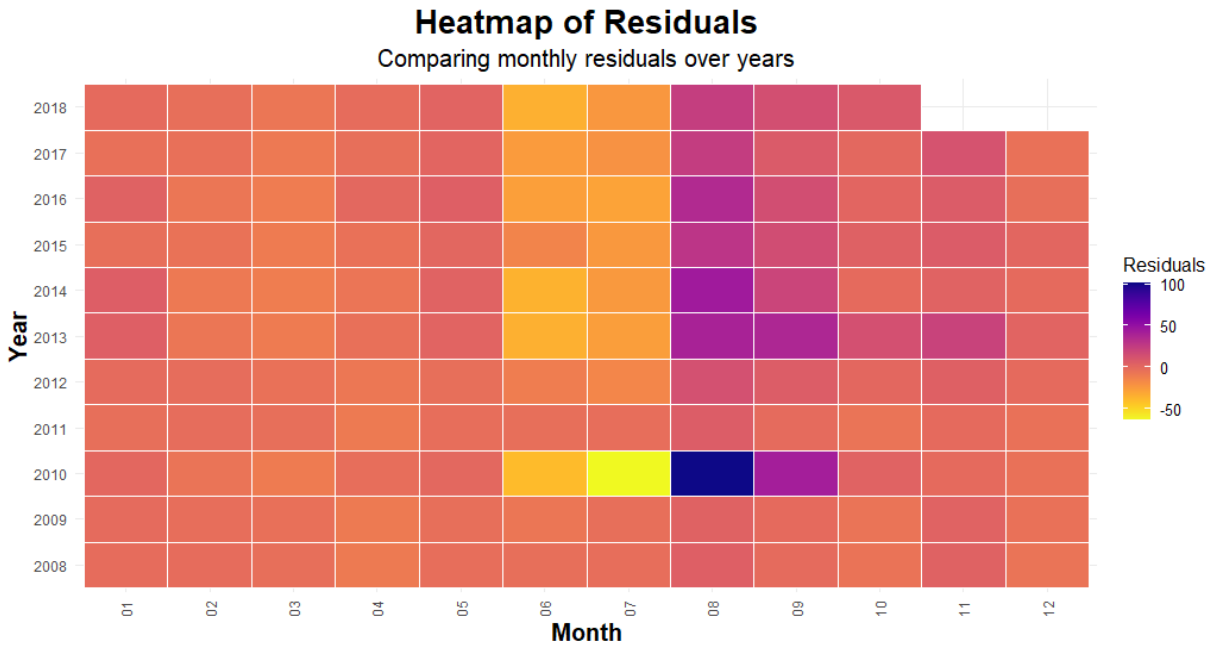

Figure 31. Heatmap of *residuals for our predictive HHH4 model*

With the heatmap in figure 31 we can visualize the residuals, which are the differences between the predicted and actual values. Residuals are an important measure for evaluating the performance of our model. Ideally, we'd want our residuals to be as small as possible, indicating that our model's predictions are close to the actual values.

Specifically in our heatmap:

The x-axis represents the 'Month', the y-axis represents the 'Year'. Each cell in the grid represents a particular month in a specific year.

The color of each cell corresponds to the value of the 'Residuals' for that month and year, as indicated by the color scale on the right. This color scale uses the 'viridis' palette, which provides a perceptually uniform scale of colors from light (low residuals) to dark (high residuals). This allows us to easily see where the residuals are highest (darkest cells) and lowest (lightest cells).

This plot allows us to see patterns in the residuals over time. If our model is performing consistently well, the colors should be roughly the same across all

months and years. If there are any noticeable patterns - for example, if certain months or years consistently have higher residuals - this could indicate areas where our model is less effective. Here we can notice some differences in the uniformity colors but nothing too dramatic.

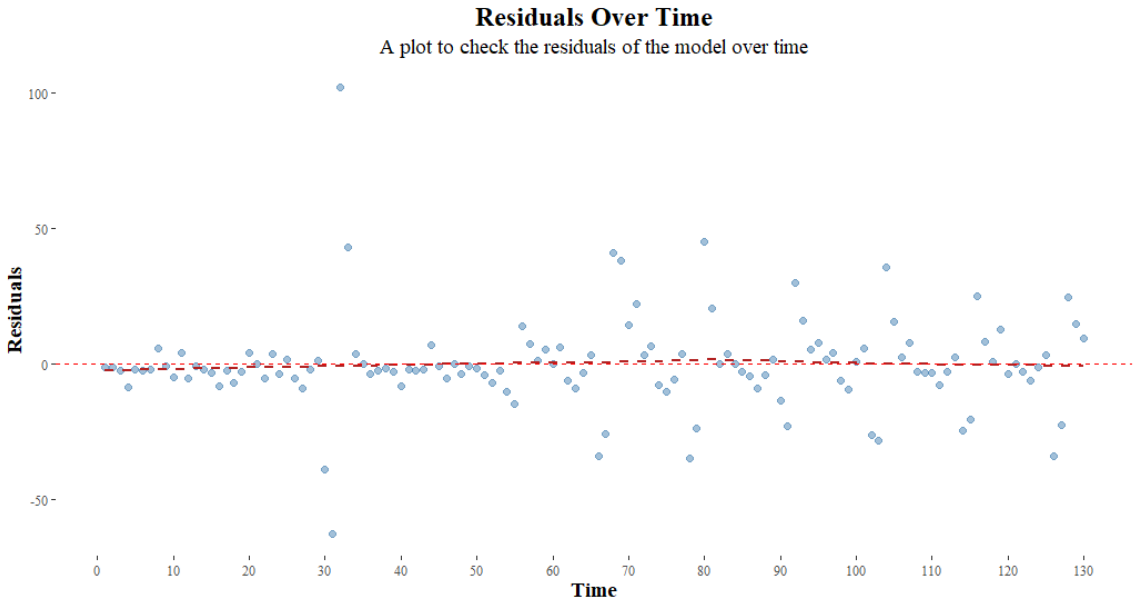

Figure 32. *Residuals over for our predictive HHH4 model*

This specific plot in figure 32 is a scatterplot of residuals over time. Residuals are the difference between observed and predicted values. In a good model, we would expect to see the residuals distributed randomly around zero, and not showing any clear patterns. This plot is used to check the assumptions of homoscedasticity (equal variance of residuals) and independence of errors.

1. **Time (x-axis):** The x-axis represents the time, or rather the sequence in which the data points occur.

2. **Residuals (y-axis):** The y-axis represents the residuals, which are the difference between the observed and predicted values. A residual of zero indicates a perfect prediction, positive values indicate under-predictions, and negative values indicate over-predictions.

3. **Red Horizontal Line:** The horizontal red dashed line at y = 0 represents the ideal residual value. If all the points (residuals) were

lying on this line, it would mean that our model has predicted all the values perfectly.

If there are patterns in the residuals - for example, if the residuals tend to be high at certain times and low at others - this could indicate that the model isn't capturing some of the time-dependent structure in the data, which might suggest avenues for improving the model. In our case again seems to have more than not values closer to 0 but again not completely.

ACF OF RESIDUALS

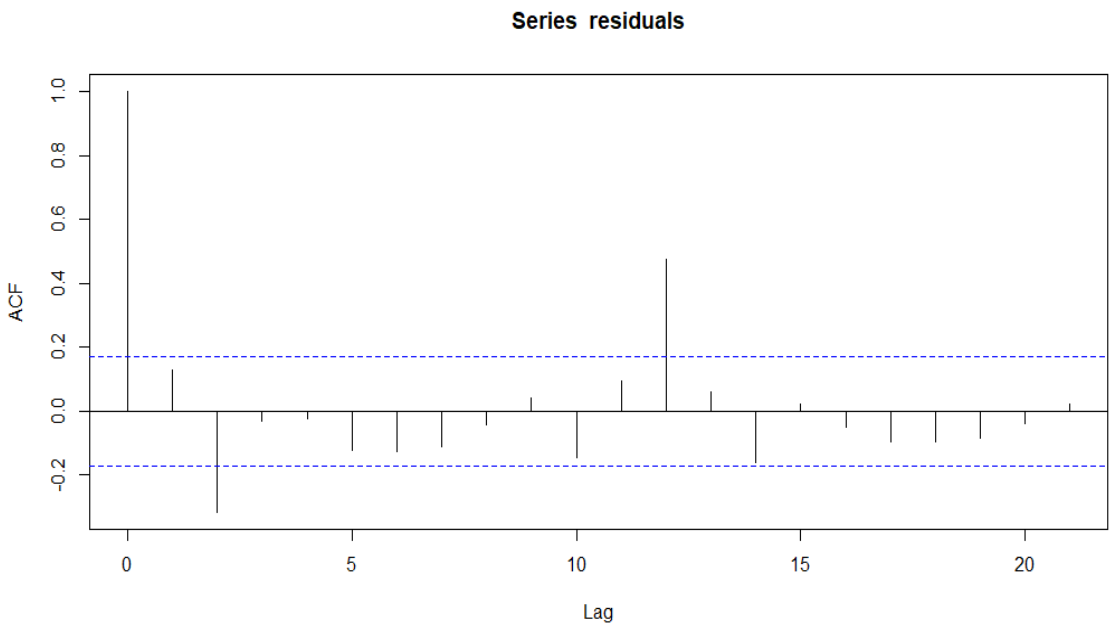

Figure 33. *ACF of residuals for our predictive HHH4 model*

When we are calling **acf(residuals)**, R calculates the correlation of the series with itself, lagged by x time units, for different values of x. The **acf()** function will then produce a plot of these correlations (the autocorrelations) against the lag times. This plot is known as the autocorrelation plot or correlogram.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

In the autocorrelation plot:

1. The **x-axis** represents the lag, i.e., the number of time points between the current observation and the previous observation.
2. The **y-axis** represents the autocorrelation coefficient, which can range from -1 to 1. A positive autocorrelation means that high values in the series tend to be followed by high values and low values tend to be followed by low values. A negative autocorrelation means that high values tend to be followed by low values and vice versa. A value near zero suggests no autocorrelation.
3. Each **vertical line (or bar)** on the plot corresponds to the autocorrelation coefficient at a specific lag. For example, the line at lag 1 shows the autocorrelation between all points and the points that immediately precede them (i.e., one time unit before them).
4. The **horizontal blue dashed lines** in the plot indicate the significance level. If the autocorrelation is within these lines, it is not significantly different from zero. In other words, it is likely that any correlation is due to chance. If the autocorrelation is above these lines, it is significantly different from zero, indicating that the autocorrelation is probably not due to chance.

For residuals of a well-fitted model, we would ideally want all autocorrelations for lags greater than zero to be near zero and not significantly different from zero. This would indicate that there is no autocorrelation in the residuals, suggesting that the model has adequately captured the information in the data. If, on the other hand, the plot shows one or more large autocorrelations, this suggests that there is information in the data that the model has not captured. In our example we can notice lags at 2 and 12 to run out of bounds.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

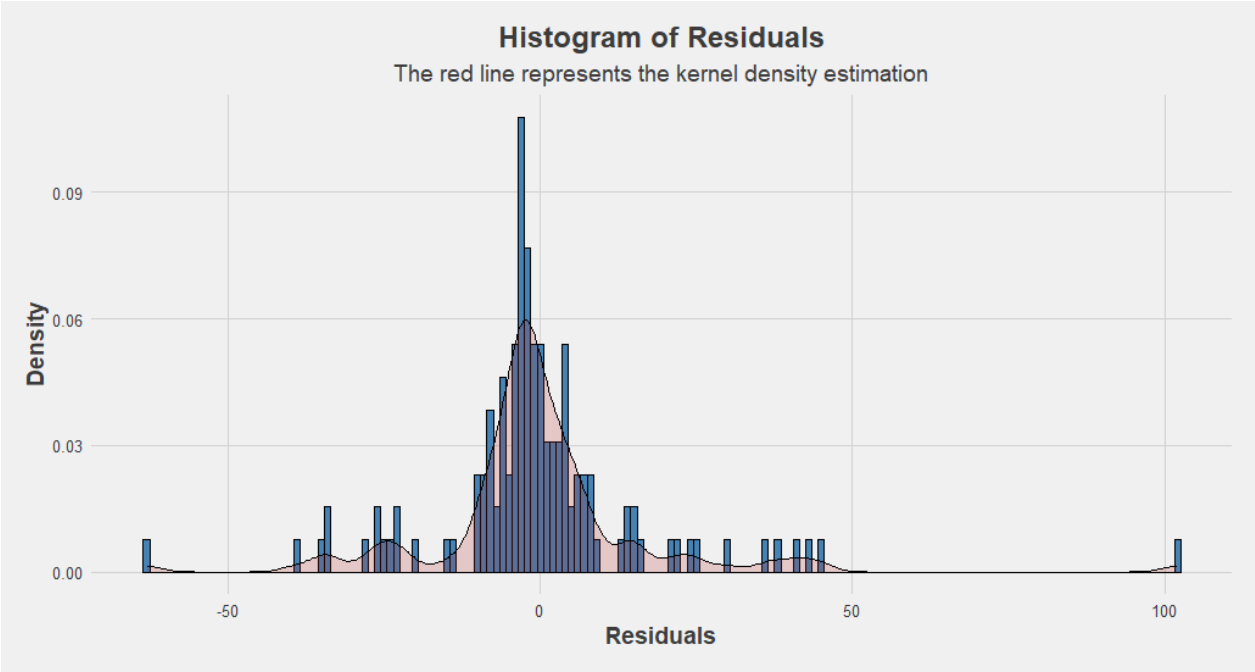

Figure 34. *Histogram of residuals for our predictive HHH4 model*

This plot in figure 34 is a histogram of residuals overlaid with a kernel density estimate. Residuals are the differences between the observed and predicted values produced by a model.

In this histogram: The bars in the histogram show the distribution of residuals. Each bar's height indicates the number of residuals (density of residuals in this case since y = ..density..) within a particular range of values. The binwidth = 1 argument defines the range This plot is a histogram of residuals overlaid with a kernel density estimate. Residuals are the differences between the observed and predicted values produced by a model.

1. **Histogram**: The bars in the histogram show the distribution of residuals. Each bar's height indicates the number of residuals (density of residuals in this case since **y = ..density..**) within a particular range of values. The **binwidth = 1** argument defines the range of these values; in this case, each bar represents a range of 1 unit of residuals. The histogram is filled in steel blue color and has black edges for better visibility.

2. **Kernel Density Estimate (KDE)**: The red area represents the kernel density estimate (KDE), which is a smoothed version of the histogram. This estimate is a way of "ironing out" the variability in the histogram to see the overall shape more clearly. It's useful for identifying the shape of the distribution, whether it is normal (bell-shaped), skewed, etc.

In terms of interpretation, here's what to look for:

- **Zero-centered residuals**: Ideally, the residuals should be centered around zero. This would mean that the model is just as likely to overestimate as underestimate the real values. If the histogram and KDE are not centered around zero, it might suggest that the model has a systematic bias (check for our model).
- **Normally distributed residuals**: Another assumption in many statistical models (including linear regression) is that the residuals are normally distributed. If the histogram and KDE look like a bell curve (i.e., the famous "normal" distribution), this is a good sign. If they are heavily skewed to one side, have multiple peaks, or have other non-normal features, it may suggest that the model's assumptions are violated, and the model might not be the best fit for the data (it resembles in our model).
- **Equal variance**: Ideally, the spread of residuals should be the same across all levels of the predicted values. This is known as homoscedasticity. If the residuals' spread increases or decreases systematically with the predicted value, this is known as heteroscedasticity and suggests that the model might not be the best fit for the data.

If the deviations are large, however, it might be worth considering a different modeling approach which is not the case with our model.

.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

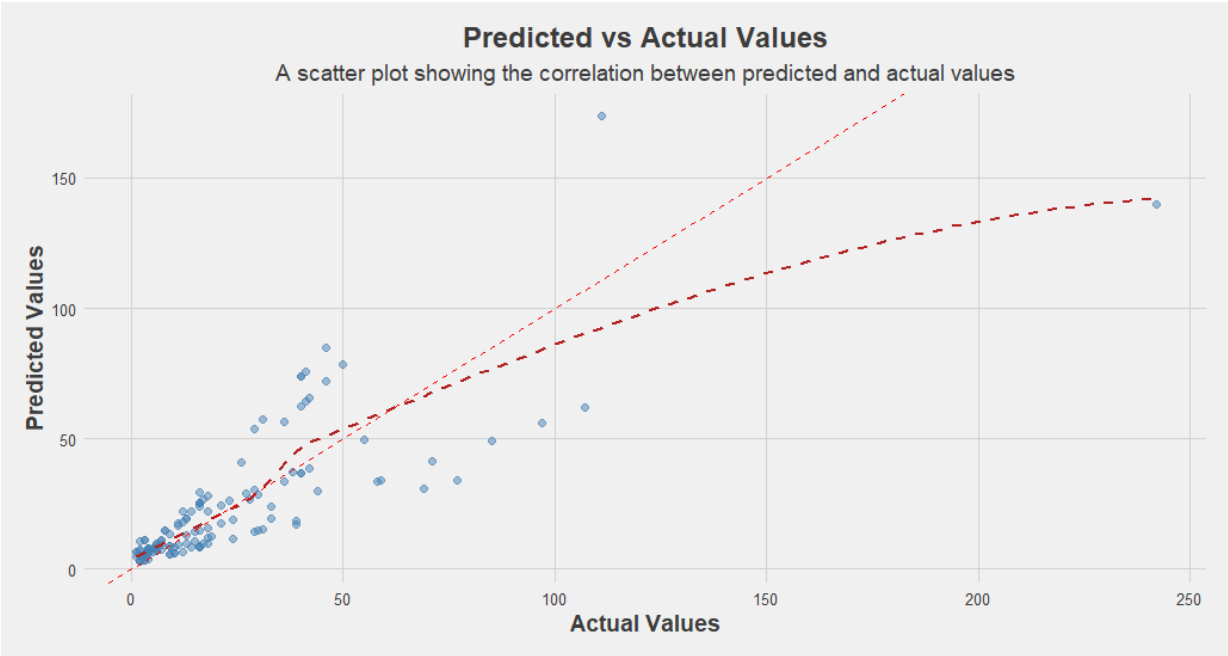

Figure 35. *Predicted vs Actual values correlaltion our predictive HHH4 model*

This scatter plot in figure 35 illustrates the relationship between the actual and predicted values generated by our model.

Points (Blue dots): Each point on the graph represents an observation. The x-coordinate of the point shows the actual value and the y-coordinate shows the predicted value for that observation.

Loess Line (Red Dashed Line): The red dashed line represents a locally estimated scatterplot smoothing (loess) line, which is a type of trend line. It shows the general pattern or relationship between the predicted and actual values. A perfect model would have all points lying exactly on this line.

Line of Equality (Red Solid Line): The solid red line (with a slope of 1 and an intercept of 0) represents the line of perfect prediction—where the actual values equal the predicted values. If a point lies on this line, it means the model's prediction for that point was exactly correct. The closer the points and the loess line are to this line, the better the model's predictions.

In terms of interpretation, we want the points and the red dashed across line to be as close as possible to the red solid line. This would mean that the model's predictions are close to the actual values. Large deviations from the line of perfect prediction indicate discrepancies between the actual and predicted values(average results).

BOXPLOT OF RESIDUALS

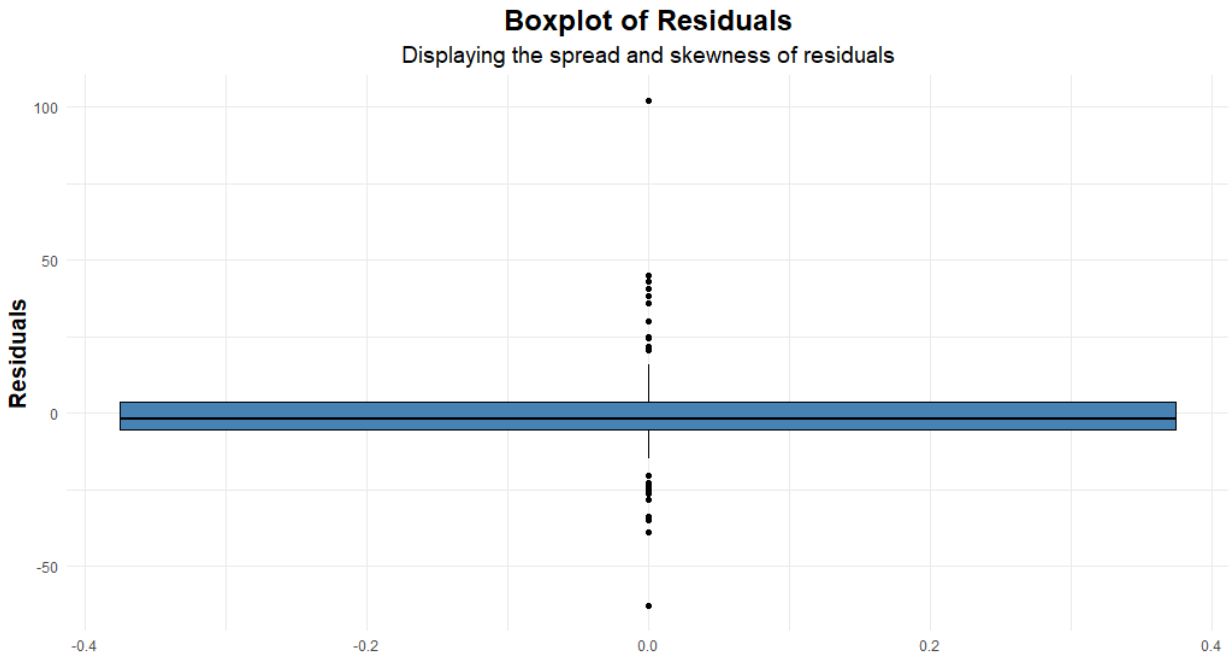

***Figure 36**. Spread and skewness of residuals for our predictive HHH4 model*

Figure 36 provides a summary of the distribution of residuals. The box represents the interquartile range (IQR), the line inside the box is the median, and the whiskers represent values within 1.5 times the IQR. Outliers, if any, are shown as individual points outside the whiskers.

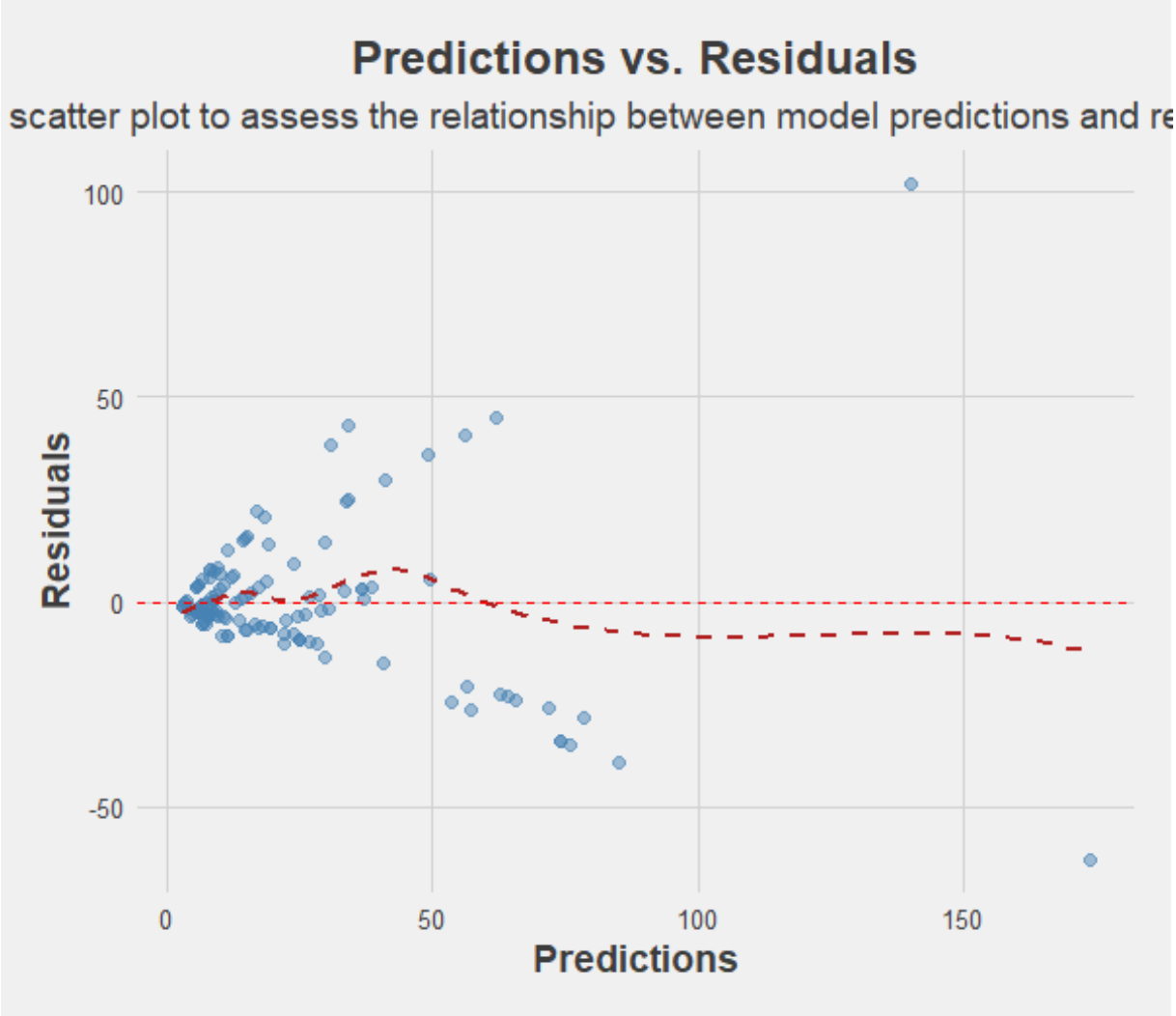

***Figure 37**. Residuals and prediction relation for our predictive HHH4 model*

As for the interpretation, figure 37 is showing the relationship between our model's predictions and the residuals. The residuals are the differences between the observed and predicted values, or in other words, they represent the error of our model's predictions.

In an ideal situation, we'd like to see no clear pattern in the residuals, and they should be scattered randomly around the zero line (the red dashed line in

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

the plot). This would mean that the model's errors are random and not systematically biased.

If there is a clear pattern in the residuals (such as a curve or a trend), it could suggest that our model is systematically over- or under-predicting at certain prediction levels, which means there might be room for improving the model. For example, if residuals tend to be positive for low predictions and negative for high predictions, it suggests that our model is underestimating in some cases and overestimating in others.

Another thing we might look for is heteroscedasticity - if the residuals get larger or smaller as the predicted value increases, it suggests the errors are not consistent across the range of predictions, which is another form of bias that could be addressed.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

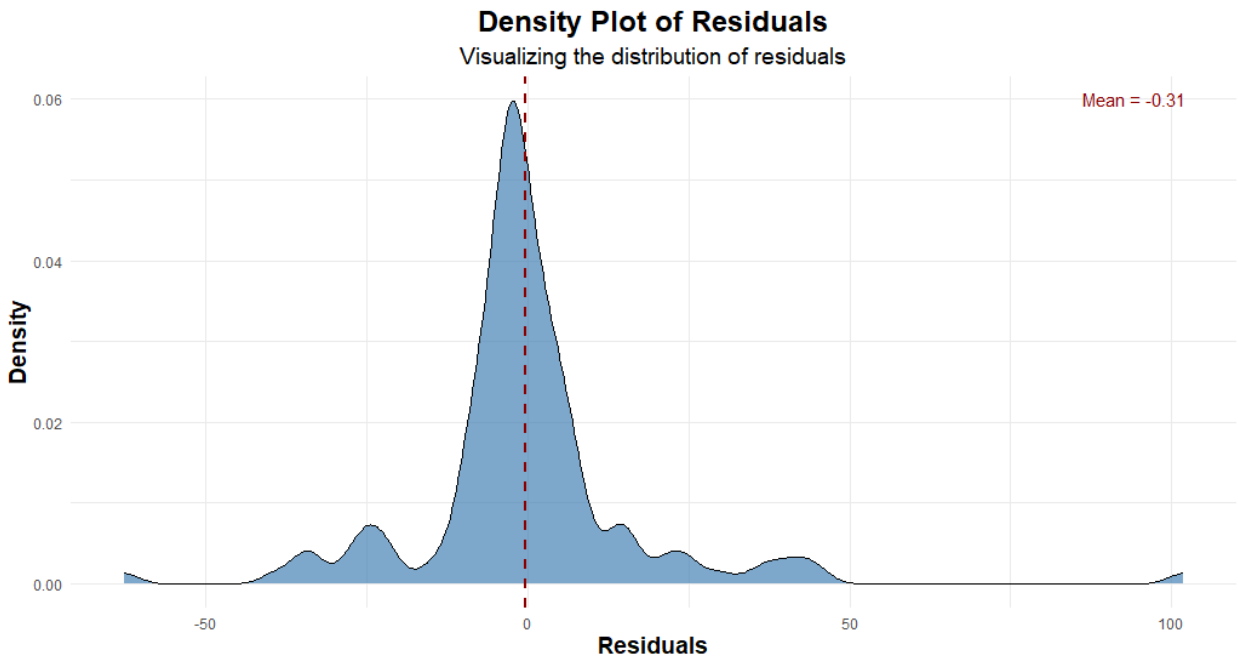

***Figure 38**. Residuals distribution for our predictive HHH4 model*

This density plot of residuals in figure 38, is used to visualize the distribution of the residuals, or the model errors.

The blue area represents the density of the residuals. This is essentially a smoothed version of the histogram, and it shows where values are concentrated. If the model is well-specified and the errors are normally distributed, we would expect this to be a symmetric bell-shaped curve (a normal distribution) centered around zero.

The red vertical dashed line represents the mean of the residuals, and its value is also displayed in the top right corner of the plot. In an ideal scenario, the mean of the residuals should be zero, indicating that the model is not systematically over- or under-predicting the outcomes. If the mean is significantly different from zero, it might indicate a bias in the predictions. In this case, the mean is very close to zero, suggesting the model is doing a good job overall, although individual predictions may still be off.

Just like before, if we see a skew in the distribution or a mean significantly different from zero, it may indicate that there is room for improving our model. For instance, the model's errors might have a consistent bias, or they might depend on the variables in a way that the model doesn't capture but our model doesn't seem to have that problem because most of the residuals gathers around the 0 value.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.3.5 Conclusion

Based on the analysis of the graphs and evaluation scores, including the mean log score, DSS, MAE, and others, it is evident that the hhh4 model, in its initial form, demonstrates strong performance in predictive modeling. The results obtained from the first model are highly encouraging.

The model exhibits a notable DSS, which implies that it effectively captures the forecasted probability distribution and closely matches the observed outcomes. This alignment showcases the model's reliability in generating accurate predictions.

When compared to other models, the hhh4 without weighted schemes model should be further evaluated and compared in Chapter 4 of my thesis.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.4 MODEL 2 (KCDE IMPLEMENTATION)

3.4.1 Method description – parametrization

1. KCDE Method Description (E.L. Ray, K.Sakrejda, S.A. Lauer, M.A. Johansson & N.G. Reich, 2016, (68))

Suppose we observe $\mathbf{z}_t = \{z_{t,1}, \dots, z_{t,D}\} \in \mathbb{R}^D$ at each point in time $t = 1, \dots, T$. Our goal is to obtain a predictive distribution for one of the observed variables, with index $d_{\text{pred}} \in \{1, \dots, D\}$, over a range of prediction horizons contained in the set $\mathcal{P}$. For example, if we have monthly data and we are interested in obtaining predictions for a range between 1 and 3 months after the most recent observation then $\mathcal{P} = \{1,2,3\}$. Let P be the largest element of the set $\mathcal{P}$ of prediction horizons.

Our prediction methodology involves the utilization of lagged observations. The maximum count of lags that can be employed for each observed variable for prediction purposes is given by $\mathrm{l}^{\max} = (l_1^{\max}, \dots, l_D^{\max})$. The highest lag that could be deployed across all variables is represented by$L = \max d l_d^{\max}$. During the process of parameter estimation, we will choose a subset of these lags to apply in the predictions. The lags we decide to use are indicated in the vector:

$$\mathrm{u} = \left(u_{1,0}, \dots, u_{1,l_1^{\max}}, \dots, u_{D,0}, \dots, u_{D,l_D^{\max}}\right), \text{ where}$$
$$u_{d,l} = \begin{cases} 0 \text{ if lag } l \text{ of variable } d \text{ is not used in forming predictions} \\ 1 \text{ if lag } l \text{ of variable } d \text{ is used in forming predictions.} \end{cases}$$

By analogy with the standard notation in autoregressive models, we define.

$$\mathrm{y}_t = \left(z_{t,d_{\text{pred}}}, \dots, B^{(P-1)} z_{t,d_{\text{pred}}}\right) \text{ and}$$
$$\mathrm{x}_t = \left(B^{(P)} z_{t,1}, \dots, B^{(P+l_1^{\max}-1)} z_{t,1}, \dots, B^{(P)} z_{t,D}, \dots, B^{(P+l_D^{\max}-1)} z_{t,D}\right)$$

Here, $B^{(a)}$ is the backshift operator defined by $B^{(a)} z_{t,d} = z_{t-a,d}$. The lengths of the vectors of y_t and x_t, and the exact lags utilized to form them, depend on $\mathcal{P}$ and $\mathrm{l}^{\max}$. However, to maintain notational clarity, we have chosen to suppress this dependence. The vector $\mathbf{y}_t$ acts as the prediction target when the most recent observation was made at time $t - P$, representing the

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

vector of observed values at each prediction horizon $p \in \mathcal{P}$. The variable $\mathbf{x}_t$ serves as the vector of all lagged covariates available for use in prediction.

To provide a tangible example, let's imagine that $\mathbf{z}_t$ contains the observed case count for month t in a dataset, the observed case count for month t, and the date on week 1 of month t, our goal is to predict the monthly case count the dataset. Then $D = 3$ and $d_{\text{pred}} = 1$. If we want to predict the monthly case counts for the two months after the most recently observation, then $p = 2$. If we specify that the model may include the two most recent observations for the case counts in our dataset, but only the time index at the most recent observation then $\mathbf{l}^{\max} = (1,1,0)$. If our current model uses only the most recently observed case counts for our dataset then $\mathbf{u} = (1,0,1,0,0)$, where the 1's are in the positions of the $\mathbf{u}$ vector representing lag 0 of the counts for our dataset and lag 0 of the counts for Iquitos. The variable $y_t^{(P)}$ is a vector containing the observed case counts for our dataset in months $t+1$ and $t+2$; $\mathbf{x}_t^{(\mathbf{l}^{\max})}$ contains the observed case counts for our dataset in months t and $t-1$ as well as the time index variable in month t.

In order to perform prediction, we regard $\{(\mathbf{y}_t, \mathbf{x}_t), t = 1 + P + L, \dots, T\}$ as a sample from the joint distribution of $(\mathbf{Y}, \mathbf{X})$. We wish to estimate the conditional distribution of $\mathbf{Y} \mid \mathbf{X}$. In order to do this, we employ kernel density estimation. Let $K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}^*, H^{\mathbf{Y}})$ and $K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}^*, H^{\mathbf{X}})$ be kernel functions centered at $\mathbf{y}^*$ and $\mathbf{x}^*$ respectively and with bandwidth matrices $H^{\mathbf{Y}}$ and $H^{\mathbf{X}}$. We estimate the conditional distribution of $\mathbf{Y} \mid \mathbf{X}$ as follows:

$$\hat{f}_{\mathrm{Y|X}}(\mathrm{y} \mid \mathrm{X} = \mathrm{x}) = \frac{\hat{f}_{\mathrm{Y,X}}(\mathrm{y}, \mathrm{x})}{\hat{f}_{\mathrm{X}}(\mathrm{x})} \quad (\mathbf{1})$$

$$= \frac{\sum_{t \in \tau} K^{\mathrm{Y,X}}\{(\mathrm{y}, \mathrm{x}), (\mathrm{y}_t, \mathrm{x}_t), H^{\mathrm{Y,X}}\}}{\sum_{t \in \tau} K^{\mathrm{x}}(\mathrm{x}, \mathrm{x}_t, H^{\mathrm{x}})} \quad (\mathbf{2})$$

$$= \frac{\sum_{t \in \tau} K^{\mathrm{Y|X}}(\mathrm{y}, \mathrm{y}_t \mid \mathrm{x}, \mathrm{x}_t, H^{\mathrm{Y,X}}) K^{\mathrm{X}}(\mathrm{x}, \mathrm{x}_t, H^{\mathrm{x}})}{\sum_{t \in \tau} K^{\mathrm{X}}(\mathrm{x}, \mathrm{x}_t, H^{\mathrm{X}})} \quad (\mathbf{3})$$

$$= \sum_{t \in \tau} w_t K^{\mathrm{Y|X}}(\mathrm{y}, \mathrm{y}_t \mid \mathrm{x}, \mathrm{x}_t, H^{\mathrm{Y,X}}) \quad (\mathbf{4}), \text{ where}$$

$$w_t = \frac{K^{\mathrm{X}}(\mathrm{x}, \mathrm{x}_t, H^{\mathrm{x}})}{\sum_{t^* \in \tau} K^{\mathrm{X}}(\mathrm{x}, \mathrm{x}_{t^*}, H^{\mathrm{X}})} \quad (\mathbf{5})$$

UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ATHENS

In Equation (1), we are simply making use of the fact that the conditional density for $\mathbf{Y} \mid \mathbf{X}$ can be written as the quotient of the joint density for $(\mathbf{Y}, \mathbf{X})$ and the marginal density for $\mathbf{X}$. In Equation (3), we obtain separate kernel density estimates for the joint and marginal densities in this quotient. In Equation (4), we rewrite this quotient by passing the denominator of Equation (3) into the summation in the numerator. We can interpret the result as a weighted kernel density estimate, where each observation $t \in \tau$ contributes a different amount to the final conditional density estimate. The amount of the contribution from observation t is given by the weight w_t, which effectively measures how similar $\mathbf{x}_t$ is to the point $\mathbf{x}$ at which we are estimating the conditional density. If $\mathbf{x}_t^{(1^{\max})}$ is similar to $\mathbf{x}_{t^*}^{(1^{\max})}$, a large weight is assigned to t; if $\mathbf{x}_t^{(1^{\max})}$ is different from $\mathbf{x}_{t^*}^{(1^{\max})}$, a small weight is assigned to t.

In kernel density estimation, it is generally required that the kernel functions integrate to 1 in order to obtain valid density estimates. However, after conditioning on $\mathbf{X}$, it is no longer necessary that $K^{\mathbf{X}}(\mathbf{x}, \mathbf{x}_t, H^{\mathbf{X}})$ integrate to 1. In fact, as can be seen from Equation (5), any multiplicative constants of proportionality will cancel out when we form the observation weights. We can therefore regard $K^{\mathbf{x}}(\mathbf{x}, \mathbf{x}_t, H^{\mathbf{x}})$ as a more general weighting function that measures the similarity between $\mathbf{x}$ and $\mathbf{x}_t$. As we will see, eliminating the constraint that $K^{\mathbf{x}}$ integrates to 1 is a useful expansion the space of functions that can be used in calculating the observation weights. However, we still require that $K^{\mathbf{Y}}$ integrates to 1 .

In Equations (1) through (5), τ is an index set of time points used in obtaining the density estimate. In most settings, we can take $\tau = \{1 + P + L, \dots, T\}$. These are the time points for which we can form the lagged observation vector $\mathbf{x}_t$ and the prediction target vector $\mathbf{y}_t$. However, the model place additional restrictions on the time points included in τ in the cross-validation procedure discussed later in parameter estimation section.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

If we wish to obtain point predictions, we can use a summary of the predictive density. For example, if we take the expected value, we obtain kernel regression:

$$
\begin{aligned}
(\widehat{\mathbf{Y}} \mid \mathbf{X} \quad &= \mathbf{x}) = \mathbb{E}_{\hat{f}_{\mathbf{Y}|\mathbf{X}}}\{\mathbf{Y} \mid \mathbf{X} = \mathbf{x}\}(\mathbf{6}) \\
&= \int \sum_{t \in \tau} w_t K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}_t, H^{\mathbf{Y}}) \mathbf{y} d\mathbf{y} \ (\mathbf{7}) \\
&= \sum_{t \in \tau} w_t \mathbf{y}_t \ (\mathbf{8})
\end{aligned}
$$

The equality in Equation (8) holds if the kernel function $K^{\mathbf{Y}}(\mathbf{y}, \mathbf{y}_t, H^{\mathbf{Y}})$ is symmetric about $\mathbf{y}_t$, or more generally if it is the pdf of a random variable with expected value $\mathbf{y}_t$.

Another alternative that we pursue is the use of smoothed observations in forming the lagged observation vectors. We use smoothed case counts on a log scale for the weighting kernels, and the unsmoothed case counts on the original scale for the prediction kernels.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

2. Parameter Estimation

Next we use cross-validation to select the variables that are used in the model and estimate the corresponding bandwidth parameters by (approximately) minimizing a cross-validation measure of the quality of the predictions obtained from the model. Formally,

$$\left(\hat{\mathbf{u}}, \hat{H}^{\mathbf{X}}, \hat{H}^{\mathbf{Y}}\right) \approx \underset{(\mathbf{u}, H^{\mathbf{X}}, H^{\mathbf{Y}})}{\operatorname{argmin}} \sum_{t^*=1+P+L}^{T} Q\left[\mathbf{y}_{t^*}, \hat{f}(\mathbf{y} \mid \mathbf{X} = \mathbf{x}_{t^*}; \mathbf{u}, H^{\mathbf{X}}, H^{\mathbf{Y}}, \{(\mathbf{y}_t, \mathbf{x}_t) : t \in \tau_{t^*}\})\right]$$

Here, Q is a loss function that measures the quality of the estimated density $\hat{f}$ given an observation $\mathbf{y}_{t^*}$. We have made the dependence of this estimated density on the the parameters $\mathbf{u}, H^{\mathbf{X}}$, and $H^{\mathbf{Y}}$, as well as on the data $\{(\mathbf{y}_t, \mathbf{x}_t) : t \in \tau_{t^*}\}$, explicit in the notation. In order to reduce the potential for our parameter estimates to be affected by local correlation in the time series, we eliminate all time points that fall within one year of t^* from the index set τ_{t^*} used to form the conditional density estimate $\hat{f}(\mathbf{y} \mid \mathbf{X} = \mathbf{x}_{t^*}; \mathbf{u}, H^{\mathbf{X}}, H^{\mathbf{Y}}, \{(\mathbf{y}_t, \mathbf{x}_t) : t \in \tau_{t^*}\})$.

We use a forward/backward stagewise procedure to obtain the set of combinations of variables and lags that are included in the final model. For each candidate model, i use the limited memory box constrained optimization procedure of [Byrd, Lu, Nocedal, and Zhu, 1995] to estimate the bandwidth parameters. The approximation in the last equation is due to the fact that this optimization procedure may not find a global minimum.

3.Copulas

Utilizing copulas to connect the marginal predictive distributions for individual prediction horizons derived from KCDE into a joint predictive distribution for the incidence trajectory across multiple time points is a valuable approach. A copula, a parametric function capturing dependence relations among random variables, facilitates the computation of joint distribution from marginal distributions.

To elaborate, the predictive densities $f^h\left(y_{t^*}^h \mid \mathbf{x}_{t^*}; \boldsymbol{\theta}^h\right)$ we can naturally yield from a model, corresponds to a specific predictive CDFs $F^h\left(y_{i^*}^h \mid\right.$

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

$\mathbf{x}_{t^*}; \theta^h)$.

Our model defines the joint cumulative distribution function CDF for $\left(Y_{r^*}^1, \dots, Y_{r^*}^{H_{r^*}}\right)$ as follows:

$$F^{H_{r^*}}\left(y_{l^*}^1, \dots, y_{l^*}^{H_{r^*}} \mid \mathbf{x}_{l^*}; \theta^1, \dots, \theta^{H_{r^*}}, \xi^{H_{l^*}}\right) =$$
$$C^{H_{l^*}}\left\{F^1\left(y_{l^*}^1 \mid \mathbf{x}_{l^*}; \theta^1\right), \dots, F^{H_{l^*}}\left(y_{l^*}^{H_{c^*}} \mid \mathbf{x}_{l^*}; \theta^{H_{l^*}}\right); \xi^{H_{l^*}}\right\}.$$

The copula function $C^{H_{i^*}}$ maps the marginal CDF values to the joint CDF value. We employ the isotropic normal copula from the R package <<copula>>. The copula function is given by

$$C^H(u_1, \dots, u_H; \xi^H) = \Phi_\Sigma\left(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_H)\right),$$

where Φ^{-1} is the inverse CDF of a univariate normal distribution with mean 0 and variance 1 , and Φ_{Σ^μ} is the CDF of a multivariate normal distribution with mean $\underline{0}$ and covariance matrix Σ^H . The isotropic specification sets $\Sigma^H = \left[\sigma_{i,j}^H\right]$, where

$$\sigma_{i,j}^H = \begin{cases} 1 \text{ if } i = j, \\ \xi_d^H \text{ if } |i - j| = d. \end{cases}$$

Intuitively, ξ_d^H captures the amount of dependence between incidence levels at future times that are d months apart. We obtain a separate copula fit for each value of H from 2 to W (note that a copula is not required for "trajectories" of length $H = 1$).

In summary, parameter estimation for the model involves two stages with copulas: firstly, estimating KCDE parameters for each prediction horizon $h = 1, \dots, H$ as described previously, and secondly, estimating copula parameters while holding KCDE parameters fixed. Although the two-stage approach may lead to some loss of efficiency compared to one-stage methods, it does reduce the computational cost of parameter estimation..

Interestingly, in our dataset, implementing and estimating copula parameters while holding KCDE parameters fixed results in a significant loss of efficiency compared to KCDE alone, with worse predictive evaluation scores. The only positive aspect is a marginal reduction in computation time

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

based on these specific data. Therefore, the copula implementation appears suboptimal for this particular dataset.

Below is a recap to the implementation of the KCDE method in R language.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.4.2 R code implementation of KCDE model

This complex and long code implementation focuses on predictive modeling using Kernel Conditional Density Estimation (KCDE) with relevant libraries, including "zoo," "kcde," and "HIDDA.forecasting." The code initiates by preparing time-indexed datasets and selecting variables for prediction. It offers bandwidth parameterization choices: "diagonal" or "full" modes for model complexity control. **Note that the diagonal approach is a more practical in the cost of a very small accuracy loss**

The code assembles kernel components, such as periodic and log-pdtmvn kernels, to capture seasonality and predictive variable relationships. This forms the foundation of the comprehensive KCDE model. Functions for discretization are introduced to manage continuous-to-discrete conversion.

The model fitting process is executed, enabling the model to learn from data over time. Results encompass predictions and log scores as indicators of model performance. Distributions, samples, point predictions, and Absolute Errors contribute to performance evaluation. The Log Score provides insight into model accuracy and proficiency.

In summary, this code explores predictive modeling using KCDE, unraveling intricate relationships in time-series data. It showcases model configuration, fitting, and evaluation through various metrics, highlighting the potency of advanced techniques in deciphering patterns within temporal data.

Notable Findings and Considerations

- Computational Time: 120 hours (full parametrization) vs. 10 minutes (diagonal version).
- Package Limitations: Some metrics weren't extractable for all models; direct comparisons with KCDE not always possible.
- Coding: KCDE required >1000 lines in R; other models used fewer (around 600).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

- Long-Term Forecasts: Computing all long-term KCDE forecasts estimated to be very time-consuming, so they were skipped.
- Model Output: KCDE's kcde() doesn’t return a dedicated class, complicating summary or fitted values extraction.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.4.3 Results

In this section, I present the outcomes of our KCDE model's performance evaluation.

DSS, LOG SCORE & AE

```
      DSS            log_score            AE
 Min.   : 5.340   Min.   : 3.309   Min.   :  2.00
 1st Qu.: 7.650   1st Qu.: 4.382   1st Qu.: 12.00
 Median : 8.292   Median : 5.383   Median : 35.00
 Mean   :10.422   Mean   : 5.448   Mean   : 52.44
 3rd Qu.: 8.681   3rd Qu.: 5.968   3rd Qu.: 54.00
 Max.   :55.084   Max.   :10.137   Max.   :298.00
```

Table 1: KCDE predictive model result values

1. **DSS (Density Sensitive Score)**: This is a loss function which measures the distance between the estimated and the true density. Lower values of DSS mean a better model, as they indicate a smaller distance. Note that it's the same metric as the Dawid-Sebastiani Score (the second one is the more formal name of the evaluation metric)

2. **Log Score**: This score measures the quality of a probabilistic prediction. It can be seen as the logarithm of the likelihood of the prediction under the true distribution. Again, a higher log score implies a better fit. However, comparing the mean log score of two different models directly may not be entirely accurate due to potential differences in their complexity and the amount of overfitting. We should also consider other metrics and visual inspections.

3. **AE (Absolute Error)**: This is the absolute difference between the predicted and the actual value. Lower values of AE imply a better model as the difference between predicted and actual values is smaller.

In summary, to judge the performance of a model, we generally want DSS, log score and AE to be as low as possible. However, these scores should be used in conjunction with other model evaluation techniques such as visual inspection, cross-validation, and other relevant domain-specific measures to fully understand the model's performance.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

VISUALIZE DSS & LOG SCORE

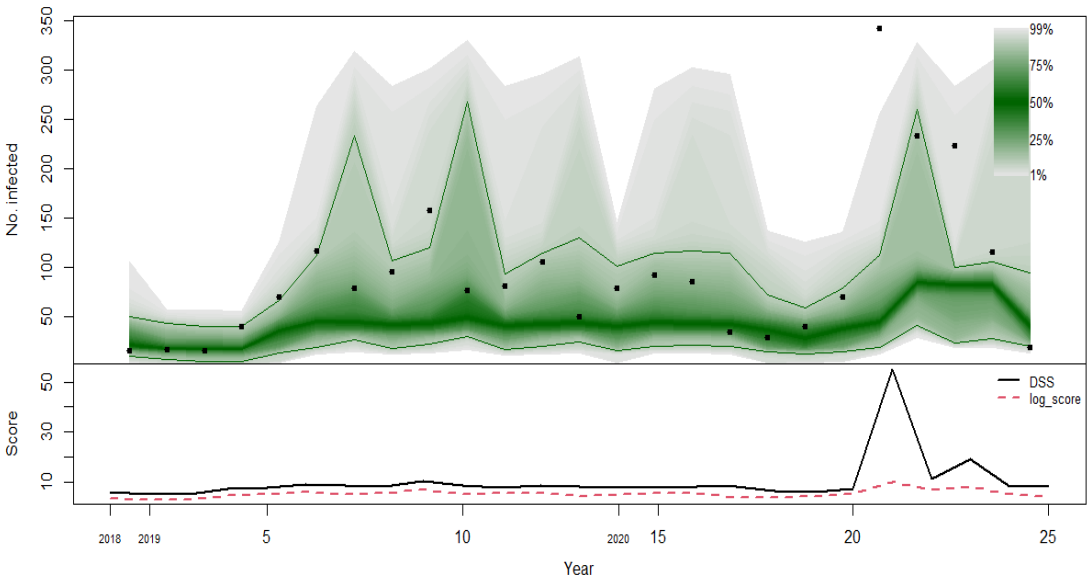

***Figure 39**. Visualize of results for our predictive KCDE model*

A version of an out-of-sample assessment plot, or "OSA" plot

1. **Quantile Fan Chart**: This is the shaded area that fans out over time. Each line in this fan represents a quantile of the forecast distribution. In our case, we have calculated the 1% to 99% quantiles, thus we are visualizing the range of these predicted values over time. As time progresses, we'd expect the fan to widen as the model's certainty about future predictions decreases.

2. **Observed Values**: These are the black dots on the plot, which represent the observed values. Ideally, these dots should fall within the fan. If many of the observed values fall outside the fan, particularly on one side, it might suggest the model has systematic bias.

3. **Scores Line Chart**: The lines overlaid on the chart represent our scores: DSS and log_score. These are evaluation metrics calculated for our model. Depending on the model, we would ideally want the DSS to

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

be as low as possible (indicating a better fit to the data), and the log score to be as high as possible (indicating better prediction accuracy).

As for what would constitute good values or problem indicators:

- For the fan chart, we'd want to see the observed values scattered evenly and falling within the fan. If they consistently fall outside or on one side, it could suggest a biased model. Also, if the fan widens too quickly, it could suggest that our model isn't confident about its predictions (which is somewhat true in our model).
- For the DSS and log score lines, lower values of DSS and higher values of log score are generally better. Sudden spikes in DSS (we can observe this at month 20) or drops in log score could indicate problems.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

This code snippet generates a visualization using the **ggplot2** package to compare the observed and predicted values of a specific variable over the time period from 2019 to 2020.

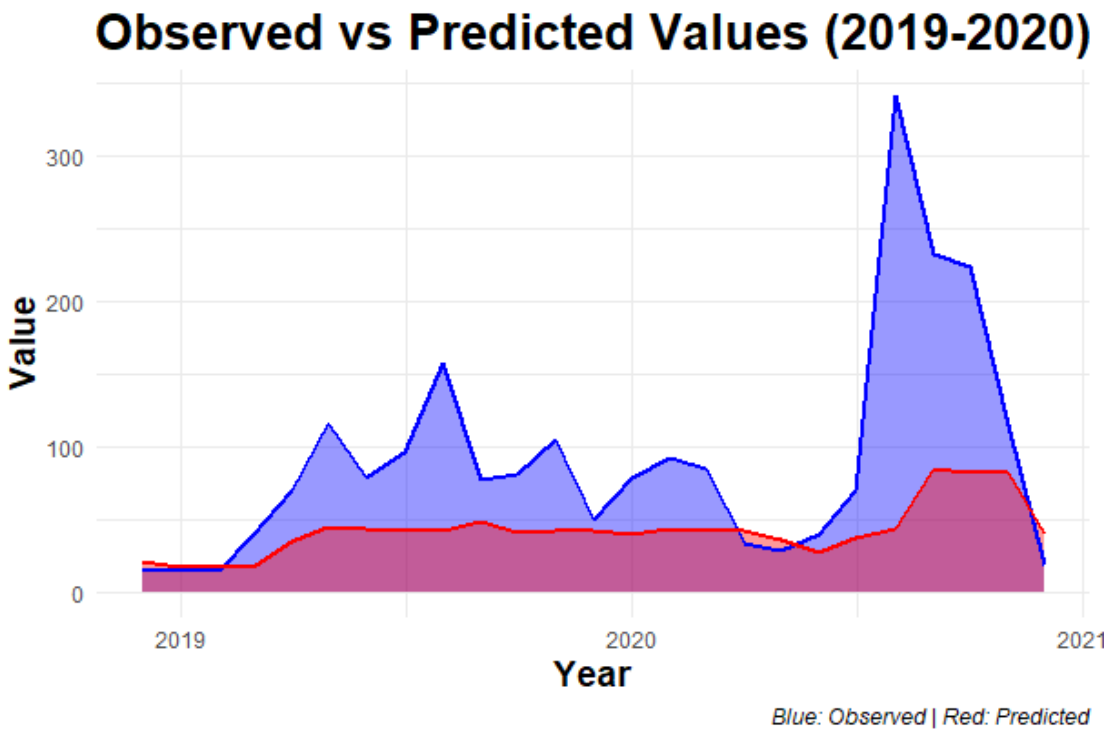

***Figure 40**. 2019-2020, Obs vs pred values for our predictive KCDE model*

Figure 40 is a comparison of observed vs. predicted values over time, with separate lines and areas for the observed (in blue) and predicted (in red) values.

The plot provides a visual comparison of the actual observed values and the values that our model predicted. This is a way to see how well our model is performing.

The blue area represents the observed values. The darker line in the middle of the blue area is the actual observed values at each point in time, and the blue area itself indicates the range of these values.

Similarly, the red area represents the predicted values. The darker line in the middle of the red area is the predicted values at each point in time, and the red area itself indicates the range of these values.

Interpreting the plot is fairly straightforward:

- If our model is perfect, the red and blue lines will overlap completely, indicating that our model's predictions are always equal to the actual observed values.
- If the red line is consistently above the blue line, it indicates that our model is consistently overestimating the values.
- If the red line is consistently below the blue line, it indicates that our model is consistently underestimating the values.
- If the red and blue lines overlap for some periods but not for others, it indicates that ourr model is sometimes accurate and sometimes not.

In addition, the divergence between the two areas (red and blue) can provide a sense of the degree of uncertainty or variance in our model's predictions. A wider red area would indicate a higher degree of uncertainty in the predictions.

When reviewing this plot, ideally, we want the red (predicted) line to align as closely as possible with the blue (observed) line, and the red area to be as narrow as possible, indicating a high degree of certainty in our model's predictions (we can see a small peak capture in 2020-2021 time period only but in general it doesn’t capture the pattern with high accuracy).

3.4.4 Evaluation metrics

The following are the evaluation metrics of our KCDE implementation, primarily employed for comparing its performance to that of other models.

MAE: 52.44

This MAE value of 52.44 means that, on average, the model's predictions are about 52.44 units away from the actual values.

RMSE: 82.43

The RMSE value of 82.43 is larger than the MAE, which is always the case, or they could be equal but never smaller. The RMSE gives a relatively high weight to large errors because the squared term grows exponentially. This means the RMSE is most useful when large errors are particularly undesirable.

MAPE: 47.95

The MAPE value of 47.95 suggests that, on average, the model's predictions are about 48% off from the actual values. This is a high error rate and indicates that the model may not be very accurate.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

RESIDUALS VS PREDICTED VALUES

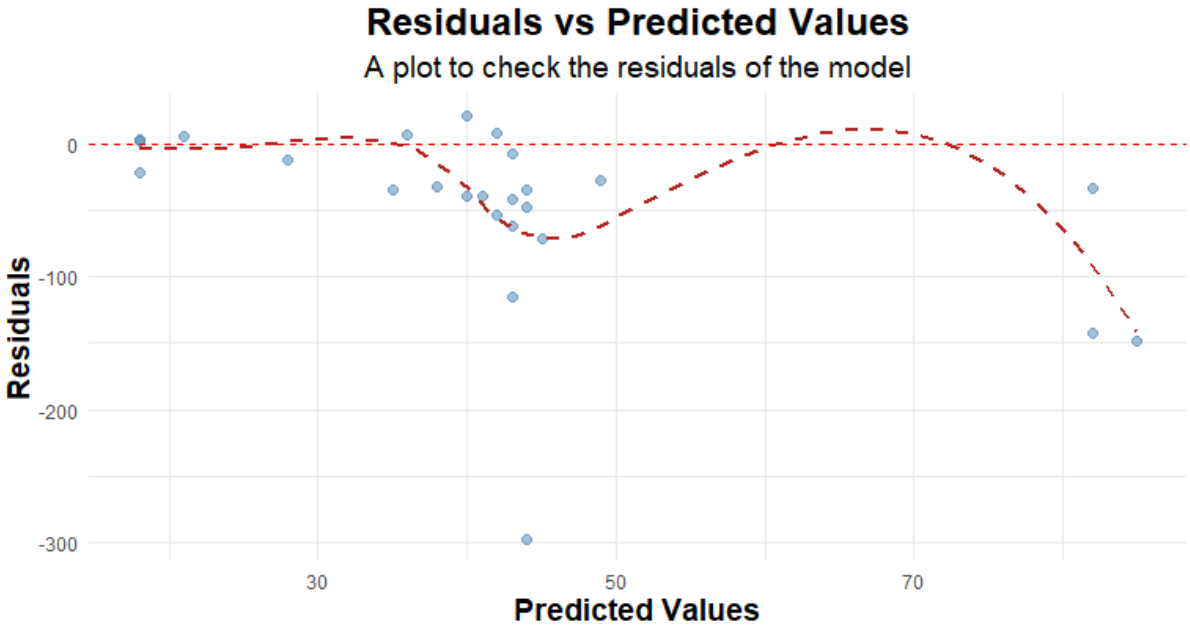

***Figure 41**. Residuals for our predictive KCDE model*

Scatter plot of the residuals in figure 41 is the difference between the observed and predicted values against the predicted values. In a well-performing model, we expect the residuals to be randomly scattered around zero, showing no discernible pattern.

Here's what we should consider when interpreting the plot:

- If the model is working well, the points should be randomly scattered around the horizontal line at 0, with no clear pattern.
- If there's a systematic pattern in the plot, it suggests that this model isn't capturing some aspect of the data.
- If the points seem to form a funnel shape (i.e., they scatter more widely either at the lower or higher end of the predicted values), it suggests heteroscedasticity. This means that the prediction errors (residuals) are not constant across different levels of the dependent variable, violating one of the key assumptions of linear regression.

- If we see a curve in the plot, it means that the model is not capturing the non-linear nature of the data.

The red horizontal line at 0 is the reference line where ideally our residuals should be randomly scattered around.

In a perfect model, residuals would be randomly scattered around 0, with no discernible pattern. They would be as close to 0 as possible and wouldn't form any discernible pattern or shape (here we can see mild pattern).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

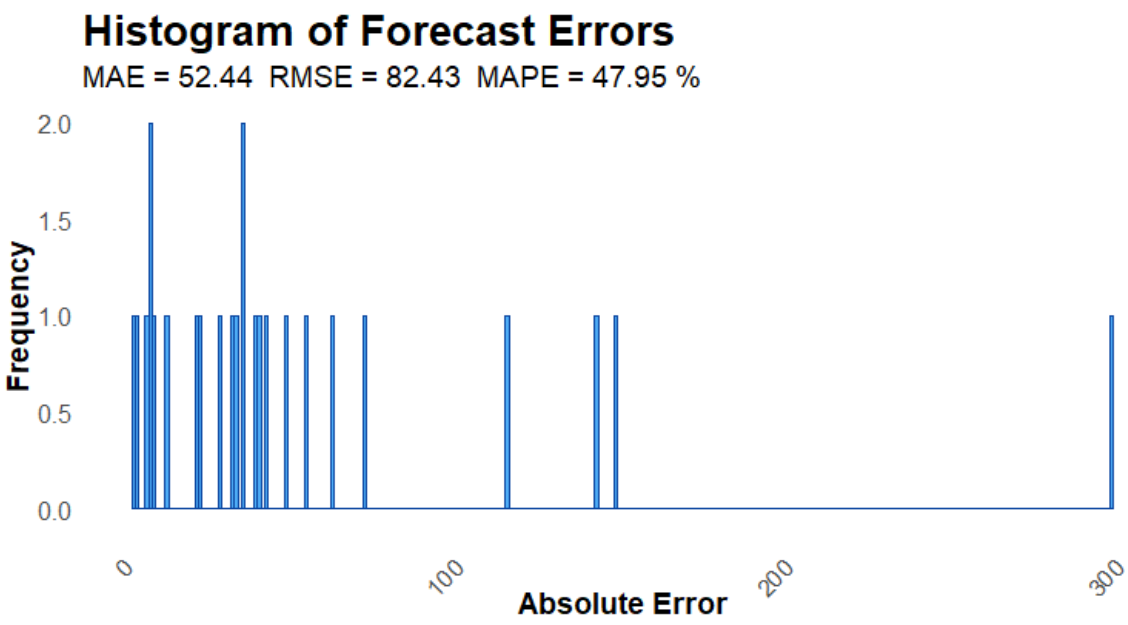

***Figure 42**. Forecast errors for our predictive KCDE model*

The histogram of forecast errors in figure 42 represents the distribution of the differences between the predicted values and the actual values. It shows the frequency or count of forecast errors falling into different bins or intervals.

In the context of forecasting, the histogram of forecast errors provides insights into the accuracy and precision of the forecasts. Here are some key points about the histogram of forecast errors:

1. Histogram Shape: The shape of the histogram indicates the distribution of forecast errors. A symmetrical distribution centered around zero is generally desirable as it suggests that the forecast errors are evenly distributed on both sides of the actual values. This indicates a balanced forecast with no significant bias.

2. Skewness: Skewness in the histogram indicates a bias in the forecasts. If the histogram is skewed to the right (positive skew), it suggests that the forecasts tend to underestimate the actual values. Conversely, if it is skewed to the left (negative skew), it indicates an overestimation of

the actual values. Ideally, we would want the histogram to be centered around zero without significant skewness.

3. Peak and Width: The peak of the histogram represents the most common range of forecast errors. A higher peak indicates a higher frequency of errors falling within that range. The width of the histogram bars represents the variability or dispersion of the forecast errors. A narrower histogram suggests lower variability and more accurate forecasts, while a wider histogram indicates higher variability and less accuracy.

4. Error Metrics: The calculated error metrics, such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE), provide quantitative measures of forecast accuracy. Lower values of MAE and RMSE indicate better accuracy, as they represent smaller average forecast errors. A lower MAPE indicates a smaller percentage difference between the forecasted values and the actual values.

In summary, when interpreting the graph, we would ideally want to see a symmetrical histogram centered around zero, indicating a balanced forecast with no significant bias. The peak of the histogram should represent a common range of forecast errors, and a narrower histogram width suggests lower variability and more accurate forecasts. Additionally, lower values of MAE, RMSE, and MAPE indicate better forecast accuracy.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

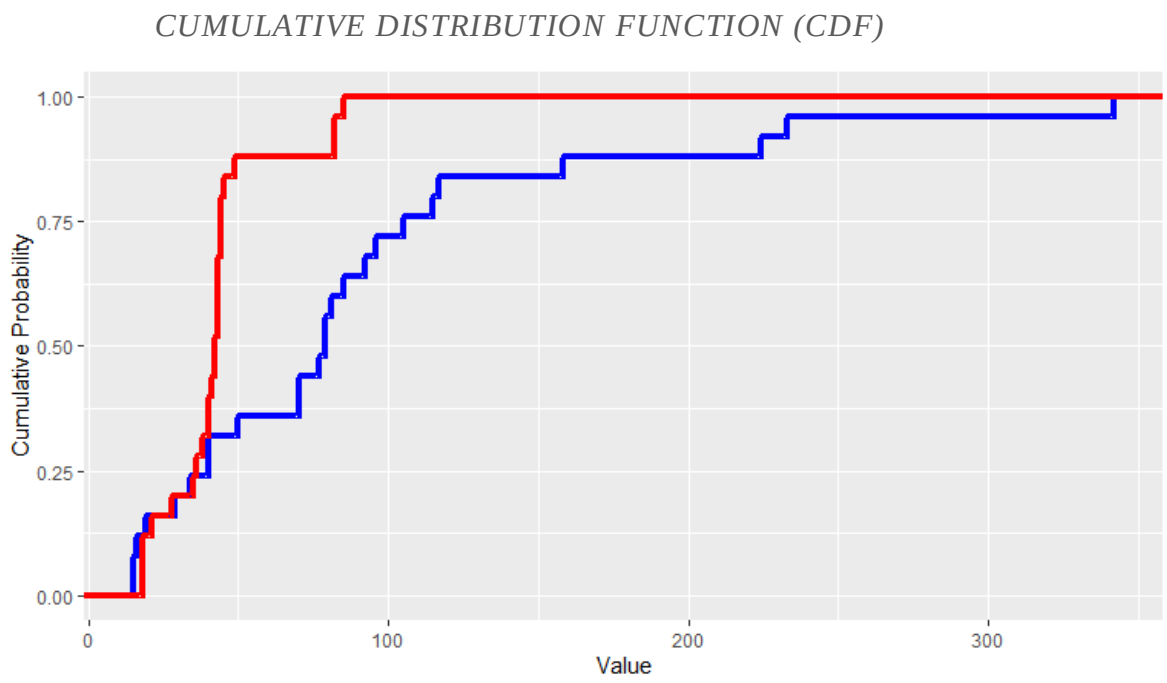

***Figure 43**. CDF plot for our predictive KCDE model*

The Cumulative Distribution Function (CDF) plot is used to visualize the cumulative probability distribution of the observed values and predicted values. It provides insights into the overall fit and alignment between the observed and predicted values.

In the CDF plot(figure43):

- X-axis: The x-axis represents the values of the variable being analyzed (e.g., the number of cases). It shows the range of values from the lowest to the highest.
- Y-axis: The y-axis represents the cumulative probability, ranging from 0 to 1. It shows the proportion or percentage of values that are less than or equal to a specific value on the x-axis.
- Blue Line: The blue line in the CDF plot represents the cumulative distribution of the observed values. It shows how the observed values accumulate as the values on the x-axis increase. The steeper the slope of the blue line, the higher the concentration of observed values at lower values.

- Red Line: The red line in the CDF plot represents the cumulative distribution of the predicted values. It shows how the predicted values accumulate as the values on the x-axis increase. The red line should ideally align closely with the blue line, indicating a good fit between the predicted and observed values.
- Comparison: Comparing the blue and red lines can provide insights into the performance of the predictive model. If the red line closely follows the blue line, it indicates that the model accurately captures the distribution of the observed values. On the other hand, deviations or gaps between the red and blue lines suggest discrepancies between the predicted and observed distributions.
- Goodness of Fit: In an ideal scenario, the red line would overlap perfectly with the blue line, indicating a perfect fit between the predicted and observed values. However, some deviation is expected due to the uncertainty inherent in predictions. A good fit is indicated by the red line closely following the blue line, suggesting accurate predictions.

By analyzing the CDF plot, we can assess the accuracy and reliability of the predictive model. A close alignment between the red and blue lines indicates a well-performing model, while significant deviations may indicate areas for improvement in the model's predictive capabilities which is our case here.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

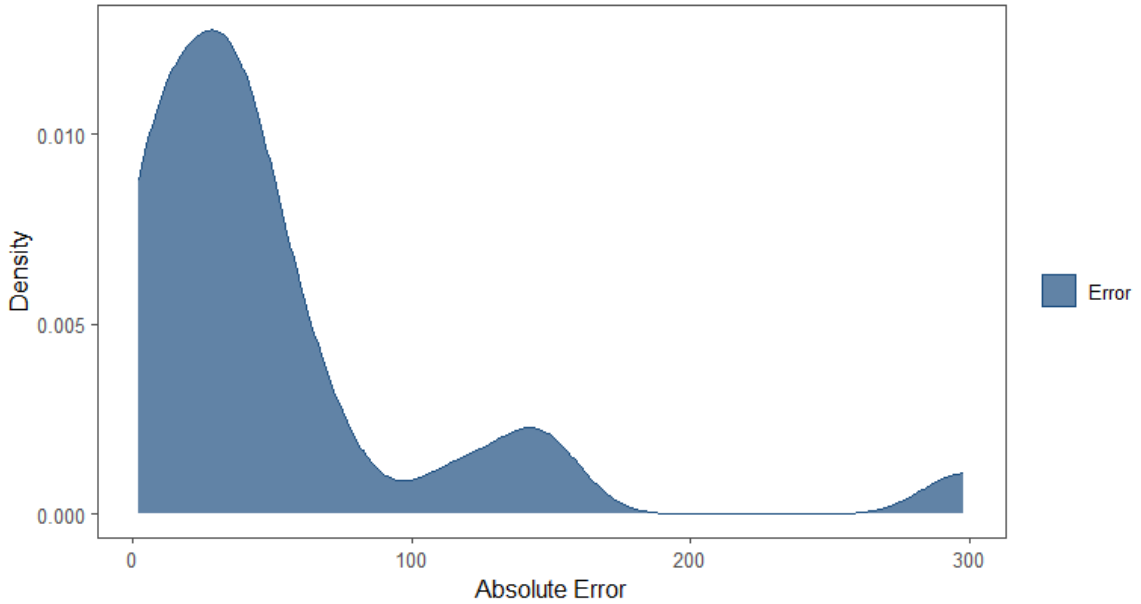

***Figure 44**. Distribution of A.Errors for our predictive KCDE model*

In this density plot(figure 44):

- X-axis: The x-axis represents the absolute error, which is the absolute difference between the observed and predicted values. It shows the range of absolute error values.
- Y-axis: The y-axis represents the density, which represents the probability distribution of the absolute error values. It shows how the absolute error values are distributed across different ranges.
- Density Plot: The density plot visualizes the distribution of the absolute error values. The shaded area represents the density of the absolute error values. The higher the density, the more concentrated the absolute error values are in that range.
- Color and Fill: The density plot is filled with a solid color (blue in this case) to highlight the distribution. The alpha parameter controls the transparency of the fill.

Interpretation: The density plot helps us understand the distribution of absolute error values. A higher density indicates that a particular range of absolute error values is more common, while a lower density suggests that values in that range are less frequent.

By examining the density plot, we can gain insights into the spread and concentration of absolute error values, which can help evaluate the predictive accuracy and precision of the model.

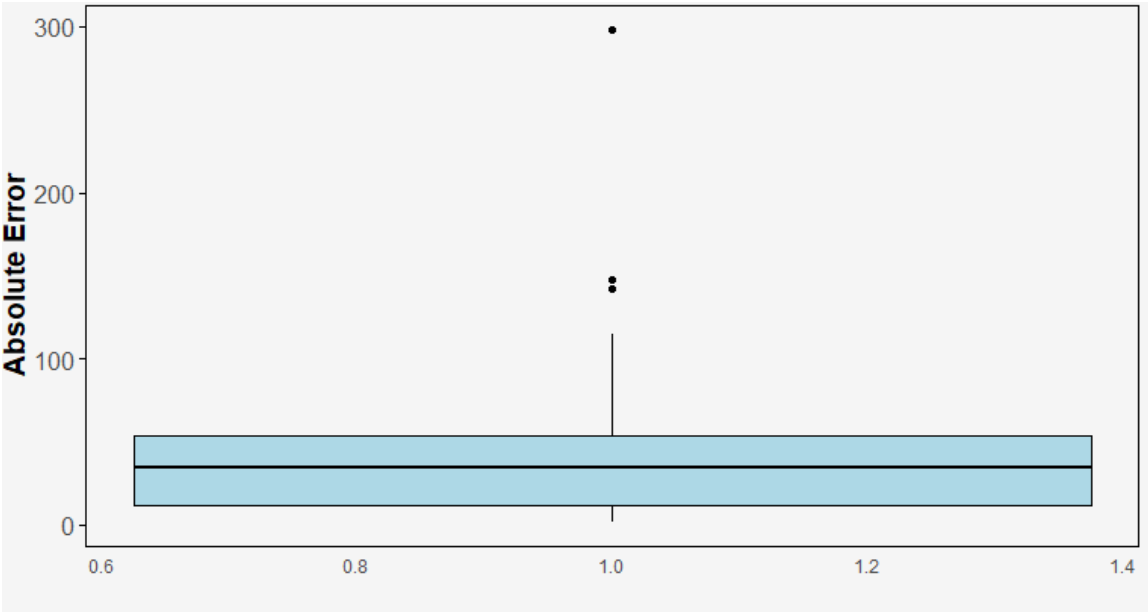

Figure 45. *Distribution of A. Errors for every quantile for our predictive KCDE model*

This box plot in figure 45 is a graphical representation of the distribution of a dataset, showing the minimum, first quartile (Q1), median (Q2), third quartile (Q3), and maximum values. It provides a visual summary of the central tendency, spread, and skewness of the dataset.

- The box plot represents the distribution of absolute errors.
- The x-axis label is left blank to focus on the y-axis values, which represent the absolute errors.

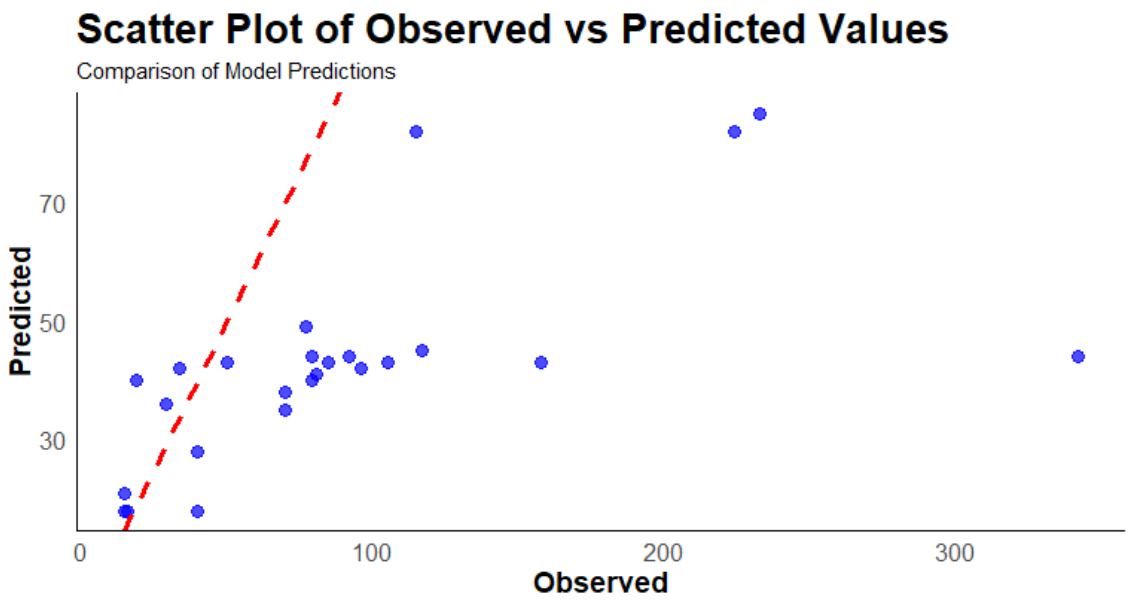

***Figure 46**. Observed vs Predicted values relation for our predictive KCDE model*

The scatter plot in figure 46 is a commonly used visualization for comparing observed and predicted values. It allows us to assess the relationship between two variables and visually examine the agreement between the observed and predicted values.

In the scatter plot, each point represents a data point with its observed value plotted on the x-axis and the corresponding predicted value plotted on the y-axis. Here are some considerations for interpreting the scatter plot:

1. Alignment with the 45-degree line: The dashed red line in the scatter plot represents a perfect 1:1 correspondence between observed and predicted values. Points that fall close to this line indicate a good agreement between the observed and predicted values.

2. Dispersion of points: The spread or dispersion of points around the dashed red line provides insights into the model's accuracy. Ideally, we would want the points to be tightly clustered around the line, indicating

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

accurate predictions. If the points are scattered widely, it suggests greater variability or inconsistency between observed and predicted values.

3. Outliers: Outlying points that deviate significantly from the general pattern can indicate potential issues with the model. These outliers may represent cases where the model fails to capture the underlying patterns or where there are other influential factors not accounted for in the model.

4. Pattern or trend: Examining the overall pattern or trend of the points can provide insights into the relationship between the observed and predicted values. For example, if the points form a clear upward or downward trend, it suggests a systematic bias in the predictions.

5. Distribution of points: The concentration of points in specific regions of the plot can provide information about the distribution of errors. For instance, if the points are more concentrated towards one end of the plot, it indicates a tendency of overprediction or underprediction by the model.

Overall, in a well-performing model, we would expect to see a tight cluster of points aligned closely to the 45-degree line, indicating a close agreement between observed and predicted values. On the other hand, a poorly performing model may exhibit scattered points, outliers, or a systematic deviation from the 45-degree line.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

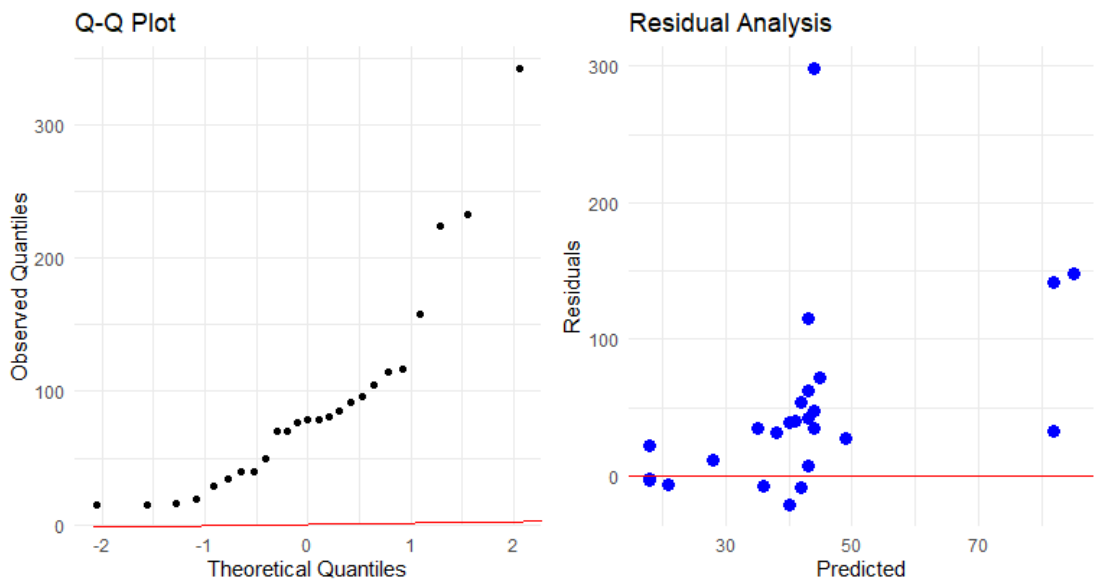

***Figure 47**. GOF and Residual analysis for our predictive KCDE model*

The Q-Q plot:

Quantile-quantile plot is a graphical tool used to assess whether a dataset follows a specific theoretical distribution, in this case, comparing the observed values and predicted values. Here's how to interpret the Q-Q plot:

Points along the diagonal line: Ideally, the points on the plot should fall along a straight line, indicating that the observed values and predicted values are normally distributed. Deviations from the diagonal line suggest departures from normality.

Deviations from the diagonal line: If the points deviate from the diagonal line, it suggests a departure from normality. For example, if the points curve upward, it indicates that the predicted values tend to be lower than the observed values. Conversely, if the points curve downward, it suggests that the predicted values tend to be higher than the observed values.

Outliers: Points that are far away from the diagonal line indicate potential outliers or extreme values that don't conform to the expected distribution.

These outliers may represent cases where the model fails to capture the underlying patterns or where there are other influential factors not accounted for in the model.

Overall pattern: The overall pattern of the points can provide insights into the distributional characteristics of the predicted values compared to the observed values. It helps to identify any systematic deviations or trends in the predictions.

Goodness of fit: A good fit is indicated when the points closely follow the diagonal line, suggesting that the predicted values align well with the observed values. Deviations from the diagonal line may indicate areas where the model can be improved.

For the Residual Analysis plot:

The residual plot is a diagnostic tool used to assess the quality of the model's predictions by examining the residuals (the differences between the observed and predicted values). Here's how to interpret the residual analysis plot:

Horizontal line at y = 0: The red horizontal line at y = 0 represents the ideal situation where the residuals are centered around zero. It indicates that, on average, the model's predictions are unbiased.

Scatter of points around the horizontal line: The scatter of points around the horizontal line provides insights into the randomness and distribution of the residuals. Ideally, the points should be scattered randomly around the line, indicating that the model's predictions have captured the underlying variability in the data.

Patterns or trends: Any discernible patterns or trends in the residuals may suggest that the model has systematic biases or is failing to capture certain patterns in the data. For example, if there is a clear pattern of increasing or decreasing residuals as the predicted values change, it suggests that the model has a bias or is missing important features.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Outliers: Outlying points in the residual plot represent observations with large discrepancies between the observed and predicted values. These outliers may indicate potential issues or data points that are poorly fit by the model.

By examining the Q-Q plot and residual analysis plot, we can assess the assumptions of the model, such as normality and independence of residuals. Deviations from the expected patterns provide valuable insights into the model's performance, potential biases, and areas for improvement.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.4.5 Conclusion

Based on the various performance metrics and plots we have examined. It is evident that the Kernel Conditional Density Estimation (KCDE) model has significant room for improvement in its predictive performance for this particular dataset.

The model exhibits a Mean Absolute Error (MAE) of 52.44, indicating that the model's predictions are, on average, about 52.44 units away from the actual values. Additionally, the Root Mean Squared Error (RMSE) of 82.43 signifies that larger errors are being given substantial weight, which is not desirable in this context. The Mean Absolute Percentage Error (MAPE) of 47.95% suggests that the model's predictions are roughly 48% off from the actual values, signifying a rather high error rate.

The scatter plot of residuals further confirms these findings. Ideally, we would want to see the residuals randomly scattered around zero, indicating a good model fit. However, the observed pattern in our residuals plot indicates that our model is not capturing some aspects of the data. The potential presence of a systematic pattern suggests issues with heteroscedasticity and possibly non-linear trends not being accounted for in the model.

Moreover, the histogram of forecast errors reveals that the errors are not symmetrically distributed around zero, suggesting the presence of a bias in the forecasts. Ideally, the histogram should be centered around zero, implying an unbiased forecast.

Finally, the density plot of absolute errors further emphasizes the issues with the model, as it exhibits a somewhat spread of absolute error values indicating higher variability and less accuracy in the model's predictions.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.5 HHH4 WITH WEIGHTED SCHEMES FOR PAST INCIDENCE

3.5.1 Methodology – parametrization

Methodology for HHH4 with weighted schemes model (Johannes Bracher & Leonhard Held, 2020, (69))

1. **Endemic-epidemic model**

Let's see the parameters of the current model and what its going to change in its weights. Given the past, the number of cases Y_{it} in unit $i = 1, \dots, m$ and week t is assumed to follow a negative binomial distribution:

$Y_{it} \mid \mathbf{Y}_{t-1}, \mathbf{Y}_{t-2}, \dots \sim \mathrm{NBin}(\lambda_{it}, \psi_i)$ (1)

with conditional mean λ_{it} and overdispersion parameter ψ_i. The conditional variance is thus $\lambda_{it} + \psi_i \lambda_{it}^2$ and reduces to λ_{it} for $\psi_i = 0$. A common simplification is to assume one global overdispersion parameter, i.e. $\psi_i = \psi$ for $i = 1, \dots, m$. Counts from different units i and j at time t are assumed to be conditionally independent given the past.

The conditional mean λ_{it} in (1) is further decomposed into

$\lambda_{it} = v_{it} + \phi_{it} \sum_{j=1}^{m} \lfloor w_{ji} \rfloor Y_{j,t-1}$, (2)

where v_{it} is referred to as the endemic component and captures infections not directly linked to observed cases from the previous week. The remaining autoregressive term in (2) is the epidemic component and describes how the incidence in region i is linked to previous cases $Y_{j,t-1}$ in units $j = 1, \dots, m$. The parameters v_{it} and ϕ_{it} are constrained to be non-negative and modeled in a log-linear fashion, for instance with sine-cosine terms to account for seasonality (Held & Paul, 2012). Long-term temporal trends and covariates such as meteorological conditions (Bauer & Wakefield, 2018; Cheng., 2016) or vaccination coverage (Herzog., 2011) could also be included.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

The coupling between units is described by weights w_{ji}, which enter in (2) after normalization: $\lfloor w_{ji} \rfloor = w_{ji} / \sum_{h=1}^{m} w_{jh}$. Unrestricted estimation of all m^2 weights is usually unstable in practice, so more parsimonious and epidemiologically meaningful parameterizations are introduced. Specifically, the weights can be based on social contact data for spread across age groups (Meyer & Held, 2017) or on the geographical distance between cases to describe spatiotemporal spread (Meyer & Held, 2014). In the latter case, the weights can be specified through a power law:

$$w_{ji} = \left(o_{ji} + 1\right)^{-\rho} \quad (3)$$

where o_{ji} is the path distance between regions j and i (with $o_{ii} = 0, o_{ji} = 1$ for direct neighbors i and j and so on) and ρ is a decay parameter to be estimated from the data. The power law formulation is motivated from human movement behavior (Brockmann., 2006) and has been found to be an efficient way of capturing spatial dependence.

2. Weighting schemes for past incidence

This method starts a singular EE model exhibiting constant endemic and epidemic parameters, denoted as v and ϕ. Subsequently, this model develop into the following mathematical model:

$$\boldsymbol{\lambda_t = v + \phi \sum_{d=1}^{p} \lfloor u_d \rfloor Y_{t-d}} \quad (1)$$

Here, the weights $\lfloor u_d \rfloor = u_d / \sum_{g=1}^{p} u_g$ are normalized and restricted to be positive. These attributes are inspired by the model of a discrete-time serial interval distribution and serve practical considerations as well. Ensuring all weights sum up to one is necessary for model identification, while maintaining positive weights is crucial for the model's proper definition. The model would not be well-defined if u_d< 0 since this might lead to negative λ_t, for which the negative binomial distribution is not defined.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

To estimate the normalized weights $\lfloor u_d \rfloor$ without additional assumptions would be a flexible approach, but it might lead to numerical instabilities and suboptimal predictions due to the increased parameters. On the other extreme, a predetermined weighting scheme can be used based on existing epidemiological knowledge like in the research of Wang (2011), which they fix the weights at specific values. To balance this out, the model is parameterizing the weight schemes in a way that they correlate with different shapes of the serial interval distribution. This is combined with a data-driven estimation of an underlying scalar parameter.

The scalar parameter is referring to a quantity that is represented by a single real number or a single value, as opposed to a vector or a matrix which consist of multiple values. In the context of the discussion, the scalar parameter refers to a single parameter value that is estimated or used in the mathematical model. It is a scalar value that influences the behavior or outcome of the model.

The model considers three parameterizations, truncated to lags $d = 1, \dots, p$ and depending on an unknown weighting parameter κ.

1. Shifted Poisson weights correspond to

$u_d = \frac{\kappa^{d-1}}{(d-1)!}\exp(-\kappa),\ \kappa > 0,$ (4)

as used by Kucharski, (2014) to analyze daily avian influenza data.

2. A linear decay of the weights (subject to a non-negativity constraint),

$u_d = \max(1 - \kappa d, 0),\ 0 < \kappa < 1,$ (2)

corresponds to a triangular serial interval distribution.

3. A geometric weighting scheme.

$u_d = (1 - \kappa)^{d-1}\kappa,\ 0 < \kappa < 1,$ (3)

is considered. The latter can also be motivated by the effect underreporting has on the autocorrelation of a firstorder autoregressive process (Bracher, 2019). Moreover, note that the formulation (1) with geometric weights (3) and $p \to \infty$ is equivalent to the negative binomial INGARCH (1,1) model:

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

$\lambda_t = \alpha + \beta Y_{t-1} + \gamma \lambda_{t-1}$,

where $\alpha = v\kappa, \beta = \phi\kappa$ and $\gamma = 1 - \kappa$ (Fokianos., 2009; Zhu, 2011).

Parameterizations (2) and (3) force the weight u_1 to be the largest and imply a decay in infectiousness. This is reasonable for weekly or monthly data, and for diseases with short serial intervals. The Poisson version of shifted weights can also capture an initial increase in infectiousness and may thus be more appropriate for longer serial intervals or daily data.

This model also can combine the multivariate formulation with the suggested weighting schemes:

$$\boldsymbol{\lambda_{it} = v_{it} + \phi_{it} \sum_{d=1}^{p} \sum_{j=1}^{m} \lfloor u_d \rfloor \lfloor w_{ji} \rfloor Y_{j,t-d}}$$

where v_{it} and ϕ_{it} now depend on unit i and time t. The weights $\lfloor u_d \rfloor$, however, are assumed to be constant across units and time.

3. Inference

Here the likelihood-based inference procedure from the R package surveillance is extended (**first model we examined)** (Meyer, 2017) to fit higher-order lag models (3)-(4) in the package hhh4addon .To optimize the likelihood given the weights $\lfloor u_d \rfloor$, the efficient and robust routine provided in surveillance for first-order models (Meyer & Held, 2014; Paul & Held, 2011; Paul., 2008) was adapted. Then this model uses the profile likelihood approach (Held & Sabanés Bové, 2020) to estimate the unrestricted weights u_d or the weighting parameter κ. This simplifies the process and eliminates the need for implementing complex derivatives of the log-likelihood function relative to the extra parameters.

The maximum of the profile likelihood function can be found using standard numerical optimization. If the weights are estimated in an unrestricted nonparametric manner, it may be necessary to try several starting values to ensure convergence. This is also the case for the triangular parameterization (2).

Under the parametric formulations (4)-(3), the weights $\lfloor u_d \rfloor$ usually become negligible after a certain order. To find a sufficiently large p, the suggestion

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

is to inspect the weights visually and plotting the order p against the AIC (or log-likelihood, as the complexity of the model does not depend on p) to check when changes become negligible. For the unrestricted estimation of the weights $\lfloor u_1 \rfloor, \dots, \lfloor u_p \rfloor$, where the number of model parameters depends on p, we choose the order based on the AIC, as is common for prediction purposes.

To take into account the uncertainty surrounding epidemiological forecasts, they should take the form of predictive distributions rather than deterministic point forecasts (Held, 2017). We consider monthly or weekly updated forecasts for different horizons $h = 1, \dots, H,$. This means that for each week t, the model is re-fitted using all data already available in order to predict the incidence in weeks $t + 1, \dots, t + H$. To forecast from the fitted model without carrying forward the uncertainty in the parameter estimates the model is using plug-in forecasts(a method where the fitted model is used directly to generate predictions without explicitly propagating the uncertainty in the estimated parameters. Instead of incorporating the uncertainty in the parameter estimates into the forecast distribution, plug-in forecasts rely on using the point estimates of the parameters obtained from the fitted model.

The term "plug-in" stems from the idea that the estimated parameter values are plugged into the model equations to generate the forecasts. These point estimates are treated as fixed values, disregarding any variability or uncertainty associated with the parameter estimates.

Plug-in forecasts are relatively straightforward to compute since they only require evaluating the model equations using the estimated parameter values. However, they do not account for the potential variability or uncertainty in the parameter estimates, which can result in overly confident or optimistic forecasts.)

Taking this into account requires fitting EE models using MCMC methods in a Bayesian framework (Bauer & Wakefield, 2018; Wakefield, 2019) with carefully chosen prior distributions.

Plug-in one-time-ahead forecast distributions are given by independent negative binomial distributions for the m units, so the predictive densities can

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

be evaluated analytically. For forecast horizons of two or more weeks ahead, predictive means, variances, and autocovariances can be still obtained analytically. The full predictive distributions, however, are no longer given by any standard distribution and are dependent across different units, as they reflect e.g. spatiotemporal dependencies arising over a longer forecast horizon. To evaluate the predictive probability masses $f(\mathbf{Y}_{t+h} \mid M)$.

So next the models implements the following procedure, which uses Rao-Blackwellization (Gelfand & Smith, 1990; Ray, 2017) for improved numerical stability:

1.Generate $k = 1, \dots, K$ samples from the predictive distribution of $(\mathbf{Y}_{t+1}, \dots, \mathbf{Y}_{t+h-1} \mid M)$.

2 For each sample $\left(\mathbf{y}_{t+1}^{(k)}, \dots, \mathbf{y}_{t+h-1}^{(k)}\right)$ and each unit $i = 1, \dots, m$, evaluate.

$$f\left(Y_{i,t+h} \mid M, \mathbf{y}_{t+1}^{(k)}, \dots, \mathbf{y}_{t+h-1}^{(k)}\right)$$

on a suitably chosen support $y_{i,t+h} \in \left\{0, \dots, Y_{i,t+h}^{(\max)}\right\}$. These one-time-ahead distributions are conditionally independent negative binomial distributions. The joint probability mass function $f\left(\mathbf{Y}_{t+h} \mid M, \mathbf{y}_{t+1}^{(k)}, \dots, \mathbf{y}_{t+h-1}^{(k)}\right)$ is thus given by the product of the unit-wise probability mass functions.

3 Compute $f(\mathbf{Y}_{t+h} \mid M) = 1/K \cdot \sum_{k=1}^{K} f\left(\mathbf{Y}_{t+h} \mid M, \mathbf{y}_{t+1}^{(k)}, \dots, \mathbf{y}_{t+h-1}^{(k)}\right)$.

As the relevant support of $\mathbf{Y}_{t+h}$ obviously gets very large for even moderate m, we usually only store the unitwise marginal predictive densities or summaries thereof (central 50% and 95% predictive intervals), supplemented with analytically obtained covariances. In all applications the default K value is $K = 1000$, which leads to very stable results.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

4. Model assessment

To assess the performance of probabilistic forecasts, proper scoring rules have become the standard in infectious disease epidemiology. A score is deemed "proper" if a forecaster has no incentive to deviate from her actual belief, and "strictly proper" if any such deviation leads to a penalty. A commonly used strictly proper score is the logarithmic score:

$$\log S(y_{\text{obs}} \mid M) = -\log[f(y_{\text{obs}} \mid M)],$$

i.e., the negative log predictive probability mass the model M assigned to the observed value $y_{\text{obs.}}$. The score is negatively oriented, i.e., lower values are better. To aggregate over several forecasts, one can use average log scores, which are still strictly proper. For multivariate h-weeksahead forecasts of $\mathbf{Y}_{t+h}$, we apply the multivariate logarithmic score (Gneiting, 2008):

$$\log S(\mathbf{y}_{t+h} \mid M) = -\log[f(\mathbf{y}_{t+h} \mid M)]$$

where M is now a model fitted based on the information available at time t. In practice, the previous logarithmic score is often standardized and divided by $m = \dim(\mathbf{y}_t)$. For one-time-ahead predictions the standardized multivariate log score can be evaluated analytically, as it is just the average of m negative binomial log-densities.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.5.2 R code implementation for the hhh4 model with weighted schemes

3.5.2.1 *Fit model and choose order*

The provided code segment involves the implementation of the hhh4 package to fit and evaluate time series models using different lag weighting schemes. Initially, the data is read and preprocessed, converting it into the required sts (spatiotemporal surveillance) object format. The code proceeds to define various lag weighting schemes, encompassing autoregressive (AR), geometric, Poisson, linear, and unrestricted lag models. Subsequently, these models are fitted to the data, with different lag orders ranging from 1 to 10 being considered. The AIC values for each model are computed and stored in a structured matrix. This process entails testing and comparing various lag weighting schemes for time series modeling, enabling the selection of the optimal order based on AIC criteria. The results are then saved to a CSV file for further analysis and accessibility later.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.5.2.2 *Describe the model*

In the subsequent section of the code, a series of time series models are fitted and evaluated based on various lag weighting schemes. The code initializes a range of controls for each scheme and then iterates through the models to fit them to the provided data. The models encompass Autoregressive (AR), geometric, Poisson, linear, unrestricted, and a specialized scheme called "siraj." The code computes Akaike Information Criterion (AIC) values for each model and the resulting improvements in AIC relative to a reference model. The script then proceeds to visualize the changes in AIC values for different lag orders using a line plot. Additionally, the color palette is defined for visually distinguishing the models. Finally, the code creates a multi-panel plot for further analysis. This part of the code effectively explores and contrasts various time series models using different lag weighting schemes and visually presents their evaluation outcomes (see appendix 3.5.2).

***Below we can see the improvement in AIC per p**

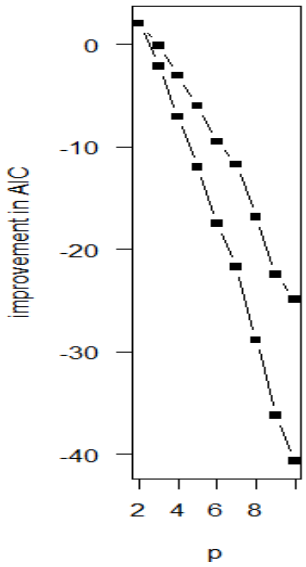

Figure 48. Improvement of AIC per p

3.5.3 Fit model

In this subsequent section of the code, a comprehensive evaluation of various time series models is performed using different lag weighting schemes. First, the code sets up the required folder structure and initializes key parameters. It also establishes controls for different lag structures and defines the timepoints for which models will be fitted. Notably, this code primarily focuses on fitting models with a minimum lag of 1.

The subsequent steps involve iteratively fitting models for each selected timepoint during the validation period. Different lag weighting schemes are explored, including Autoregressive (AR1), geometric, Poisson, linear, unrestricted, and a scheme based on the work of Siraj (2017). For each scheme, specific control settings are tailored and applied to the model fitting process. These models are fitted to the data by adjusting the lag parameters and other relevant parameters. Special attention is paid to accommodating initial conditions, especially for the first timepoint.

Throughout the fitting process, each model is evaluated against the data. As models are fitted and assessed for each individual timepoint, the code prints the current timepoint to track progress. Overall, this section of the code is responsible for systematically fitting various models with different lag weighting schemes to the data for thorough evaluation (see appendix 3.5.3).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.5.4 Obtain forecast fuction

The next code segment basically calculates and introduces a custom function named **forecasting_nStepAhead**. This function is pivotal for generating forecasts over multiple time steps in a time series context. The function's purpose, key inputs, and essential functionalities are as follows:

- **Function Purpose**: The **forecasting_nStepAhead** function is crafted to deliver forecasts for a designated number of time steps into the future. It calculates and compiles diverse predictive metrics, providing insights into the model's predictive performance.
- **Input Parameters**: The function is parameterized with critical inputs: the fitted model object (**fit**), the original time series object (**stsObj**), the reference time point for forecasting (**tp_cond**), the forecast horizon (**horizon**), and the number of simulation iterations (**n_sim**).
- **Model Type Handling**: The function adeptly differentiates between two model types, "hhh4lag" and "hhh4," adapting its computations accordingly.
- **Single-Step Forecast**: In cases where the forecasting horizon is 1, the function performs a single-step ahead forecast utilizing specialized functions (**oneStepAhead_hhh4lag** or **oneStepAhead**). It computes log scores, indicating the model's accuracy for this immediate prediction.
- **Multi-Step Forecasts**: For longer forecasting horizons, the function simulates diverse sample paths, drawing from the fitted model. It subsequently evaluates log scores for each simulated path.
- **Predictive Metrics**: Essential predictive metrics are calculated for each unit, encompassing predictive means, variances, quantiles of the forecast distribution, and prediction intervals derived from cumulative distribution functions.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

- **Accuracy Assessment**: Unit-wise log scores and Probability Integral Transform (PIT) values are determined to assess the model's prediction accuracy on a per-unit basis.
- **Output Format**: The function generates a comprehensive output list, encapsulating critical predictive information. This includes predictive means, variances, quantiles, observed values, unit-wise log scores, unit-wise PIT values, and a multivariate log score.

This function's capability to provide forecasts spanning multiple time steps offers invaluable insights into future trends and patterns within time series data. By employing simulations and assessing diverse metrics, it empowers analysts to make informed decisions based on the model's predictions and their associated uncertainties (see appendix 3.5.4).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.5.5 Actual forecast

Model Evaluation and Forecasting

In this section of the code, a meticulous evaluation of diverse time series models takes center stage. The code's progression unfolds as follows:

1. **Setup and Initialization**: The code begins by establishing the essential infrastructure, creating the necessary folders for log scores and forecasts. Key parameters such as the maximum forecasting horizon and timepoints are defined. The number of units in the dataset is also determined.

2. **Data Frame Template Creation**: A template data frame is crafted to hold the detailed results of the evaluation. It includes fields for data set information, unit details, prediction horizons, lag structures, prediction times, and various prediction metrics. This template acts as a scaffold for accumulating the evaluation results.

3. **Results Storage Initialization**: Containers for detailed results and log scores are initialized. These structures will store comprehensive evaluation data for each lag structure.

4. **Iterative Evaluation**: For each predefined lag structure (AR1, end, geometric, linear, Poisson, unrestricted, Siraj), the code iterates through the evaluation process. It steps through different timepoints during the validation period.

5. **Model Fitting for Different Lag Structures**: Within each iteration, the code selects the appropriate model configuration based on the current lag structure. The model fitted during the exploratory phase is adapted for evaluation. It is fitted to the data, incorporating relevant adjustments for lag parameters, initial conditions, and other settings.

6. **Forecasting and Evaluation**: The fitted model is used for forecasting over various prediction horizons. Predictions are compared against the

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

actual data to evaluate the model's performance. Log scores, prediction intervals, and other assessment metrics are computed.

7. **Results Aggregation**: The evaluated metrics, along with model and dataset information, are integrated into the template data frame. This forms a detailed record of the model's performance under the specific lag structure.

8. **Progress Tracking and Logging**: During the iterative evaluation, the code prints out progress updates, indicating the current lag structure and timepoint being evaluated.

9. **Results Storage and Export**: After evaluation for each lag structure, the detailed evaluation results and log scores are saved as CSV files for future access and analysis.

This section of the code systematically conducts thorough evaluations of multiple time series models under various lag weighting schemes. The primary objective is to gauge the models' effectiveness in forecasting and capturing the underlying patterns within the observed data.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.5.6 Results

LOG SCORES PER MODEL PER HORIZON

	hhh4_log_ar1	hhh4_log_pois	hhh4_log_lin	hhh4_log_geom	hhh4_log_unres	hhh4_log_siraj
h1	4.946424	4.946437	4.946437	4.946437	4.946437	5.480640
h2	5.413138	5.412441	5.412441	5.412441	5.412441	5.503106
h3	5.611944	5.611159	5.611159	5.611159	5.611159	5.542784
h4	5.700006	5.700226	5.700226	5.700226	5.700226	5.526894
h5	5.739203	5.739665	5.739665	5.739665	5.739665	5.529473
h6	5.762800	5.761352	5.761352	5.761352	5.761352	5.541936
h7	5.834524	5.836663	5.836663	5.836663	5.836663	5.585535
h8	5.949142	5.949327	5.949327	5.949327	5.949327	5.638422

Table 2. Log scores per model per horizon

The log scores observed for Horizon 1 are closely aligned with our original HHH4 model, which demonstrated impressive performance. It is logical and somewhat expected to see an increase in log scores as we expand the horizon points. This correlation suggests that as we look further into the future, our predictions inherently become less certain, thus leading to higher log scores.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Model	MAE	RMSE	R_Squared	MAPE	MAD	MSE	MPE	MD	MAPE.1
ar1	55.714	83.224	-0.2518261	0.8936543	51.67237	6926.234	0.4563257	55.26819	0.8936543
pois	55.76565	83.32461	-0.2548545	0.8947496	51.90371	6942.99	0.4566843	55.20559	0.8947496
lin	55.76565	83.32461	-0.2548545	0.8947496	51.90371	6942.99	0.4566843	55.20559	0.8947496
geom	55.76565	83.32461	-0.2548545	0.8947496	51.90371	6942.99	0.4566843	55.20559	0.8947496
unres	55.76565	83.32461	-0.2548545	0.8947496	51.90371	6942.99	0.4566843	55.20559	0.8947496
siraj	78.2803	102.7528	-0.9082454	1.647426	78.91441	10558.14	1.371734	55.60274	1.647426

Table 3. Various evaluation metrics per model

Brief explanation

- MAE (Mean Absolute Error): The average of the absolute differences between the predicted and actual observations. Lower values indicate better model performance.
- RMSE (Root Mean Squared Error): The square root of the average squared differences between the predicted and actual observations. Lower values are better, and it gives higher penalties to large prediction errors compared to MAE.
- R_Squared: The proportion of the variance in the dependent variable that is predictable from the independent variable(s). Higher values (closer to 1) indicate better model fit to the data.
- MAPE (Mean Absolute Percentage Error): The average of the absolute percentage differences between the predicted and actual observations. Lower values are better.
- MAD (Mean Absolute Deviation): The average of the absolute differences between each data value and the mean. Lower values are better.
- MSE (Mean Squared Error): The average of the squared differences between the predicted and actual observations. Lower values are better.

- MPE (Mean Percentage Error): The average of the percentage differences between the predicted and actual observations. Lower absolute values are better.
- MD (Mean Deviation): The average of the differences between the predicted and actual observations, considering their direction. Lower absolute values are better.

Upon reviewing the range of evaluation metrics, it is evident that the AR1 model offers the most reliable and accurate performance. It consistently outperforms other models across all metrics, with the lowest error rates (including MAE, RMSE, MAPE, MAD, MSE, MPE, and MD), and the highest R_Squared value, suggesting an excellent fit to the data. Thus, based on our evaluation, we choose the AR1 model for future predictions.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

	Model_Horizon	Mean	Median	SD	IQR
ar1_h1	ar1_h1	4.946424	4.933334	1.342634	1.337943
ar1_h2	ar1_h2	5.413138	5.131235	1.672874	1.483349
ar1_h3	ar1_h3	5.611944	5.400922	1.619892	1.204967
ar1_h4	ar1_h4	5.700006	5.413594	1.558851	1.551322
ar1_h5	ar1_h5	5.739203	5.580408	1.42493	1.801704
ar1_h6	ar1_h6	5.7628	5.718708	1.352345	1.787955
ar1_h7	ar1_h7	5.834524	5.870275	1.293472	1.737532
ar1_h8	ar1_h8	5.949142	5.965116	1.27212	1.643889
pois_h1	pois_h1	4.946437	4.933347	1.34265	1.337949
pois_h2	pois_h2	5.412441	5.131395	1.672719	1.48108
pois_h3	pois_h3	5.611159	5.401486	1.620266	1.205028
pois_h4	pois_h4	5.700226	5.411594	1.559218	1.543601
pois_h5	pois_h5	5.739665	5.583372	1.427607	1.794329
pois_h6	pois_h6	5.761352	5.719334	1.351518	1.803208
pois_h7	pois_h7	5.836663	5.862702	1.293829	1.741842
pois_h8	pois_h8	5.949327	5.966043	1.272911	1.662158
lin_h1	lin_h1	4.946437	4.933347	1.34265	1.337949
lin_h2	lin_h2	5.412441	5.131395	1.672719	1.48108
lin_h3	lin_h3	5.611159	5.401486	1.620266	1.205028
lin_h4	lin_h4	5.700226	5.411594	1.559218	1.543601
lin_h5	lin_h5	5.739665	5.583372	1.427607	1.794329
lin_h6	lin_h6	5.761352	5.719334	1.351518	1.803208
lin_h7	lin_h7	5.836663	5.862702	1.293829	1.741842
lin_h8	lin_h8	5.949327	5.966043	1.272911	1.662158
geom_h1	geom_h1	4.946437	4.933347	1.34265	1.337949
geom_h2	geom_h2	5.412441	5.131395	1.672719	1.48108
geom_h3	geom_h3	5.611159	5.401486	1.620266	1.205028
geom_h4	geom_h4	5.700226	5.411594	1.559218	1.543601
geom_h5	geom_h5	5.739665	5.583372	1.427607	1.794329
geom_h6	geom_h6	5.761352	5.719334	1.351518	1.803208
geom_h7	geom_h7	5.836663	5.862702	1.293829	1.741842
geom_h8	geom_h8	5.949327	5.966043	1.272911	1.662158
unres_h1	unres_h1	4.946437	4.933347	1.34265	1.337949
unres_h2	unres_h2	5.412441	5.131395	1.672719	1.48108
unres_h3	unres_h3	5.611159	5.401486	1.620266	1.205028
unres_h4	unres_h4	5.700226	5.411594	1.559218	1.543601
unres_h5	unres_h5	5.739665	5.583372	1.427607	1.794329
unres_h6	unres_h6	5.761352	5.719334	1.351518	1.803208
unres_h7	unres_h7	5.836663	5.862702	1.293829	1.741842
unres_h8	unres_h8	5.949327	5.966043	1.272911	1.662158
siraj_h1	siraj_h1	5.48064	5.261915	1.247932	0.9905685
siraj_h2	siraj_h2	5.503106	5.288356	1.265433	1.002608
siraj_h3	siraj_h3	5.542784	5.403307	1.074886	0.7798922
siraj_h4	siraj_h4	5.526894	5.355231	0.8983858	0.770281
siraj_h5	siraj_h5	5.529473	5.404982	0.8447875	0.7981912
siraj_h6	siraj_h6	5.541936	5.525591	0.8038562	0.6824465
siraj_h7	siraj_h7	5.585535	5.594944	0.761674	0.5603263
siraj_h8	siraj_h8	5.638422	5.629079	0.6820524	0.4834424

Table 4. Mean log score for each model for each horizon

As we expect the mean log mean score and the other evaluation metrics for every +1 horizon is getting worse. Overall, the best model is again the AR1 model for every horizon point.

BOXPLOT: DISTRIBUTION OF PREDICTED MEAN PER MODEL

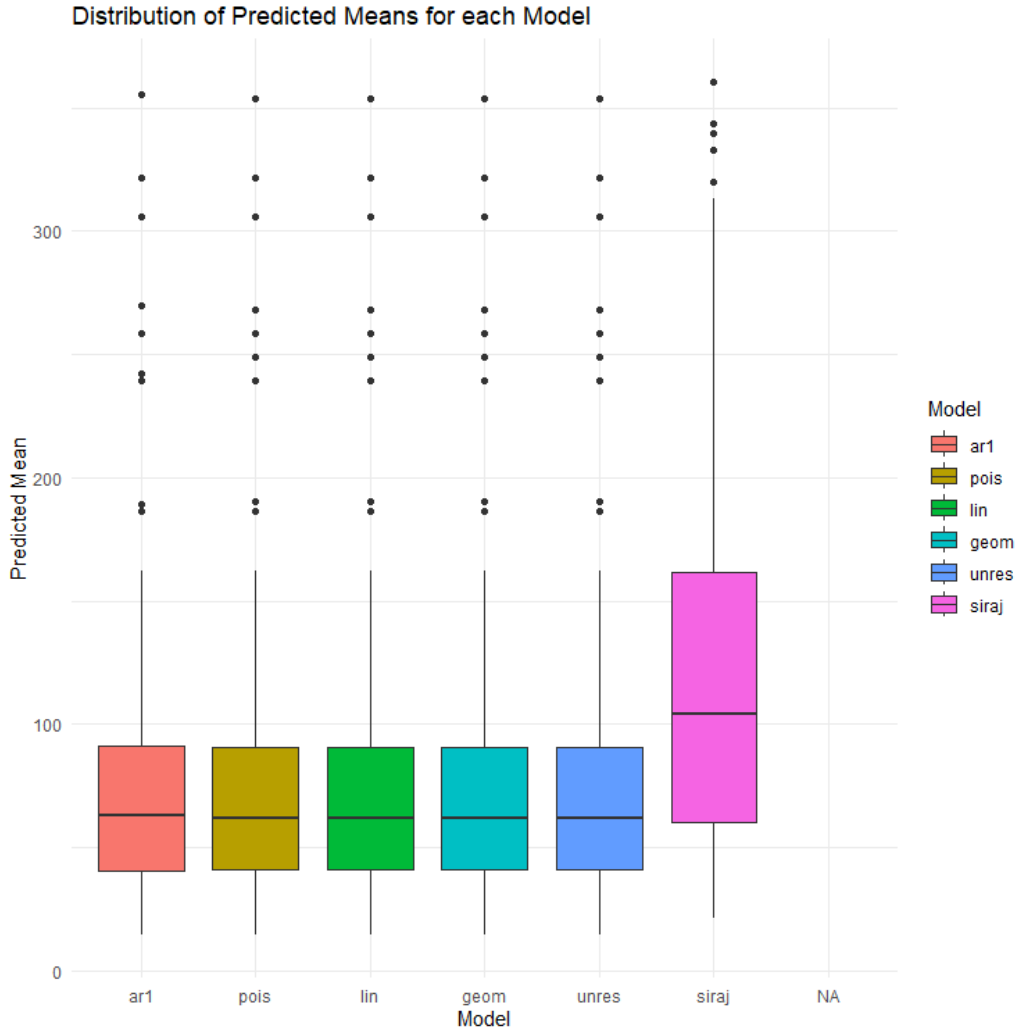

***Figure 49**. Predicted mean for each distribution for our predictive with weighted schemes model*

In this resulting boxplot (figure 49), each model (represented by a different lag structure) have its own box showing the distribution of its predicted means. The variation in the box sizes and their median lines gives a visual representation of how each model's predictions vary. Small differences between each model as we expect.

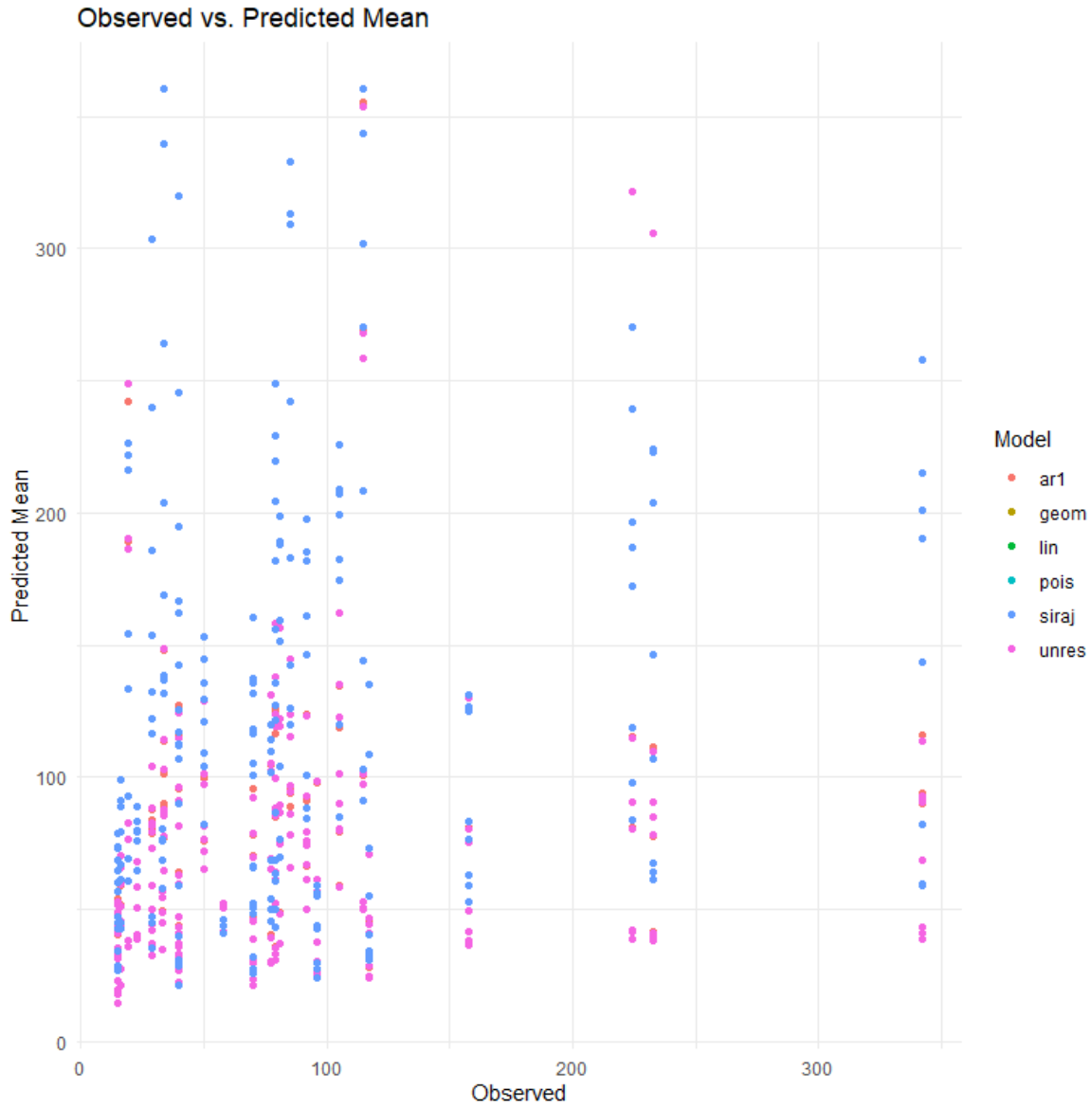

***Figure 50**. Predicted vs Actual mean for each distribution for our predictive with weighted schemes model*

Figure 50 visualizes how accurately each of our models predicts the observed data. Each point represents an observation, with the x-coordinate being the actual observed value and the y-coordinate being the model's prediction. The different colors represent different models.

Ideally, we'd like all the points to lie along the line where x equals y, because this would mean that our predictions perfectly match the observations. The closer a model's points are to this line, the more accurate the model is.

We can also compare the models by looking at which one's points are generally closer to this line. This way, we can identify which model performs best at predicting the observed data.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

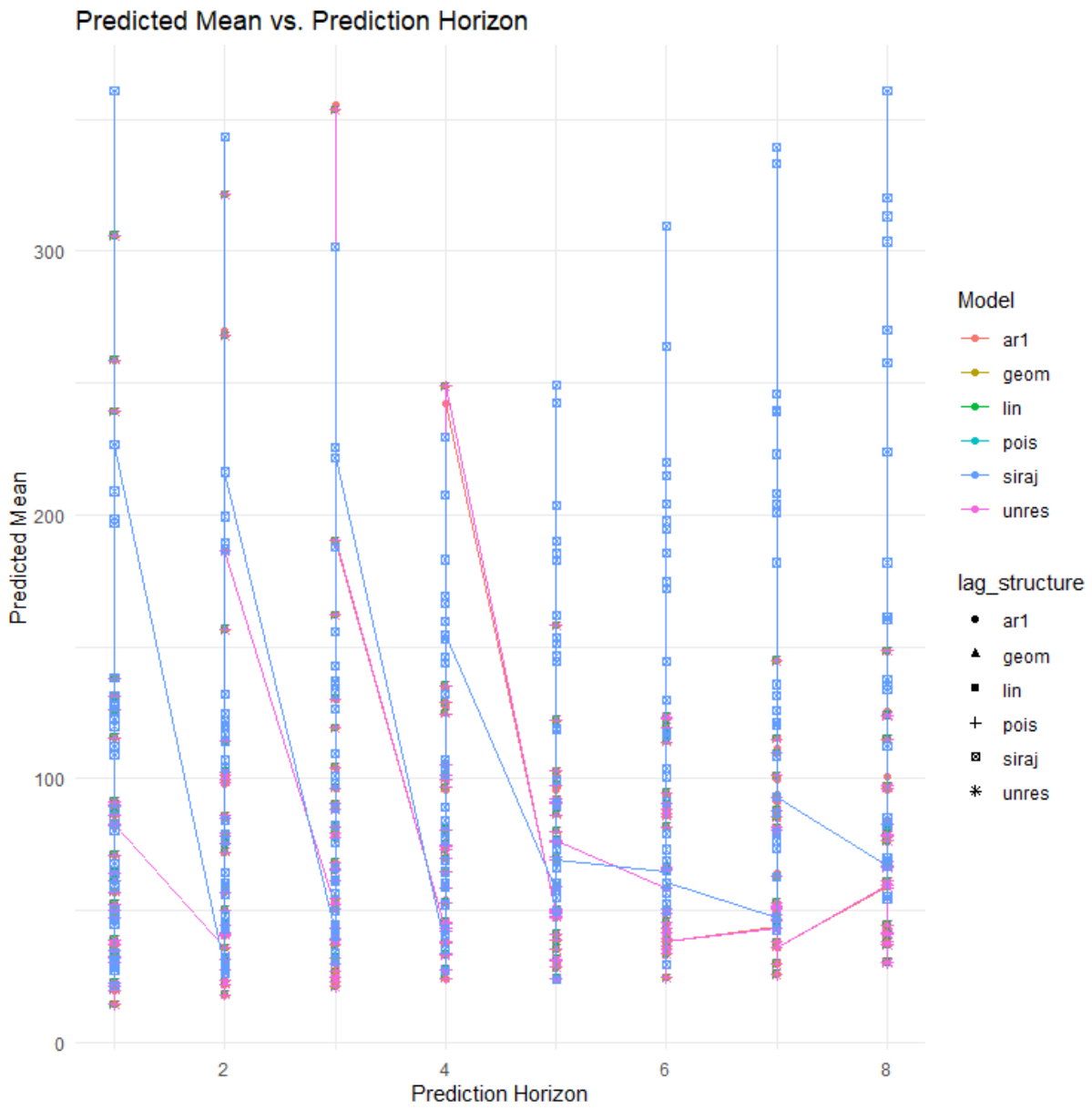

***Figure 51**. Prediction per horizon prediction change for each distribution for our predictive with weighted schemes model*

Figure 51 gives a visual representation of how each model's predictions change as the prediction horizon increases. We can compare the models by looking at how close the predicted values (the lines) are to the actual observations.

This line plot shows how the predicted mean values from our models change as we try to predict further into the future, represented by the prediction horizon on the x-axis. Each line represents a different model, with different colors for each one.

The y-axis shows the mean value that each model predicts for each horizon. Points on the line correspond to the actual predictions made by the model at each horizon. By comparing the lines, we can see how each model's predictions change as we try to predict further into the future, and which models predict higher or lower values on average.

Ideally, we want our models to maintain a high degree of accuracy even as the prediction horizon increases. Therefore, the model whose line stays closest to the actual observed data points is typically performing the best.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.5.7 Naïve model forecast with weighted schemes

"The naive forecast predictive model is a simple approach used to make predictions based solely on historical data (69). It assumes that future values will be the same as the most recent observed value. This method is straightforward and requires minimal computation, making it suitable for quick estimations. However, the naive model doesn't consider any underlying patterns, trends, or seasonality in the data. As a result, it may not perform well when faced with complex or volatile datasets, and it's often used as a benchmark to evaluate the performance of more advanced forecasting methods.

In contrast, the naive forecast predictive model with weighted schemes is an extension of the traditional approach. It acknowledges seasonality by introducing seasonal weights, giving recent observations more importance than distant ones. This allows the model to capture short-term fluctuations while maintaining the simplicity of the naive approach. By incorporating seasonality, the model recognizes that certain time periods may have more influence on future predictions than others. Despite its ability to capture seasonality, the model doesn't consider serial dependence. It is implemented using a negative binomial generalized linear model (GLM) with categorical covariates such as the calendar month. In the multivariate case, additional covariates like district are included.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

R code explanation

Here's a breakdown of the process (appendix 3.5.7):

1. **Model Functions:** The code defines two functions, **naive_forecast_glmnb** and **naive_forecast_glmnb_multiv**, for implementing the naive forecast with seasonal weights using a Negative Binomial distribution. The second function caters to multivariate time series forecasting.

2. **Data Preparation:** The code uses a time series dataset and sets parameters like the prediction time, maximum forecast horizon, and frequency of the data.

3. **Forecasting Process:** The code iterates through specified time points to perform the weighted naive forecasts. It calculates seasonal weights and fits a GLM with a Negative Binomial distribution to the training data.

4. **Results Storage:** The code stores forecasted values, prediction intervals, observed values, and log scores in structured data formats.

5. **Output:** The code generates log scores and summary statistics for the weighted naive forecasts, providing insights into the model's predictive accuracy.

By employing weighted schemes, this code illustrates how the naive forecast model can be enhanced to account for seasonality. The model's forecasts and log scores are collected for evaluation, helping to gauge its performance in capturing seasonally influenced fluctuations in the data (see appendix 3.5.7).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Results

LOG SCORES PER HORIZON

h1	h2	h3	h4	h5	h6	h7	h8
7.88	7.97	8.02	8.05	8.07	8.08	8.1	8.14

Table 5. Log scores of our naıve model per horizon

Much worse values compared to our previous models, which is expected.

MAE, RMSE, MAPE & MAD

MAE	RMSE	R_Squared	MAPE	MAD	Pred_Var	LB50	UB50
62.25	89.23	-inf	62.12	62.25	646.91	9.74	33.92

Table 6. Naïve model predictive evaluation metrics

The metrics calculated for the naive forecasting with weighted schemes model are higher (worse) than the corresponding metrics for our other methods, including the previous three main models. This suggests that the naive forecasting model with weighted schemes is not performing as well as the other methods in terms of accuracy and predictive power.

MODEL PREDICTION ACCURACY

Model prediction accuracy is a measure of how well a predictive model's forecasts align with the actual observed values. It provides an understanding of the model's performance in making predictions

We can use this equation for our model to get this metric

```
Prediction accuracy = 1 - (mae / mean_observed)
```

PREDICTION ACCURACY: 0.28

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

A prediction accuracy of 0.277 indicates that the model's predictions are off by approximately 72.29% on average. This means that the model's predicted values deviate from the observed values by a significant amount.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.5.8 Conclusion

The Autoregressive model (AR1) appears to be the most effective predictive model among the evaluated methods. This conclusion is drawn based on the consideration of several statistical metrics that are commonly used to evaluate the performance of predictive models.

First, the Mean Absolute Error (MAE), which measures the average magnitude of the errors in a set of predictions, without considering their direction, is lowest for the AR1 model at 55.71, compared to all other models. A lower MAE suggests the model predictions are closer to the actual observed values.

Second, the Root Mean Square Error (RMSE), which is a quadratic scoring rule that measures the average magnitude of the error and gives a relatively high weight to large errors, is also lowest for AR1 at 83.22.

Third, the Mean Absolute Percentage Error (MAPE), which expresses accuracy as a percentage, and is defined as the average absolute percent difference between actual and predicted values, is lower for AR1 (0.8936543) compared to the other models. This indicates a smaller average relative error, suggesting AR1 is more accurate on a percentage basis.

Lastly, the Mean Square Error (MSE), which is similar to RMSE but does not take the root of the sum of squared differences, is also the lowest for AR1 (6926).

The other models have higher values for these metrics, indicating a worse fit to the data.

The Naïve Forecast has the highest values for these metrics, showing that it is the least accurate model.

In conclusion, the AR1 model is the most accurate of the evaluated models for this specific prediction task. It is important to remember that these results might not be generalizable to other prediction tasks, and each task should be evaluated separately. Different models might work better depending on the specific characteristics of the data and the task at hand.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.6 TIME-SERIES MODEL (ARIMA – SARIMA)

We should run a simple Arima - Sharima time series model to check the possible positive difference that each of the previous specialized methods on our dengue data. The methodology was analyzed earlier in Chapter 2.

3.6.1 R code implementation

A similar example in chapter 2 was made but here is a more detailed explanation of the process of the implementation of this predictive method to our data (see appendix 3.6).

1. The code begins by loading the required R packages, **forecast** and **ggplot2**.
2. A data frame named **data** is created with two columns: "Time" representing dates from January 2008 to December 2020, and "NumValue" containing a series of numerical values.
3. The data in the "NumValue" column is converted into a time series object **ts_data** using the **ts()** function. The time series frequency is set to 12, indicating monthly data, and the start is specified as January 2008.
4. The variable **max_order** is defined with a value of 5. This value determines the maximum order of the ARIMA parameters (p, d, q) that will be explored in the grid search.
5. The code initializes variables **best_aic** with infinity (**Inf**) and **best_order** with a placeholder value of **(0, 0, 0)** to keep track of the best model's AIC value and corresponding ARIMA order.
6. The code enters a series of nested loops to perform a grid search over possible combinations of ARIMA parameters. The loops iterate over values of p, d, and q from 0 to **max_order**.
7. Within the innermost loop, the sum of p, d, and q is checked against **max_order**. If the sum is within the limit, an ARIMA model is fit to the time series data using the specified ARIMA order **(p, d, q)**. The

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

method argument is set to "ML" for maximum likelihood estimation, and **include.mean** is set to **FALSE** to exclude the mean term.

8. The AIC value is calculated for the current ARIMA model using the **AIC()** function.
9. If the current model's AIC is lower than the previously recorded **best_aic**, the **best_aic** is updated, and the **best_order** is set to the current ARIMA order.
10. After the grid search is complete, the code prints the best ARIMA order and the corresponding AIC value to the console

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

3.6.2 Results

BEST ARIMA order

BEST ARIMA order: (0, 1, 4)

Explanation of the Best ARIMA Order

In our specific case, the output indicates that the best ARIMA order found through the grid search is (0, 1, 4). Let's break down what this ARIMA order means:

- **p (AutoRegressive terms): 0**
    - The number of autoregressive terms, denoted as "p," is 0. This indicates that our model doesn't include any lagged values of the time series to predict the current value. In other words, the past values of the time series are not being used for prediction in this component.
- **d (Integrated, Differencing): 1**
    - The "d" component represents differencing, which is used to make the time series stationary. A value of 1 for "d" means that we've applied first-order differencing to our time series. This involves subtracting each value from its previous value. Differencing helps remove trends or seasonality, making the series more amenable to modeling.
- **q (Moving Average terms): 4**
    - The number of moving average terms, denoted as "q," is 4. This indicates that our model takes into account the last four forecast errors to improve predictions. A moving average model captures the relationship between the current value and the errors of past predictions.

In summary, an ARIMA(0, 1, 4) model for our dengue French data suggests that no autoregressive terms are used, first-order differencing has been

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

applied to achieve stationarity, and the model considers the last four forecast errors to enhance its predictions. This specific ARIMA order has been determined as the best fit through a grid search based on the AIC criterion. Lower AIC values indicate a better balance between model fit and complexity, making ARIMA(0, 1, 4) a suitable choice for modeling our data.

AIC OF THE BEST ORDER

BEST AIC: 1538.35

ACTUAL VS PREDICTED DENGUE FEVER CASES

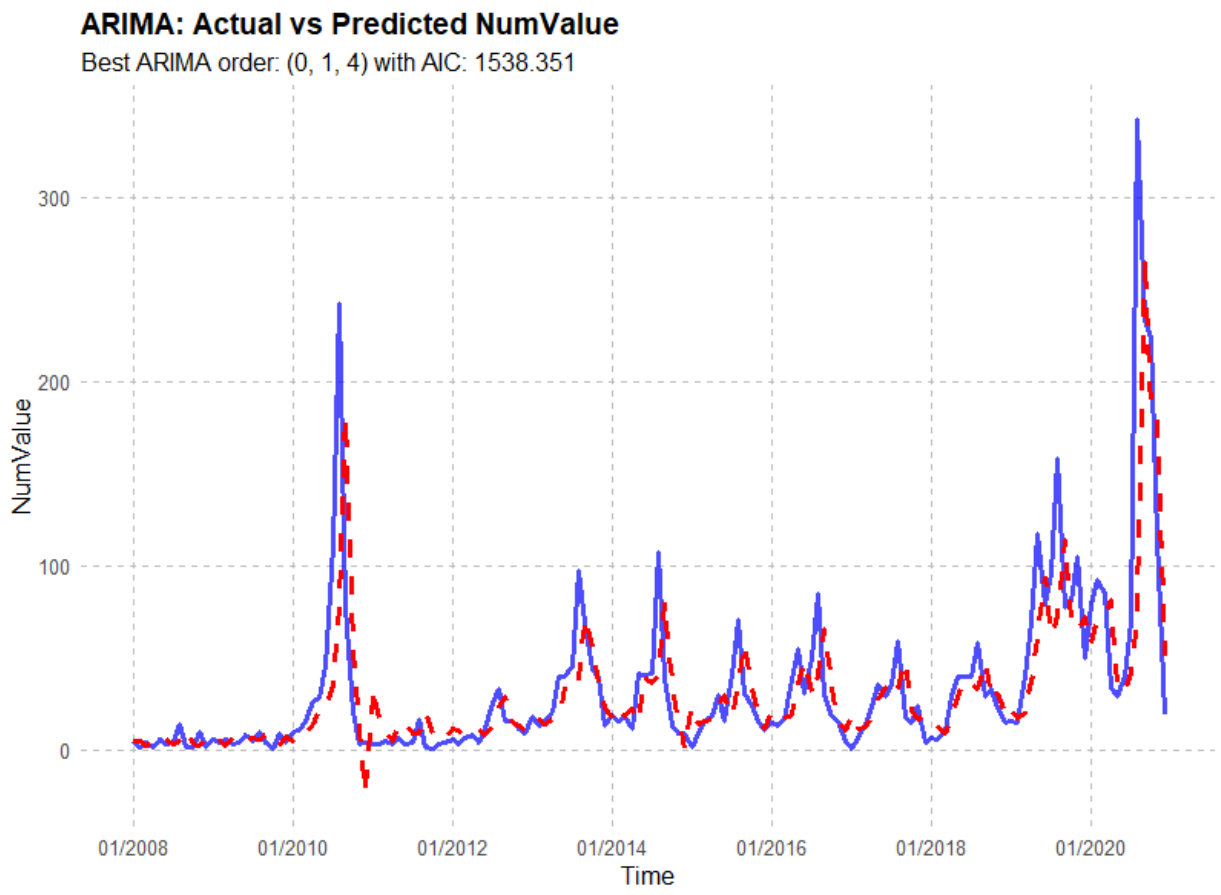

***Figure 52**. Actual vs predicted values for our ARIMA model*

1. **Blue line ("NumValue")**: This is the actual data from our time series. It represents the actual values of "NumValue" from January 2008 to December 2020.
2. **Red line ("ARIMA")**: This is the in-sample prediction of the ARIMA model fitted on our data. The "in-sample" prediction means that the ARIMA model is using all available data up to each point in time to forecast the next value.

The red line is very close to the blue line, this means that the ARIMA model is doing a good job at predicting the "NumValue" in our time series within the sample period. This is much better because for each prediction the model take into consideration all the previous real values results of the dengue fever, that's the reason its so close to reality

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

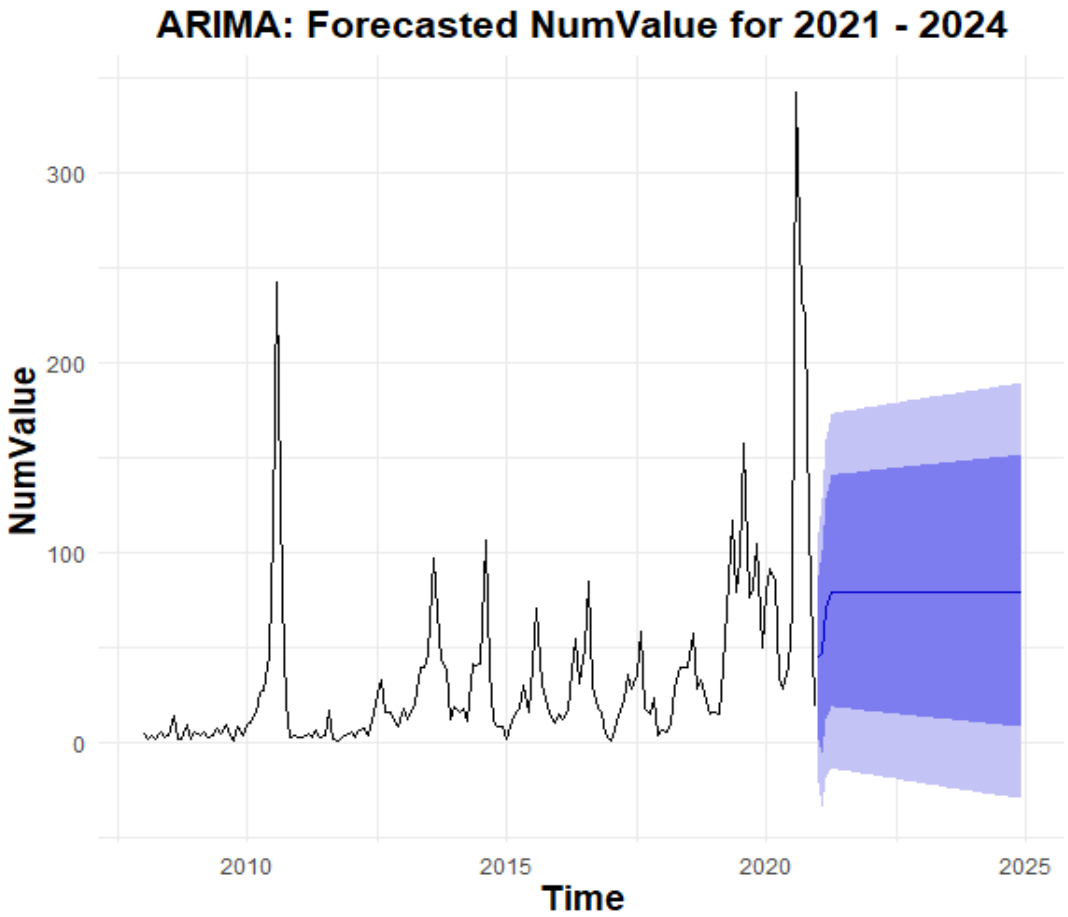

***Figure 53**. Forecast values (2021-2024) for our ARIMA model*

Out-of-sample forecast of our time series data from 2021 to 2024 using an ARIMA model.

Here is an explanation of the elements in the plot:

1. **X-axis (Time)**: This represents the time period from 2021 to 2024. Each point on this axis corresponds to a specific time.
2. **Y-axis (NumValue)**: This represents the value we are trying to forecast. Each point on this axis corresponds to the value at a specific time.

3. **Blue Line**: This line represents the forecasted values generated by the ARIMA model. The trend observed in the line is what the ARIMA model has predicted for future values based on the past data.

4. **Grey Shade**: This is the confidence interval for our forecast. It represents the range within which the true future values are likely to fall, with a certain level of confidence. It's usually set at a 95% confidence level by default, meaning we expect that future observations will fall within this range 95% of the time. The width of the shaded area can give us an idea of the uncertainty of the forecast: a wider area signifies more uncertainty.

In a basic ARIMA model, like the ARIMA(0, 1, 4) mentioned, the primary focus is on capturing the autocorrelations and the general trends in the data. It includes differencing to make the series stationary and moving average terms to account for the lagged errors. However, ARIMA models are not inherently designed to handle seasonality unless explicitly specified. As a result, they might struggle to capture recurring patterns that occur across specific periods, like the seasonal fluctuations in dengue fever cases that might occur during certain months or times of the year. This can lead to a smoother prediction line that doesn't account for the seasonal spikes.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

BEST SARIMA MODEL FIT

```
Series: ts_data
ARIMA(3,1,1)(0,0,2)[12]

Coefficients:
         ar1      ar2      ar3      ma1    sma1    sma2
      0.6770  -0.0442  -0.1791  -0.9272  0.4255  0.1576
s.e.  0.0947   0.0972   0.0905   0.0544  0.0993  0.0881

sigma^2 = 1018:  log likelihood = -755.32
AIC=1524.64   AICc=1525.4   BIC=1545.94

Training set error measures:
                   ME     RMSE      MAE       MPE     MAPE      MASE        ACF1
Training set 2.955007 31.18698 15.97656 -52.44569 105.7084 0.6458799 -0.02241628
```

Table 7. Best SARIMA model summary

BEST AIC: 1524.65

Better AIC than our arima counterpart.

ACUTAL VS PREDICTED VALUES FOR SARIMA MODEL

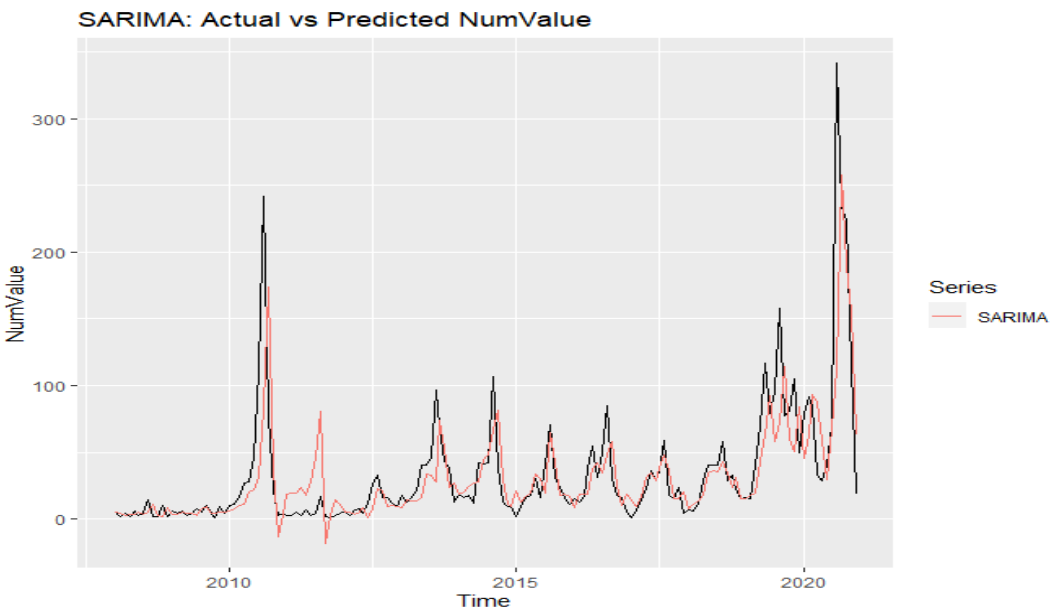

***Figure 54**. Actual vs Predicted values for our SARIMA model*

The reason that the SARIMA and actual data lines are very close could be that the SARIMA model is doing a good job of capturing the underlying patterns in the data. SARIMA models can capture both trend and seasonality, making them well-suited to many time series datasets. This includes patterns in the data that persist from one step to the next, which might be why the predictions are very close to the actual values.

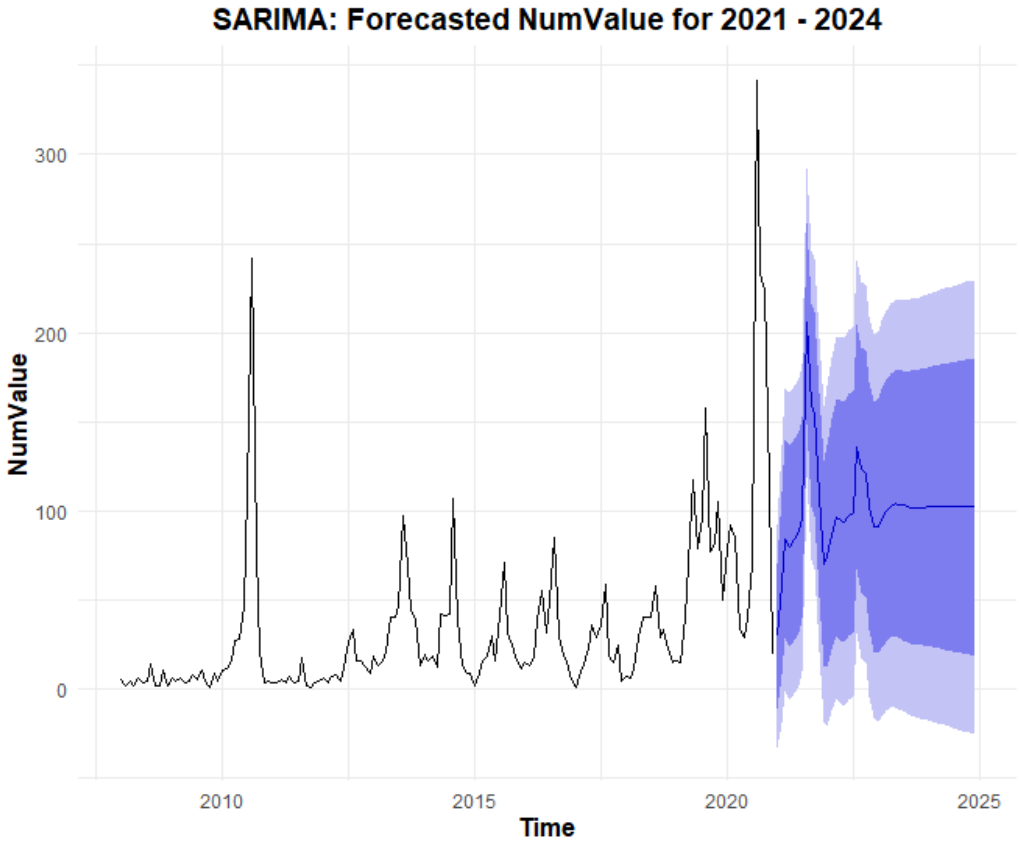

***Figure 55**. Forecast values (2021-2024) for our SARIMA model*

SARIMA (Seasonal ARIMA) models, are an extension of the basic ARIMA models and specifically consider the seasonal component in the data. They include additional seasonal terms (P, D, Q, and s) that allow them to capture patterns that repeat over fixed intervals (e.g., months, quarters, years). When there is clear seasonality in the data, SARIMA models are more suitable. In our case, the SARIMA model is likely able to capture and predict the seasonal spikes in dengue fever cases because it's explicitly accounting for the seasonal patterns.

In summary the SARIMA model is capturing the spikes and fluctuations in our dengue fever data better than the basic ARIMA model, it's to the seasonality component. SARIMA models are designed to handle these kinds of patterns, making them more effective when dealing with data that exhibits clear seasonal behavior.

Evaluation metrics results for our best time series model SARIMA (3,1,1) (0,0,2)

SARIMA LOG SCORE: 4.84

ARIMA LOG SCORE: 4.90

SARIMA RMSE: 31.19

SARIMA MAE: 15.98

The lower the log score, the better the model's predictive performance. The SARIMA model outperformed the ARIMA model in terms of log score, indicating that it provides more accurate predictions. In summary, the SARIMA (3,1,1) (0,0,2) model generally outperformed the basic ARIMA model across various evaluation metrics, indicating better predictive accuracy.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

4 MODELS SELECTION COMPARISON METHODS ANALYSIS

In the upcoming section of this thesis, we will delve deeper into the performance assessment of various predictive models that we have utilized in this study.

Some quick notes (recap)

Data Split:

- **Training Period:** 2008-2018
- **Test Period:** 2019-2020

Dataset Note:

- Only one region (France) included. So w_{ji} variable (geographical weight) was omitted

Model Forecasting:

- All models predicted a +1 horizon for consistent comparisons.

We will compare the performance of these models based on several key predictive accuracy metrics. The models being compared include 'HHH4', 'KCDE', 'AR1', 'Pois', 'Siraj', and 'Naïve'. These models were selected due to their potential to handle the challenges inherent in the prediction tasks associated with our data.

The key metrics we'll focus on are Mean Log Score (MLS), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). These metrics provide different perspectives on the accuracy and reliability of the predictions made by each model.

- **Mean Log Score (MLS)**
- **Root Mean Squared Error (RMSE)**
- **Mean Absolute Error (MAE)**

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

- **Mean Absolute Percentage Error (MAPE)**.

Additionally, we'll also take into account the Akaike Information Criterion (AIC) and Dawid-Sebastiani Score (DSS). These are statistical measures used to compare the goodness-of-fit of different statistical models.

- **AIC**
- **DSS**

In the upcoming analysis, we will calculate and compare these metrics for each model using the R programming language. Furthermore, we will visualize these metrics using a data ranking table for better comparative understanding, and summarize our findings in a coherent manner.

Our ultimate goal in this section is to identify the model that offers the best predictive performance for our specific dataset and to provide clear and concise insights into its performance compared to other models. Let's proceed with the analysis.

Table for the comparison

Results and ranking

Model	Mean_Log_Score	RMSE	MAE	MAPE	Rank_MLS	Rank_RMSE	Rank_MAE	Rank_MAPE
HHH4	4.92	17.41	10.27	0.65	1	1	1	1
KCDE	5.45	82.42	52.44	47.95	4	3	2	5
AR1	4.95	83.22	55.71	0.89	2.5	4	3	2.5
Pois	4.95	82.32	55.77	0.89	2.5	2	4	2.5
Siraj	5.48	102.75	78.28	1.65	5	6	6	4
Naive	7.88	89.23	62.25	62.12	6	5	5	6

Table 8. Final collective results

When analyzing the performance of predictive models, there are several key metrics to consider. The most important of these are Mean Log Score (MLS), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE).

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

The Mean Log Score (MLS) is a measure of the accuracy of a model where lower values indicate better accuracy. In this case, the HHH4 model has the best MLS, only slightly lower than AR1, Pois, Lin, Geom, and Unres models. Its performance is significantly better than that of the KCDE, Siraj, and Naïve model.

RMSE is a measure that squares the average difference between the predicted and actual values. This has the effect of giving a relatively high weight to large errors. This means the RMSE should be more useful when large errors are particularly undesirable. In our case, the HHH4 model has the lowest RMSE value, indicating it tends to produce fewer large errors in its predictions.

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction. It's the average over the test sample of the absolute differences between prediction and actual observation where all individual differences have equal weight. The HHH4 model has the lowest MAE, meaning that on average, its predictions are closer to the actual values compared to other models.

MAPE measures the size of the error in percentage terms. It is calculated as the average absolute percent error for each time period minus actual values divided by actual values. The HHH4 model has the lowest MAPE, suggesting that its predictions are, on average, closer to the actual values in terms of percentage difference.

AIC: Based on the provided AIC values, the model with the lowest AIC is the preferred model. In this case, the model "hhh4" has the lowest AIC value of 910. A lower AIC indicates a better fit of the model to the data. Therefore, based on AIC, the "hhh4" 1st model is considered the best model.

DSS: Finally, based on the DSS values, a higher DSS indicates a better model performance compared to the naive forecast. In this case, the "kcde" model has the highest DSS value of 10.422, which is higher than the DSS value of 7.118514 for the "hhh4" model. Therefore, based on DSS, the "kcde" model is considered the best model, but only one metric is not enough.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

Taken together, these metrics suggest that the our first HHH4 model is the best predictive model among those evaluated. Despite not having the absolute lowest MLS, its performance across all other key metrics—particularly those that emphasize the magnitude of errors—is superior. In other words, while HHH4 may not always make the most accurate predictions, it makes consistently close predictions and is less likely to produce a very large error. This stability and reliability make it a robust choice for prediction tasks for similar structured data as ours.

Now, let's dive deeper into the specifications of our best model. Understanding the formula and parameters behind HHH4 is crucial for comprehending why it performs so well. Let's explore the intricacies of this model and how each parameter contributes to its predictive capabilities.

HHH4 MODEL FORMULATION:

$$
\begin{aligned}
\lambda_t &= v_t + \phi_t Y_{t-1} \\
\log(v_t) &= \alpha^{(\nu)} + \gamma^{(\nu)} \sin(2\pi t/\omega) + \delta^{(\nu)} \cos(2\pi t/\omega) \\
\log(\phi_t) = \quad & \alpha^{(\phi)} + \sum_{k=1}^{2} \gamma_k^{(\phi)} \sin(2\pi k t/\omega) \\
& + \sum_{k=1}^{2} \delta_k^{(\phi)} \cos(2\pi k t/\omega),
\end{aligned}
$$

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

COEFFICIENTS SUMMARY TABLE

Coefficient	Estimate	Std_Error
ar.1	-0.32	0.11
ar.sin((2 * π * t/12))	0.36	0.12
ar.cos((2 * π * t/12))	-0.59	0.17
ar.sin((4 * π * t/12))	0.22	0.1
ar.cos((4 * π * t/12))	-0.1	0.16
ar.sin((6 * π * t/12))	-0.3	0.1
ar.cos((6 * π * t/12))	-0.45	0.14
end.1	-1.68	1.31
end.sin((2 * π * t/12))	1.39	0.74
end.cos((2 * π * t/12))	3.77	1.41
end.sin((4 * π * t/12))	-3.63	1.38
end.cos((4 * π * t/12))	-1.05	0.48
end.sin((6 * π * t/12))	2.79	1.15
end.cos((6 * π * t/12))	0.88	0.41
overdisp	0.16	0.03

Table 9. Coefficients estimates and std errors

Coefficients explanation from Table 9:

Autoregressive Terms:

ar.1 = -0.32: This is the autoregressive term for a lag of 1 month. An autoregressive coefficient describes the relationship between an observation and a number of lagged observations (past values).

Seasonal Terms:

These coefficients capture cyclical patterns in the data, especially important for diseases that have seasonal variations:

ar.sin(2 * π * t/12) = 0.36: This coefficient captures a sinusoidal component with a 12-month (yearly) period. A positive value indicates that there is an upward trend in dengue incidents during certain months of the year

ar.cos(2 * π * t/12) = -0.59: The cosine component, when negative, indicates a downward trend

*(2,4,6) they will complete two full cycles over 12 months. This can capture patterns in the data that repeat every 6 months

Together, these sine and cosine terms can capture a full cycle of upward and downward trends in dengue incidents across a year.

Endemic Terms (vt, end.1):

end.1 = -1.68: This term captures non-seasonal trends in the data.

Overdispersion:

overdisp = 0.16: Overdispersion is a term that captures excess variability in the data that's not explained by the model. In count data (like the number of disease incidents), A value of 0.16 indicates a certain level of overdispersion, which means the observed variability in dengue cases is slightly higher than what the model expects. This can be due to various unaccounted factors.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

5 FINAL THOUGHTS

The journey through this thesis has been one of discovery, analysis, and synthesis of an array of predictive models applied mainly to forecast Dengue fever cases in France from 2008 to 2020. It has provided a comprehensive overview of the complexities of infectious diseases, their devastating impacts over history, and our constant struggle to mitigate their effects.

Starting with the rich history of pandemics, from the Black Death to the ongoing COVID-19 pandemic, we have realized that disease outbreaks are complex events, intricately intertwined with socio-economic and ecological factors. Understanding these patterns allows us to better predict future disease trends and manage the impact of such events.

Predictive statistical modelling has been a primary focus of this work, and rightly so. The field has demonstrated its potential in providing reliable and informative forecasts that could support public health policies and interventions. Our exploration of various types of predictive models highlighted the depth and breadth of techniques available in the modern statistical toolkit.

Yet, the value of a model lies in its effectiveness, suitability, and accuracy when applied to real-world scenarios. The evaluation of predictive models for Dengue fever cases has been illuminating. It has showcased how each model, from HHH4 to KCDE, with their distinct methodological assumptions, strengths and weaknesses, brings unique insights into the data and the underlying disease processes.

The HHH4 model, with its impressive performance across a range of metrics, demonstrated the value of robust, consistent predictions. The KCDE model, with the highest DSS value, emphasized the importance of incorporating information about previous disease trends into the prediction process. The ARIMA model underlined the benefits of using time-series analysis for infectious disease forecasts.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

This work has emphasized that the choice of predictive model should be based on a holistic evaluation. It isn't simply about which model has the highest performance in a single metric, but which one provides the most valuable, reliable, and interpretable results for the specific task at hand.

The path of predictive modelling for infectious diseases is not without challenges, such as data quality and quantity, model assumptions, and overfitting. But with an understanding of these potential pitfalls, and the rigorous application of statistical methodologies, predictive modelling can play a vital role in public health, informing strategies to combat the spread of diseases like Dengue fever.

As we stand at the forefront of data-driven decision making, our mission should be to continually refine these predictive models, drive forward our understanding of disease processes, and use these tools to foster a healthier, safer world. This thesis is but a small steppingstone towards that goal.

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

6 BIBLIOGRAPHY

1. URL:https://www.healthdirect.gov.au/what-is-a-pandemic [section 1.1]
2. URL: publichealth.columbia.edu [section 1.1]
3. URL:https://www.medicalnewstoday.com/articles/pandemic-vs-endemic [section 1.1]
4. URL:https://intermountainhealthcare.org/blogs/topics/live-well/2020/04/whats-the-difference-between-a-pandemic-an-epidemic-endemic-and-an-outbreak/ [section 1.1]
5. URL:https://www.who.int/emergencies/diseases/novel-coronavirus-2019 [section 1.1]
6. URL:https://www.who.int/westernpacific/activities/preparing-for-pandemics [section 1.1]
7. Chan, M "WHO's Role and Responsibilities in Health Research. Bulletin of the World Health Organization", 2010 [section 1.1]
8. URL:https://www.cdc.gov/flu/pandemic-resources/national-strategy. [section 1.1]
9. URL:https://health.mo.gov/emergencies/panflu/panfluplanphases [section 1.1]
10. URL:https://www.ncbi.nlm.nih.gov/books/NBK143061/ [section 1.1]
11. Barry, J. M. "The Great Influenza: The Story of the Deadliest Pandemic in History, 2005 [section 1.2]
12. Honigsbaum, M. "The Pandemic Century: One Hundred Years of Panic, Hysteria, and Hubris. W. W. Norton & Company", 2019 [section 1.2]
13. McMillen, C. W "Pandemics: A Very Short Introduction. Oxford University Press", 2016 [section 1.2]
14. Osterholm, M. T., & Olshaker, M "Deadliest Enemy: Our War Against Killer Germs (Updated and Expanded Edition)", 2020 [section 1.2]
15. Khan, K., Arino, J., Hu, W., Raposo, P., Sears, J., Calderon, F.. & Hayman, J "Spread of a novel influenza A (H1N1) virus via global airline transportation. New England Journal of Medicine", [section 1.2]

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

16. URL:https://www.who.int/news-room/spotlight/history-of-vaccination [section 1.3]
17. URL:https://www.who.int/news-room/fact-sheets/detail/influenza-(seasonal), [section 1.3]
18. URL: https://www.cdc.gov/flu/about/viruses/types.htm, [section 1.3]
19. Chen, E. Y., & Zhu, Z. "Influenza A virus infection: epidemiology, pathogenesis, and host immune responses. Influenza and Other Respiratory Viruses", [section 1.3]
20. URL:"https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3291411/", [section 1.3]
21. Gubler, D. J., & Kuno, G "Dengue and Dengue Hemorrhagic Fever (2nd ed.)", 2013 [section 1.4]
22. Sotelo, E., & Fernandez, R. (Eds.) "Dengue Virus: Detection, Diagnosis, and Control", 2014 [section 1.4]
23. Halstead, S. B. "Dengue. The Lancet", 2017 [section 1.4]
24. World Health Organization "Dengue: Guidelines for Diagnosis, Treatment, Prevention and Control.", 2009 [section 1.4]
25. URL:https://climate-adapt.eea.europa.eu/en/metadata/case-studies/reducing-the-risk-of-local-dengue-transmission-in-france, [section 1.4]
26. URL:https://www.lshtm.ac.uk/newsevents/expert-opinion/dengue-france, [section 1.4]
27. URL: https://www.ncbi.nlm.nih.gov/pmc [section 1.4]
28. URL: https://hastie.su.domains/Papers/ESLII.pdf [section 2.2]
29. Fox, J. "Applied Regression Analysis and Generalized Linear Models", 2015 [section 2.2]
30. Scott, S., & Wild, C. J." Applied Logistic Regression. Cambridge University Press", 2015 [section 2.2]
31. Agresti, A "Foundations of Linear and Generalized Linear Models. Wiley", 2015 [section 2.2]
32. URL:https://mybiostats.files.wordpress.com/2015/03/time-series-analysis-and-its-applications-with-examples-in-r.pdf [section 2.2]
33. Shumway, R. H., & Stoffer, D. S "Time Series Analysis and Its Applications: With R Examples", 2017 [section 2.2]

34. URL:https://www.mastersindatascience.org/learning/machine-learning-algorithms/decision-tree/, [section 2.2]
35. Iau, G., & Scornet, E. "Random Forests for Big Data. Journal of Machine Learning Research", 2016 [section 2.2]
36. Kuhn, M., & Johnson, K "Applied Predictive Modeling. Springer", 2013 [section 2.2]
37. URL:Graph:https://vitalflux.com/classification-model-svm-classifier-python-example/, [section 2.2]
38. URL: https://www.youtube.com/watch?v=_PwhiWxHK8o "SVM Tutorial by Alex Smola (video lecture)', [section 2.2]
39. URL:https://machinelearningmastery.com/naive-bayes-for-machine-learning/, [section 2.2]
40. URL: https://www.coursera.org/specializations/deep-learning, [section 2.2]
41. Nielsen, M. "Neural Networks and Deep Learning. Determination Press", 2015 [section 2.2]
42. URL:https://towardsdatascience.com/a-gentle-introduction-to-gradient-boosting, "Towards Data Science - A Gentle Introduction to Gradient Boosting", [section 2.2]
43. URL:https://www.analyticsvidhya.com/complete-guide-parameter-tuning-gradient-boosting, [section 2.2]
44. URL: Source Link "Mathematical and Statistical Estimation Approaches in Epidemiology, "Compartmental Models (SIR, SEIRS, SIRS)", [section 2.3]
45. URL: Source Link "University of Michigan, "Compartmental Models in Epidemiology", [section 2.3]
46. URL: Source Link "The Santa Fe Institute, "Introduction to Agent-Based Modeling", [section 2.3]
47. URL: Source Link "Agent-Based Models", [section 2.3]
48. Steven F. Railsback and Volker Grimm "Agent-Based and Individual-Based Modeling: A Practical Introduction", [section 2.3]
49. URL: Source Link "BioMed Central, "Spatial modeling in epidemiology: a review of methods and applications", [section 2.3]

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

50. URL: Source Link “Towards Data Science, "An Introduction to Bayesian Inference in PyStan”, [section 2.3]
51. John K. Kruschke “Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan”, [section 2.3]
52. URL: Source Link “University of Cambridge, Notes on the Generalized Linear Model”, [section 2.3]
53. URL: Source Link “University of Bristol, Introduction to Generalized Linear Models”, [section 2.3]
54. URL: Source Link “Nature, "Machine learning for predicting and understanding infectious diseases”, [section 2.3]
55. URL: Source Link “NIH, "Machine Learning Approaches to Predict Infectious Disease Outbreaks”, [section 2.3]
56. Luo, L., “Forecasting dengue fever in China using ARIMA model. Public Health”, 2017 [section 2.4]
57. Hladish, T. J., “Forecasting dengue vaccine demand in disease-endemic and non-endemic countries.”, 2018 [section 2.4]
58. Ramadona, A. L., “Spatial modeling of dengue fever considering socioeconomic and environmental factors in Bandung City, Indonesia. Spatial and Spatio-temporal Epidemiology”, 2019 [section 2.4]
59. Gao, Z. “Predicting dengue fever outbreaks in China using a Bayesian hierarchical model”, 2018 [section 2.4]
60. Thompson, R. N., “Improved inference of time-varying reproduction numbers during infectious disease outbreaks”, 2019 [section 2.4]
61. Reiner, R. C. “A systematic review of mathematical models of mosquito-borne pathogen transmission: 1970-2010”, 2013 [section 2.4]
62. Apinonkul, B. “The use of generalised linear mixed models to predict incidence of dengue infection in Southern Thailand. PLoS Neglected Tropical Diseases”, 2016 [section 2.4]
63. Poovorawan, Y.,“Dynamic changes in viral load patterns in dengue patients exhibiting different viral clearance profiles.”, 2011 [section 2.4]
64. Lopes, L. Q., “Spatio-temporal analysis of dengue: Insights from agent-based modeling and simulation”, 2016 [section 2.4]
65. Steve Selvin, “Statistical Methods in Epidemiologic Research”, 2011 [section 2.5]

ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

66. Christl Donnelly and Simon Cauchemez, “Analysis of Infectious Disease Data”, [section 2.5]
67. Leonhard Held, Michael Ho¨hle and Mathias Hofmann, “A statistical framework for the analysis of multivariate”, Statistical Modelling 2005, pages [187-199], [section 3.3]
68. Evan L. Ray, Krzysztof Sakrejda, Stephen A. Lauer, Michael A. Johansson & Nicholas G. Reich, “ Infectious disease prediction with kernel conditional density estimation”, Wiley statistics in medicine 2016, [section 3.4]
69. Johannes Bracher & Leonhard Held, “Endemic-epidemic models with discrete-time serial interval distributions for infectious disease prediction”, International journal of forecasting 2020, [section 3.5]