UNIVERSITY OF BERGAMO

Department of Engineering and Applied Sciences 2021-2022, (XXXV CYCLE)

UNIVERSITÀ
DEGLI STUDI
DI BERGAMO

DEEP-LEARNING AND STATISTICAL METHODS IN SMARTPHONE-BASED EARTHQUAKE EARLY WARNING SYSTEMS

Thesis For The Degree Of Doctor Of Philosophy

Presented by

Massoda Tchoussi Frank Yannick

Under the guide of

Thesis advisor: Prof. Francesco Finazzi

Doctoral Higher Education School

Contents

List of Figures

List of Tables

Aknowledgement

Above all, with heartfelt humility, I express my profound gratitude to my Father and GOD, through my Lord Jesus Christ, who is eternally blessed. ABBA, your unwavering love, support, and divine guidance have been my steadfast anchor throughout this transformative journey. I am immeasurably grateful for your unwavering presence in my life, never forsaking me, and for the boundless strength and care you have bestowed upon me. All honor and glory belong to you, for you are the source of all blessings and the guiding light in my life.

To my beloved mother, Tonga Tchoussi Marie Florentine, my deepest appreciation extends to the divine for the blessing of your life and the tremendous influence you've had in molding me into the person I am now. This accomplishment is devoted to you as a testament of my unending love and profound gratitude for the abundance of love and wisdom you've imparted upon me.

My deepest and most sincere gratitude is extended to Blandine Kamtchou, affectionately known as Madame Ngatchou, who has been nothing short of a second mother to me. Her enduring love, selfless sacrifices, and unshakeable faith in my potential have served as a guiding light during the most challenging periods of my life. Her boundless support and steadfast dedication to my well-being are gifts for which I am eternally thankful.

While I could conclude my acknowledgments there, I am ever-mindful of the significance of honoring those who have played an integral role in my growth and success. To my beloved family from Italy and Cameroon, I extend my deepest appreciation for your invaluable advice, unwavering support, and unwavering encouragement. Your love and presence have been a constant source of comfort, motivation, and inspiration, propelling me forward on this journey of self-discovery.

I extend my sincere gratitude to my supervisor, Prof Francesco Finazzi, whose exceptional mentoring and unwavering guidance have been instrumental in shaping this work. Your expertise, patience, and encouragement have nurtured my academic growth, and I am truly grateful for your support on this challenging journey.

I would also like to thank my esteemed reviewers, Prof Giada Adelfio, and Dr.Robert Steed, for their fair and honest evaluation of this work. Your insightful feedback and constructive criticism have been invaluable in refining and enhancing this research, leading to its continued growth and development.

To all my friends and colleagues who have stood by my side throughout this arduous journey, thank you for your camaraderie, encouragement, and unwavering belief in my abilities. Your collective support has been an unwavering source of strength and inspiration, pushing me to strive for excellence.

I wholeheartedly acknowledge and appreciate the countless others who, though not mentioned here, have contributed to this work in various ways. Your support, prayers, and belief in me have been deeply meaningful, and I am truly grateful for the impact you have had on my life's journey.

To each and every one of you, from the depths of my heart, I extend my sincerest gratitude. This work would not have been possible without your love, encouragement, and unwavering belief in me. I am truly blessed and profoundly honored to have each of you in my life.

This is also for you, my future wife, though you are not yet here. I dedicate this achievement to our future and the boundless love that will bind us together. May our lives intertwine in a symphony of dreams, laughter, and love as we build a beautiful life together. As I embark on this new chapter, I carry with me the hope of meeting you and creating a life filled with joy, fulfillment, and unending love. Until that blessed day comes, know that you are always in my thoughts, and I eagerly await the moment we shall be united in love and purpose. With fervent anticipation and love, this is for you, my dear future wife.

May the abundant blessings and grace of God be upon each one of you, illuminating your paths and enriching your lives with boundless love, unbridled joy, and abundant prosperity, all in the mighty name of Jesus.

Chapter 1

Introduction

For countless millennia, the Earth has experienced seismic upheavals known as earthquakes, and their occurrence will continue into the future. While some earthquakes transpire in remote, sparsely populated regions with minimal consequences, others strike in close proximity to densely inhabited urban areas, subjecting both the residents and the vital infrastructure to violent shaking. Although we cannot prevent earthquakes from transpiring, it is within our capabilities to mitigate the impact of their intense shaking, thereby reducing the number of fatalities, injuries, and property damage they cause.

An earthquake occurs when the ground undergoes a shaking motion due to the propagation of seismic waves within the Earth's rocks. These waves are generated by the release of tremendous energy resulting from the fracturing of the Earth's crust. Depending on the magnitude of the energy released, seismic waves can induce subtle tremors that are barely perceptible, causing slight vibrations in dishes or rustling leaves on trees. Alternatively, earthquakes can manifest as colossal jolts that violently propel everything in their path, leading to the rupture and collapse of buildings, the rupture of dams, and a plethora of catastrophic events that profoundly impact society.

The consequences of earthquakes can be devastating, resulting in significant human suffering and widespread destruction. Beyond the immediate physical harm caused by collapsing structures and flying debris, earthquakes can trigger secondary hazards such as landslides, tsunamis, and fires, further compounding the destruction. The socio-economic repercussions can be immense, as critical infrastructure, including transportation networks, communication systems, and utilities, may be severely impaired or rendered inoperable. The disruption to vital services, such as healthcare, emergency response, and supply chains, can exacerbate the impacts and hamper recovery efforts.

The recognition of the destructive potential of earthquakes has driven researchers and engineers to develop strategies aimed at mitigating their consequences and increasing society's resilience. Anti-seismic engineering practices have emerged as a crucial aspect of earthquake mitigation, focusing on the construction of buildings and infrastructure designed to withstand seismic forces. These structures incorporate innovative designs,

materials, and construction techniques that can dissipate and absorb the energy generated during seismic events, minimizing the risk of collapse and reducing overall damage. Simultaneously, urban planning and zoning regulations now integrate seismic hazard assessments to ensure that new developments are located away from known fault lines or on stable ground, thereby minimizing exposure to seismic risks. Such proactive measures play a significant role in creating safer built environments that can better withstand the impact of earthquakes [1].

Additionally, earthquake early warning systems (EEWSs) have emerged as essential tools for mitigating the effects of earthquakes. EEWSs leverage advanced technologies, including seismometers and communication networks, to detect and analyze the initial seismic waves generated during an earthquake. By rapidly assessing the characteristics of these initial waves, EEWSs can estimate the potential magnitude and intensity of the forthcoming shaking, providing valuable seconds to minutes of warning before the arrival of more destructive seismic waves. This advance notice allows individuals, organizations, and emergency responders to initiate protective actions, such as taking cover, halting critical operations, activating emergency protocols, and evacuating vulnerable areas. Ultimately, these systems save lives, reduce injuries, and minimize the socio-economic impacts of earthquakes [2].

Traditional EEWSs encompass various detection and alerting mechanisms, including front-detection systems and single-station systems. Front-detection systems rely on the rapid transmission of seismic data from stations located closest to the earthquake's epicenter to centralized processing centers. These centers analyze the data and generate alerts that propagate outward to regions at a greater distance from the epicenter, providing a timely warning to potentially affected areas [3, 4, 5, 6]. Single-station systems, on the other hand, utilize seismic data from a single station or sensor to estimate the earthquake's characteristics and issue corresponding alerts [7, 8, 9, 10, 11]. While these conventional approaches have proven effective, their widespread adoption has been limited by the high costs associated with implementation and maintenance, hindering their global deployment.

In response to these limitations, the Earthquake Network (EQN) citizen science initiative [12, 13] has emerged as a pioneering solution, deploying the first crowdsourced EEWS based on smartphone networks. The EQN initiative encourages active participation from citizens by installing the EQN smartphone application on their Android, iOS, or Huawei devices. Leveraging the built-in accelerometers of these smartphones, the EQN application records vibrations potentially induced by earthquakes. The captured data is then transmitted to a central server, where real-time analysis of the information from the smartphone network takes place. Subsequently, alerts are issued when earthquakes are detected, providing users with timely notifications. Since its establishment in 2012, the EQN initiative has gained the active participation of over 12 million citizens worldwide and has successfully detected more than 6,000 earthquakes globally. Notably, other technology companies, such as Google, have also recently introduced similar smartphone-based services, expanding the reach of crowdsourced EEWSs [14].

However, smartphone-based EEWSs face a significant challenge in the form of false alerts. False alerts occur when the smartphone network mistakenly detects events unrelated to genuine earthquakes. These false triggers can arise from various sources, including natural phenomena like thunderstorms or human activities such as explosions or exuberant celebrations during sporting events. The occurrence of false alerts within EEWSs presents a particular predicament since once an alert is triggered, it cannot be retracted. This situation can lead to unnecessary stress for citizens and potentially erode trust in the reliability of the EEWS over time.

To address these challenges, ongoing research and development efforts are focused on refining and enhancing the capabilities of smartphone-based EEWSs. Two primary approaches are being explored: statistical modeling and deep learning. The first approach utilizes a statistical model to regulate the false alarm probability, manage the probability of missed detection, classify detections as true or false earthquakes, and estimate the epicenter and depth of an earthquake if it is classified as genuine. This methodology relies on a statistical parametric model, statistical hypothesis testing, and Monte Carlo simulation. By incorporating the well-established spatio-temporal dynamics of seismic waves into the statistical model, the classification of EQN detections and the estimation of earthquake parameters are improved. Given the real-time requirements of the application, it is crucial for the classification and earthquake parameter estimation processes to be completed within one to two seconds of computation time.

In contrast, the second approach leverages deep learning techniques, which have gained significant prominence in various domains. Specifically, convolutional neural networks (CNNs), commonly employed in computer vision applications, are adapted to classify EQN detections within the EEWS. By transforming the smartphone data collected by the EQN network into digital images that capture the spatial structure of the network and convey information regarding the propagation of seismic waves, CNNs can effectively classify detections as true or false earthquakes. Various CNN architectures are explored to identify the most effective approach in terms of the aforementioned relevant probabilities. Real-time processing and response times remain critical considerations for this approach, ensuring that alerts are issued promptly to users [15, 16, 17].

By embracing these two distinct approaches, this thesis aims to contribute to the ongoing pursuit of reducing the occurrence of false alerts within smartphone-based EEWSs, ensuring greater accuracy and reliability in earthquake detection and classification. Through the fusion of statistical modeling and deep learning techniques, researchers endeavor to enhance the overall performance and effectiveness of EEWSs, ultimately bolstering public trust and confidence in these crucial systems. These advancements hold the potential to save lives, protect infrastructure, and minimize the socio-economic impacts of earthquakes on communities worldwide.

Chapter 2

Overview of Earth internal structure and earthquakes

This chapter provides an overview of Earth's internal structure and its relationship to earthquakes. It explores the geological processes that give rise to earthquakes, the different types of seismic waves, and the factors influencing the severity of ground shaking. Additionally, it examines the role of tectonic plate boundaries, fault lines, and seismic activity in earthquake occurrence. Furthermore, the chapter discusses the principles and functioning of earthquake early warning systems, highlighting their significance in mitigating the impacts of earthquakes. By understanding the fundamental aspects of Earth's structure and seismic activity, we can better comprehend the causes and effects of earthquakes, enabling us to develop more effective strategies for earthquake mitigation and disaster preparedness.

2.1 Internal structure of the earth

Understanding earthquakes begins with gaining insight into seismology, the field that studies the internal structure of the Earth and its relation to seismic activity. This section provides a basic overview of Earth's internal structure, the causes of earthquakes, and the terminology associated with seismic events.

The Earth has a mass of approximately 5.9×10^{29} kilograms and is roughly spherical, with an equatorial radius of 6370 kilometers and a polar diameter of 6350 kilometers. The slight difference of about 20 kilometers between the equatorial and polar radii is due to the Earth's higher rotational velocities at the equator.

The Earth's interior can be divided into three main layers that overlap: the crust, the mantle, and the core. Each of these layers possesses distinct physical and chemical properties. The crust is the outermost layer and varies in thickness, ranging from approximately 25 to 40 kilometers beneath continents to around 5 kilometers beneath the oceans. In certain young mountain ranges, the crust can reach thicknesses of 60 to 70 kilometers. The crust consists of different layers, with a basaltic layer underlying a granitic layer, depicting the complex internal structure of the crust in continental regions.

Beneath the crust lies the mantle, which is further divided into the upper mantle and the lower mantle. The upper mantle extends to a depth of approximately 650 kilometers and is followed by the lower mantle, which reaches depths of roughly 2850 kilometers. The lower mantle is predominantly homogeneous in terms of chemical composition, except near its bottom boundary. Notably, no earthquakes are known to occur in the lower mantle. The mantle exhibits an average temperature of about 4000 degrees Fahrenheit (2200 degrees Celsius), with cooler temperatures near the crust and higher temperatures at greater depths. The materials in the mantle are in a vicious and semi-molten state. When subjected to rapid stress, such as that caused by seismic waves, the mantle behaves as a solid. However, under long-term stress, it can flow slowly, resembling a fluid-like behavior.

The core of the Earth is composed of two distinct regions: the outer core and the inner core. The outer core is a liquid layer approximately 2260 kilometers thick. Being in a liquid state, it does not transmit S-waves, and at the core-mantle boundary, known as the Gutenberg discontinuity, the velocity of S-waves drops to zero. The outer core consists mainly of molten iron. In contrast, the inner core, also known as the solid core, is a solid region consisting of compacted nickel-iron material. It is believed to have a relatively constant temperature exceeding 5000 degrees Fahrenheit (2800 degrees Celsius).

Understanding the internal structure of the Earth is crucial for comprehending the mechanisms behind earthquakes. The interactions and dynamics within these layers give rise to seismic activity, as well as the propagation and behavior of seismic waves. By studying the Earth's structure and seismic processes, scientists can gain valuable insights into the causes and effects of earthquakes, enabling the development of more effective strategies for earthquake prediction, mitigation, and disaster preparedness.

2.1.1 Crust

The Earth's crust, the outermost layer on which we reside, accounts for less than 1% of the Earth's volume [18, 19]. Due to its exposure to the oceans and atmosphere, the crust tends to be cooler than the underlying layers. The division of the crust into shifting tectonic plates allows for the escape of heat from the Earth's interior. Chemically, the crust shares properties with andesite, while its mineral composition is primarily dominated by feldspar, quartz, and pyroxene, making up around 41%, 12%, and 11%, respectively, of the minerals found in the continental crust [20, 21]. The boundary between the crust and the mantle is typically defined by the Mohorovičić discontinuity, characterized by a variation in seismic velocity [22, 23, 24]. The temperature of the crust increases with depth, reaching levels between 200 and 400 °C near the interface with the underlying mantle [20, 21]. Locally, the temperature gradient can be as high as 30°C per kilometer in the upper crust, while it tends to decrease in the deeper sections.

Minerals, natural components produced by the Earth, exhibit distinct structures and chemical compositions. With over 4600 known minerals, they form the building blocks of the Earth's crust. Some minerals, such as gold and diamonds, are rare and valuable [25, 26], while others like quartz are more common. Elements,

which constitute approximately 100 fundamental units of matter, combine to form minerals either in their pure state or through chemical bonding with other compounds [20, 21].

The Mohorovičić discontinuity, named after the seismologist who discovered it in 1909, marks the lower boundary of the Earth's crust, typically found around 30 kilometers below the surface on average. However, the thickness of the crust can vary widely, ranging from approximately 5 to 65 kilometers. Although the crust's thickness is not random, it correlates strongly with the Earth's surface topography. The continental and oceanic crusts differ significantly in their thickness and composition. The isostasy principle governs the crust's thickness, describing the physical, chemical, and mechanical differences that enable the crust to "float" on the underlying, more ductile mantle [27, 28]. The upper part of the continental crust primarily consists of igneous rocks, such as granite or quartz diorite, while the lower part includes basalt, a denser igneous rock commonly erupted from volcanoes. In contrast, the oceanic crust is predominantly composed of basalt [29]. According to Airy's isostatic equilibrium principle, the upper mantle material beneath the solid crust is in approximate equilibrium, with individual crustal sections sinking into the underlying mantle until the buoyant force matches their weight [30]. Although the Earth's crust is not always in perfect isostatic balance, most movements aim to bring the system closer to equilibrium.

Geologists use the term SIMA to refer to the rocks of the oceanic crust, which primarily consist of silicates and magnesium [29]. Basalt, a specific type of rock, is commonly found in the oceanic crust, with a density of approximately 3 grams per cubic centimeter. At mid-ocean ridges, where tectonic plates diverge, oceanic crust continuously forms. Young oceanic crust is created as lava rises and cools at these rifts on the Earth's surface. The age and density of the oceanic crust increase with distance from the mid-ocean ridges [31, 32]. Conversely, oceanic crust is destroyed in subduction zones, mirroring the creation process at mid-ocean ridges [33, 34]. During subduction, the denser oceanic lithosphere, including the crust, sinks beneath the less dense continental lithosphere. As the lithosphere subducts, it becomes more flexible and ductile, eventually participating in mantle convection. This process allows the recycling of mantle minerals, which can later emerge as lava, forming crust at mid-ocean ridges and volcanoes. Oceanic crust is significantly younger than continental crust, primarily due to subduction [35, 36]. The Ionian Sea, a part of the eastern Mediterranean basin, is known for hosting the oldest oceanic crust. Geologists study oceanic crust by exploring the ocean floor, employing submersibles, and examining ophiolites—portions of oceanic crust uplifted above sea level through tectonic processes. Ophiolites, sometimes visible as dikes in continental crust, provide accessible samples for scientists compared to the oceanic crust located at the ocean's depths.

Granite, in its various forms, constitutes the majority of the continental crust. Geologists often refer to the rocks that make up the continental crust as "sial." The prevalent minerals in the continental crust are silicates and aluminum, collectively referred to as sial [37, 38, 39]. Sial is less dense than sima, with a density of approximately 2.7 grams per cubic centimeter. However, it can be substantially thicker, reaching up to 70 kilometers in certain regions [29]. Similar to oceanic crust, the continental crust is formed through plate tectonics. Orogenesis, or mountain-building, uplifts the continental crust along convergent plate boundaries

where tectonic plates collide. The thickest portions of the continental crust are found in the world's highest mountain ranges, such as the Himalayas and the Andes, as a result of this process. These regions represent only a fraction of the overall continental crust, which exhibits uneven distribution. Cratons, the oldest and most stable sections of the continental lithosphere, are commonly located deep within continents. Cratons can be classified into two categories: shields, where ancient basement rocks are exposed at the surface, and platforms, where sediment has accumulated on top of the basement rock. Both shields and platforms provide valuable insights into the early history and formation of the Earth. Unlike oceanic crust, certain portions of the continental crust have persisted since the early stages of the Earth, as the subduction process rarely destroys and recycles them.

2.1.2 Mantle

The Earth's mantle lies beneath the crust and extends from a depth of 30 km. It constitutes approximately 82% of the Earth's total mass. The mantle consists of two primary regions: the upper mantle and the lower mantle. The upper mantle is divided into two distinct layers: the solid lithosphere, which interacts with the Earth's crust, and the asthenosphere, a viscous plastic layer that exhibits properties between those of a solid and a liquid. The asthenosphere is subject to convective currents, which play a crucial role in driving the dynamics of tectonic plates. The lower mantle is located between the upper mantle and the core. Despite the extremely high temperatures at this depth, the upper mantle remains solid due to the intense pressure preventing rock fusion. Additionally, there are two additional regions of interest: the transition zone, which lies between the upper and lower mantles, and the D" (D double-prime) layer, where the mantle meets the outer core.

Two significant discontinuities mark the boundaries of the Earth's mantle: the Mohorovičić discontinuity (Moho) separating the mantle from the crust, and the Gutenberg-Wiechert discontinuity [40] marking the boundary with the core. These boundaries are characterized by significant changes in seismic wave velocities and, consequently, density. The variations in density can arise from either compositional changes or abrupt shifts in the physical state of the materials. However, near the crust-mantle and mantle-core boundaries, drastic changes in the physical state are unlikely, suggesting a rapid change in material composition.

The Repetti surface, a second-order discontinuity, demarcates the division between the upper and lower mantles. By studying the velocity-depth function of seismic waves, it is observed that the velocity curves in the lower mantle rise continuously and smoothly, while the velocity alterations in the upper mantle are more pronounced and irregular [30, 41]. This observation implies that as depth increases, our understanding of the "fine structure" of velocity changes becomes less certain. Furthermore, if the observed velocity-depth function accurately represents the mantle's behavior, it supports the theory that there is no significant material differentiation below a depth of approximately 800-1000 km. This suggests that the chemical composition of the mantle does not differ significantly below this depth. Instead, the abrupt boundary observed at the Gutenberg-Wiechert discontinuity is attributed to a change in the physical state of the materials. Extensive studies on the structure and physical conditions of the mantle [42, 43] indicate that the mantle is homogenous

from 900 km to 2900 km or to the edge of the Earth's core. Birch discovered that this zone corresponds to a transition from ultrabasic silicate material beneath the crust to materials modified by high pressure in the mantle. This transition is marked by a sharp increase in the adiabatic incompressibility-to-density ratio between 300 and 800 kilometers in depth.

The predominant rocks in the Earth's mantle are silicates, a large class of substances characterized by a common silicon and oxygen structure. Olivine, garnet, and pyroxene are examples of common silicates found in the mantle [44, 45]. Magnesium oxide is another significant rock type present in the mantle. Other constituents of the mantle include calcium, salt, potassium, aluminum, and iron. The temperature of the mantle increases from approximately 1000°C (1832°F) near its border with the crust to 3700°C (6692°F) near its boundary with the core. Heat and pressure in the mantle generally increase with depth. This increase in temperature is quantified by the geothermal gradient, which is typically around 1°F every 70 feet of depth or 25°C per kilometer. The viscosity of the mantle also exhibits considerable variation. While the majority of the mantle is solid rock, it becomes less viscous near tectonic plate boundaries and mantle plumes. In these regions, the mantle is characterized by a decreased viscosity. Some rocks in the mantle possess plasticity, allowing them to undergo gradual deformation over millions of years under extreme pressure and depth. The distribution of heat and material within the mantle significantly influences the Earth's topography. Mantle activity, driven by convection currents, is responsible for plate tectonics, earthquakes, seafloor spreading, volcanoes, and orogeny (mountain-building).

Direct investigation of the mantle has not been possible, even with advanced drilling technology. The deepest drillings have not reached beyond the crust. However, geologists employ xenolith analysis to gain insights into the mantle. Xenoliths are rock inclusions trapped within other rocks. Diamonds are among the most informative xenoliths concerning the mantle. Diamonds form in the upper mantle, at depths of at least 150 kilometers (93 miles), under specific conditions. Above certain depths and pressures, carbon crystallizes as graphite rather than diamond. Intense volcanic eruptions associated with rock formations known as kimberlites and lamprolites bring diamonds to the surface. However, geologists are more interested in the xenoliths contained within the diamonds themselves. These inclusions provide valuable information about the mantle's composition. Through diamond intrusions, scientists have been able to gain insight into the lower mantle to depths of up to 700 kilometers below the Earth's surface. Xenolith studies suggest that rocks in the deep mantle may consist of subducted seabed slabs that are approximately three billion years old. Diamond intrusions have also revealed the presence of water, ocean sediments, and carbon within the mantle. Modern geologists and seismologists can create mantle maps using advanced technologies. These maps display seismic velocities and provide insights into the deep structures of the Earth's crust. By analyzing the body waves of thousands of earthquakes with magnitudes of at least 5.5, mantle mapping can pinpoint tectonic plate movements and the locations of ancient subducted material. Many geologists believe that mantle maps hold the potential to provide evidence of mantle plume structures and their motions. Mantle plumes, upwellings of hot rock from the mantle, are believed to give rise to volcanic "hot spots" that are unrelated to plate tectonics. As a mantle plume approaches the upper mantle, it transforms into a diapir. This molten material heats the

asthenosphere and lithosphere, leading to volcanic eruptions. While tectonic activity at plate boundaries is primarily responsible for heat loss, volcanic eruptions have only a minimal impact.

Studies of volcanic lava, diamond pipes, meteorites, and mineral and rock analyses have provided scientists with insights into the mantle's composition. Research suggests that the first few hundred miles of the mantle may consist mainly of eclogite and peridotite, which are predominantly complex calcium, sodium, and aluminum silicate minerals with small amounts of iron and magnesium silicate minerals. Denser variants of these minerals are believed to exist in the transition zone. At great depths, only iron, magnesium, and silicon oxides are expected to be present in the mantle due to the extreme pressure conditions. However, the lateral and vertical composition variations in the mantle are still not fully understood.

Upper Mantle

The upper mantle extends up to approximately 410 kilometers below the Earth's surface. Although primarily solid, the more ductile regions of the upper mantle contribute to tectonic activity. The upper mantle can be divided into two sections: the lithosphere and the asthenosphere.

The lithosphere, reaching a depth of around 100 kilometers, constitutes the solid outer layer of the Earth. It comprises the brittle uppermost part of the mantle and the crust. The lithosphere is characterized as both the coldest and the most rigid layer of the Earth. The most well-known aspect of the lithosphere is its involvement in tectonic activity, which arises from the interaction between enormous lithospheric plates known as tectonic plates [46, 47]. The lithosphere consists of several major tectonic plates, including the North American, Caribbean, South American, Scotia, Antarctic, Eurasian, Arabian, African, Indian, Philippine, Australian, Pacific, Juan de Fuca, Cocos, and Nazca plates. Beneath the Moho, the mantle is predominantly composed of peridotite, a rock primarily composed of the minerals olivine and pyroxene [48, 49]. The lithospheric mantle is relatively cooler and stiffer than the layers below.

Below the lithospheric mantle lies the asthenosphere, situated between 100 and 410 kilometers beneath the Earth's surface. The asthenosphere is characterized by its extreme heat and pressure, which cause rocks to undergo partial melting, resulting in a semi-molten state. Compared to the lithosphere or lower mantle, the asthenosphere exhibits greater ductility. Ductility refers to the ability of a solid material to stretch or deform under stress. The lithosphere-asthenosphere boundary (LAB), where geologists and rheologists (scientists who study the flow of matter) distinguish between the two layers of the upper mantle, is typically more viscous than the lithosphere. Plate tectonics, which encompasses continental drift, earthquakes, mountain formation, and volcanism, is driven by the slow movement of lithospheric plates "floating" on the asthenosphere. Although the asthenosphere can be considered as molten material, it is not a liquid, and the movement of tectonic plates is not a result of floating. Only the plate boundaries and hotspots are prone to instability [50].

Transition Zone

Between depths of 410 and 660 kilometers lies the transition zone of the mantle. Rocks within this zone undergo significant changes, not melting or decaying but undergoing transformative modifications in their crystalline structure. The rocks become exceedingly dense, and this zone acts as a barrier to substantial material exchanges between the upper and lower mantles. Some geologists propose that the higher density of rocks in the transition zone inhibits further sinking of subducted lithospheric slabs. These large fragments of tectonic plates remain stationary within the transition zone for millions of years before eventually mixing with other mantle rocks, returning to the upper mantle as part of the asthenosphere, erupting as lava, becoming part of the lithosphere, or forming new oceanic crust during seafloor spreading. However, other rheologists and geologists argue that subducted slabs may penetrate the transition zone and reach the lower mantle. Evidence suggests the transition zone exhibits permeability and material exchange between the upper and lower mantles [51, 52].

One of the most significant features of the transition zone is the presence of water. The amount of water stored in the mantle's transition zone is estimated to be equivalent to the water in the Earth's oceans. However, the water in the transition zone does not exist in the familiar liquid form. Instead, it is present as hydroxide. Hydroxide is an ion composed of hydrogen and oxygen with a negative charge. Rocks such as ringwoodite and wadsleyite, with their crystalline structures, trap hydroxide ions in the transition zone. These minerals are formed from olivine under extremely high pressures and temperatures. Ringwoodite and wadsleyite undergo crystallographic changes with increasing temperature and pressure at greater depths, releasing hydroxide as a "melt." The melt particles ascend towards minerals capable of storing water. This provides a mechanism for a continuous supply of water to the transition zone. Geologists believe that water from the Earth's surface is transported to the mantle through the process of subduction. Subduction occurs when a denser tectonic plate sinks or melts beneath a less dense plate. Most subduction events involve an oceanic plate subducting beneath a continental plate. Along with rocks and minerals from the lithosphere, substantial amounts of water and carbon are also transported to the mantle. Through mantle convection, volcanic eruptions, and seafloor spreading, water and hydroxyl ions are recycled back to the upper mantle, crust, and atmosphere.

Lower Mantle

The lower mantle extends from a depth of 660 kilometers to approximately 2700 kilometers beneath the Earth's surface. It is hotter and denser compared to the upper mantle and transition zone. Unlike the upper mantle, the lower mantle is less ductile due to the intense pressure, despite the high temperatures. While heat often leads to rock melting, the lower mantle remains rigid as a result of the extreme pressure exerted. Geologists hold differing views on the structure of the lower mantle. Some propose that subducted lithospheric slabs settle in this region, while others suggest the lower mantle remains entirely immobile, without convection [53].

D" (D-Double Prime)

D", or "D double-prime," is a shallow region located beneath the lower mantle. In some areas, D" forms an almost imperceptibly thin boundary with the outer core, while in others, it exhibits significant concentrations of iron and silicates. Geologists and seismologists have also observed extensive melting in certain regions. The movement of elements within D" is influenced by interactions between the lower mantle and outer core. The iron present in the outer core contributes to the formation of igneous intrusions known as diapirs, which create dome-shaped geologic structures. Similar to a lava lamp, the iron diapir releases heat and may generate a large bulging pulse of material or energy. As this energy propagates upward, it can lead to the formation of mantle plumes and eventually cause them to erupt into the lower mantle and transition zone. The core-mantle boundary (CMB), located at the base of the mantle, marks the end of the mantle and the beginning of the liquid outer core. This transition is known as the Gutenberg discontinuity [54].

2.1.3 Core

The Earth's core is a highly dense and scorching hot region located beneath the crust and mantle. It consists of two distinct layers: the outer core and the inner core. The core is situated approximately 2900 kilometers below the Earth's surface and has a radius of about 3485 kilometers. The formation of the core dates back to the early stages of Earth's history, around 4.5 billion years ago, during a period known as the iron catastrophe. At that time, the Earth was a hot, homogeneous ball of rock. As the planet's temperature reached the melting point of iron, which is about 1,538°C (2800°F), the molten material differentiated, and denser elements, such as nickel and iron, sank towards the center, forming the core. This process is known as planetary differentiation and played a crucial role in shaping the Earth's structure [55].

The core is primarily composed of iron, with a significant amount of nickel. These elements form alloys, such as nickel-iron (NiFe), which are highly prevalent in the core [56, 57]. The core also contains other elements, including sulfur and smaller amounts of precious metals like platinum and gold, as well as siderophile elements that dissolve in iron. The presence of sulfur in the core helps explain why the core is not significantly denser, as it is a relatively light element compared to iron and nickel. The exact temperature of the core is challenging to determine precisely due to various factors such as pressure, composition, and the Earth's rotation. However, estimates suggest that temperatures in the core range from 4400°C (7952°F) to 6000°C (10800°F) [58].

Outer Core

The outer core is a liquid layer surrounding the inner core. It is approximately 2,200 kilometers thick and composed mainly of liquid nickel and iron. The temperatures in the outer core range between 4500°C and 5500°C (8132°F and 9932°F), making it extremely hot and hostile to human existence. The outer core's high temperatures keep the nickel-iron alloy in a molten state, and its low viscosity allows for vigorous convection currents. These convective movements of liquid metal generate Earth's magnetic field through a process known as the dynamo effect. The churning motion of the outer core's liquid metal is responsible for the creation and maintenance of the planet's magnetic field, which plays a crucial role in protecting the Earth

from harmful solar radiation and facilitating navigation using compasses [58].

Inner Core

The inner core lies at the center of the Earth's core and is a solid, spherical region with a diameter of approximately 1220 kilometers. Despite the incredibly high temperatures, the inner core remains solid due to the immense pressure exerted on it by the surrounding layers. The pressure prevents the iron atoms from transitioning into a liquid state. The inner core's temperature reaches around 5200°C (9392°F), and the pressure is estimated to be nearly 3.6 million atmospheres. The extreme conditions in the inner core give rise to unique properties of the iron crystals, which are structured in a hexagonally close-packed (hcp) configuration. The alignment of these iron crystals plays a role in Earth's magnetic field, as seismic waves travel faster in a north-south direction compared to an east-west direction due to the crystal orientation. Over time, the inner core slowly expands by about a millimeter per year as the entire Earth cools. This expansion occurs as portions of the liquid outer core freeze and crystallize, causing the inner core to grow slightly larger. This process is influenced by mantle activity, with areas near subduction zones experiencing more pronounced growth. Despite the expansion of the inner core, it is not expected to freeze entirely as the Earth's interior continues to generate heat through radioactive decay, which slows down the crystallization process. It is estimated that it would take approximately 91 billion years for the entire core to solidify [59, 60].

Geoscientists have discovered that the inner core itself has an innermost region, which exhibits different characteristics compared to the rest of the inner core. The crystals in this region have an east-west orientation, in contrast to the north-south orientation of the rest of the inner core. This unique feature is believed to have formed about 500 million years ago due to significant geologic shifts. The precise structure and properties of the crystals in this inner-inner core are still a subject of investigation, and it is suggested that they may differ from the hexagonal close-packed structure of the rest of the inner core [58, 59, 60].

Our understanding of the Earth's core is primarily based on indirect methods, including seismic studies, laboratory experiments simulating extreme temperature and pressure conditions, analysis of meteorites, and computer modeling. Seismic waves generated by earthquakes provide valuable information about the core's structure, as the behavior of these waves changes as they pass through different layers of the Earth. By analyzing seismic data, scientists have been able to infer the presence of the liquid outer core and the solid inner core. Additionally, laboratory experiments and computer simulations help researchers understand the behavior of materials under high-pressure and high-temperature conditions similar to those found in the core. The study of meteorites, which are remnants of early planetary formation, provides insights into the composition and processes involved in core formation. The combination of these methods allows scientists to gain a deeper understanding of the Earth's core, its composition, and its dynamic processes.

2.2 Continental Drift

One of the earliest hypotheses put forth by geologists for the movement of continents through time is called continental drift. Although the science of plate tectonics has now replaced the hypothesis of continental drift, it played a significant role in our understanding of Earth's geological history. The physicist Alfred Wegener is most closely associated with the concept of continental drift. In the early 20th century, Wegener proposed the idea that continents were not fixed in their positions but instead "drifted" across the Earth's surface, occasionally colliding with each other and forming mountain ranges and oceans. He referred to this process as continental drift [61, 62].

Wegener, who was originally trained as an astronomer, supported his theory of continental drift using evidence from various fields such as biology, botany, and geology. One of the key pieces of evidence he presented was the "fit" between the coastlines of South America and Africa. He noticed that the rock strata on the eastern coast of South America matched up with those on the western coast of Africa, as if they were once connected. Additionally, Wegener observed similarities in fossils and geological formations across these continents, further supporting his idea of continental drift.

Despite presenting compelling evidence, Wegener's theory of continental drift was initially met with skepticism by the scientific community. One of the main challenges was the lack of a mechanism to explain the movement of continents. Wegener hypothesized that the rotation of the Earth might have caused the continents to move towards or away from each other. However, this explanation was proven to be incorrect. It was only later, with the development of the theory of plate tectonics, that a viable mechanism for continental movement was proposed.

Plate tectonics, the prevailing theory in geology today, explains the movement of continents through the interactions of large rigid plates that make up the Earth's surface. These tectonic plates, which float on the semi-fluid asthenosphere beneath them, constantly move and interact with one another. This movement is driven by the convective currents in the underlying mantle.

Seafloor spreading zones and enormous rift valleys are some of the most active regions of plate tectonics. At seafloor spreading zones, molten rock rises from deep within the Earth, creating new seafloor (oceanic crust) along the boundaries of existing plates. One of the most prominent seafloor spreading regions is found along the enormous underwater mountain ranges known as mid-ocean ridges. As new crust is formed, the continents on either side of the ridge gradually move apart. For example, the Mid-Atlantic Ridge separates the North American and Eurasian tectonic plates, causing them to drift apart at a rate of approximately 2.5 centimeters per year.

In addition to seafloor spreading, other processes such as subduction play a crucial role in plate tectonics. Subduction occurs when one tectonic plate sinks beneath another into the mantle. This process is responsible for the formation of deep-sea trenches and volcanic arcs. It also contributes to the recycling of crustal material,

as old oceanic crust is eventually consumed back into the mantle.

In conclusion, while the hypothesis of continental drift laid the foundation for our understanding of plate tectonics, it has been replaced by the more comprehensive theory that explains the dynamic nature of Earth's surface. Plate tectonics provides a mechanism for the movement of continents and helps us comprehend the geological processes that have shaped our planet over millions of years.

2.3 Plate Tectonics

The concept of continental drift, which suggested that continents were able to move across the Earth's surface, was initially met with skepticism by scientists due to the solid nature of the ocean floor. However, this idea laid the groundwork for the development of the modern theory of plate tectonics. Plate tectonics proposes that the Earth's surface is divided into several large, unbroken pieces called plates, which are in constant motion relative to one another. These plates include six continental-sized plates (African, American, Antarctic, Australia-Indian, Eurasian, and Pacific) and approximately 14 smaller subcontinental plates (Caribbean, Cocos, Nazca, Philippine, etc.) that make up the Earth's crust [63, 64].

The movement between plates occurs primarily along narrow zones near their boundaries, where deformation is concentrated. Plate movements can occur slowly and continuously through aseismic deformation or suddenly and violently through seismic deformation, resulting in earthquakes. Consequently, earthquakes tend to be concentrated near plate boundaries, where most deformation occurs.

Plate tectonics is primarily a kinematic theory, concerned with describing the geometry of plate movements rather than explaining the underlying causes. However, the massive size and motion of the plates require strong driving forces. The most widely accepted explanation for the origin of plate movement is based on the thermomechanical equilibrium of Earth's interior. The mantle, which is in contact with the heated outer core below and the relatively cooler crust above, exhibits a temperature gradient. As a result, denser (cooler) material overlies less dense (hotter) material, causing the mantle's density to vary with temperature. Warmer, less dense material rises, while colder, denser material sinks due to gravity. This convective process drives the movement of the plates and is known as mantle convection.

2.3.1 Plate Boundaries

Understanding the types of plate boundaries helps us comprehend the movements associated with plate tectonics and provides insights into earthquake behavior along these boundaries.

Spreading Ridge Boundaries

Spreading ridges and rifts are boundaries where plates diverge or move away from each other. The formation of new plate material occurs at these spreading ridges as molten rock from the mantle rises to the surface and cools. The plates at spreading ridges "grow" in this manner. Spreading rates vary from approximately 2

to 18 cm per year, with the Pacific Ocean ridges exhibiting the highest rates and the Mid-Atlantic Ridge having the lowest. Globally, the production rate of new marine crust is estimated to be around 3.1 km^2 per year [65, 66].

When mantle material reaches the surface, it cools and becomes magnetized, acquiring a polarity based on the direction of Earth's magnetic field at that time. This remnant magnetism preserves a record of the Earth's changing magnetic field over geological timescales, including periods of magnetic reversals. By measuring the magnetic field perpendicular to a spreading ridge, an erratic pattern of magnetic strength can be observed. These magnetic anomalies help date different materials and contribute to identifying the geometry and movement of plates, validating the theory of plate tectonics.

Subduction Zone Boundaries

In certain locations, where two plates converge, the consumption of plate material must balance the production of fresh material at spreading ridges. This occurs at subduction zone boundaries, where one plate subducts beneath the other.

Subduction zones frequently occur along the margins of continents. The denser oceanic crust sinks beneath the lighter continental crust due to its colder and thicker nature. A trench forms at the plate boundary, which is known as the subduction zone or trench boundary when the convergence rate is high. Earthquakes occur along the sloping Benioff zone, where the subducting and overriding plates meet. When the convergence rate is slower, sediments accumulate on top of the crustal rock, creating an accretionary wedge that obscures the trench [67, 68].

As the subducting plate sinks, it undergoes heating and becomes less brittle. At depths below approximately 700 km, the plate becomes ductile, resulting in the absence of earthquakes at such depths. Partial melting of the subducting plate can generate magma, which rises to the surface, giving rise to volcanic activity. Mountain ranges can also form when moving plates carrying continents collide. For example, the collision between the Australia-Indian plate and the Eurasian plate created the two crustal layers that make up the Himalayas. The ongoing collision between the plates carrying Africa and Europe is gradually reducing the size of the Mediterranean Sea and may eventually lead to the formation of a mountain range through continued collision.

Transform Fault Boundaries

Transform faults occur when plates slide horizontally past each other without creating or consuming crust. These faults are often found offsetting spreading ridges. Magnetic anomaly offsets and preserved surface scarps serve as markers for these transform faults. However, the term "transform fault" specifically refers to the segment of the fracture zone between the spreading ridges. Magnetic anomaly offsets can extend for thousands of kilometers across fracture zones. The parts of the fracture zone beyond the transform fault move in the same direction on either side, resulting in no relative motion. These dormant sections of the

fracture zone can be thought of as fossil faults that are not currently generating earthquakes.

The theory of plate tectonics provides a valuable framework for understanding and describing changes on Earth's surface, as well as the distribution of volcanoes and earthquakes. By considering the three different types of plate movements, plate tectonics explains the creation of new crust and the consumption of old crust. However, it does not account for all tectonic seismicity, as intraplate earthquakes (occurring within a plate and away from its boundaries) remain a topic of ongoing research [69].

2.4 Faults

While plate tectonics provides a framework for understanding the overall movement of plates at plate boundaries, the specific behavior of tectonic plates can be more complex and intricate when examined at a local scale. In some regions, plate boundaries may be stretched out, resulting in the presence of smaller platelets or microplates trapped between larger plates. Conversely, in other areas, plate boundaries may be well-defined and easily recognizable. It is at these boundaries, along faults, where the movement between different regions of the Earth's crust occurs. Faults are disruptions or fractures in the Earth's crust that can extend from the surface to depths of several tens of kilometers, with varying lengths ranging from a few meters to hundreds of kilometers. Faults may be evident on the surface through topographic features or may be concealed and difficult to detect. It is important to note that the presence of a fault does not automatically indicate the occurrence of earthquakes, as faults can be inactive or movement may occur without seismic activity. Conversely, the absence of visible faulting does not preclude the possibility of earthquakes, as most earthquakes do not result in fault rupture reaching the Earth's surface.

2.4.1 Fault Surface Features

Faults can exhibit a range of surface features depending on their characteristics and the type of movement they accommodate. Some faults may appear as prominent lines or valleys on the Earth's surface, while others may create noticeable steps or offsets. When one block of the Earth's crust has moved relative to another, an exposed fault surface, known as a fault scarp, may be present. The sliding motion along fault surfaces can result in various effects on the surrounding rock. In some cases, the movement of one rock body against another can generate a smooth and polished surface called slickenside. This surface often exhibits linear features, known as slip lineations, which are marks left behind from fault movement. Alternatively, faulting can crush and grind the adjacent rock, transforming it into fault gouge (a powdered form) or fault breccia (a collection of fragmented chunks). However, many faulting events occur deep underground and are not visible on the Earth's surface. While faults can generate earthquakes, the faults themselves may not be directly observable [70, 71].

2.4.2 Fault Classification

Faults can be classified based on the direction of slip and the associated types of stress they accommodate. The movement on faults is influenced by the orientation and magnitude of stresses within the Earth's crust.

Three main types of faults are commonly recognized: normal faults, reverse faults, and strike-slip faults.

Normal faults occur when the rock above the fault (hanging wall) moves down relative to the rock below the fault (footwall). This type of faulting is associated with extensional stresses, where the crust is being pulled apart. Reverse faults, on the other hand, involve the opposite movement, with the hanging wall moving up relative to the footwall. Reverse faults occur under compressional stresses, where the crust is being squeezed together. In some cases, reverse faults with a low dip angle are referred to as thrust faults. Thrust faults can result in significant horizontal displacement and are often associated with large-scale tectonic features, such as mountain belts, as observed in the European Alps [72, 73].

Strike-slip faults involve horizontal movement along the fault plane, parallel to the strike of the fault. The relative lateral displacement of materials on either side of the fault further distinguishes strike-slip faults. Right-lateral strike-slip faults occur when the material on the opposite side of the fault appears to move to the right when viewed from nearby. Conversely, left-lateral strike-slip faults involve material on the opposite side moving to the left when observed in close proximity [74, 75].

2.4.3 Fault Geometry

The geometry of a fault is described using conventional geologic terminology. Although the surface of a fault may appear irregular, it can generally be approximated as a planar surface when examined closely. The orientation of the fault plane is expressed by its strike and dip. The strike represents the intersection of the fault plane with a horizontal plane and is described by its azimuth relative to true north. The dip angle indicates the downward slope of the fault plane, measured perpendicular to the strike. A vertical fault would have a dip angle of 90°. These strike and dip measurements provide crucial information about the spatial orientation and characteristics of faults [76, 77].

2.5 Seismic Hazard

The significance of earthquakes to society stems from their potential to wreak havoc and cause immense destruction. These geological phenomena can result in massive loss of life, city-wide devastation, and even trigger tsunamis. While it is not possible to prevent earthquakes from occurring in densely populated regions, we can implement measures to mitigate their impact. Hazard mitigation is an integral part of earthquake-resistant design in new constructions. Additionally, Earthquake Early-Warning Systems (EEWSs) play a crucial role in minimizing damages, and their implementation is explored in the following section.

2.5.1 Ground Shaking

Seismic waves, which propagate from an earthquake's epicenter, traverse through the earth's crust at high speed, causing trembling at the ground surface. The magnitude and duration of ground shaking at a specific location depend on factors such as the earthquake's size, location, and local geological features. Areas close to a powerful earthquake's epicenter experience severe ground tremors. Ground shaking is considered the

most significant of all seismic hazards, as it forms the basis for other seismic risks. Soil deposits, though not the primary medium for seismic waves, can significantly influence ground shaking characteristics due to their filtering effect. Local variations in soil properties can lead to dramatic differences in ground shaking intensity within a small area.

2.5.2 Structural Hazard

The most visually striking and memorable earthquake damage is the collapse of structures. Structural damage is a major cause of both loss of life and economic loss in many earthquakes. From unreinforced masonry buildings in underdeveloped regions to modern architecture in urban areas, various types of structures are vulnerable to earthquakes. However, not all structural collapses result in fatalities and property loss. Earthquake-induced structural damage may lead to fatalities due to falling items within structures or damage to interior infrastructure such as pipes and lighting systems.

Over time, seismic design standards in building codes have evolved, focusing on earthquake-resistant design to ensure both strength and flexibility. Accurate ground motion predictions have become crucial for designing buildings that can withstand seismic forces effectively.

2.5.3 Liquefaction

One of the most astounding examples of earthquake destruction is liquefaction, where soil deposits lose their strength and behave like fluids. Liquefaction can cause soil to lose its ability to stabilize or support structures, resulting in catastrophic consequences. Liquefaction typically occurs in wet soils near rivers, bays, and bodies of water. It involves various processes, such as flow failures when the soil strength falls below the required level for static stability.

Lateral spreading is another related phenomenon where small displacements occur during earthquake shaking, potentially causing harm to bridges and their components. Level-ground liquefaction is characterized by sand boils caused by the rapid rush of groundwater to the surface, indicating strong groundwater pressures that may lead to subsidence and differential settlements.

2.5.4 Landslides

Powerful earthquakes can trigger landslides, ranging from small to extremely massive slides. In some cases, landslides have tragically buried entire cities and villages. Other earthquake-induced landslides can damage or destroy bridges and other infrastructure, leading to significant economic losses. Landslides can be caused by liquefaction events or the collapse of moderately stable slopes under seismic shaking.

2.5.5 Lifeline Hazard

Lifelines refer to interconnected networks of critical infrastructure that provide essential services for public health and trade. These networks include telecommunications, water and sewage systems, transportation,

oil and gas distribution, and waste storage facilities. Lifelines are vital for modern industrial societies, and their failure after an earthquake can lead to severe economic losses and impact the environment and people's quality of life.

The disruption and financial losses caused by lifeline failures can far exceed the expense of rebuilding damaged facilities. Properly functioning lifelines are crucial for emergency response and rescue efforts following a damaging earthquake. Failure to maintain lifelines can have severe consequences for both local and regional economies, as evidenced by past earthquakes in various parts of the world.

2.5.6 Tsunami and Seiche Hazards

Rapid vertical seabed motions during earthquakes can generate long-period sea waves known as tsunamis. These waves travel great distances across open waters at high speeds, making them challenging to detect. However, as they approach shallower coastal regions, the waves' height rises, resulting in catastrophic destruction when they hit the shore.

Seiches are waves generated in confined bodies of water due to an earthquake's seismic waves. These waves have lengthy periods and can be detected far from the earthquake's epicenter. Seiches can lead to the oscillation of water levels in lakes, reservoirs, and other enclosed water bodies, causing localized flooding and damage to structures near the water's edge.

It is crucial for societies in earthquake-prone regions to understand and prepare for the various seismic hazards mentioned above. Through comprehensive risk assessment, improved construction practices, early warning systems, and effective emergency response plans, the impact of earthquakes can be mitigated, ultimately saving lives and reducing economic losses.

2.6 Earthquake Geometry and Location

The accurate description and determination of an earthquake's location is crucial for understanding its impact and assessing potential hazards. When an earthquake occurs, it is the result of rock rupturing along a fault plane. The initial seismic waves and rupture originate from a point known as the focus or hypocenter. This focus is located at a certain depth below the Earth's surface, and the rupture propagates along the fault plane from this point at a speed of approximately 2 to 3 km/s. The epicenter, on the other hand, is the location on the Earth's surface directly above the focus. The distances between an observer or site and the epicenter and focus are referred to as epicentral distance and focal distance or hypocentral distance, respectively.

Determining the location of an earthquake can be a complex process. The initial epicentral position is relatively easy to calculate using the arrival times of p- and s-waves at a network of at least three seismographs. The preliminary location is estimated based on the time difference between the first arrivals of p- and s-waves using the formula:

$$d = \frac{\Delta t_{p-s}}{\frac{1}{v_s} - \frac{1}{v_p}} \tag{2.1}$$

where Δt_{p-s} is the time difference between p- and s-wave arrivals, and v_p and v_s are the velocities of p-waves and s-waves, respectively. The velocities of p-waves in bedrock typically range from 3 to 8 km/s, while s-wave velocities vary from 2 to 5 km/s. Using this equation, the epicentral distance can be calculated at each individual seismograph station. The uncertainty in the epicenter's location is represented by a circle with a radius equal to the epicentral distance around each seismograph station.

To narrow down the epicenter's location, additional data from at least three seismograph stations is required. By plotting the circles representing the epicentral distances from multiple stations on a map, the likely location of the epicenter is indicated by the points of intersection between the circles. More advanced techniques, involving three-dimensional seismic velocity models of the Earth and numerical optimization methods, are used to achieve greater accuracy in determining the epicenter or hypocentral location. The precision of these methods depends on factors such as the number, quality, and distribution of seismograph stations, as well as the accuracy of the seismic velocity model used in the calculations [78].

2.7 Earthquake Waves

When an earthquake occurs, it initiates a rapid rupture along a fault, causing the sudden displacement of rock masses. As this rupture propagates, it releases the accumulated strain energy in the form of seismic waves that radiate outward from the fault. Each location along the fault's rupture surface contributes to the overall seismic wave pattern, resulting in a complex wave train that interferes with itself as it travels through the Earth [79]. While direct observation of fault slide movement is challenging, seismic waves provide crucial information about the earthquake's source and the mechanics of the rupture process.

2.7.1 Body Waves

Seismic waves can be broadly categorized into body waves and surface waves. Body waves travel through the Earth's interior, and they are further divided into two main types: Primary (P) waves and Secondary (S) waves. P-waves, also known as compressional waves, resemble sound waves in their motion. As they pass through the rock, they cause the particles to oscillate back and forth along the direction of wave propagation. P-waves are the fastest seismic waves, traveling at speeds between 5 and 7 kilometers per second (km/s). Due to their high speed and frequency, P-waves are often the first waves to be detected by seismographs, providing initial information about an earthquake's occurrence. One remarkable property of P-waves is their ability to travel through both solid and fluid materials, making them capable of penetrating the Earth's crust and core as well as bodies of water.

S-waves, on the other hand, are known as shear waves. Unlike P-waves, S-waves cause particles to move perpendicular to the direction of wave propagation. This means that as S-waves travel through the rock, they

induce motion up and down or side to side. S-waves travel at slightly lower speeds than P-waves, typically between 3 and 4 km/s. These waves can only propagate through solid materials, as they rely on the material's shearing stiffness for their motion. When encountering fluids, such as the Earth's outer core, S-waves are completely absorbed and do not pass through.

The time difference between the arrivals of P-waves and S-waves at a seismograph station is used to calculate the distance from the earthquake's epicenter to that station. By comparing the arrival times at multiple seismograph stations, scientists can triangulate the earthquake's epicenter and determine its location on the Earth's surface.

2.7.2 Surface Waves

Surface waves, as the name suggests, travel along the Earth's free surface. These waves result from interactions between body waves and the Earth's surface, and they are generally more pronounced at locations further from the earthquake's epicenter. Surface waves are responsible for causing the majority of ground movements and damages observed in populated areas during earthquakes.

The two main types of surface waves are Love waves (L-waves) and Rayleigh waves (R-waves). Love waves are horizontal waves that move the ground from side to side, similar to the motion of a snake slithering on the ground. These waves travel at speeds ranging from 2 to 4.4 km/s and are the first type of surface waves to arrive at a seismograph station. On the other hand, Rayleigh waves are rolling waves that combine characteristics of both P and S waves. They cause the ground to move in an elliptical motion, with both vertical and horizontal components. Rayleigh waves have velocities similar to Love waves, and they arrive at a seismograph station after the Love waves.

At distances greater than approximately twice the thickness of the Earth's crust from the earthquake's epicenter, surface waves dominate the ground motion. Due to their relatively slow but powerful motion, surface waves are responsible for causing significant damage to buildings and infrastructure in densely populated areas affected by earthquakes.

Overall, the study of seismic waves and their properties is essential for understanding earthquake mechanics, assessing seismic hazards, and designing earthquake-resistant structures to mitigate the impact of future earthquakes on society.

2.8 Earthquakes Recording and Measurements

Gathering data is the first crucial step in analyzing any natural phenomenon, and earthquakes are no exception. To effectively study earthquakes, methods for differentiating between them are needed. Two main approaches can be employed: subjective assessment of ground shaking intensity and damage, or objective measurements of various seismic parameters such as ground vibration amplitude or energy release during an earthquake.

2.8.1 Strong Ground Motion Measurements: Seismograms and Accelerograms

A variety of instruments are available for measuring ground motion, each serving specific purposes. Seismograms are recordings used to measure relatively mild ground motion, while accelerograms are utilized to represent strong ground vibrations and their impact on structures.

A simple yet illustrative example of a seismograph is the mass-spring-damper single-degree-of-freedom (SDOF) device depicted in Figure 2.1. This device consists of a stylus attached to a mass, and a revolving drum is coupled to the seismograph housing. A parallel spring and dashpot connect the mass to the housing, which is attached to the ground. During an earthquake, the mass's motion differs from the ground's motion due to the properties of the spring and dashpot. As a result, the stylus leaves a trace on the revolving drum, indicating the relative displacement between the mass and the ground. In a typical seismograph station, three seismographs are often set up to capture motion in two perpendicular horizontal directions and the vertical axis.

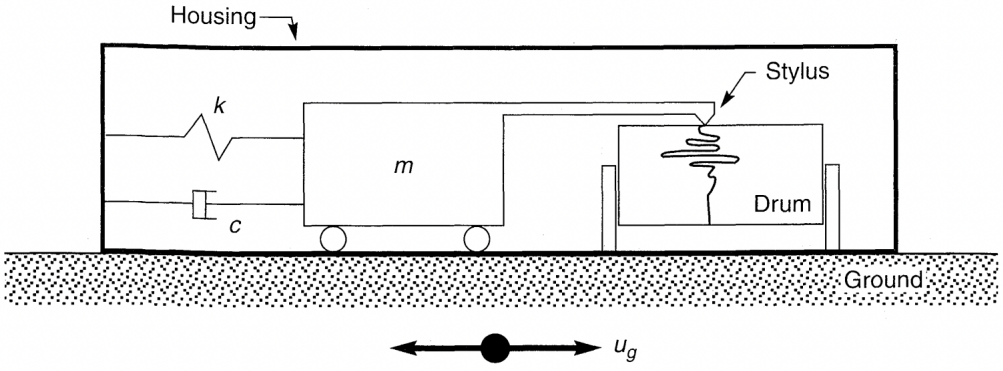


Figure 2.1: Simple seismograph of the mass-spring-dashpot variety. The earth is securely attached to the house. The stylus leaves a track on the revolving drum that represents the relative displacement between the mass and the ground as the earth trembles. The majority of contemporary instruments are more intricate than this one. Image taken from [69]

Seismographs can measure various ground motion characteristics. Understanding the dynamic response of a basic seismograph helps grasp how this is achieved. The equation of motion for such a system is given by:

$$m\ddot{\mu} + c\dot{\mu} + k\mu = m\ddot{u}_g \tag{2.2}$$

where u is the ground displacement, and u_g is the seismograph trace displacement (the distance between the seismograph and the ground).

The displacement response ratio (the ratio of trace displacement to ground displacement amplitude) for ground displacement with a simple harmonic motion at a circular frequency w_g is:

$$\frac{\mid \mu \mid}{\mid \mu_g \mid} = \frac{\beta^2}{\sqrt{(1-\beta^2)^2 + (2\zeta\beta)^2}} \tag{2.3}$$

where $\beta = w_g/w_0$ is the tuning ratio, w_0 is the undamped natural circular frequency, and $\zeta = \frac{c}{2}\sqrt{km}$ is the damping ratio. The trace amplitude equals the ground motion amplitude when ground motion frequencies are significantly higher than the seismograph's natural frequency (large values of β).

The damping ratio determines the lowest frequency for which this equality holds (within a specified range of accuracy). SDOF displacement seismographs are often designed with damping ratios in that range because the frequency response is flat, and phase angles are retained at damping ratios of 60% [80].

Similarly, the acceleration response ratio, which is the ratio of the amplitude of the ground's acceleration to the amplitude of the trace, is given by:

$$\frac{\mid \mu \mid}{\mid \mu_g \mid} = \frac{1}{w_0\sqrt{(1-\beta^2)^2 + (2\zeta\beta)^2}} \tag{2.4}$$

For frequencies significantly below the seismograph's natural frequency (i.e., low values of β), the trace amplitude is proportional to the ground acceleration amplitude. With a 60% damping, a seismograph can accurately measure accelerations up to around 55% of its natural frequency. Most modern seismographs have natural frequencies of around $25Hz$, damping ratios close to 60%, and exhibit flat responses (constant acceleration response ratios) up to frequencies of approximately $13Hz$. These devices can function both as accelerographs and displacement seismographs, monitoring accelerations at frequencies much below and displacements well beyond their native frequency range. Richter utilized the Wood-Anderson seismograph [81], which employed a tiny mass eccentrically strung on a thin tungsten torsion wire, to create the first earthquake magnitude scale. With a ground motion magnification of roughly 3000, the optical recording was made possible by a mirror fastened to the wire. The damped natural period was 0.8 seconds, and electromagnetic damping at 80% critical was applied.

Most modern seismographs use electronic transducers, or seismometers, to detect motion and generate continuous analog electrical signals, which are recorded and processed later. Accelerometers are the most common type of accelerograph used today. These electronic transducers generate output voltages proportional to acceleration. Different types of accelerometers are available, including servo accelerometers, piezoelectric accelerometers, and geophones. Servo accelerometers use a mass suspended from a spring to measure acceleration. The housing's acceleration generates a restoring force that pulls the mass back towards its equilibrium position, and this force is inversely proportional to the acceleration. Piezoelectric accelerometers detect

accelerations by applying a mass to a piezoelectric material, which generates an electrical charge proportional to the acceleration. Piezoelectric materials are particularly useful for high-frequency measurements due to their stiffness and high natural frequencies. Triaxial accelerometers can also detect three orthogonal acceleration components using a common time base, enabling the calculation of any direction's acceleration from the three components.

Instrument shelters are used to protect seismographs, accelerometers, and auxiliary equipment from damage. Accuracy in timekeeping is crucial for seismographs or accelerographs, especially when monitoring multiple motion components or comparing ground motion at different locations. Modern instruments often synchronize with standard time services that broadcast radio time signals, ensuring accurate timekeeping. Universal Coordinated Time (UTC), which is the scientific equivalent of Greenwich Mean Time, serves as the standard time reference for the world.

Seismoscopes are another type of ground motion detector, which are relatively affordable compared to sophisticated seismographs [82]. Seismoscopes use conical pendulums with a metal stylus attached to a suspended mass to record ground motion on smoked glass plates, resulting in two-dimensional records. The seismoscope trace can be used to compute accelerograms, providing valuable data for seismological analyses.

2.8.2 Intensity of an Earthquake

One of the oldest ways to describe an earthquake's size is through its intensity. Earthquake intensity is a qualitative measure of the earthquake's effects at a specific location, based on recorded damage and the responses of the local population. Historical accounts of qualitative earthquake effects have been used to estimate the locations and sizes of pre-instrumental earthquakes, those that occurred before modern seismic instruments were developed. Intensity assessments are also essential in characterizing the recurrence rates of earthquakes with different magnitudes at various locations, a crucial step in assessing seismic hazards. Intensity values can be used to calculate strong ground motion levels, compare earthquake consequences in different places, and estimate earthquake loss. The Rossi-Forel (RF) intensity scale, developed in the 1880s, has been widely used, but the modified Mercalli intensity (MMI) scale, revised in 1931 to better suit California's conditions, has largely replaced it in English-speaking countries [83]. The Japanese Meteorological Agency (JMA) uses its own intensity scale, and the Medvedev-Sponheuer-Karnik (MSK) scale is used in central and Eastern Europe.

Determining earthquake intensity often involves post-earthquake interviews with witnesses. In some seismically active regions, permanent observers are trained to provide objective and unbiased assessments of ground tremors. Intensity observations are crucial as they help quantify ground shaking distribution, especially in regions where human observers and structures are more widely dispersed than seismological observatories with instruments.

2.8.3 Magnitude of an Earthquake

To represent earthquakes more objectively, it is essential to translate seismograms into a quantitative measure that characterizes the earthquake's size. The key to this challenge lies in normalizing measurements. Many earthquake magnitude scales have been developed based on measured characteristics of ground shaking.

Richter Local Magnitude

For shallow, local earthquakes in southern California with epicentral distances less than approximately 600 km, Charles Richter developed a magnitude scale in 1935 using a Wood-Anderson seismometer [84]. The Richter local magnitude (ML) scale is determined based on the greatest trace amplitude (measured in micrometers) recorded on a Wood-Anderson seismometer located 100 km from the earthquake's epicenter. The Richter local magnitude is defined as the logarithm (base 10) of this maximum trace amplitude. The Richter local magnitude (ML) scale, although widely used, is not always the most suitable scale for describing an earthquake's size.

Surface Wave Magnitude

The Richter local magnitude scale does not differentiate between various waveforms. Other magnitude scales have been developed based on specific wave types. Surface waves often dominate the motion at great epicentral distances since body waves have typically been attenuated and dispersed significantly. The surface wave magnitude scale, developed by Gutenberg and Richter [85], is based on the amplitude of Rayleigh waves with a period of 20 seconds. The magnitude of the surface wave is calculated using the formula:

$$M_s = \log A + 1.66 \log \Delta + 2.0 \tag{2.5}$$

where A represents the largest ground displacement in micrometers, and Δ represents the epicentral distance in degrees (360° equals the circumference of the Earth). The surface wave magnitude is most commonly used when describing the size of moderate to large, distant earthquakes (beyond approximately 1000 km) with shallow focal depths (less than about 70 km).

Body Wave Magnitude

For deep-focus earthquakes, surface waves are often too small to allow reliable assessment of the surface wave magnitude. The body wave magnitude (Gutenberg, 1945) is a global magnitude scale based on the amplitude of the first p-wave cycles, which are less significantly impacted by focal depth [86]. The body wave magnitude is defined by:

$$M_b = \log A - \log T + 0.01\Delta + 5.9 \tag{2.6}$$

where T is the p-wave's period (often one second), and A is the p-wave's amplitude in micrometers. The amplitude of higher-mode Rayleigh waves with a one-second period may also be used to determine body wave magnitude, and the resultant magnitude, m_{bLg}, is frequently used to characterize intraplate earthquakes.

Other Instrumental Magnitude Scales

Additional magnitude scales have been suggested, utilizing different sections of the instrumental record. For example, the coda of an earthquake motion, which comprises backscattered waves following the passage of the initial (unreflected) body and surface waves, can be used to assess earthquake size [87]. Aki (1969) developed the coda magnitude, M_c, which can be calculated based on specific coda characteristics. Duration magnitude, M_D, is another scale used to describe small earthquakes that may be of greater interest to seismologists than engineers, as it is based on the entire duration of the earthquake [88]. In Japan, the Japanese Meteorological Agency uses long-period waves to calculate the MIMA local magnitude scale for Japanese earthquakes.

Moment Magnitude

Magnitude scales based on ground-shaking levels often saturate for extremely large earthquakes, where ground shaking causes less discernible effects compared to smaller earthquakes. Moment magnitude is a scale that avoids this limitation by basing earthquake size on the seismic moment, which is a direct indicator of the variables responsible for rupture along the fault. It is the only magnitude scale not subject to saturation. The moment magnitude is given by the formula:

$$M_w = \frac{\log M_0}{1.5} - 10.7 \tag{2.7}$$

where M_0 is the seismic moment in dyne-cm.

2.8.4 Earthquake Energy

Understanding the energy released during an earthquake is often difficult to comprehend due to the sheer magnitude of the numbers involved. The relationship used to determine the total seismic energy released during an earthquake is given by:

$$\log E = 11.8 + 1.5M_s \tag{2.8}$$

where E is measured in ergs. Later research [89] demonstrated that this relationship holds true for moment magnitude as well. This relationship indicates an increase in seismic energy of $10^{1.5}$ or 32 times for every unit change in magnitude. Consequently, a magnitude 5 earthquake would release roughly 0.001 times as much energy as a magnitude 7 earthquake, highlighting the inefficiency of minor earthquakes in dissipating strain energy compared to massive earthquakes. Combining Equations 2.7 and 2.8, we find that the relationship between the energy released during an earthquake and the seismic moment is linear.

The amount of energy released during earthquakes can be staggering. While a single erg is a small unit of energy (1 erg = 10^{-7} J), the energy released in an atomic bomb of the size used at Hiroshima (20,000-ton TNT equivalent) would correspond to a magnitude 6 earthquake. The 1960 Chile earthquake [90, 91] with a moment magnitude (M_w) of 9.5 released as much energy as 178,000 such atomic bombs.

Understanding earthquake magnitudes and the energy they release is crucial for earthquake hazard assessments,

risk management, and building codes in seismically active regions. Accurate and comprehensive earthquake measurements enable scientists and engineers to mitigate earthquake-related risks and improve the resilience of structures and communities in earthquake-prone areas.

Chapter 3

Earthquake early-warning systems

Earthquake early warning (EEW) systems have emerged as a critical tool in seismic risk reduction, providing valuable seconds to tens of seconds of advance notice before an earthquake's ground shaking reaches a specific location. Nations such as Japan, Taiwan, and Mexico have embraced this technology due to its effectiveness in rapidly detecting the onset of an earthquake, assessing its magnitude, pinpointing its location, estimating the expected peak ground motion in surrounding areas, and issuing timely warnings to vulnerable populations.

While predicting the exact time and location of a future devastating earthquake remains challenging, EEW systems have shown that it is possible to forecast the impact of strong ground shaking in nearby areas while an earthquake is still unfolding. This capability is made feasible by the fact that earthquakes generate multiple types of seismic waves that travel at different velocities. Among these, P (primary) waves are the first to arrive at seismic monitoring stations, moving at a speed of approximately 6.5 kilometers per second. Although they generally have relatively small amplitudes and are less likely to cause structural damage, P-waves provide crucial information regarding the earthquake's size and location.

In contrast, S (secondary) waves, which follow the P-waves, travel at a slower speed of around 3.5 km/s but possess significantly greater power, making them capable of inflicting severe damage to buildings and structures. EEW systems utilize the data obtained from the early-arriving P-waves to estimate the anticipated maximum shaking and promptly disseminate this information to locations located farther away from the epicentral zone.

The key advantage of earthquake early warning lies in the difference between the arrival times of P and S waves, known as the S-P time. This time difference serves as a measure of a specific point's distance from the earthquake's epicenter. In areas near the epicenter, the S-P time is minimal, offering little to no warning time and rendering them part of the early warning system's blind zone. However, as one moves further away from the epicentral zone, the warning time significantly increases, affording valuable seconds for preparedness and mitigation actions. It is in these regions that early warning systems prove most effective in providing advance notice of major earthquakes originating at a distance.

The rapid communication of this seismic information from the source to distant locations is possible because the speed of information transmission is nearly instantaneous, far faster than the propagation of the destructive earthquake waves. As a result, EEW systems play a crucial role in enhancing earthquake preparedness, enabling the implementation of safety measures, and potentially saving lives and reducing damages in earthquake-prone regions.

Continued research and development in this field promise to refine the capabilities of earthquake early warning systems further, improving their accuracy, coverage, and reliability. By leveraging these advancements, more regions across the globe can benefit from these life-saving technologies, bolstering resilience in the face of seismic hazards and strengthening societal responses to earthquakes.

3.1 Classical Earthquake Early-Warning Systems

Earthquake Early-Warning (EEW) systems are built upon modern seismic networks, capable of detecting ground movements, processing waveform data in real-time, and analyzing the incoming signals to provide advance warnings of potential earthquakes. While the concept of earthquake early warning has been around for a long time, the technological components needed for its successful implementation have only recently become available. The development of classical EEW systems predates the mobile applications and artificial intelligence that have become more prevalent in recent years [92, 93].

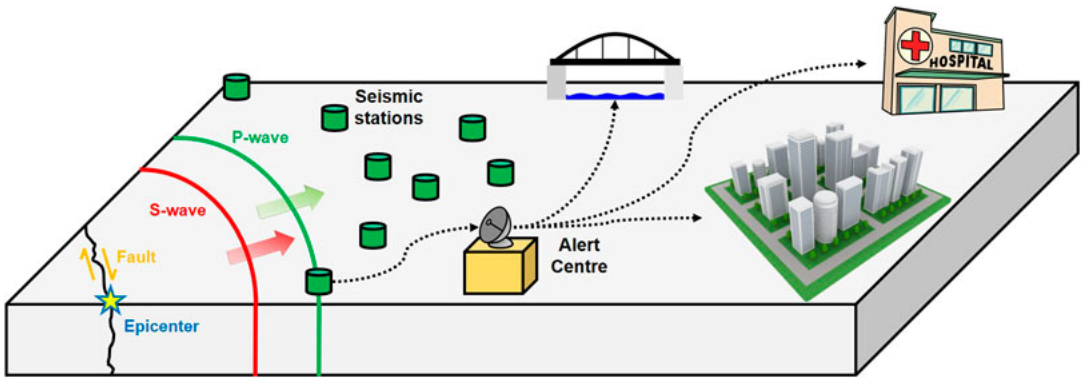


Figure 3.1: Visual representation of a classical earthquake early-warning system based on a network of seismometers. Adapted from [94].

The classical EEW systems primarily relied on the instrumentation and procedures of the time, focusing on detecting P (Primary) waves and front detection techniques. The early concepts proposed by J.D. Cooper after the 1868 Hayward Fault earthquake in San Francisco included the use of a bell at City Hall to sound an alarm if ground vibrations exceeding a specific level were detected along earthquake faults [4]. These early ideas laid the foundation for defining the key characteristics of earthquake early-warning systems that are still valid today. Some of these defining conditions include the limitations of warning proximity to the earthquake source, the improved effectiveness of systems with denser instrumental networks, and the automation of broadcasting warnings and response actions [4].

3.1.1 Front Detection

Front detection, one of the methodologies used in classical EEW systems, involves identifying dangerous earthquakes by detecting destructive ground shaking at a single site and providing a warning before the arrival of the earthquake energy. The idea was first proposed for San Francisco after the 1868 earthquake, where telegraph wires would be used to ring a bell above the city to warn of impending ground shaking [95]. Although this specific system was not implemented, it laid the groundwork for front detection, where seismic sensors are placed in front of potential alert recipients.

The Japanese railway networks installed seismometers along their lines in the late 1960s, which would activate and cut off electricity to trains when severe ground shaking was detected above a predetermined threshold. However, this method only offers warning after significant ground shaking has already begun [92]. To increase warning time, additional devices were placed between the train tracks and the large offshore earthquakes along Japan's east coast [92]. Front detection, which involves detecting an earthquake in one location and sending a warning to another, is the basis for all warning systems that rely on networks. This method can provide meaningful warning times, often tens of seconds, when the seismic sources are at a considerable distance from populated areas. Mexico City's Seismic Alert System (SAS) is an example of front detection, which sends warnings from equipment along the subduction zone's shore to the city, offering a 60-second warning to Mexico City [93].

3.1.2 Using P-Waves

P-waves, being the primary and less destructive waves compared to S (Secondary) waves, play a significant role in classical EEW systems. The time interval between the arrival of P-waves and S-waves determines the amount of advance warning that can be provided. The interval increases with the distance from the earthquake's epicenter and typically ranges from 60 to 90 seconds for large, distant earthquakes [96].

Several P-wave characteristics have been developed for early warning purposes. One of the earliest methods, pioneered by Nakamura in 1988, involved using the predominant period of the first few seconds of the P-wave to estimate the earthquake's magnitude. Another technique, developed by Kanamori in 2005, determined the frequency content of the P-wave using the τ_c technique over a defined time frame, providing a scaling relationship between c and magnitude [92, 4]. Additionally, the amplitude of the P-wave can be utilized to calculate earthquake risk, with peak displacement, velocity, or acceleration of the P-wave's initial few seconds being correlated with magnitude [6, 97, 98].

3.1.3 Onsite Warning

Onsite warning, also known as single-station warning, involves measuring seismic energy at a single point and issuing a warning of impending ground shaking at the same location. This can be achieved by detecting the P-wave and forecasting the peak shaking, which relies on combining different P-wave characteristics. Scaling relationships between P-wave amplitude and peak ground-shaking are one of the simplest methods for

providing warnings. However, to enhance reliability, a combination of frequency and amplitude information is typically used to avoid false positives or missed alerts [92, 6].

The UrEDAS system, developed by Nakamura in Japan, is considered the precursor of both general and onsite earthquake early-warning systems. It calculates source parameters using three seconds of the P-wave and estimates magnitude using the dominant period. By placing a single, three-component seismometer at a user's location, UrEDAS can initiate an on-site alert based on established criteria, offering quick and reliable warnings [92, 99]. Another onsite technique, the $\tau_c - Pd$ approach, combines two P-wave characteristics to estimate peak ground velocity and magnitude when a station triggers on the arrival of a P-wave [6, 100, 98].

3.1.4 Regional EEWSs

Regional EEW systems incorporate multiple elements, often combining front detection, P-wave analysis, and onsite warning techniques within a seismic network. These networks enable the combination of data from several stations, leading to more precise forecasts of ground shaking distribution over the impacted region. The use of onsite processing techniques can deliver the quickest alerts, while data from multiple stations help improve earthquake location accuracy and reduce uncertainty in magnitude estimations.

While front detection offers longer warning durations when a communication network is established for gathering seismic data and alerting users, there may still be a blind zone near the epicenter where no warning can be given due to transmission and processing delays. However, genuine on-site solutions, where seismometers are placed at user locations, can eliminate these telemetry delays, but they may be more prone to false or missed alerts [92].

3.1.5 Challenges of Classical EEWSs

Despite the considerable efforts made in developing and implementing classical EEW systems, challenges remain. Public education and awareness of these systems are essential, as lack of funding and low utilization rates can hinder their effectiveness. Moreover, the cost of implementing and maintaining such systems can be substantial, amounting to millions of dollars, making it difficult for countries with limited economic resources but high seismic activity to adopt them. Therefore, there is a growing need for low-cost alternatives to enable wider implementation of EEW systems and enhance seismic risk reduction efforts. Additionally, continuous research and innovation are necessary to overcome technical challenges and improve the overall efficiency and reliability of classical earthquake early-warning systems [93, 92, 4, 96, 6, 97, 98].

3.2 Smartphone-based earthquake early warning systems

Early warning systems employ real-time monitoring and alert mechanisms to mitigate the potential dangers posed by both natural and human-induced events. In the case of hazardous earthquakes, earthquake early warning (EEW) systems promptly notify the public and first aid organizations. These systems typically rely on networks of stations equipped with sensors that continuously measure seismic and GPS data. By

swiftly detecting an earthquake, timely alerts can be disseminated to the population, enabling them to take precautionary measures even before substantial shaking occurs.

To ensure the effectiveness of EEW systems, a substantial number of stations located near prominent fault traces are required. The construction, maintenance, and operation of these stations can involve significant financial investment, amounting to millions of dollars. In recent times, crowdsourcing systems leveraging cell phone networks have emerged for various social and environmental applications. For instance, the urban heat island effect [101] has been examined by utilizing temperature measurements from smartphones [102]. Additionally, the iSPEX project [103] aims to assess certain characteristics of atmospheric particles by employing spectropolarimeters installed on numerous cellphones.

Studies have proposed the utilization of smartphone-based crowdsourced Earthquake Early Warning (EEW) systems, with simulations being primarily employed to assess their detection capabilities [104]. Commercially available smartphones are equipped with built-in accelerometers and GPS sensors, which can be leveraged for monitoring ground motion and detecting potential earthquakes. These smartphone-based EEW systems have the potential to be constructed using a vast network of hundreds of millions of devices, requiring minimal upkeep. However, there are notable differences between smartphone-based EEW systems and traditional EEW systems. Firstly, the low-cost sensors present in smartphones are not specifically designed for earthquake detection. Secondly, a crowdsourced EEW system relies on a diverse range of sensors and heterogeneous smartphones sourced from multiple suppliers.

3.2.1 The Earthquake Network project

Cellphones' accelerometer sensors have been utilized to measure the ground motion resulting from earthquakes, as demonstrated by studies such as [105] and [106]. The Earthquake Network project aims to establish and oversee a global Earthquake Early Warning (EEW) system utilizing smartphone networks, where smartphones are freely provided by individuals worldwide. This EEW system, known as the Earthquake Network, is designed to operate efficiently within the latencies of mobile telecommunications technologies, enabling real-time earthquake detection and swift alerts to the population. Launched in 2013, the initiative has evolved into a global network encompassing millions of cellphones to date.

The Earthquake Network project shares similarities in scope with the Quake Catcher Network (QCN) project [107] and the Community Seismic Network (CSN) monitoring system [108]. However, QCN and CSN currently rely on Internet-connected computers equipped with either built-in or plugged-in accelerometer sensors. These differences in hardware, software design, and data collection methods arise from the distinct nature of computers compared to cellphones, despite their shared scope. The Earthquake Network project takes advantage of the inherent capability of cellphones, which come with built-in accelerometer sensors. By simply downloading and installing the Earthquake Network app from the Google Play store (https://play.google.com/store/apps/details?id=com.finazzi.distquake), cellphones seamlessly become part of the Earthquake Early Warning (EEW) system

While this study does not delve into the intricate details of the hardware and software architecture of the Earthquake Network Early Warning (EEW) system, we provide a brief overview of its key components below. By installing the Earthquake Network application on a smartphone, the device transforms into an EEW node. Once connected to a power source to prevent battery drainage, and deemed stationary without systematic motion, the smartphone enters the active state following sensor calibration.

An active smartphone transmits an active signal to the server every 30 minutes while continuously monitoring its acceleration. The network's status is determined by the number and geographic distribution of active nodes, as indicated by the active signals. If an active smartphone detects vibration, it sends a vibration signal to the server along with its current latitude and longitude. The server broadcasts an earthquake alert if there is a high volume of received signals relative to the number of active smartphones in a specific location.

In summary, smartphones are utilized to detect ground shaking, and an alert is issued as soon as an earthquake is detected. This system enables individuals residing farther away from the epicenter to receive warnings before being affected by the damaging seismic waves.

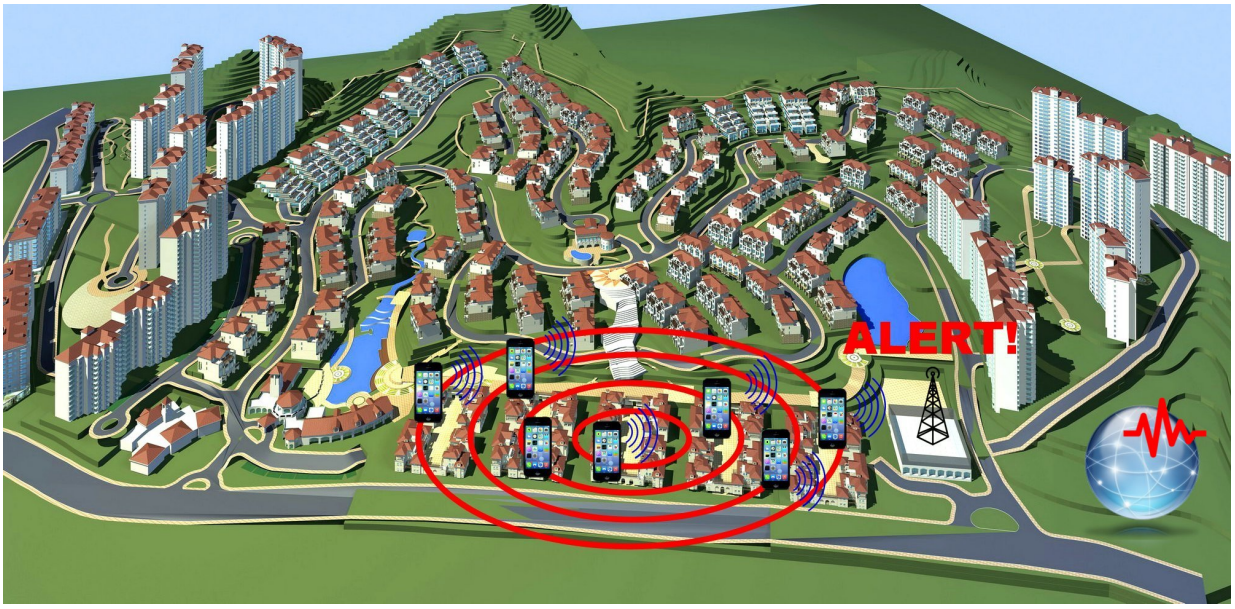


Figure 3.2: Visual representation of EQN. It uses the accelerometer in smartphones to recognize the shaking brought on by an earthquake in real time and sent an alert to user not to close to the epicenter. The image is taken from the website of the earthquake network project (`https://sismo.app/`).

The seminal work by [109], introduces a statistical methodology for real-time earthquake detection. However, the study does not incorporate the spatial dimension of the smartphone network, leading to potential false alarms in the detection algorithm. Moreover, this methodology lacks the capability to estimate crucial earthquake parameters such as epicentre and depth. In contrast, [110] models the detection capabilities of the Earthquake Network (EQN) within a probabilistic framework. The research reveals instances where the smartphone network failed to detect certain relatively strong earthquakes. Notably, the EQN prioritizes

users in close proximity to the detected location when sending out earthquake alerts to the entire network. However, this approach is inadequate as an earthquake may occur within the user-specified warning radius of the smartphone application yet still be too minor for the user to perceive. Conversely, a significant earthquake may impact a broader region even if the user has selected a radius that is too small. The literature extensively covers methods for estimating epicentre and magnitude in EEW systems ([111]; [112]), but it remains unclear if these methods can be effectively applied to the Earthquake Network EEW system. Taken together, these considerations and findings underscore the need for enhancements in the methods and algorithms employed by EQN.

3.2.2 Performance of EQN

EQN detection performance

The performance of EQN in terms of detection latency, false detection rate, and missed earthquake detections was evaluated using a dataset of 550 detections from Chile, USA, and Italy. These countries were chosen because they had at least 10 detections, possessed accurate location information in their national catalogues, and had coverage of low magnitude earthquakes with available accelerometric data. Accurate locations were crucial for latency estimation, while catalogues with low magnitude earthquakes were necessary for sensitivity and false detection rate estimates, as smartphone detections can detect earthquakes as small as M2 (Kong et al., 2020a). In addition, accelerometric data from scientific-grade stations near each detection location was utilized for consistency checks against waveform data.

The association process involved matching EQN detections in time and space with hypocenters from national catalogues. An earthquake was considered the source of the detection if the theoretical arrival time of the P-wave at the detection location fell within a specific time window before and after the detection time. This association process successfully linked 535 out of 550 detections. For further analysis, visual inspection of ground motion was conducted whenever an accelerometric station was available within 20 km of the detection location (410 out of 550 detections). This examination led to the association of four additional detections, including one that was linked to a distant M3.8 earthquake at a distance of 350 km. The false detection rate was determined to be approximately 2%, as some detections could not be associated with any known earthquake.

The 539 associated detections aligned with previous studies on smartphone sensor detectability (Kong et al., 2019). It was observed that half of the detections were related to earthquakes below M4. While these events may not have generated significant shaking necessitating early warnings, comparison with independent data indicated that nearly all EQN detections were likely felt, making them relevant for rapid public information, even for those of very low magnitude.

Assessing EQN's rate of missed earthquakes is more complex compared to traditional seismic monitoring networks due to the network's geometry being influenced by spatio-temporal variations in population distribution and the dynamic nature of app installations and deletions, as well as the number of active

Country	Chile	USA	Italy	Total
Detections	458	70	22	550
Detections associated with known earthquakes	449	70	20	539
Available accelerometers records	328	69	13	410
Magnitude (min; max)	1.4; 7.1	2.2; 7.1	2.4; 5.1	1.4; 7.1
System detection delay w.r.t. origin time (min; median; max in s)	4.8; 17.2; 209.0	4.3; 8.1; 42.5	3.4; 7.3; 11.0	3.4; 15.4; 209.0
System detection delay w.r.t. passing of triggering seismic wave (min; median; max in s)	0.5; 4.3; 12.1	2.0; 4.6; 10.2	1.8; 4.5; 5.9	0.5; 4.3; 12.1
False detection rate (%)	2.0	0.0	9.1	2.0
Nighttime/daytime ratio	2.7	11.3	8.0	3.1
Source of catalogue	CSN	USGS	INGV	

Table 3.1: Summary statistics of earthquake detections. Associated detections are the number of EQN detections for which it was possible to identify the causative earthquake. The accelerometers record column gives the number of detections for which accelerometer data is available within 20 km of the detection location. Detection delays of the EQN system were computed with respect to the earthquake origin time and the theoretical arrival time of the most likely causative seismic phase. False detection rate is the ratio between the number of false detections and the total number of detections while the nighttime/daytime ratio is computed considering that day (7:00 a.m. - 10:59 p.m.) lasts twice the night. CSN: Centro Sismologico Nacional, Chile. INGV: Istituto Nazionale Geologia e Vulcanologia, Italy. USGS: United States Geological Survey. Source [113]

smartphones. As a result, EQN's detectability generally increases at night when more phones are charging. The rate of EQN earthquake detections was found to be 3.1 times higher at night compared to daytime. Despite this variability, the study observed that in Italy, where the number of app users remained stable during the analysed period (approximately 45,000 users), the two largest earthquakes (M5.1 and M4.9) were both detected, along with four out of six earthquakes in the M4.5-4.9 range.

Latency of earthquake detections from a dynamic monitoring network

The detection latency for earthquakes, defined as the time between the earthquake's origin and the issuance of an alert, is minimized when the hypocenter is in close proximity to regions with a high adoption of the EQN app. This accounts for the median detection time of approximately 7-8 seconds in Italy and the USA, where all identified earthquakes occurred on land and at shallow depths within the Earth's crust (less than 40 km). In contrast, in Chile, where a considerable portion of the detected earthquakes were offshore or at intermediate depths, the median detection time was 17 seconds (table 3.1).

A limited comparison of earthquake detection times can be conducted between EQN and ShakeAlert, an operational EEW system designed to cover the West Coast of the USA with 1,700 seismic stations ([114]; [115]). Both systems detected four earthquakes: the M7.1 Ridgecrest mainshock and three of its aftershocks

Magnitude	Origin time	ShakeAlert detection delay (s)	EQN detection delay (s)	EQN detection distance (km)
7.1	July 6th, 2019 03:19:53.04	6.9	40.0	188
4.5	October 15th, 2019 05:33:42.81	5.6	7.2	3
3.8	December 5th, 2019 08:55:31.65	5.7	5.4	10
3.9	December 12th, 2019 08:24:32.6	6.8	10.4	20

Table 3.2: Detection latencies for the 4 earthquakes detected by both ShakeAlert and EQN. These 4 earthquakes were detected in California and they followed the M7.1 Ridgecrest mainshock. ShakeAlert detection times were retrieved from Chung et al. (2020) for the M7.1 Ridgecrest earthquake in California and from http://earthquake.usgs.gov for the others. Source [113]

ranging in magnitude from 3.8 to 4.5. Excluding the mainshock case, EQN exhibited slightly longer latencies on average, with a difference of 1.6 seconds (averages of 7.6 seconds for EQN and 6.0 seconds for ShakeAlert). This difference in latency is relatively small considering the disparity in technology levels between the two systems.

The Ridgecrest earthquake sequence highlights how EQN's performance can rapidly change due to sudden app adoption. The sequence began with a M6.4 foreshock occurring 36 hours before the mainshock. However, the foreshock went undetected by EQN due to a lack of users in California at that time. Subsequently, a sufficient number of EQN installations took place in the Los Angeles (LA) area (not in the epicentral region) as a result of the foreshock, enabling the detection of the mainshock in LA, located 200 km south of its epicenter. The detection latency of 40 seconds (as shown in Table 2) can be attributed to the seismic wave propagation time from the epicenter to LA. The mainshock led to additional EQN installations at shorter epicentral distances, resulting in a significant drop in EQN detection latency to 8 seconds (median times) for the 27 subsequent aftershocks with magnitudes ranging from M2.7 to M4.6 (refer to Table 3.2).

To assess EQN's inherent latency, the wave propagation time from the epicenter to the EQN's detection location is subtracted from the alert issuance latency, considering the most probable causative seismic phase. This estimation takes into account cumulative processing and transmission delays but may overestimate latency since it assumes that acceleration (the monitored parameter) occurs at the onset of the seismic phase, whereas it usually happens later. Therefore, the minimum and median latencies (0.5 seconds and 4.3 seconds respectively, as shown in Table 1) represent the best detection latencies achievable by the EQN system. These fast detection times are noteworthy considering the relatively low investment cost of EQN.

In summary, in regions where there is a significant user base for the EQN app, the detection latency of EQN for crustal earthquakes is comparable to latencies observed in systems like ShakeAlert (5-8 seconds, as indicated in Table 3.2). In the best-case scenario, the latency could be as low as a couple of seconds, demonstrating the system's potential for rapid detection.

EQN warning times

Warning time is defined as the time delay between the publication of an alert and the arrival of the S-wave at the locations of users experiencing a specific target intensity. A larger warning time provides users with more time to prepare for incoming shaking. It is calculated based on the slower and stronger S-wave, assuming that the P-wave is imperceptible and that, from the user's perspective, the delay represents the time between alert issuance and the perceived tremor. The warning time assumes that the maximum intensity begins with the onset of the S-wave.

To calculate warning times, target intensities of 4 (largely observed), 5 (strong), or 6 (slightly damaging) were considered for earthquakes worldwide greater than M4.5 in Italy and the USA, and greater than M5 in other regions. Intensity values based on radial distance were estimated using intensity predictive equations (IPE) according to the validity domain of the respective IPE. Region-specific IPE were used for crustal earthquakes in the Western USA [116] and Italy [117], while the same IPE [118] was used for all other regions, including deeper earthquakes. The earthquake parameters used in this global dataset were obtained from the US Geological Survey (USGS) to maintain consistency.

According to these estimations, out of the 72 detected earthquakes greater than M4.5 or M5, EQN provided early warnings for target intensity 4 for 53 (74%) earthquakes, which occurred in 11 countries across North, Central and South America, Europe, and Asia (Figures 3.3 and 3.4). Among these, 18 events also received warnings for target intensity 5, and two earthquakes had warnings for target intensity 6: the M6.4 earthquake on November 26th, 2019, in Albania, and the M6.2 earthquake on July 26th, 2019, in Panama. As expected, warning times increased with increasing magnitude for a given target intensity, and for a given earthquake, warning times decreased with increasing target intensities. For earthquakes larger than M6, estimated warning times were typically more than 10 seconds for target intensity 4 and more than 5 seconds for target intensity 5 (see Figure 3.3), providing enough time for users to take protective measures.

The warning time for target intensity 6 in the Panama earthquake was too short for individual protective action. However, for the Albania earthquake, which occurred at night and resulted in 51 fatalities, a warning time of 6.9 seconds for intensity 6 was estimated using the IPE. This estimation accounted for a detection delay of 5.1 seconds after the earthquake occurred and the location of the detection 20 km from the epicenter. Based on the IPE, the isoseismal for intensity 6 was 34 km from the epicenter, compared to 45 km based on an empirical intensity-distance curve derived from approximately 4,000 eyewitness reports collected for this event [119]. This implies that the warning times derived from the IPE may have been underestimated by about 2 seconds for intensity 6, resulting in a warning time for "slightly damaging" shaking exceeding 8 seconds. Considering the spatial distribution of EQN users at the time of the earthquake, assuming 100% delivery success and neglecting the transmission delay of the alert, it is estimated that 1,005 users received the early warning for intensity 6, 231 for intensity 5, and 632 for intensity 4. With approximately 800,000 inhabitants living within 40 km of the epicenter, the proportion of warned individuals remains small in this.

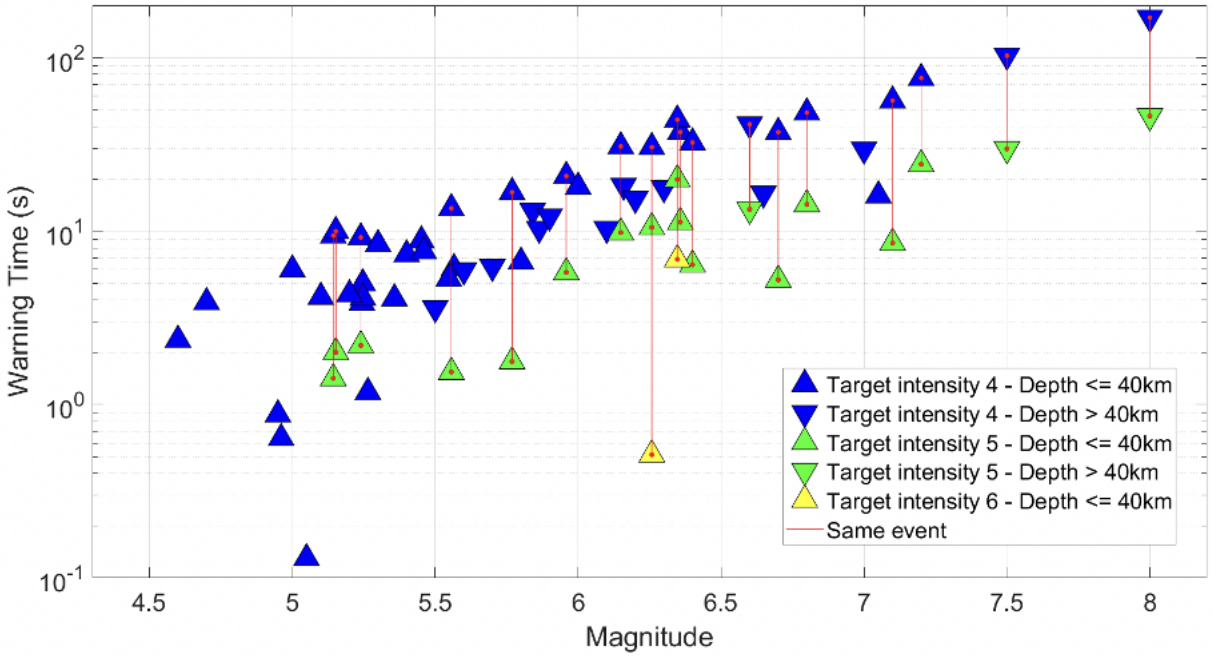


Figure 3.3: Estimated warning times for the 53 earthquakes detected worldwide with magnitude equal or greater than 4.5 with positive warning time. Blue, green, and yellow triangles depict warning times for target intensities 4, 5, and 6, respectively. Crustal and deep earthquakes are shown by triangles and inverted triangles, respectively. Warning times related to the same event are connected by red lines. For sake of clarity, magnitude is altered by a random shift of +/-(0.03, 0.06) for earthquakes sharing the same magnitude. Source [113]

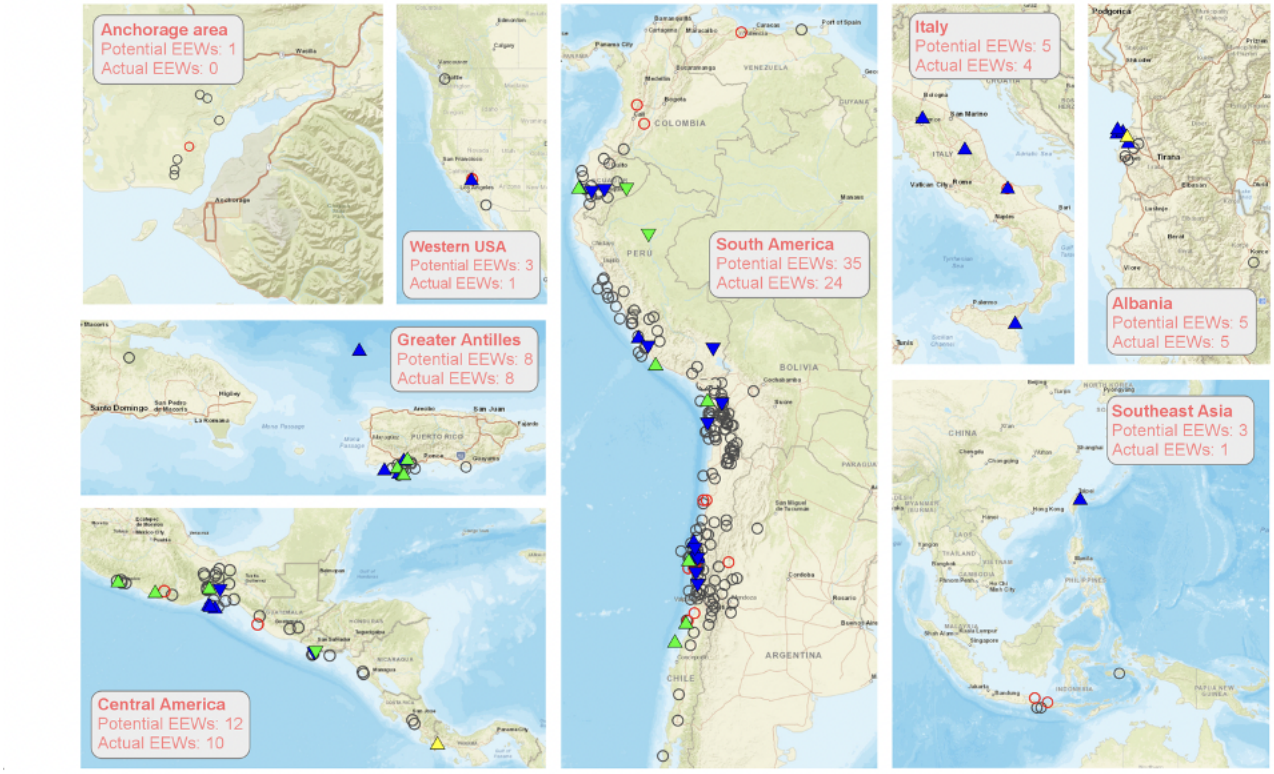


Figure 3.4: . Geographical distribution of the 53 earthquakes for which a positive warning time is determined, shown as triangles (see Figure 3 for legend). All other EQN detected earthquakes of magnitude M4.5 or above are represented by circles - in red when the maximum onshore intensity reached or exceeded intensity 4 (for which an EEW is theoretically possible) and in grey otherwise. The number of EEW in the legends indicates the number of positive warning times at intensity 4. Source [113]

Do EQN users take protective actions after a warning

To assess the effectiveness of EQN in individual risk reduction, an online survey was conducted among EQN users in the felt area of the M8 Peru earthquake on May 26th, 2019. This earthquake had a focal depth of 120 km and triggered two EQN detections, one in Peru and one in Ecuador. Alerts were issued to 599 users for intensity 5 and 54,228 users for intensity 4.

The survey received responses from 2,625 self-selected participants who were over 18 years old. Among them, over two-thirds stated that they were located between 500 and 1,000 km from the epicenter, which includes the capital cities of Quito and Lima. The majority of respondents (82%) reported previous earthquake experiences, and 25% stated that they had received an EQN earthquake early warning before. Overall, 72% of respondents were convinced or strongly convinced of the usefulness of the app, which aligns with previous studies highlighting public expectations for Early Earthquake Warning (EEW) systems.

Out of the 2,625 respondents, 1,663 had the EQN app at the time of the earthquake, while others installed it following the event. Among those who already had the app, 34% received the EQN notification before feeling the shaking, as expected from a Pre-Earthquake Early Warning (PEEW) system. Another 35% received the notification after feeling the shaking, 11% received the notification but did not feel the quake, 14% felt the shaking but did not receive the notification, and 6% neither received the notification nor felt the quake.

Importantly, among the users who received the notification before feeling the shaking, 79% understood that a tremor was about to occur. This indicates a good comprehension of what an early warning is. However, when asked about their reaction, only 25% performed the recommended "drop, cover, and hold" action. Instead, a major concern for users was to warn nearby relatives (55%) or notify others who were not in immediate proximity through social media (22%). Additionally, 35% of respondents chose to wait for the shaking to occur. These findings are consistent with a study by Nakayachi et al. in 2019, which showed that following an EEW in Japan, people tended to mentally prepare rather than take immediate safety actions.

This study, based on self-selected participants and a single case, confirms that a low-cost smartphone-based PEEW system like EQN can provide an actual early warning to some users, even though the dissemination delay of alerts may vary. Despite the lack of information about the magnitude and some users experiencing what they perceived as false alerts (alerts for real earthquakes that are not felt at their location), satisfaction and trust levels remain high. In fact, the survey revealed that 82% of users would appreciate being informed about an incoming earthquake, even if it did not reach damaging levels of intensity.

However, in its current setup, although the meaning of the notification is often understood, it only leads to appropriate protective actions in a minority of cases. This could be because the app does not address the expressed priority need of informing loved ones who may not have the app. Nevertheless, the fact that EQN is highly valued by most of its users suggests that despite its limitations in providing a guaranteed early warning or estimating an event's magnitude systematically, such a service that combines early warning

and rapid detection of felt earthquakes is a step forward in public earthquake information. [120], [113], [121] Contains valuable information related to the analysis and survey conducted.

3.3 A Comparative Analysis of Classical and Smartphone-Based Earthquake Early Warning Systems: Advantages and Disadvantages

Classical Earthquake Early Warning Systems (EEWS) and Smartphone-based Earthquake Early Warning Systems are two distinct approaches to earthquake detection and warning issuance, each with its own unique advantages and disadvantages.

Classical EEWS, being well-established and widely used, relies on specialized seismic sensors and accelerometers that are specifically designed for earthquake monitoring. The advantage of these dedicated sensors lies in their high accuracy and sensitivity, allowing them to provide precise data on seismic events. These systems are often part of robust and reliable seismic monitoring networks that have a long history of successful earthquake detection. Their real-time monitoring capabilities enable timely warnings to be issued to relevant authorities and the public, potentially saving lives and mitigating damage.

However, one of the primary disadvantages of classical EEWS is the considerable cost associated with building and maintaining such a network of specialized sensors. The infrastructure required for classical EEWS can be financially prohibitive for regions and countries with limited resources, restricting its accessibility and implementation in certain areas. Additionally, classical EEWS may face challenges in reaching remote or less populated regions, limiting their coverage and effectiveness in providing early warnings to all at-risk locations.

Furthermore, the lead time for issuing warnings from classical EEWS can be affected by the distance of a location from the earthquake's epicenter. Seismic waves take time to propagate through the Earth, resulting in potential delays in issuing timely warnings to areas far from the source of the seismic event.

On the other hand, Smartphone-based Earthquake Early Warning Systems offer a more accessible and cost-effective alternative. The widespread use of smartphones globally means that smartphone-based EEWS has the potential to reach a larger population. By utilizing the built-in sensors, such as accelerometers, on smartphones, these systems can leverage crowdsourced data from a vast number of devices for real-time earthquake monitoring.

The low cost associated with smartphone-based EEWS makes it an attractive option, especially for regions with limited resources. It can be quickly implemented and scaled up, providing a more inclusive and widespread coverage for earthquake monitoring and early warning dissemination. Additionally, smartphone-based EEWS has the potential to deliver personalized alerts directly to individuals, ensuring that people receive relevant

information based on their location and proximity to the seismic event.

However, smartphone-based EEWS also faces its own set of challenges. The accuracy and sensitivity of smartphone sensors may not match that of specialized seismic instruments, potentially leading to less precise earthquake detection and warning issuance. Moreover, the reliability of smartphone-based EEWS heavily depends on the number of active smartphones in a given area and their ability to continuously collect and transmit data. In regions with poor or limited network connectivity, the timely transmission of data may be hindered, affecting the system's effectiveness in providing timely warnings.

Another concern with smartphone-based EEWS is related to privacy. Crowdsourced data collection from smartphones raises privacy issues, and some users may be hesitant to participate in an EEWS that collects data from their devices, potentially limiting the scope and coverage of the system.

In conclusion, both classical EEWS and smartphone-based EEWS have their strengths and limitations. Classical EEWS provides highly accurate and reliable earthquake monitoring but may be financially prohibitive and less accessible. Smartphone-based EEWS, on the other hand, offers cost-effective and widespread coverage but may face challenges related to sensor accuracy, reliability, network connectivity, and privacy concerns. Integrating both approaches and leveraging their respective strengths could lead to a more comprehensive and effective earthquake early warning system, enhancing disaster preparedness and risk reduction efforts worldwide.

Aspect	Classical EEWS	Smartphone-Based EEWS
Technology	Relies on traditional seismometers and sensors deployed in fixed locations.	Utilizes smartphones equipped with built-in sensors (e.g., accelerometers) for data collection.
Cost	Generally expensive due to the installation and maintenance of dedicated hardware.	Cost-effective as it leverages existing smartphones, reducing infrastructure expenses.
Scalability	Limited scalability due to fixed sensor locations and higher costs.	Highly scalable as it can harness the vast number of smartphones in use globally.
Data Density	Data density depends on the distribution of fixed sensors.	Higher data density due to the ubiquitous nature of smartphones.
Speed of Warning	May provide early warnings, but limited by sensor distance to the earthquake epicenter.	Can deliver real-time warnings even in remote areas, leveraging data from nearby smartphones.
Flexibility	Fixed sensor networks may lack adaptability and maneuverability.	Flexible and adaptable as smartphones can be easily relocated and redistributed.
Accessibility	Access to data limited to specific authorities and researchers.	Data accessibility extended to a broader population with smartphones.
Reliability	Reliant on the functionality of fixed sensors and communication infrastructure.	Subject to potential signal and network disruptions in case of large-scale events.
Public Participation	Limited involvement of the general public.	Encourages public participation and citizen science by using personal smartphones.
Disaster Preparedness	Effective for disaster preparedness in monitored areas.	Enhances disaster preparedness and early response capabilities even in previously unmonitored regions.

Table 3.3: Comparative Analysis of Classical and Smartphone-Based Earthquake Early Warning Systems

Chapter 4

Statistical methods applied to EQN

Statistics is a versatile field with wide-ranging applications, playing a pivotal role in numerous disciplines and industries. Its primary purpose is to extract valuable insights and make informed decisions based on data analysis. By employing statistical techniques, we can unravel patterns, trends, and relationships that exist within complex datasets. One fundamental aspect of statistics is its ability to draw meaningful conclusions from data. By utilizing various statistical methods and models, we can derive valuable information that enhances our understanding of natural phenomena. Whether it's studying the behavior of biological systems, analyzing economic trends, or investigating social phenomena, statistics provides a quantitative framework to accurately describe and interpret the intricacies of the world around us.

Moreover, statistics aids in comprehending the nature and variability of different phenomena through quantitative observations. By collecting and analyzing data, we can quantify the extent of variability, identify outliers, and discern underlying patterns. This quantitative understanding allows researchers and practitioners to make informed decisions, design effective strategies, and address challenges in fields as diverse as healthcare, finance, engineering, and social sciences. In addition to its descriptive capabilities, statistics also plays a crucial role in making predictions and inferences. Maximum likelihood estimation, one of the powerful statistical tools, allows us to estimate unknown parameters based on observed data. This estimation technique finds applications in various domains, such as estimating earthquake parameters like geographical position and depth. By accurately estimating these parameters, we can enhance our understanding of seismic activity and improve disaster preparedness measures.

Furthermore, hypothesis testing is a key statistical tool used to distinguish between true and false claims or hypotheses. In the context of earthquakes, hypothesis testing enables us to evaluate the validity of seismic events by comparing observed data with expected patterns. This rigorous methodology helps differentiate genuine earthquakes from noise or false signals, ensuring reliable earthquake detection and monitoring systems. Overall, statistics provide a robust framework for exploring, analysing, and interpreting data in a wide range of fields. Its applications are far-reaching, enabling us to make evidence-based decisions, formulate strategies, and contribute to advancements in science, technology, and society as a whole.

4.1 Overview of inferential statistics

Statistical inference is the process of drawing conclusions or making predictions about a population based on information obtained from a sample. It involves using the properties of the sample, such as sample mean or sample proportion, to estimate or make inferences about unknown parameters of the population.

In the context you described, the random variable X represents the population under investigation, and its probability distribution is described by $f(x)$ with some unknown parameter θ. The parameter θ can be a single value or a vector of values, and it belongs to the parameter space Ω.

While the functional form of $f(x)$ is assumed to be known, the specific value of θ is unknown. The goal of statistical inference is to use the information from a sample to estimate or make inferences about the unknown parameter(s) θ. This can involve estimating the value of θ, testing hypotheses about θ, or constructing confidence intervals to quantify the uncertainty associated with the parameter(s).

Estimation involves determining the value of a parameter, denoted by θ, based on sample data. The goal is to use the available information from the sample to estimate the unknown population parameter. There are different methods for estimation, such as point estimation and interval estimation. Point estimation aims to provide a single value that is considered the best estimate of the parameter. In contrast, interval estimation provides a range of values within which the parameter is likely to fall.

The problem of Testing of the Hypothesis: Hypothesis testing involves making decisions or drawing conclusions about the population based on sample data. It is used to test the validity of a claim or hypothesis about the value of a population parameter. The process involves formulating null and alternative hypotheses, selecting an appropriate statistical test, collecting sample data, and making a decision about whether to accept or reject the null hypothesis based on the observed data.

Both estimation and hypothesis testing are crucial components of statistical inference, as they allow us to make inferences about population parameters based on sample data. These concepts are widely used in various fields, including scientific research, social sciences, economics, and many others.

4.1.1 Point estimation

The probability of the sample, denoted as $P(x_1, x_2, ..., x_n)$, is given by the product of the individual probabilities of each random variable in the sample. Since the random variables $x_1, x_2, ..., x_n$ are independent and identically distributed, their joint probability can be expressed as the product of their individual probabilities.

If the random variables are continuous, the joint probability density function (PDF) is given by:

$$P(x_1, x_2, ..., x_n) = f(x_1)f(x_2)...f(x_n) \tag{4.1}$$

where $f(x)$ represents the PDF of the random variable X. If the random variables are discrete, the joint probability mass function PMF is given by:

$$P(x_1, x_2, ..., x_n) = p(x_1)p(x_2)...p(x_n) \tag{4.2}$$

where $p(x)$ represents the PMF of the random variable X. In both cases, the product indicates that the probability of each individual random variable occurring is multiplied together to determine the joint probability of the entire sample. This concept is fundamental in probability theory and plays a crucial role in statistical inference, as it allows us to reason about the likelihood of observing a particular sample given the underlying distribution of X.

A statistic, denoted as T, is a function that is calculated based on the sample values $(x_1, x_2, ..., x_n)$ and does not depend on the unknown parameter θ. The statistic T itself is a random variable because it is calculated from a random sample, and it has its own probability distribution. This probability distribution is known as the sampling distribution of T. The sampling distribution describes the variability of the statistic T when different samples are drawn from the population. It helps us understand the properties and behavior of the statistic in relation to the population parameter θ. When we use the statistic T to estimate the unknown parameter θ, T is referred to as an estimator or a point estimator of θ. The value of T calculated from a specific sample is called its estimate. Estimators play a crucial role in statistical inference as they provide a way to make inferences about the population parameter based on the information contained in the sample. While we aim for the difference between the estimator T and the true parameter θ to be as small as possible, it's important to recognize that T is a random variable, and its value can vary from one sample to another. As a result, we cannot guarantee that T will be exactly equal to θ. Instead, we focus on selecting estimators that, on average, provide estimates that are close to the true parameter value. The quality of an estimator is typically assessed based on its bias, variance, and mean squared error MSE. A good estimator has low bias, meaning that, on average, it is not systematically overestimating or underestimating the parameter. It also has low variance, indicating that its estimates do not vary greatly between different samples. The MSE takes into account both bias and variance, and a good estimator has the lowest possible MSE. In statistical inference, our goal is to choose estimators that are unbiased, efficient (have low variance), and consistent (converge to the true parameter as the sample size increases). By selecting estimators with these desirable properties, we can have confidence that our estimates will be close to the true parameter value with a high probability.

Properties of an estimator

Unbiasedness An estimator T is said to be unbiased if the expected value (or mean) of T, denoted as $E(T)$, is equal to the true parameter value θ for all possible values of θ in the parameter space Ω. In mathematical terms, it can be expressed as:

$$E(T) = \theta \tag{4.3}$$

for all $\theta \in \Omega$. This means that, on average, the estimator T provides estimates that are equal to the true parameter value. Unbiased estimators are desirable because they do not systematically overestimate or underestimate the parameter. It's important to note that unbiasedness is a property of the estimator itself, not of a particular estimate obtained from a specific sample. Different samples may yield different estimates, but if the estimator is unbiased, the average of these estimates over many samples will be equal to the true parameter value θ. When considering the class of unbiased estimators, it is often desirable to select the estimator with the smallest variance. This class of estimators is known as Minimum Variance Unbiased (MVU) estimators. The reason for choosing the estimator with the smallest variance is rooted in the interpretation of variance as a measure of concentration or spread around the mean. In this context, the smaller the variance, the more concentrated the estimator is around the true parameter value θ. Chebyshev's inequality [122] provides a lower bound on the probability that an estimator T deviates from the true parameter value θ by a given amount ϵ. According to the inequality:

$$P\{[T-\theta] \leq \epsilon\} > 1 - \frac{var(T)}{\epsilon^2} \tag{4.4}$$

This inequality tells us that the probability of T being within a certain distance ϵ from θ is greater than or equal to 1 minus the variance of T divided by ϵ^2. Hence, by choosing the estimator with the smallest variance, we increase the lower bound of the probability of T being close to θ. In other words, the smaller the variance of the estimator, the higher the probability that it provides estimates concentrated around the true parameter value θ. Therefore, within the class of unbiased estimators, the one with the smallest variance is considered to be the best choice.

Consistency A sequence of estimators $T_n, n = 1, 2, ...$ of a parameter θ is said to be consistent if, as n approaches infinity and for any $\epsilon > 0$:

1. T_n converges in probability to θ, denoted as

$$P\{|T_n - \theta| > \epsilon\} \to 0 \tag{4.5}$$

2. T_n almost converge surely to θ denoted as

$$P\{|T_n - \theta| \leq \epsilon\} \to 1 \tag{4.6}$$

Efficiency If T and T' are two unbiased estimators of a parameter θ, each having a finite variance, T is said to be more efficient than T if V(T) > V(T'). The relative efficiency of T relative to T is defined as:

$$Eff(\frac{T}{T'}) = \frac{V(T')}{V(T)} \tag{4.7}$$

This relative efficiency is used to assess the efficiency of an unbiased estimator by comparing its variance with the Cramer-Rao lower bound (CRB) [123]. The Cramer-Rao lower bound is a theoretical lower bound

on the variance of any unbiased estimator for a parameter. It represents the minimum achievable variance for unbiased estimators. If an estimator achieves the Cramer-Rao lower bound, it is said to be efficient. By comparing the variance of an estimator with the CRB, we can determine how close the estimator is to the best possible estimator in terms of efficiency. A higher relative efficiency value indicates that the estimator has a smaller variance compared to the CRB and is considered more efficient. In a regular situation where the regularity conditions of the Cramer-Rao inequality hold for the family $\{f(x,\theta) \in \Omega\}$, an unbiased estimator T^* of θ is called the most efficient estimator if its variance $V(T^*)$ equals the Cramer-Rao bound (CRB). In this case, the efficiency of any other unbiased estimator T of θ is defined as:

$$Eff(T) = \frac{V(T^*)}{V(T)} \tag{4.8}$$

where T^* represents the most efficient estimator. The efficiency of an estimator T is a relative measure that compares its variance with the variance of the most efficient estimator T^*. A higher efficiency value indicates that the estimator T has a smaller variance compared to T^*, implying that it is more efficient in terms of achieving lower estimation variability. By comparing the efficiency of different unbiased estimators to the most efficient estimator, we can evaluate their relative performance in terms of variance and assess how close they are to the theoretical lower bound provided by the Cramer-Rao bound.

In addition to the previous definition of efficiency, we can define efficiency in another way known as asymptotic efficiency. This definition is applicable to consistent estimators that are asymptotically normally distributed. Among this class of estimators, the one with the minimum asymptotic variance is referred to as the "most efficient estimator." It is also known as the best asymptotically normal (BAN) estimator or consistent asymptotically normal efficient ($CANE$) estimator. If we denote the asymptotic variance of the BAN estimator T^* as $avar(T^*)$, then the efficiency of any other estimator T (within the class of asymptotically normal estimators) can be defined as:

$$Eff(\frac{T}{T^*}) = \frac{avar(T^*)}{avar(T)} \tag{4.9}$$

Here, $avar(T)$ represents the asymptotic variance of the estimator T. The efficiency in this context compares the asymptotic variances of different estimators. The higher the efficiency value, the smaller the asymptotic variance of the estimator T compared to the most efficient estimator T^*. In other words, it indicates how well the estimator T performs relative to the best asymptotically normal estimator in terms of reducing estimation variability in the limit as the sample size increases. By comparing the efficiency of different asymptotically normal estimators to the most efficient estimator, we can assess their relative performance and determine which estimator achieves the smallest asymptotic variance, making it the most efficient within this class.

4.1.2 Maximum likelihood estimation

When we encounter a dataset, our primary objective is to gain insights and summarize it in a meaningful way. Descriptive statistics, such as the sample mean, sample variance, and other measures [124, 125], serve as useful tools for achieving this goal. However, it is important to remember that the sample we have collected represents a random subset of the larger population of interest, which follows a certain distribution governed by specific parameters. Consequently, we become intrigued by the task of estimating and evaluating these population parameters.

To estimate population parameters, a multitude of methods have been developed. Each method offers a unique approach to estimation and has its own advantages and limitations. Among these methods, we find the method of moments, which matches sample moments with their corresponding population moments, aiming to provide parameter estimates [126, 127]. Another prominent method is maximum likelihood estimation, which seeks to find the parameter values that maximize the likelihood function based on the observed data [128, 129]. Bayes estimators, on the other hand, incorporate prior information and employ Bayesian inference to obtain estimates [130, 131]. Additionally, minimax estimation focuses on minimizing the maximum possible estimation error [132, 133].

These methods for estimating population parameters have received extensive attention and have found applications across various fields. Researchers have contributed significantly to advancing our understanding of these methods and their properties. They have explored the theoretical foundations, investigated the properties of estimators, and developed techniques to assess the performance of different estimation methods. By studying and applying these estimation techniques, we gain valuable insights into the underlying population and can make informed decisions based on our data.

In particular, maximum likelihood estimation (MLE) is a widely used statistical method that aims to estimate the parameters of a probability distribution by maximizing the likelihood function. In particular, MLE seeks the parameter values, denoted by $\boldsymbol{\theta}$, that optimize the likelihood function based on the observed sample data.

Consider a random sample of size n, denoted as $x_1, x_2, \ldots, x_n$, drawn from an unknown probability distribution. The likelihood function, denoted by $L(\cdot)$, captures the probability of observing the given sample for different values of the parameter(s) $\boldsymbol{\theta}$. Mathematically, the likelihood function is defined as:

$$L(\boldsymbol{\theta}; x_1, x_2, \ldots, x_n) = f(x_1, x_2, \ldots, x_n; \boldsymbol{\theta}), \tag{4.10}$$

where $f(x_1, x_2, \ldots, x_n; \boldsymbol{\theta})$ represents the joint probability density function (pdf) or probability mass function (pmf) of the random sample. It is important to emphasize that the random variables $x_1, x_2, \ldots, x_n$ are functions of the parameter(s) $\boldsymbol{\theta}$, which is why we denote the likelihood function as $L(\boldsymbol{\theta})$.

Under the assumption that the random sample is independent and identically distributed (iid), we can express the likelihood function as the product of the individual pdfs/pmf:

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} f(x_i; \boldsymbol{\theta}), \tag{4.11}$$

where $f(x_i; \boldsymbol{\theta})$ represents the pdf/pmf of each random variable x_i.

To estimate the parameter(s) $\boldsymbol{\theta}$, we seek the values that maximize the likelihood function. In practice, it is often more convenient to work with the logarithm of the likelihood function, denoted as $\ln L(\boldsymbol{\theta})$. Fortunately, maximizing $\ln L(\boldsymbol{\theta})$ yields the same parameter estimates as maximizing $L(\boldsymbol{\theta})$. This property arises from the fact that the logarithm is a monotonically increasing function.

Now, let's consider a concrete example to illustrate the concept of MLE. Suppose we have a random sample drawn from a normal distribution with an unknown mean μ and variance σ^2 (standard deviation σ). The pdf of the normal random variable is given by:

$$f(x; \mu, \sigma^2) = (2\pi\sigma^2)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right). \tag{4.12}$$

Based on this distribution, we can construct the likelihood function:

$$L(\mu, \sigma^2; x_1, \ldots, x_n) = (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2\right). \tag{4.13}$$

Taking the logarithm of the likelihood function, we obtain the log-likelihood function:

$$l(\mu, \sigma^2; x_1, \ldots, x_n) = -\frac{n}{2}\ln(2\pi) - \frac{n}{2}\ln(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2. \tag{4.14}$$

To find the maximum likelihood estimates $\hat{\mu}$ and $\hat{\sigma^2}$, we solve the system of equations:

$$\frac{\partial l(\mu, \sigma^2)}{\partial \mu} = 0, \ \frac{\partial l(\mu, \sigma^2)}{\partial \sigma^2} = 0. \tag{4.15}$$

After solving this system, we obtain the estimators for μ and σ^2:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i, \ \hat{\sigma^2} = \frac{1}{n}\sum_{i=1}^{n}(x_i-\hat{\mu})^2. \tag{4.16}$$

These estimators represent the values of μ and σ^2 that maximize the likelihood function and, consequently, provide the best fit to the observed data based on the assumption of a normal distribution.

In summary, maximum likelihood estimation (MLE) is a powerful statistical technique for estimating unknown parameters by maximizing the likelihood function. By finding the values of the parameters that maximize the likelihood, we can obtain estimators that best describe the underlying probability distribution of the observed data. In the case of the normal distribution example, the estimators $\hat{\mu}$ and $\hat{\sigma^2}$ are obtained by calculating the sample mean and sample variance, respectively. MLE serves as a fundamental tool in statistical inference and finds applications in various fields of study.

The Maximum Likelihood Estimator (MLE) possesses several desirable properties, which make it a widely used and valuable estimation method. The key properties of the MLE are as follows:

1. Consistency: The MLE is a consistent estimator, meaning that as the sample size increases, the MLE converges to the true parameter value. In other words, as the sample size grows larger, the MLE becomes increasingly accurate in estimating the unknown parameter.

2. Asymptotic Normality: Under certain regularity conditions, the MLE follows an asymptotic normal distribution. This means that as the sample size increases, the distribution of the MLE approaches a normal distribution with a mean equal to the true parameter value and variance determined by the Fisher Information.

3. Efficiency: The MLE is an efficient estimator, which means it achieves the Cramer-Rao lower bound (CRB) for the variance of unbiased estimators. In other words, among all unbiased estimators, the MLE has the smallest variance, making it the most efficient in terms of reducing estimation variability.

4. Invariance: The MLE possesses the invariance property, which means that if θ is the MLE of the parameter θ, then any function of θ, denoted as $g(\theta)$, can be estimated by applying the same function to θ. This property allows for straightforward estimation of derived quantities or transformations of the parameter.

5. Robustness: The MLE is robust to violations of certain distributional assumptions. Even if the assumed distribution is not exactly correct, the MLE can still provide consistent and asymptotically normal estimates, as long as the model is specified correctly in terms of the form of the likelihood function.

These properties make the MLE a powerful and reliable estimation technique in statistical inference. It is widely used in various fields, including economics, biostatistics, engineering, and social sciences, among others.

4.1.3 Hypothesis testing

Maximum likelihood estimation (MLE) is a powerful tool for estimating parameters based on a random sample. However, it does not provide us with a way to draw conclusions about the estimated parameters. This is where hypothesis testing comes into play. Hypothesis testing is a statistical technique that allows us to make judgments about population parameters based on data from a random sample [134, 135, 136, 137, 138, 139].

Hypothesis testing involves two hypotheses: the null hypothesis (H_0) and the alternative hypothesis (H_1). The null hypothesis is the statement that we want to test, usually assuming no significant difference or no effect. On the other hand, the alternative hypothesis is the hypothesis accepted when we observe evidence against the null hypothesis, indicating a departure from it. It is important to note that we never know for certain which hypothesis is true.

To conduct hypothesis testing, we follow a series of steps:

1. Determine the null and alternative hypotheses, typically based on mathematical formulations. These hypotheses often relate to specific characteristics or descriptive measures of the population.

2. Take a random sample from the population of interest. This sample serves as the basis for drawing inferences about the population parameters.

3. Calculate a sample statistic, also known as a test statistic, which provides information about the null hypothesis. The choice of test statistic depends on the nature of the hypothesis being tested and the available data.

4. Compare the calculated test statistic with a critical value or critical region. If the statistical result is compatible with the null hypothesis, meaning it falls within the critical region, we do not reject H_0. However, if the test statistic value falls outside the critical region, we reject H_0 in favour of H_1.

The crucial steps in hypothesis testing involve calculating the test statistic and determining if there is sufficient evidence to reject the null hypothesis. This is done by understanding the distribution of the test statistic under the null hypothesis, as the test statistic is a random variable that depends on the random sample.

The last two steps, rejecting or accepting the null hypothesis, are carried out by defining a rejection region. This region represents an interval in which we reject the null hypothesis. The critical value, derived from the distribution of the test statistic under H_0, is the threshold that separates the rejection and acceptance regions in the domain of the test statistic.

The determination of the rejection region depends on the statistical distribution under the null hypothesis, the alternative hypothesis, and the desired level of error. When the null hypothesis is true, the critical regions are typically located in the tails of the test statistic distribution. These tails may be the lower tail, the upper tail, or both, depending on the nature of the alternative hypothesis.

Hypothesis testing involves the possibility of two types of errors. A Type I error occurs when we reject the null hypothesis even though it is true. The probability of committing a Type I error is denoted by α and is often referred to as the significance level of the test. The value of α represents the maximum tolerable level of error and is commonly set to values such as 0.05, 0.01, or 0.1. The critical value is determined as the quantile (under H_0) that corresponds to a significance level of α.

On the other hand, a Type II error occurs when we fail to reject the null hypothesis when it is actually false. The probability of committing a Type II error is denoted by β, and its complement $1 - \beta$ represents the power of the test. The power of the test indicates the probability of correctly rejecting the null hypothesis when the alternative hypothesis is true. It is important to note that the Type II error and power calculation

must be performed for each alternative hypothesis of interest, as we are considering multiple possibilities for departures from the null hypothesis.

4.2 Application of statistical tools to EQN

Smartphone-based earthquake early warning systems (EEWSs) are emerging as a complementary solution to classic EEWSs based on expensive scientific-grade instruments. Smartphone-based systems, however, are characterized by a highly dynamic network geometry and by noisy measurements. Thus, there is a need to control the probability of false alarms and the probability of missed detection.

This section proposes a statistical methodology to address this challenge and to jointly estimate earthquake parameters such as epicentre and depth in near real-time. The methodology takes into account the unique characteristics of smartphone-based systems, including the dynamic network geometry and the presence of measurement noise.

To validate the effectiveness of the methodology, it is tested using data obtained from the Earthquake Network citizen science initiative. This initiative implements a global smartphone-based EEWS, utilizing a network of smartphones to detect and report earthquake events. The rich dataset collected from this network allows for a comprehensive evaluation of the proposed methodology.

The experimental results demonstrate the capability of the statistical methodology to accurately estimate earthquake parameters in near real-time within smartphone-based EEWSs. By effectively addressing the challenges posed by dynamic network geometry and noisy measurements, the proposed approach contributes to reducing the probability of false alarms and missed detections, thereby improving the overall reliability and performance of smartphone-based systems.

This section represents a significant advancement in the development of smartphone-based EEWSs. By leveraging statistical techniques and the ubiquity of smartphones, these systems have the potential to expand the coverage and accessibility of early warning capabilities, ultimately enhancing earthquake preparedness and response efforts on a global scale.

4.2.1 Problem formalization

The earthquake detection algorithm utilized by EQN incorporates a dynamic number of triggers denoted as k_j, where j represents the index of a specific detection. Each trigger is characterized by a feature vector $\mathbf{x}_i = (t_i, \text{lat}_i, \text{lon}_i)$, where $t_i \in \mathbb{R}$ denotes the triggering time, and $(\text{lat}_i, \text{lon}i) \in \mathcal{S}^2$ represents the spatial coordinates of the smartphone on a sphere embedded in $\mathbb{R}^3$. By collecting all triggers, we construct a matrix $\mathbf{X} = (\mathbf{x}1', \ldots, \mathbf{x}k_j{}')'$, and the feature space is defined as $\mathcal{X} = \bigcup k = 1^{\infty}\mathcal{X}_k$, where $\mathcal{X}_k = \mathbb{R}^k \times (\mathcal{S}^2)^k$ corresponds to the feature space for a given number of triggers k. Now, let us introduce the label space $\mathcal{Y} = -1, 1$, where each earthquake detection is assigned a label y. Specifically, $y = 1$ indicates a false detection,

while $y = -1$ signifies a true earthquake detection. The primary objective is to learn a hypothesis map $h : \mathcal{X} \to \mathcal{Y}$ that effectively approximates the true label y for any given data point $\mathbf{X}$. However, due to the intricate spatio-temporal dynamics of seismic waves and the complex spatial distribution of smartphones during an earthquake, the hypothesis map h inherently exhibits a highly nonlinear nature.

To address this challenge, a statistical parametric model $f : \mathcal{X} \to \Theta$ is adopted to discern whether a given data point $\mathbf{X}$ is generated by a true earthquake. Here, $\Theta = \mathbb{R}^s$ represents the parameter space, with $s \ll k_j$ denoting the size of the parameter vector $\boldsymbol{\theta}$. Consequently, the hypothesis map can be expressed as $h(\mathbf{X}) = g(f(\mathbf{X})) = g(\boldsymbol{\theta})$, where g is a function that maps the parameter space to the label space.

In the context of earthquake early warning (EEW) systems, precise control over two crucial parameters becomes essential: the probability α of missed detections (referring to true earthquakes that go undetected) and the probability β of false detections (representing detections unrelated to actual earthquakes). To effectively tackle this challenge, a 0/1 loss function is employed, given by $L((\mathbf{X}, y), g) = \mathbb{I}(y \neq \hat{y})$, where $\hat{y}$ represents the predicted label. The ultimate goal is to learn a function g that minimizes the Bayes risk, as expressed by the optimization problem:

$$\hat{g} = \arg\min_{g} \; E\left[L((\mathbf{X}, y), g)\right]. \tag{4.17}$$

However, solving (4.17) directly necessitates knowledge of the joint probability distribution $p(\mathbf{X}, y)$. Since obtaining this distribution is often infeasible in practice, an alternative approach is adopted. Rather than relying on the explicit knowledge of $p(\mathbf{X}, y)$, it is recognized that EQN allows for the simulation of detections under various smartphone geometries and earthquake parameters. As a result, there exists inherent variability in $\mathbf{X}$ and the number of triggers k_j.

To tackle the problem, a dataset $\mathcal{D} = (\mathbf{X}^{(1)}, y^{(1)}), \ldots, (\mathbf{X}^{(m)}, y^{(m)})$ is assumed to be available, which serves as a representative sample of $p(\mathbf{X}, y)$. Leveraging this dataset, the empirical risk can be defined as:

$$L(g \mid \mathcal{D}) = \frac{1}{m} \sum_{j=1}^{m} L\left((\mathbf{X}^{(j)}, y^{(j)}), g\right),$$

and the function g can be learned by solving the following minimization problem:

$$\hat{g} = \arg\min_{g} \; L(g \mid \mathcal{D}). \tag{4.18}$$

Notably, solving (4.18) is equivalent to solving two distinct minimization problems, depending on the specific emphasis placed on the probabilities of missed and false detections. In certain contexts, a missed detection can have a more severe negative impact than a false detection, while in other scenarios, the opposite holds true. Thus, two additional minimization problems for learning g can be formulated as follows:

$$\hat{g} = \arg\min_{g} \alpha(g) + \beta(g), \tag{4.19}$$

$$\hat{g} = \arg\min_{g} \alpha + \beta(g), \tag{4.20}$$

where it is explicitly acknowledged that the probabilities of missed and false detections are dependent on the function g. The choice of which minimization problem to address depends on the specific requirements and priorities of the EEW system in question.

4.3 Statistical parametric model and classification

In this section, we present a comprehensive statistical parametric model to describe the characteristics of a generic data point $\mathbf{X}$ in earthquake detection. The model encompasses the triggering time observed by a smartphone sensing an earthquake and incorporates various parameters and random components to capture the complex nature of seismic events.

The triggering time t_i for each smartphone is modeled as the sum of an expected triggering time t_i^* and a random component ϵ_i. This can be expressed mathematically as:

$$t_i = t_i^* + \epsilon_i, \quad i = 1, ..., k_j, \tag{4.21}$$

where k_j represents the number of triggers for the j-th earthquake detection. The random component ϵ_i is assumed to follow a normal distribution with mean 0 and variance σ_ϵ^2, denoted as $\epsilon_i \sim N(0, \sigma_\epsilon^2)$.

The expected triggering time t_i^* is determined by the seismic wave propagation and the distance between the hypocenter (the point of origin of the earthquake) and the smartphone location. It can be expressed as:

$$t_i^* = \frac{D_{i,H}}{v} + t_O, \tag{4.22}$$

where $D_{i,H}$ represents the distance between the hypocenter and the location of the smartphone, t_O represents the earthquake origin time, and v corresponds to the seismic wave speed. To obtain the value of v, the seismic wave speed, we refer to the AK135 model, also known as AK135F. This model is a widely recognized and respected global seismic model that provides a detailed representation of the seismic wave velocity structure within the Earth's interior. It offers comprehensive 1D radial profiles of seismic wave velocities at different depths. Notably, it encompasses essential seismic discontinuities, including the Mohorovičić discontinuity (Moho) and the lithosphere-asthenosphere boundary [140].

The distance $D_{i,H}$ is calculated using the formula:

$$D_{i,H} = \sqrt{d_E^2 + 4R(R - d_E)\sin^2\left(\frac{D_{i,E}}{2R}\right)}, \tag{4.23}$$

where $D_{i,E}$ denotes the distance between the epicenter (specified by latitude lat_E and longitude lon_E) and the location of the smartphone. The parameter d_E represents the depth of the earthquake, which lies within the range of 0 to 500 kilometers. The term R corresponds to the radius of the Earth (approximately 6371 kilometers).

It is important to note that the model assumes that all smartphones within a certain radius detect either the primary seismic wave with a speed of $v = 7.8$ km/s or the secondary wave with a speed of $v = 4.5$ km/s. This assumption is justified by the fact that earthquake detection is based on smartphones located within a relatively small area with a radius of 30 km. The random component ϵ_i accounts for the differences between the expected and observed triggering times. It captures various factors, including the delay in smartphone detection and potential deviations in the seismic wave velocity from the expected value.

Collectively, equations (4.21)-(4.23) define the statistical model f, and the parameter vector $\boldsymbol{\theta} = (lat_E, lon_E, d_E, t_O, \sigma_\epsilon^2) \in \Theta = \mathcal{S}^2 \times [0, 500] \times \mathbb{R} \times \mathbb{R}^+ \subset \mathbb{R}^6$ represents the space of model parameters. This parametric model provides a comprehensive framework for capturing the key characteristics and dynamics of earthquake detection based on the observed smartphone triggering times. By estimating the parameter vector $\boldsymbol{\theta}$, we can infer valuable insights about the earthquake events and improve our understanding of their spatiotemporal patterns.

4.3.1 Model estimation

The model estimation process relies on the maximum likelihood method, which allows us to find the most likely values for the model parameters based on the observed data. Specifically, for a generic earthquake (EQN) detection, the log-likelihood function is formulated using the joint probability distribution of the time differences $\Delta t_i = t_i - \hat{t_i^*}$, where t_i represents the observed triggering time and t_i^* denotes the expected triggering time. The log-likelihood function can be expressed as:

$$l(\boldsymbol{\theta}; \boldsymbol{X}) = -\frac{k}{2}\ln 2\pi - \frac{k}{2}\ln \sigma_\epsilon^2 - \frac{1}{2\sigma_\epsilon^2}\sum_{i=1}^{k}\Delta t_i^2, \tag{4.24}$$

where k represents the total number of observations. In this formulation, we assume that the time differences Δt_i are independent. This assumption is reasonable since smartphones do not share a common clock, and detection delays are influenced by various local factors, such as the smartphone's location, floor level, accelerometer sensitivity, and so on.

To obtain the maximum likelihood estimates of the model parameters lat_E, lon_E, d_E, and t_O, we aim to minimize the sum of squared time differences as follows:

$$\underset{lat_E, lon_E, d_E, t_O}{\arg\min} \sum_{i=1}^{k} \Delta t_i^2. \tag{4.25}$$

Due to the non-linearity of Equation (4.23), a closed-form solution for (4.25) cannot be obtained. Therefore, we employ numerical optimization techniques, specifically the BFGS Quasi-Newton method, to find the estimates. Multiple runs of the optimization algorithm are performed with random initial values for lat_E, lon_E, d_E, and t_O to avoid converging to local minima. The minimization in (4.25) is made possible by computing the expected triggering time t_i^* using Equations (4.22) and (4.23) for each proposed set of model parameter values, and then comparing it with the observed triggering time t_i.

Upon convergence, the BFGS Quasi-Newton method also provides the Hessian matrix, which represents the observed Fisher information matrix. Since the maximum likelihood estimates are obtained through a minimization problem, the inverse of the Hessian matrix yields the variance-covariance matrix of the three parameters (lat_E, lon_E, and d_E). From the variance-covariance matrix, standard errors for the parameter estimates can be easily computed.

Finally, the maximum likelihood estimate of the variance σ_ϵ^2 is obtained using Equation (4.26):

$$\hat{\sigma}\epsilon^2 = \frac{1}{k} \sum i = 1^k (\hat{\Delta} t_i - \hat{\mu})^2, \tag{4.26}$$

where $\hat{\Delta} t_i = t_i - \hat{t}_i^*$ is calculated by replacing the maximum likelihood estimates of latitude, longitude, and depth into Equations (4.22) and (4.23), and $\hat{\mu}$ represents the mean of the $\hat{\Delta} t_i$ values. By utilizing the maximum likelihood estimation method, we can obtain the most likely parameter values for the statistical model, allowing us to make accurate inferences about earthquake events and gain a better understanding of their spatiotemporal patterns.

4.3.2 EQN detection classification

Among all elements of $\boldsymbol{\theta}$, the parameter that carries information about how the EQN detection should be classified is σ_ϵ^2. Indeed, $\hat{\sigma}_\epsilon^2$ tends to be small when the earthquake is true (and triggering times follow the seismic wave dynamic) while $\hat{\sigma}_\epsilon^2$ tends to be large when the detection is not related to an earthquake event. This implies that $g(\boldsymbol{\theta})$ reduces to $g\left(\sigma_\epsilon^2\right)$.

In this work, g is chosen to be a statistical hypothesis test on σ_ϵ^2. The system of hypothesis is given by

$$\begin{cases} H_0 : \sigma_\epsilon^2 = \delta \\ H_1 : \sigma_\epsilon^2 > \delta. \end{cases} \tag{4.27}$$

Null hypothesis is rejected when the variance is higher than expected, namely when smartphone triggering times do not follow the propagation law of primary or of secondary seismic waves. As customary in statistical

hypothesis testing, the probability α is fixed and it represents the probability to reject the null hypothesis when it is actually true (namely it is the probability to miss a true earthquake).

The test statistic is

$$T = df\frac{{\sigma_\epsilon}^2}{\delta}, \tag{4.28}$$

which, under the null hypothesis, is distributed as a chi-square with $k-4$ degrees of freedom (df), where 4 is the number of estimated parameters in equation (4.25). The null hypothesis is rejected if $\hat{T} > q_{(1-\alpha),df}$, where $\hat{T}$ is obtained replacing σ_ϵ^2 with $\hat{\sigma}_\epsilon^2$ in (4.28), while $q_{(1-\alpha),df}$ is the $(1-\alpha)$-quantile of a chi-square distribution with df degrees of freedom, usually called critical value. In practice, an EQN detection is a true earthquake unless data bring enough evidence that the detection is actually false.

Since we do not know which seismic wave is detected by the smartphones, two models f are estimated: one with $v = 7.8$ km/s and one with $v = 4.5$ km/s in (4.22). This brings to two estimated values for σ_ϵ^2 and two hypothesis tests are implemented. The detection is classified as a false earthquake if the null hypothesis is rejected under both tests, otherwise the earthquake is classified as true.

Algorithm 1 EQN detection classification and earthquake parameters estimation

1: Initialisations: the number of times N the initial values of lat_E, lon_E, d_E and t_O are randomised when solving (4.25). The degrees of freedom df.
2: **for** $z = 1, \dots, N$ **do**
3: Sample lat_E, lon_E, d_E and t_O from uniform distributions.
4: Solve the minimization problem in (4.25).
5: Compute $\hat{\sigma}_\epsilon^2$ in (4.26).
6: Let $lat_{E,z}$, $lon_{E,z}$, $d_{E,z}$, $t_{O,z}$ and $\hat{\sigma}_{\epsilon,z}^2$ be the estimates for the z-th iteration.
7: Let $u_z = \sum_{i=1}^{k} \Delta t_i^2$ when the model parameters are $lat_{E,z}$, $lon_{E,z}$, $d_{E,z}$, $t_{O,z}$.
8: **end for**
9: Solve $z^* = \arg\min_z u_z$.
10: Maximum likelihood estimates of model parameters are lat_{E,z^*}, lon_{E,z^*}, d_{E,z^*}, t_{O,z^*} and $\hat{\sigma}_{\epsilon,z^*}^2$.
11: Compute the quantile $q_{1-\alpha,df}$ of a chi-square distribution with df degrees of freedom.
12: Set the vector $\boldsymbol{w}$ in (4.29) and the vector $\boldsymbol{\phi}$ in (4.30) replacing ${\hat{\sigma_\epsilon}}^2$ with $\hat{\sigma}_{\epsilon,z^*}^2$.
13: Compute $\boldsymbol{w}'\boldsymbol{\phi}$ and classify the EQN detection using the rule in (4.31).
14: Return the classification $\hat{y}$.
15: **if** $\hat{y} = -1$ **then**
16: Return the estimated earthquake parameters lat_{E,z^*}, lon_{E,z^*}, d_{E,z^*}, t_{O,z^*}.
17: **end if**

It is worth noting that the statistical hypothesis test is equivalent to a linear map. Indeed, setting

$$\boldsymbol{w} = \left(\frac{1}{\delta}, 1\right)', \tag{4.29}$$

$$\boldsymbol{\phi} = \left(df{\hat{\sigma}_\epsilon}^2, -q_{(1-\alpha),df}\right)', \tag{4.30}$$

then $g = \boldsymbol{w}'\boldsymbol{\phi}$ and the earthquake detection classification is based on the following rule

$$\hat{y} = \begin{cases} -1 & \text{if } \boldsymbol{w}'\boldsymbol{\phi} < 0 \\ 1 & \text{else.} \end{cases} \tag{4.31}$$

Finally, δ is obtained solving the problem

$$\hat{\delta} = \arg\min_{\delta} \alpha + \beta(\delta). \tag{4.32}$$

Algorithm 1 summarises the steps for classifying a EQN detection and for estimating the earthquake parameters in case the detection is classified as a true earthquake.

4.4 Simulation study

To solve the minimization problem in Equation (4.32) without a closed-form solution, we employ a Monte Carlo simulation. The simulation aims to generate a data set $\mathcal{D}$ and minimize the objective function in Equation (4.32). In this simulation, we simulate a total of 1000 true EQN detections and 1000 false EQN detections using the actual locations of 1000 smartphones in the EQN network in Lima, Peru.
To evaluate the performance of the EQN detection algorithm, we fix the probability of missed detection at $\alpha = 0.01$. We vary the parameter δ from 0.1 to 1.5 with a step size of 0.1. For each value of δ, we compute the quantity $\beta(\delta)$ by estimating the model f and implementing the hypothesis test in Equation (4.28) for all data points $\boldsymbol{X}^{(j)}$ in $\mathcal{D}$. Finally, we determine the value $\hat{\delta}$ that minimizes $\beta(\delta)$.

4.4.1 Simulation of true detections

For simulating a true earthquake, the following aspects are taken into account: the earthquake epicentre and depth, the arrival time of the seismic wave at the smartphone locations, the earthquake detectability by the smartphone and the error on the triggering time. Lastly, we account for the fact that smartphones may detect events unrelated to the earthquake.

The epicentre locations (lon_E, lat_E) is simulated uniformly inside the coordinates box $[-12.39°, -11.74°]$ for latitude and $[-77.17°, -76.66°]$ for longitude. The box encompasses the EQN network of Lima. On the other hand, the earthquake depth is simulated uniformly in the range $[0, 100]$ km independently of the earthquake epicentre.

The arrival time of the seismic wave at each smartphone location is simulated from (4.21) assuming $t_O = 0$ and $v = 7.8$ km/s. Only 70% of the smartphones are made triggering because of the earthquake. For these smartphones, the error on the triggering time is simulated from a zero mean normal distribution with variance $\sigma_\varepsilon^2 = 1.67$. Such variance guarantees that the 1-st and the 99-th percentiles of the error distribution are around -3 and 3 s, respectively, which are realistic values for an error on the triggering time.

Of the remaining 30% of smartphones which do not trigger, 6% are made triggering at random with a triggering time uniformly generated in the range $[0, 12]$ s. This implies that, when the earthquake is detected by the EQN detection algorithm, the list of triggering smartphones may include triggers unrelated with the earthquake dynamic.

Once the list of triggering smartphones is defined and sorted by triggering time, the EQN detection algorithm is applied to the list. The algorithm stops when the detection condition is satisfied, and the sub-list of triggers that concurred to the earthquake detection is given as output.

Figure 4.1 shows an example of a simulated true earthquake. Two separated regions can be visually identified, one with triggering smartphones (those that concurred to the detection), and another with non-triggering smartphone not yet reached by the seismic waves.

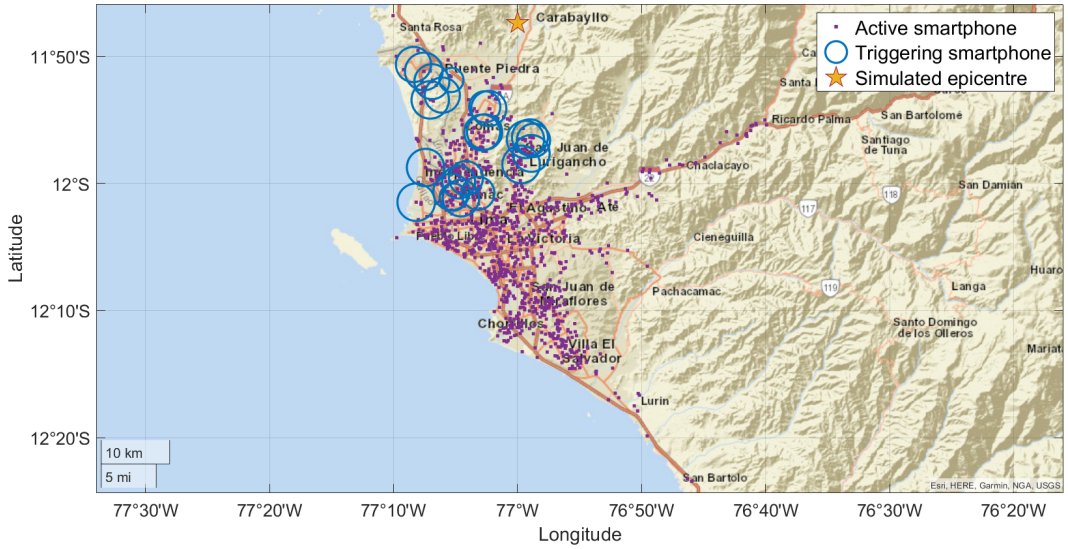


Figure 4.1: Simulated true earthquake detection based on the EQN smartphone network of Lima (Peru). The diameter of circles is proportional to the triggering time.

4.4.2 Simulation of false detections

To simulate a false detections, we assume that smartphones trigger at random with a triggering time which does not follow the law of the seismic wave propagation. Only 30% of the smartphones are made triggering and their the triggering time is uniformly sampled in the range $[0, 12]$ s.

Figure 5.12 shows an example of a simulated false EQN detection. Contrary to true earthquakes, no specific spatial pattern on the triggers is observed.

4.4.3 Simulation results

The minimization of (4.32) is attained when $\hat{\delta} = 0.6$ and β is found to be equal to 0.008 (conditionally on $\alpha = 0.01$). Figure 5.15 shows the empirical distributions of $\hat{\sigma}_\epsilon^2$ for both true and false simulated EQN detections when $\delta = 0.6$ and $\alpha = 0.01$. Although the detection classification is based on the hypothesis test (and not directly on $\hat{\sigma}_\epsilon^2$), the overlapping between distributions suggests that classification errors are possible.

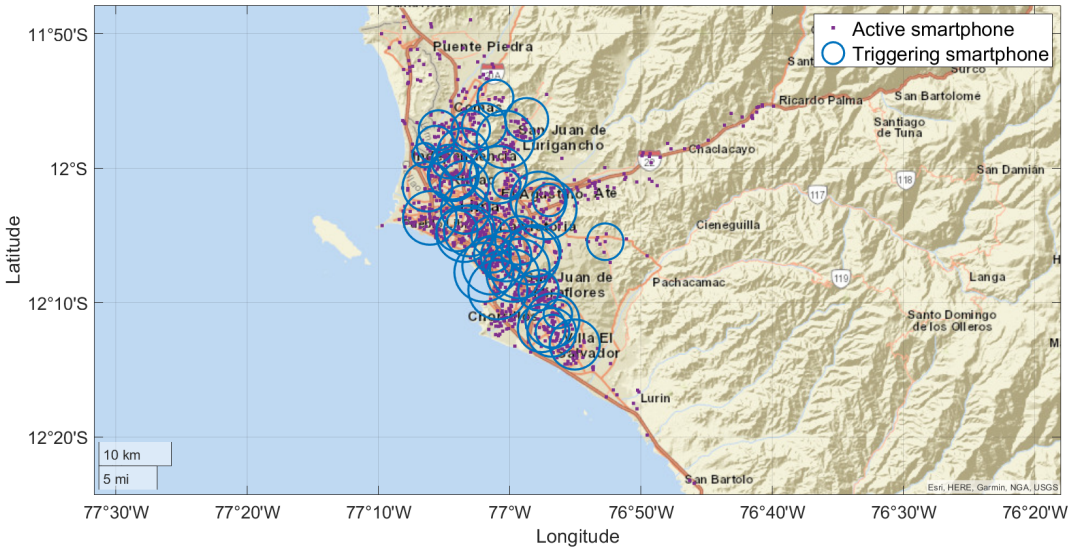


Figure 4.2: Simulated false earthquake detection based on the EQN smartphone network of Lima (Peru). The diameter of circles is proportional to the triggering time.

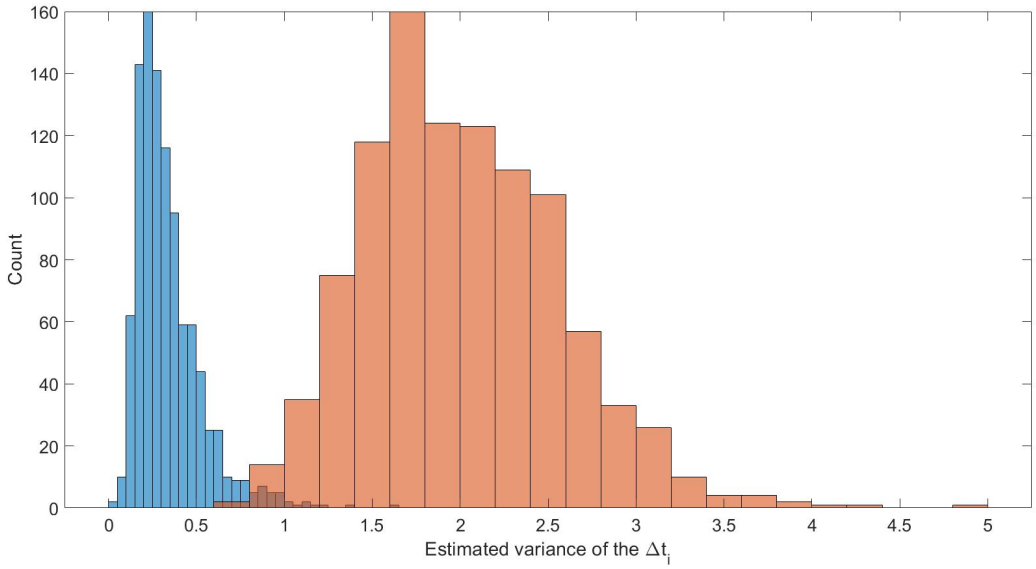


Figure 4.3: Empirical distributions of $\hat{\sigma}_\epsilon^2$ under simulated true detections (blue histogram) and under simulated false detections (red histogram) when $\delta = 0.6$ and $\alpha = 0.01$.

A by-product of detection classification are the estimates on the earthquake parameters. Figure 4.4 shows box-plots of errors on earthquake epicentre and depth. Both errors have a median of around 18 km, suggesting that, along with the detection classification (true/false), the model output can be exploited to provide preliminary estimates of the earthquake parameters.

4.5 Real data example

The methodology developed in this work is applied on a true and on a false detections detection made by EQN. As a true earthquake, the event occurred near Genova (Italy) on October 4, 2022, at 21:41:10.5 UTC

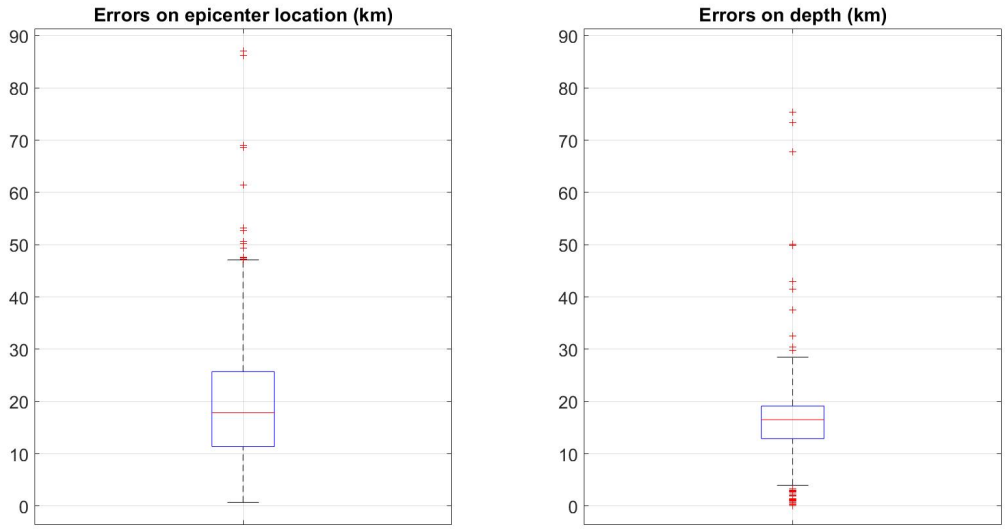


Figure 4.4: Box plot of the errors on epicentre location (lat_E, lon_E) (left) and box plot of the errors on earthquake depth d_E (right) for the 1000 simulated true earthquake detections.

is considered. Figure 4.5 depicts the triggering smartphones ($n = 21$), while estimation and classification results are reported in Table 4.1 for $v = 7.8$ and $v = 4.5$ km/s, respectively.

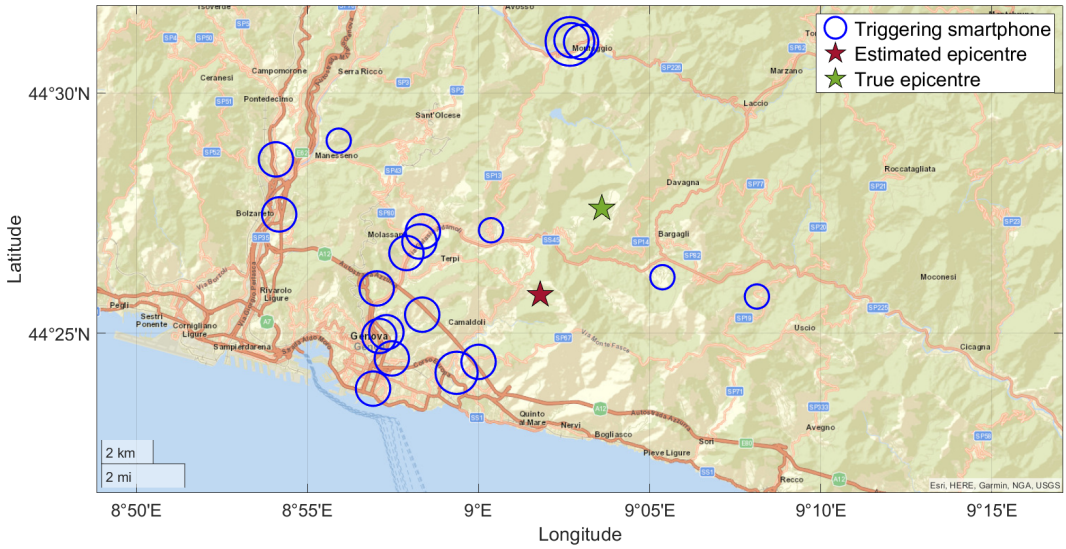


Figure 4.5: EQN triggers for the earthquake occurred on October 4, 2022 close to Genoa (Italy). The diameter of circles is proportional to the triggering time.

For both seismic wave velocities, we can observe that latitude and longitude are accurately estimated, while the error on depth is not negligible. Nonetheless, the true values are within the 99% confidence intervals evaluated from the standard errors on the model parameters. Additionally, the earthquake is classified as true under both velocities since both observed test statistics are lower than the test critical value. This happens because triggers are close to the epicentre, and primary and secondary seismic waves are nearly concurrent.

The estimation and classification results were obtained in less than 1 s using an Intel(R) Core(TM)

	Real	$v = 7.8$ km/s Estimated	Error	$v = 4.5$ km/s Estimated	Error
Latitude (°)	44.46	44.43 [44.38, 44.47]	0.03	44.43 [44.40, 44.45]	0.02
Longitude (°)	9.06	9.06 [9.01, 9.11]	0.00	9.03 [9.03, 9.09]	0.03
Depth (km)	8.00	0.01 [0.00, 8.36]	7.99	0.01 [0.00, 3.80]	7.99
Estimated variance	-	0.57	-	1.03	-
Test statistic value	-	17.18	-	31.02	-
Critical value	-	34.80	-	34.80	-
Classification	-	True earthquake	-	True earthquake	-

Table 4.1: Detection classification and earthquake parameters estimation for the EQN detection near Genova (Italy) assuming v equal to 7.8 and 4.5 km/s. The number of triggering smartphones is $n = 21$. In brackets, 99% confidence intervals. Real earthquake parameters are taken from the website of the European-Mediterranean Seismological Centre (`www.emsc-csem.org`).

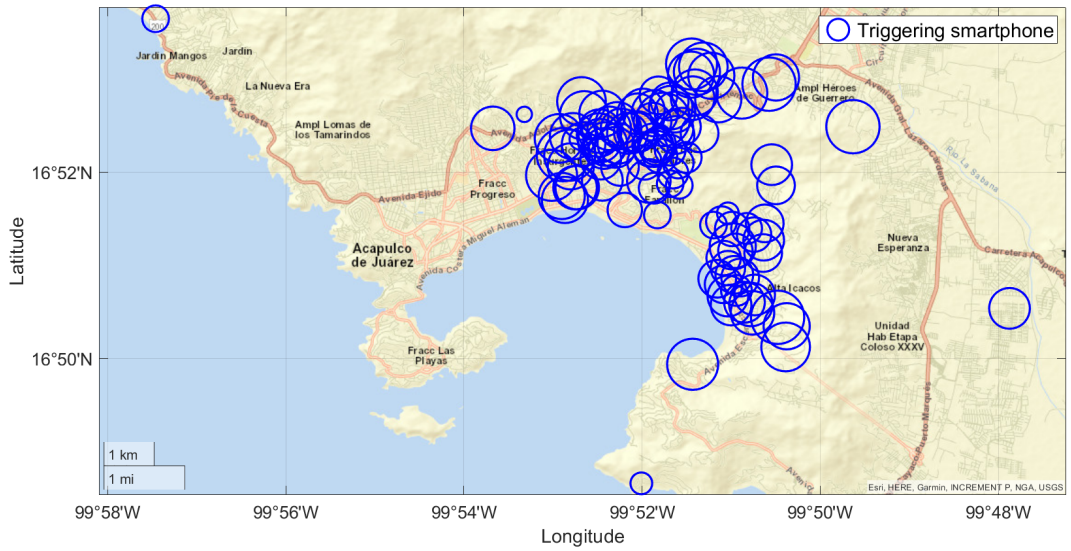


Figure 4.6: Triggers for the false EQN detection occurred on September 25, 2022, close to Acapulco (Mexico). The diameter of circles is proportional to the triggering time.

i7-9750H CPU @2.60GHz, suggesting that the approach can be adopted for real-time applications.

Figure 4.6 shows, instead, the $n = 108$ triggers of a false detection occurred near Acapulco (Mexico) on September 25, 2022, at 09:55:45 UTC. In this case, the computed test statistics are 1039.7 and 1026.0 for $v = 4.5$ and 7.8 km/s, respectively, while the critical value is 141.62. H_0 is rejected in both cases and the detection is claimed as false. In this particular case, the detection was caused by a strong lightning bolt. The speed of sound, however, is around 0.3 km/s, a value much smaller than the speed of primary and secondary

seismic waves.

Furthermore, we conducted several tests on a sample of 207 real earthquakes extracted from the EQN system, with the aim of accurately estimating the earthquake's position and classifying it as true or false. To achieve this, we employed two distinct approaches, each yielding valuable insights.

In the first approach, we adopted a randomized technique wherein 100 initial values were generated for each earthquake, centered around the EQN's current estimation. Subsequently, from this ensemble of initial values, the best result was selected for analysis. The outcomes were striking, as we observed that this statistical approach demonstrated superior performance in estimating the median initial time t_0 when compared to EQN's estimation. On the contrary, EQN exhibited better accuracy in predicting the actual location of the earthquake event. To visually capture and present these findings, Figures in 4.7 were devised, presenting histograms depicting the discrepancies in time and distance between the real earthquake event and the predictions made by both the EQN and the statistical hypothesis.

Regarding the classification of earthquakes as true or false using the simulation-derived variance threshold of 0.6, the results revealed that a considerable number of earthquakes, specifically 56 out of the 207 tested, were classified as false positives. This indicates that the statistical approach might be overly conservative in its identification of true seismic events.

On the other hand, when employing a variance threshold of 3.5 for a 99% quantile derived from the same set of 207 earthquakes, a substantial difference was observed compared to the threshold of 0.6. The variance threshold of 3.5 allowed for a more lenient classification of earthquakes as true positives. This variation in the thresholds can be attributed to certain factors that were not implemented during the simulation process. These factors, though significant for a comprehensive understanding of earthquake patterns, might have been omitted due to their complexity and challenges in implementation.

Nonetheless, one crucial advantage of this statistical approach is its capability to classify earthquakes as true positives in quasi-real time. This rapid and almost real-time identification of seismic events holds immense value for earthquake early warning systems and seismic hazard assessments. By promptly identifying genuine seismic events, this approach has the potential to provide timely warnings to vulnerable regions, helping communities and authorities undertake appropriate preparedness and response measures.

In the second approach, we took a different path by keeping the initial values fixed at EQN's predictions while enforcing a constraint on the depth to ensure it did not exceed 10 km. The results of this approach were interesting. Figures 4.8 vividly portray the improvements achieved through this constrained scenario, particularly evidenced by the increased number of earthquakes accurately classified as true seismic events. Specifically, this approach classifies approximately 49% of the samples as true earthquakes, showcasing an improvement in the location estimation while observing a slight deterioration in the estimation of the origin

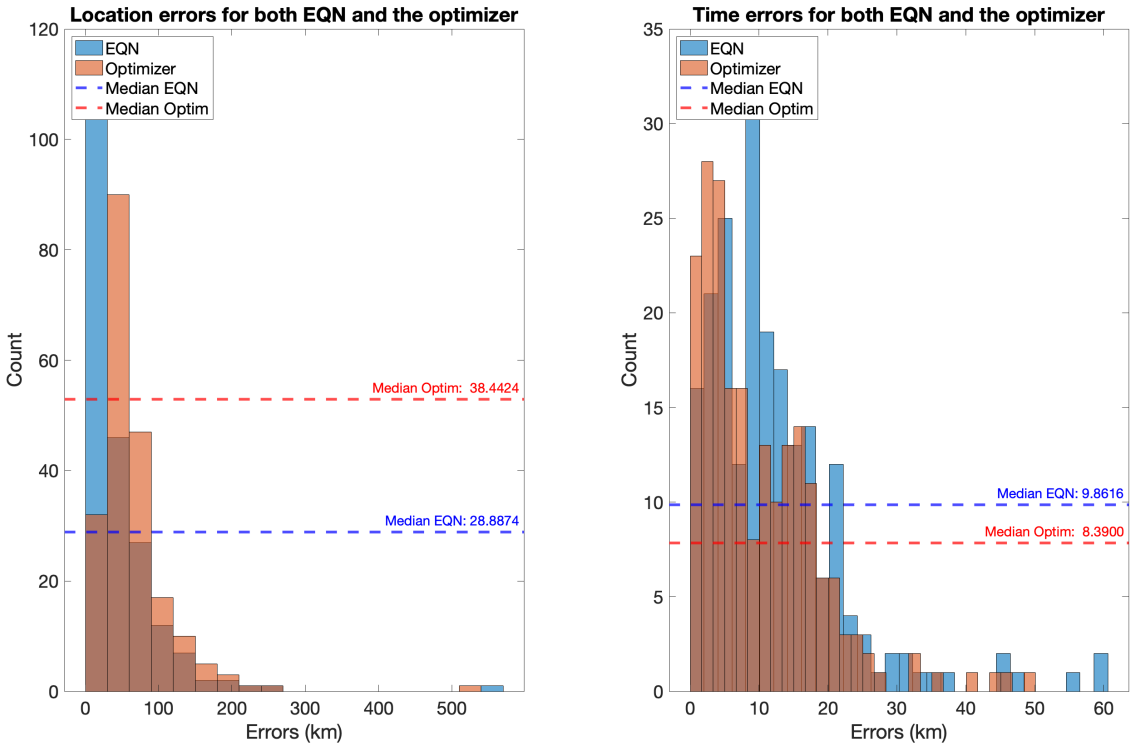


Figure 4.7: In this figure, we present a comparison between the Earthquake Query Network (EQN) and a statistical hypothesis for earthquake prediction for the first approach. The right histogram shows the discrepancy in time between the actual earthquake occurrence and the respective predictions made by EQN (in blue) and the statistical hypothesis (in red). Similarly, the left histogram displays the variation in distance between the actual earthquake location and the corresponding predictions generated by EQN (in blue) and the statistical hypothesis (in red). This analysis provides valuable insights into the performance and accuracy of the two approaches in estimating earthquake timings and locations.

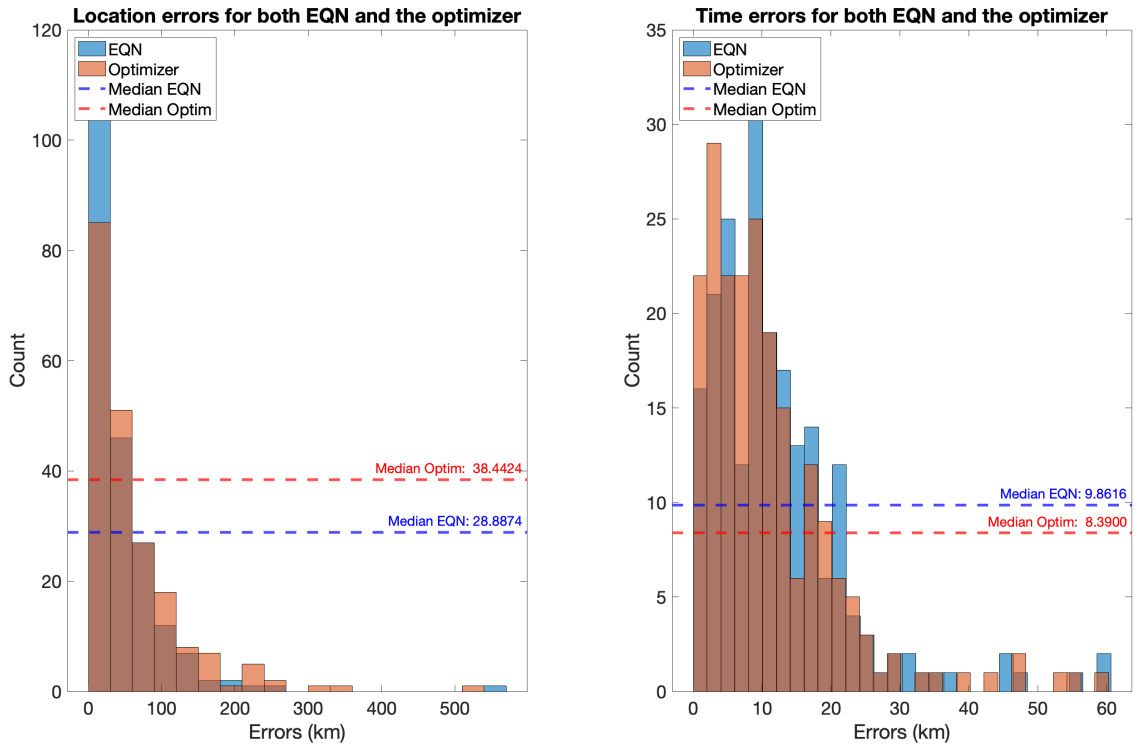


Figure 4.8: In this figure, we present a comparison between the Earthquake Query Network (EQN) and a statistical hypothesis for earthquake prediction for the second approach. The right histogram shows the discrepancy in time between the actual earthquake occurrence and the respective predictions made by EQN (in blue) and the statistical hypothesis (in red). Similarly, the left histogram displays the variation in distance between the actual earthquake location and the corresponding predictions generated by EQN (in blue) and the statistical hypothesis (in red). This analysis provides valuable insights into the performance and accuracy of the two approaches in estimating earthquake timings and locations. The median for the times error for the statistical approach is 9.3900.

time, albeit still lower than that of EQN.

However, as with any scientific study, it is crucial to acknowledge the inherent limitations and potential biases that might affect the results. A notable constraint in our evaluation process was the lack of real false alarms, attributable to the paucity of available data within the EQN system. While our primary focus revolved around assessing the system's detection performance and its ability to accurately associate earthquakes, the absence of real false alarms could impact the overall reliability of this statistical approach when applied to real-world scenarios. Addressing this limitation would entail incorporating real-world false alarm data in future testing, significantly enhancing our understanding of the system's capacity to distinguish between non-earthquake vibrations and genuine seismic events. Consequently, this would bolster the effectiveness and trustworthiness of the statistical approach in practical earthquake early warning situations.

4.6 Discussion and conclusion

The methodology developed in this section allows to classify detections made by smartphone-based earthquake early warning systems between true (related to a real earthquake) and false. This is done analysing the

information content of the smartphone triggers that contributed to the detection.

With respect to classic classification problems, the data point describing the triggers has a varying dimension which depends on the smartphone network geometry. The proposed solution is based on two steps. First, a statistical parametric model is used to convert the data point into a parameter vector with fixed (and small) dimension. Second, a hypothesis test is implemented for classification.

While we do not claim our choices of f and g to be optimal, both steps are based on well established statistical methods. With respect to the specific choice of g, it is worth discussing that a simpler alternative is the linear map $g^* = \boldsymbol{\delta}'\boldsymbol{\phi}$, with $\boldsymbol{\delta} = (\delta, 1)'$ and $\boldsymbol{\phi} = \left(1, -\hat{\sigma}_\epsilon^2\right)'$. In this case, the classification is based on the more intuitive comparison $\hat{\sigma}_\epsilon^2 \lessgtr \delta$. This simpler solution, however, does not take into account either the actual number of triggers for the specific detection (10 or 1000 makes a difference on the uncertainty of $\hat{\sigma}_\epsilon^2$) nor the fact that the distribution of σ_ϵ^2 is known under the null hypothesis (that the detection is related to a true earthquake). Using hypothesis testing, we are thus able to retain part of the information which is lost when $\boldsymbol{X}$ is synthesized with $\boldsymbol{\theta}$.

4.6.1 Conclusion

Classification and earthquake parameters estimation are performed in near real-time, making the statistical methodology suitable to be implemented in operational systems. On the other hand, the methodology does not fully exploit the information available on the EQN system. Specifically, the modelling is only on the triggering smartphones, while the active non-triggering smartphones are ignored. Knowing, at the EQN detection time, which smartphones have not (yet) triggered may better constraint epicentre and depth, thus improving their estimates.

Additionally, for a EEWS like EQN that works globally, it would be important to study if the data set $\mathcal{D}$ generated by the Monte Carlo simulation is actually a representative sample of $p(\boldsymbol{X}, y)$. If not, the observed α and β probabilities might deviate from the expected ones.

Finally, a limit of the approach proposed by this section is that the statistical methodology is applied downstream of EQN detections. Ideally, the detection, the classification and the earthquake parameter estimation problems should be jointly addressed in a unified approach. In this regard, the vast literature on wireless sensor networks may help propose a solution under the real-time constraint. These open problems, along with the estimation of the earthquake magnitude, will be the focus of future works.

Chapter 5

Deep-learning method

5.1 Artificial neural networks and deep learning

Neural networks are a fascinating field of study that draws inspiration from the complex and interconnected nature of neurons in the brain. Artificial neural networks (ANNs) attempt to replicate the functionality of biological neurons, albeit at a more abstract and computational level. To comprehend the workings of ANNs, it is essential to delve into the fundamental components and processes that underpin both biological and artificial neural systems.

In the human brain, neurons serve as the building blocks of the nervous system. They are highly interconnected elements responsible for transmitting and processing information. Neurons consist of several crucial elements: dendrites, the cell body (also known as the soma), and an axon. Dendrites act as branches that receive electrical signals from other neurons. These signals are then transmitted to the cell body, where they are summed and evaluated. If the combined input signals surpass a certain threshold, an electrical signal is generated and propagated along the axon, a long fiber-like structure, to other connected neurons. The points of contact between the axon of one neuron and the dendrites of another are known as synapses. It is at these synapses that information is exchanged through the transmission of chemical or electrical signals.

Artificial neural networks simulate this complex interconnectedness by employing nodes as the equivalent of biological neurons. Nodes receive input signals and perform computations on them. In ANNs, synapses are represented by weights, which assign importance or influence to each input signal. Each input is multiplied by its corresponding weight before being passed to the node. These weighted inputs are then combined through arithmetic addition to yield an activation value for the node. This activation value is subsequently compared to a predefined threshold to determine whether the node should activate and propagate its signal to the next layer of nodes.

ANNs learn by adjusting the weights assigned to synapses, a process often referred to as training. During training, an artificial neural network receives a set of input data along with their corresponding output labels, forming training pairs. In a supervised learning paradigm, the network uses this training data, which acts as external stimuli, to learn the underlying function that connects the inputs to the outputs. By making

predictions about the output labels based on the input data, the network attempts to approximate the correct outputs. The accuracy of these predictions is then assessed by comparing them to the provided output labels. Any discrepancies between the predicted and actual outputs are used to calculate an error, which quantifies the network's performance.

The objective of training an ANN is to minimize this error by iteratively adjusting the weights. Through a process called backpropagation, the error is propagated backward through the network, and the weights are updated accordingly. The adjustment of weights aims to alter the network's predictions in a mathematically determined manner, so that they align more closely with the desired output labels. This iterative weight adjustment continues until the network achieves satisfactory performance on the training data.

It is important to emphasize that the training process allows an artificial neural network to generalize its knowledge from the seen or training data to unseen or test data. The network aims to learn underlying patterns and relationships within the training data, so it can make accurate predictions on new, previously unseen data. The potential and utility of neural networks extend beyond their ability to learn from data. Deep learning, a subfield of machine learning, explores the power and capabilities of neural networks with multiple layers. The specific way in which nodes are interconnected within a neural network defines its architecture. The architecture plays a vital role in determining the network's ability to capture and represent complex relationships in the data. However, it is important to note that even the most sophisticated architecture cannot compensate for insufficient or poor-quality data. For an artificial neural network to effectively learn complex, highly non-linear patterns present in the data, an ample amount of diverse and relevant data must be available for training.

5.1.1 Single-input neuron

Neural networks offer a powerful framework for modeling complex relationships and learning from data. At the heart of these networks lies the single-unit neuron, which serves as a foundational building block. While simple in structure, the single-unit neuron provides essential insights into the workings of more sophisticated models.

Figure 5.1 depicts a single-input neuron, where the input variable x is multiplied by the weight parameter w and summed with an offset known as bias (distinct from the statistical concept of bias). The resulting sum is denoted as z and serves as the input to an activation function, also referred to as a transfer function. In biological terms, the weight parameter corresponds to the strength of a synapse, the combination of the summer and the activation function represents the cell body, and the output of the neuron is represented by the axon, denoted as a.

Mathematically, the output a of the neuron can be calculated using the following equations:

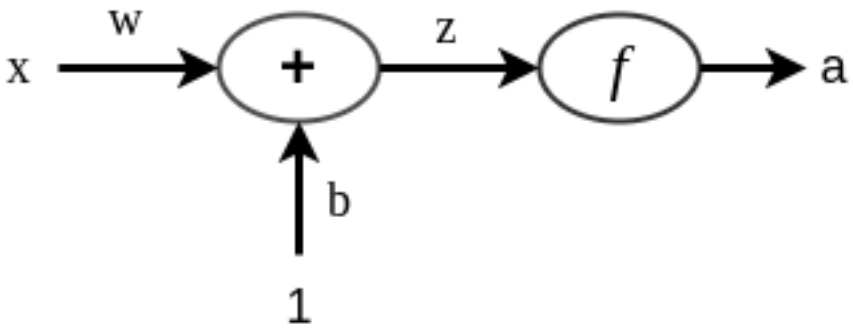


Figure 5.1: Single-input neuron.

$$\begin{cases} z = wx + b, \\ a = f(z) \end{cases} \tag{5.1}$$

In Equation 5.1, the input x is multiplied by the weight w and summed with the bias term b to obtain z. The activation function $f(\cdot)$ is then applied to z to generate the output a of the neuron.

Alternatively, we can consider the bias as a weight with an input value of 1. This allows us to rewrite Equation 5.1 in a more concise form:

$$\begin{cases} z = \mathbf{w}^T\mathbf{x}, \\ a = f(z) \end{cases} \tag{5.2}$$

Here, $\mathbf{w}$ is a vector of weights $[b; w]^T$, and $\mathbf{x}$ is a vector of inputs $[1; x]$. All other variables are scalar. The adjustable parameters of this model are the weights w and b, while the choice of activation function is a design consideration. The specific activation function employed has a significant impact on the output of an artificial neuron, as we will explore in later sections.

It is important to highlight that neural networks are not limited to single-unit neurons. They are composed of interconnected layers of neurons, forming a complex network architecture. Each neuron in a layer receives inputs from the previous layer, performs its computations using weights and biases, and passes the output through the activation function. This process continues through multiple layers until the final output layer produces the overall output of the network.

By adjusting the weights and biases of the neurons, neural networks have the ability to learn and adapt to input data. This learning process typically involves providing the network with labeled training data, a technique known as supervised learning. The network then compares its predictions with the actual labels, and the resulting error is used to update the weights and biases. This iterative procedure, commonly referred

to as backpropagation, enables the network to refine its predictions and improve its performance over time.

It is worth noting that neural networks can exhibit far greater complexity than the single-unit neuron architecture discussed here. Deep neural networks, for example, consist of multiple hidden layers, allowing them to learn intricate and hierarchical representations of data. These networks have achieved remarkable success in a wide range of applications, including image recognition, natural language processing, and speech recognition. [141].

5.1.2 Threshold logic unit

When it comes to understanding general terminology, the single-input neuron serves as the simplest building block. However, if we aim to construct more complex models, such as the perceptron, the Threshold Logic Unit (TLU) emerges as the fundamental component. TLUs possess only one computational unit, similar to the single-unit neuron, but they offer the advantage of accommodating multiple inputs, enabling learning from a greater number of examples.

Figure 5.2 illustrates the structure of a TLU. Each input x_i is associated with a weight $w_{1,i}$. In the weight index, the value of 1 signifies the computational unit, as there is only one in this case, while i represents the input index. In the TLU, there are n inputs, allowing for a more extensive learning capability.

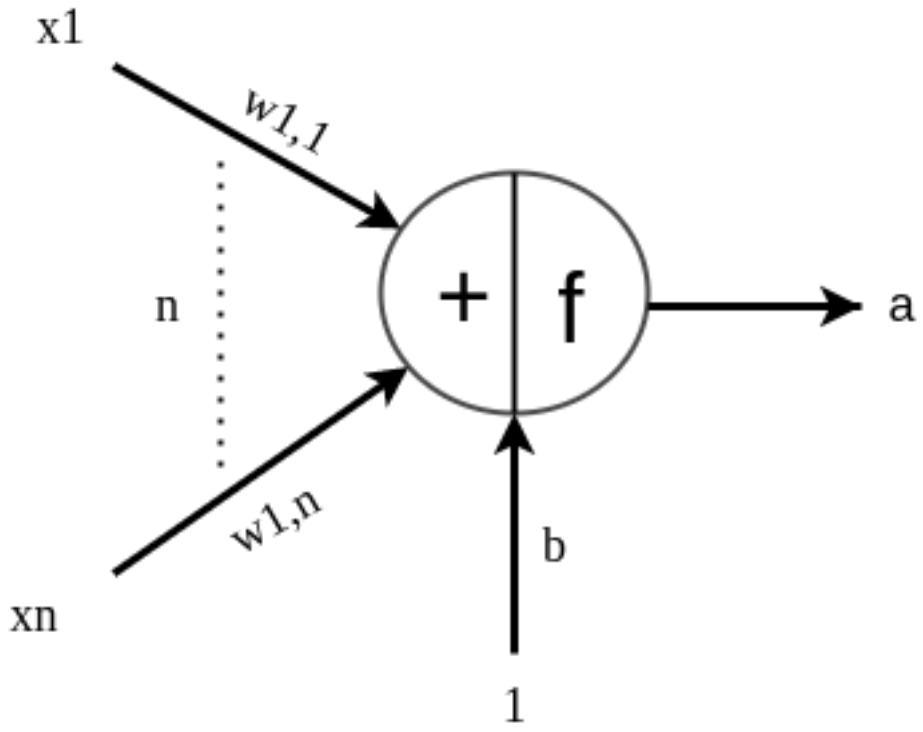


Figure 5.2: Linear Threshold Unit.

The observations made in Equation 5.2 remain valid in this context. However, there are some differences to note. The weight vector $\mathbf{w}$ now belongs to $\mathbb{R}^{1\times(n+1)}$, and the input matrix $\mathbf{X}$ belongs to $\mathbb{R}^{m\times(n+1)}$, assuming we have m training examples. This results in the updated Equation 5.3.

$$\begin{cases} z = \mathbf{w}^T\mathbf{X}, \\ a = f(z) \end{cases} \tag{5.3}$$

In Equation 5.3, z represents the weighted sum of inputs, where $\mathbf{w}^T$ denotes the transpose of the weight vector $\mathbf{w}$, and $\mathbf{X}$ is the input matrix. The activation function $f(\cdot)$ is typically the Heaviside step function or the sign function, which acts as the TLU's activation mechanism.

The adjustable weight vector $\mathbf{w}$ plays a crucial role in the TLU's functionality, as it determines the strength of the connections between inputs and the computational unit. The process of adjusting these weights is discussed in the subsequent sections, as it forms a key aspect of the learning process in neural networks.

TLUs, with their ability to process multiple inputs, provide a foundation for constructing more sophisticated models. They offer enhanced learning capabilities, enabling the exploration of more complex patterns and relationships in the data. This is achieved by adjusting the weights and employing appropriate activation functions.

5.1.3 Perceptron

The Perceptron, introduced by Rosenblatt in 1958, is a foundational concept in neural networks ([142, 143]). It consists of a single layer of Threshold Logic Units (TLUs), with each neuron connected to all the inputs. The general architecture of a perceptron with n inputs and r outputs is depicted in Figure 5.3. The equation 5.3 that describes the behavior of TLUs remains valid, but with the distinction that the output is now an r-dimensional vector, leading to equation 5.4. Additionally, it is important to note that since the perceptron is composed of TLUs, the activation functions f are typically the Heaviside step function or the sign function.

The weight vector $\mathbf{w}$ in the perceptron model is adjusted using the perceptron learning rule. This learning rule, inspired by Donald Hebb's work, suggests that the connection between two biological neurons becomes stronger when one neuron frequently activates the other ([144, 145]). In the perceptron learning rule, the weights are updated based on the errors observed for each example in the training data. The connection weights from the inputs that could have contributed to the correct prediction are strengthened for every output neuron that made an incorrect prediction.

The perceptron is commonly employed in classification settings, which involve supervised learning problems where the output labels belong to distinct categories. Examples include classifying between cats and dogs, distinguishing between men and women, and many others. A popular example is the classification of handwritten digits in the MNIST dataset. It has been demonstrated that the perceptron will always converge

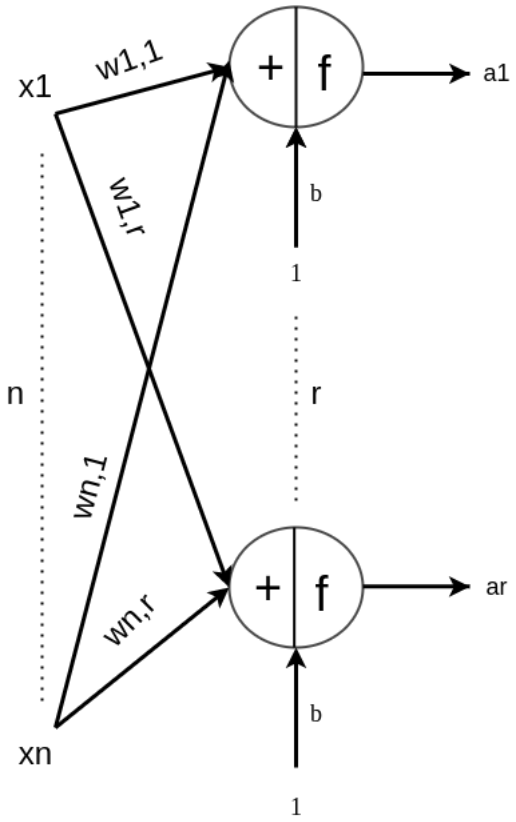


Figure 5.3: A visual representation of a perceptron.

to weights that ensure the correct classification of two distinct labels in a finite number of iterations, given that these weights exist and the two output labels are linearly separable ([146, 147]).

The behavior of the perceptron can be described using the following equations:

$$\begin{cases} \mathbf{z} = \mathbf{w}^T \mathbf{X}, \\ \mathbf{a} = f(\mathbf{z}) \end{cases} \tag{5.4}$$

In Equation 5.4, $\mathbf{z}$ represents the weighted sum of inputs, where $\mathbf{w}^T$ denotes the transpose of the weight vector $\mathbf{w}$, and $\mathbf{X}$ is the input matrix. The activation function $f(\cdot)$ is applied element-wise to $\mathbf{z}$ to obtain the output vector $\mathbf{a}$. Each element of $\mathbf{a}$ corresponds to the activation of an individual neuron in the perceptron.

The perceptron's ability to learn and adjust its weights based on training data allows it to perform classification tasks effectively. By iteratively updating the weights, the perceptron improves its ability to correctly classify inputs into distinct categories.

5.1.4 Fully connected network

The perceptron, although powerful for handling linearly separable problems, falls short when confronted with tasks where distinct classes are not easily separated along a linear boundary, such as the XOR problem ([148]). To tackle more complex problems, the multi-layer perceptron (MLP) was developed, introducing a

more sophisticated architecture.

An MLP consists of multiple layers of perceptrons, with each layer receiving the outputs from the preceding layer as inputs. The final layer in an MLP is the output layer. Figure 5.4 provides a general representation of an MLP's architecture. Considering the input layer as layer 0, we can observe that each neuron in layer $l-1$ is fully connected to every neuron in the next layer l through a set of weights indexed as $w_{i,o,l}$. Here, i represents the index of the source neuron, and o represents the index of the destination neuron. The notation $w_{i,o,l}$ is general and accounts for the varying number of source and destination neurons between layer $l-1$ and layer l. Each neuron's output can be denoted as $a_{i,l}$, where i corresponds to the index of the neuron in layer l. We can also rewrite each input x_i as $a_{i,0}$. This perspective allows us to conceptualize the weights as being attached to the outputs of the computational units in layer $l-1$. For example, if we assume there are L layers in the perceptron, the first layer consists of r computational units, the $(L-1)$-th layer is composed of S computational units, and the final layer (output layer L) contains t computational units. Figure 5.4 provides a visual representation of the notation framework for the various layers.

Importantly, it should be noted that each layer has its own activation function, denoted as f_L.

Equation 5.4 remains applicable in this context, with the distinction that the activation function f is now a composition function, given by $f = f_L(f_{L-1}(\dots f_1(\mathbf{z})))$. This composition allows for the sequential transformation of the input through each layer, ultimately producing the output vector $\mathbf{a}$.

$$\mathbf{a} = f(\mathbf{z}) \tag{5.5}$$

To adjust the weights in each layer, the backpropagation algorithm is commonly used. This algorithm, described in more detail in the training section, operates as a gradient-based optimization method ([149, 150]). The backpropagation algorithm involves computing predictions (forward pass) for each training instance, calculating the error, performing a reverse pass through each layer to determine the contribution of each connection's error, and finally, making slight modifications to the connection weights to decrease the overall error (gradient descent step).

In summary, the multi-layer perceptron (MLP) addresses the limitations of the perceptron by introducing a more complex architecture capable of handling nonlinearly separable problems. Through the sequential transformation of inputs across multiple layers of perceptrons, the MLP enables the representation of intricate patterns and relationships. The weights within each layer are adjusted using the backpropagation algorithm, which leverages gradient-based optimization to iteratively improve the model's performance.

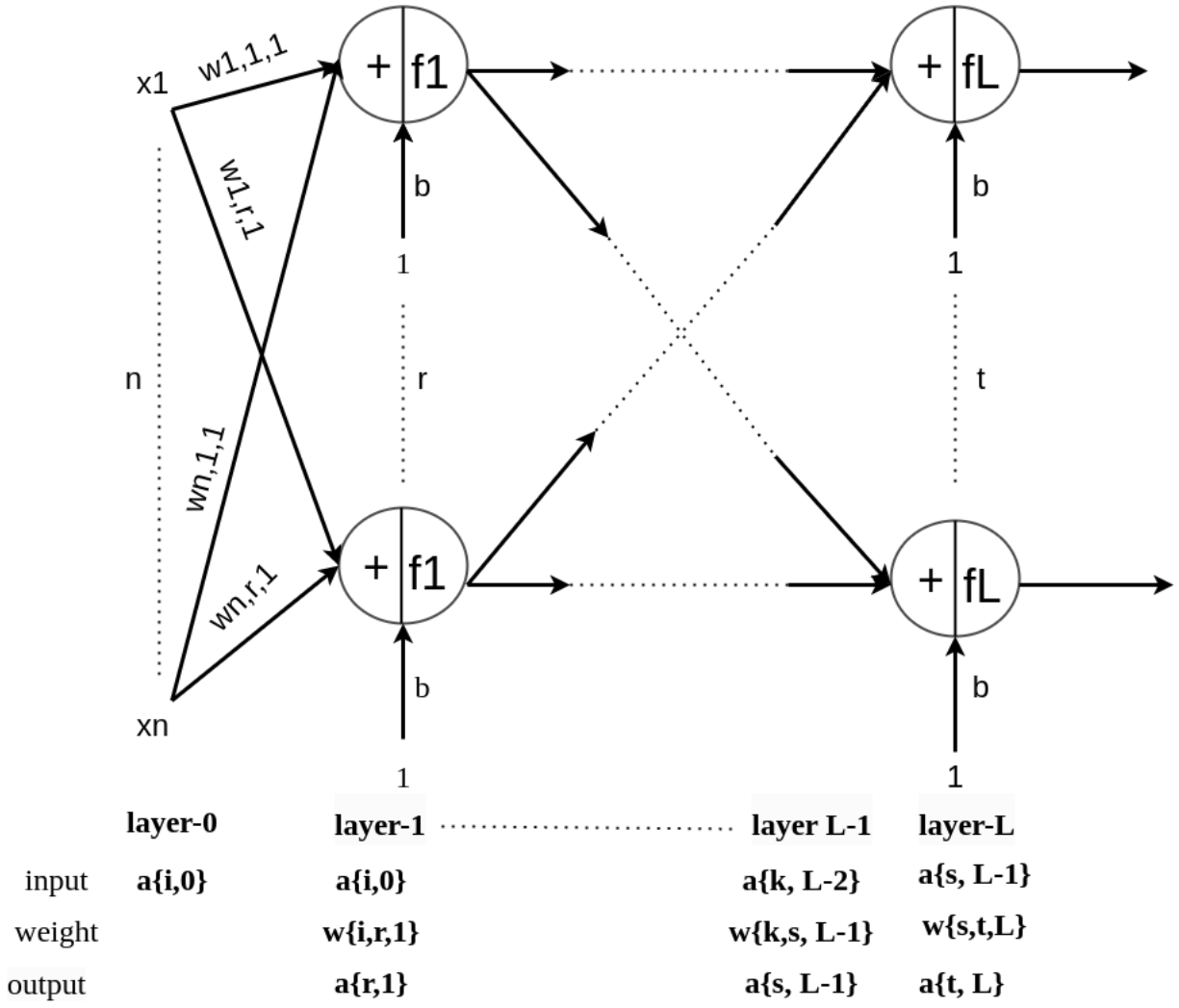


Figure 5.4: A visual representation of a multi-layer perceptron with a notation framework for the various layers.

5.1.5 Activation functions

Activation functions are pivotal components in neural networks, significantly impacting the behavior of individual neurons and their capacity to capture intricate non-linear patterns within the input space. They play a crucial role in enabling neural networks to model and learn from complex data that linear parametric models may struggle to represent effectively. In this section, we will delve into a comprehensive exploration of several commonly employed activation functions, thoroughly examining their unique characteristics and applications.

Heaviside activation function

The hard limit as the name suggest, set the output to 0 if the inputs is less than 0, and to 1 otherwise. This activation function is used in the perceptron for classifying into 2 distinct categories.

$$f(z) = \begin{cases} 0, & \text{if } z < 0 \\ 1, & \text{if } z >= 0 \end{cases} \tag{5.6}$$

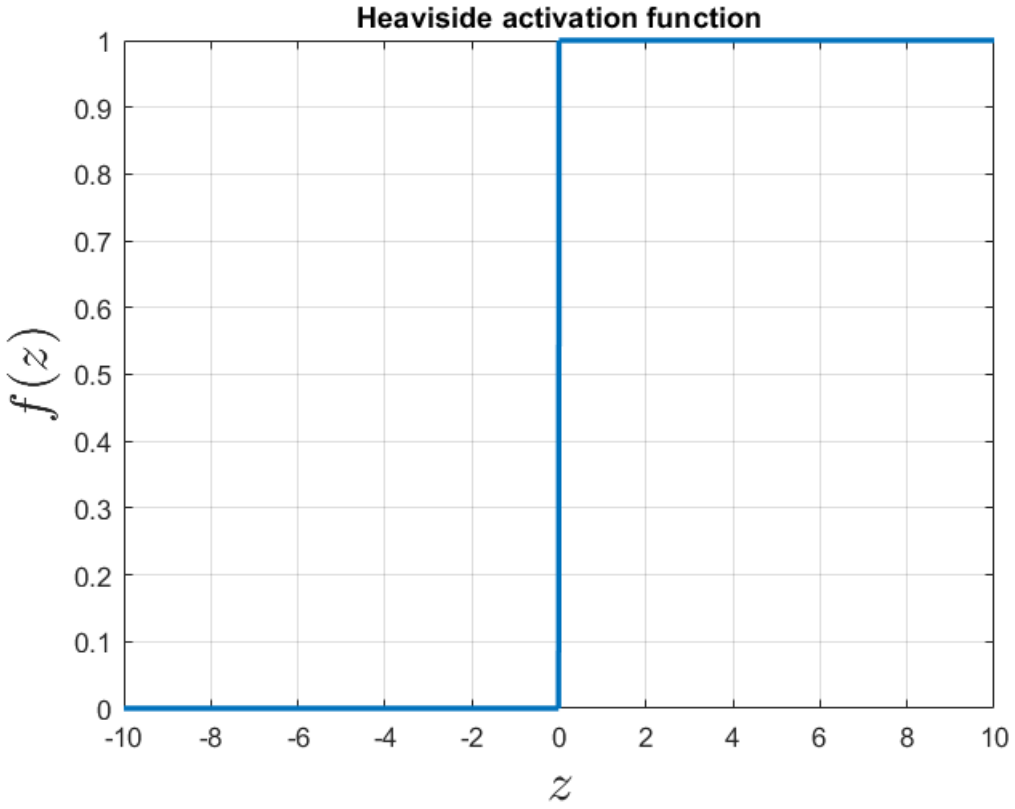


Figure 5.5: Heaviside activation function.

This activation function is commonly used in the perceptron algorithm for binary classification tasks. The decision boundary is set at 0, and the hard limit activation function assigns a class label of 0 or 1 based on whether the input is below or above the threshold, respectively.

The hard limit activation function is depicted in Figure 5.8, where the output abruptly changes from 0 to 1 at the threshold. It is a step-like function that exhibits discontinuity at the threshold point. Due to its discontinuous nature, it is not differentiable, which can pose challenges for optimization algorithms that rely on derivatives.

The hard limit activation function is simple and interpretable, making it suitable for problems that require a clear decision boundary. However, its lack of differentiability limits its applicability in more complex neural network architectures that rely on gradient-based optimization algorithms.

Linear activation function

The linear activation function is a fundamental component in neural networks and is commonly used in various machine learning tasks. It is a simple function that preserves the input value without any transformation or non-linear mapping. In this section, we will explore the characteristics and applications of the linear activation function.

The linear activation function, as the name suggests, maintains a linear relationship between the input and output values. It is defined as:

$$f(z) = z \tag{5.7}$$

This means that the output of the function is equal to the input value. The linearity of this activation function makes it particularly useful in regression settings, where the goal is to predict continuous real-valued outputs instead of discrete labels. By using the linear activation function in the last layer of a Multi-Layer Perceptron (MLP), we can directly map the input to the output without any additional transformations.

The linear activation function is visually represented in Figure 5.6. As shown in the plot, the function has a constant slope of 1, indicating that the output value is directly proportional to the input value. This linearity allows the model to learn and approximate a linear relationship between the input features and the target variable.

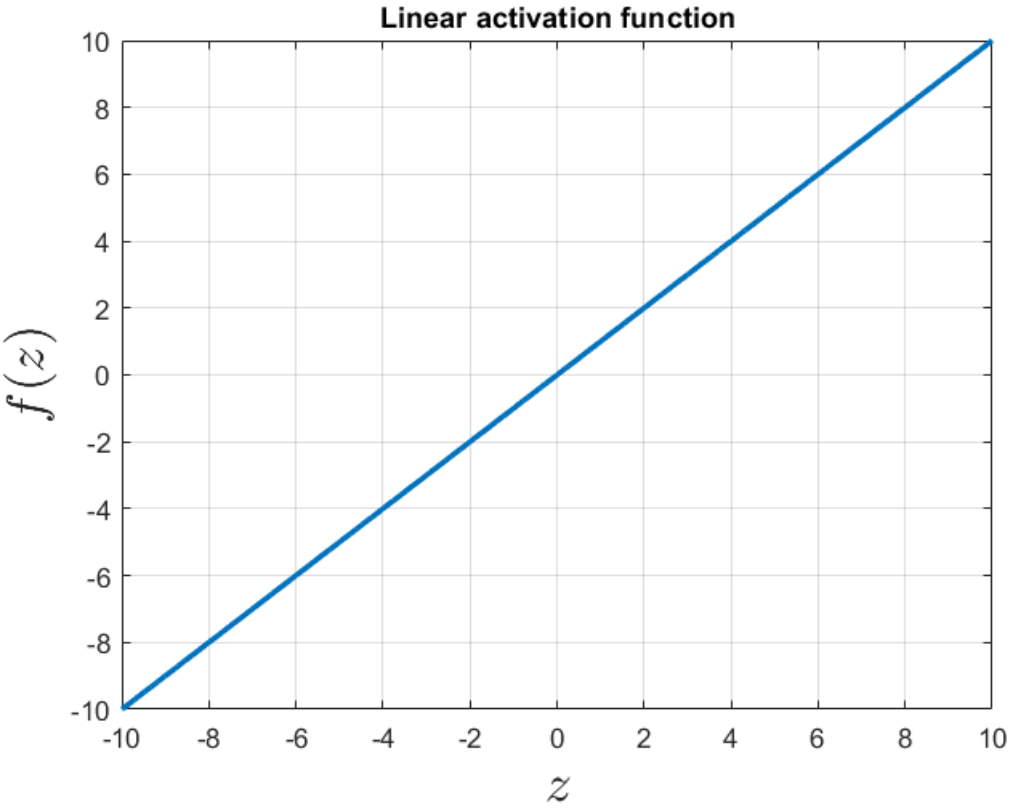


Figure 5.6: Visualization of the linear activation function.

The main advantage of the linear activation function is its simplicity and computational efficiency. Since it does not introduce any non-linear transformations, the gradients are constant and easily calculated. However, the linearity of this function limits the model's ability to capture complex patterns and non-linear relationships in the data. Therefore, it may not be suitable for tasks that require more intricate representations.

In summary, the linear activation function is a straightforward and useful tool in regression settings, where the objective is to predict continuous outputs. It provides a direct mapping between the input and output values without any non-linear transformations. However, its linearity restricts its ability to model complex patterns, and other non-linear activation functions may be more appropriate for tasks that involve intricate relationships in the data.

Logistic activation function

The logistic activation function, also known as the sigmoid function, is widely used in neural networks. It takes an input value from negative infinity to positive infinity and squashes it within the range of 0 to 1. Due to its continuous and differentiable nature, the logistic function is particularly suitable for multi-layer perceptrons trained with backpropagation. In binary classification tasks, the outputs of the logistic activation in the last layer can be interpreted as probabilities, as they sum to 1.

The logistic activation function is typically denoted by $\sigma(z)$ and is defined as:

$$\sigma(z) = \frac{1}{1 + \exp^{-z}} \tag{5.8}$$

The logistic function produces an S-shaped curve, as depicted in Figure 5.8. It smoothly maps the input values to the interval between 0 and 1, allowing for non-linear transformations of the data. The logistic activation function is commonly used in neural networks for tasks such as binary classification, where the goal is to assign inputs to one of two classes.

Despite its usefulness, the logistic activation function is not without its limitations. One significant challenge is the vanishing gradient problem. When the input to the logistic function becomes very large (either positive or negative), the output saturates to 1 or 0, respectively, resulting in gradients close to zero. This can hinder the learning process during backpropagation, as the small gradients prevent effective weight updates in earlier layers. Several techniques, such as weight initialization and using alternative activation functions, have been proposed to mitigate the vanishing gradient problem ([151, 152]).

Overall, the logistic activation function is a popular choice in neural networks due to its smoothness and interpretability. It is well-suited for tasks where the output needs to be transformed into a probability or where non-linear relationships between inputs and outputs are crucial. However, it is essential to be aware of its limitations, particularly the vanishing gradient problem, and consider alternative activation functions for specific scenarios.

Softmax activation function

The softmax activation function is a generalization of the logistic function and is commonly used in multi-class classification tasks [153, 154]. While the logistic function is suitable for binary classification, the softmax

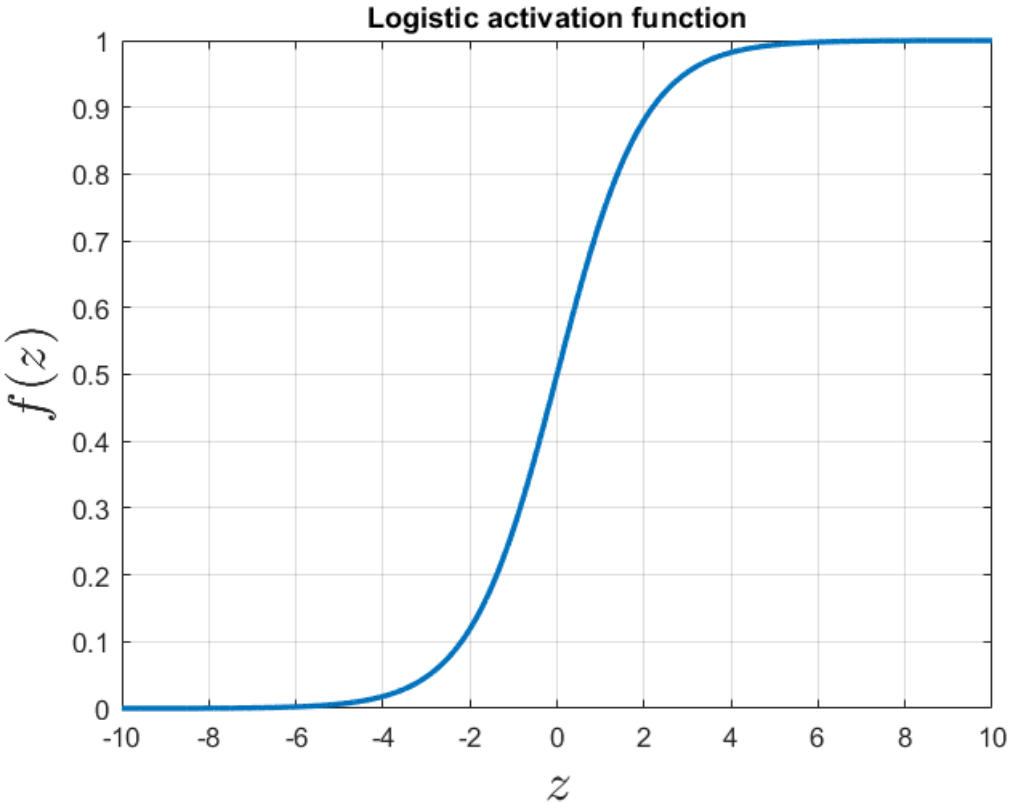


Figure 5.7: Logistic activation plot.

function extends its application to scenarios with multiple classes.

Given a neural network with K classes, the softmax activation function computes the probability distribution over the classes. For a vector of inputs $\boldsymbol{z} = [z_1, z_2, \ldots, z_K]$, the softmax function outputs a vector $\sigma(\boldsymbol{z}) = [\sigma(\boldsymbol{z})_1, \sigma(\boldsymbol{z})_2, \ldots, \sigma(\boldsymbol{z})_K]$, where each element represents the probability of the corresponding class. The softmax function is defined as:

$$\sigma(\boldsymbol{z})_j = \frac{\exp(z_j)}{\sum_{k=1}^{K} \exp(z_k)} \tag{5.9}$$

for $j = 1, 2, \ldots, K$.

The softmax function normalizes the inputs using the exponential function, ensuring that the output probabilities sum up to 1. By exponentiating the inputs, the softmax function amplifies larger values and suppresses smaller values, resulting in a well-behaved probability distribution.

The softmax activation function is crucial for multi-class classification tasks, as it allows the network to assign probabilities to each class and make informed predictions. During training, the softmax function is commonly used in the output layer of the network, where the predicted probabilities are compared to the true class labels to compute the loss and update the network's weights.

The softmax function has several desirable properties. It ensures that the predicted probabilities are non-negative and sum up to 1, making them interpretable as class probabilities. Additionally, the softmax function is differentiable, enabling the use of gradient-based optimization algorithms, such as backpropagation, to train the network.

By incorporating the softmax activation function in the output layer, neural networks can effectively handle multi-class classification problems and provide meaningful probabilistic predictions for each class. This enables applications such as image recognition, natural language processing, and speech recognition, where distinguishing between multiple classes is essential.

Hyperbolic tangent activation function (tanh)

The hyperbolic tangent activation function, commonly referred to as $\tanh(z)$, is another popular choice in neural networks. Similar to the logistic function, it exhibits an S-shaped curve, but it squashes the input values within the interval of -1 to 1. The $\tanh(z)$ function is defined as:

$$\tanh(z) = 2\sigma(2z) - 1 \tag{5.10}$$

As depicted in Figure 5.8, the $\tanh(z)$ function ranges from -1 to 1, making it advantageous for normalizing the outputs of computational units around zero. This normalization property can aid in the convergence of the backpropagation algorithm, as it helps balance the updates to the weights during training. However, it should be noted that the $\tanh(z)$ function does not fully address the vanishing or exploding gradient problem, which can still occur in deep neural networks.

The $\tanh(z)$ function shares the desirable properties of being continuous and differentiable, which are crucial for gradient-based optimization algorithms like backpropagation. It is commonly used as the activation function in the hidden layers of neural networks, enabling them to model non-linear relationships and capture complex patterns in the data.

By squashing the input values to a range between -1 and 1, the $\tanh(z)$ function provides more flexibility in capturing both positive and negative activations compared to the logistic function, which is bounded between 0 and 1. This expanded range allows for richer representations of data and can enhance the expressive power of neural networks.

In summary, the $\tanh(z)$ activation function combines the benefits of being continuous, differentiable, and normalizing the outputs around zero. It is particularly useful in neural networks for tasks that require non-linear transformations and can contribute to improved convergence during training. However, like other activation functions, it is important to consider the potential limitations, such as the vanishing or exploding gradient problem, when designing and training deep neural networks.

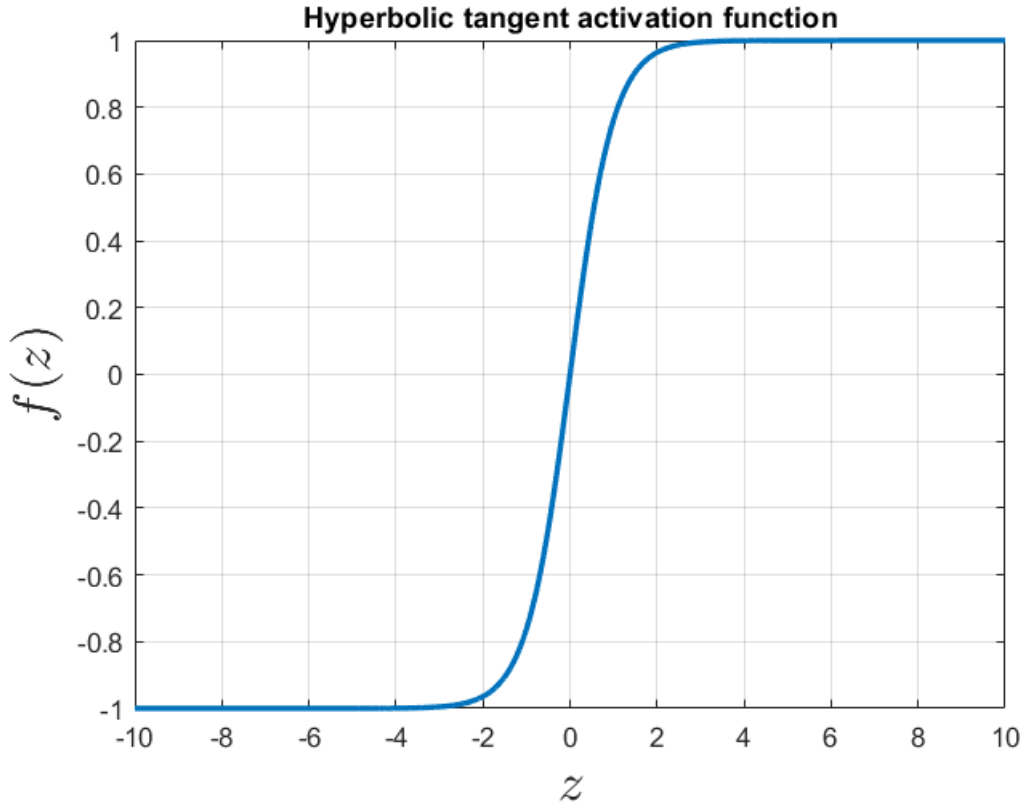


Figure 5.8: Plot of the hyperbolic tangent activation function.

Rectifier Linear Unit activation function

The rectified linear unit (ReLU) activation function is a widely used activation function in neural networks. It is continuous but non-differentiable at the point where the input is zero. The ReLU function has been found to work well in practice and is computationally efficient. One of its key advantages is that it does not suffer from saturation, which helps alleviate the vanishing gradient problem.

The ReLU activation function is defined as:

$$\text{ReLU}(z) = \max(0, z) \tag{5.11}$$

As shown in Figure 5.8, the ReLU function outputs the input value if it is positive, and outputs zero for negative values. This simple thresholding operation makes ReLU computationally efficient and allows the network to easily learn sparse representations. ReLU has been successful in various deep learning applications, primarily due to its ability to model non-linear relationships effectively.

However, the ReLU activation function is not without its limitations. One common issue is the "dying ReLU" problem, where some neurons become inactive during training. When a neuron's input is consistently negative, the ReLU function outputs zero, causing the gradient to be zero during backpropagation. Once a neuron dies, it is unlikely to recover its ability to contribute to the network's learning process. To mitigate

this problem, alternative activation functions have been proposed, such as the exponential linear unit (ELU) [155] and the leaky ReLU [156], which allow small negative activations to avoid neuron death.

In addition to choosing appropriate activation functions, the initialization of weights in each layer is crucial for training a multi-layer perceptron. Certain weight initialization techniques, such as the Glorot initialization and He initialization [157, 158], have been found to alleviate the vanishing gradient problem and promote efficient learning in deep networks.

Various techniques have been developed to address the vanishing/exploding gradient problem. One notable technique is batch normalization [159], which aims to normalize the intermediate inputs of each layer during training. By mitigating the issue of internal covariate shift, where the distribution of inputs to each layer changes due to weight updates in the previous layer, batch normalization helps stabilize training and improve generalization performance. Moreover, batch normalization has been shown to combat the problem of overfitting, which occurs when a model becomes too complex and fails to generalize well to unseen data [160, 161].

In summary, activation functions, weight initialization techniques, and normalization methods are essential components in addressing the vanishing/exploding gradient problem and improving the performance and generalization of neural networks. By carefully choosing appropriate activation functions and employing regularization techniques, such as batch normalization, practitioners can train deep neural networks more effectively and achieve better results on various tasks.

5.2 Training a neural network

In a supervised learning setting, the objective is to learn a function that maps input data to corresponding output labels. This function can be represented as $y = f(\mathbf{x}) + \epsilon$, where $\mathbf{x}$ represents the input, y represents the output, f captures the underlying relationship between the input and output, and ϵ represents the random noise or error. The function f can be parametric or non-parametric. In parametric models, f is assumed to have a specific form with a fixed set of parameters, while in non-parametric models, no assumptions are made about the functional form of f.

Deep neural networks, as a type of parametric model, consist of multiple layers of interconnected computational units called neurons. Each neuron applies a nonlinear activation function to a linear combination of its inputs, allowing the network to capture complex relationships between the input and output. The parameters of a deep neural network, often referred to as weights, are learned from the training data using an optimization algorithm such as backpropagation [162, 163, 164].

Backpropagation is an iterative algorithm that computes the gradients of the network's weights with respect to a chosen loss function, indicating the direction and magnitude of weight updates. The basic idea behind

backpropagation is to propagate the error backwards through the network, layer by layer, to update the weights in a way that minimizes the difference between the predicted outputs and the true outputs. The gradient descent algorithm is commonly used to perform the weight updates based on the computed gradients.

However, standard gradient descent suffers from certain limitations. For instance, it can get stuck in local minima or converge slowly due to small learning rates. To address these issues, various optimization techniques have been proposed. One popular technique is stochastic gradient descent (SGD), which updates the weights using a randomly selected subset of training samples at each iteration. This can result in faster convergence and help avoid getting stuck in local minima.

Another important consideration is the choice of the loss function, which measures the discrepancy between the predicted outputs and the true outputs. Commonly used loss functions include mean squared error (MSE) for regression problems and cross-entropy for classification problems. The choice of the loss function depends on the nature of the problem and the desired properties of the model.

In addition to optimization techniques, several regularization methods have been developed to prevent overfitting, which occurs when the model performs well on the training data but fails to generalize to new, unseen data. Regularization techniques such as L1 and L2 regularization, dropout, and early stopping help control the complexity of the model and improve its generalization performance.

Furthermore, the initialization of the weights in a deep neural network is crucial. Poor initialization can lead to vanishing or exploding gradients, hindering the training process. Techniques such as Xavier and He initialization provide effective strategies for initializing the weights to facilitate better gradient flow and improve convergence.

In recent years, advancements in deep learning have led to the development of more sophisticated optimization algorithms, including adaptive methods such as Adam, RMSprop, and Adagrad, which dynamically adjust the learning rates based on the gradient history. These algorithms often exhibit faster convergence and better generalization performance compared to standard gradient descent.

Overall, the training of deep neural networks involves a careful balance of optimization algorithms, regularization techniques, weight initialization strategies, and choice of loss functions, with the goal of minimizing the difference between predicted outputs and true outputs while avoiding overfitting and achieving good generalization performance.

5.2.1 Gradient descent with momentum

Gradient descent with momentum [165] is an optimization algorithm that enhances the standard gradient descent by incorporating a momentum term. The main idea behind this approach is to make large steps in the direction where the previous gradients consistently point while taking smaller steps in directions

with oscillating or conflicting gradients. By adding a momentum element to the gradient descent update, the algorithm can accumulate gradients from previous iterations, leading to faster convergence and better handling of saddle points.

The update equations for gradient descent with momentum can be defined as follows:

$$\begin{cases} \boldsymbol{m}_t = \mu \boldsymbol{m}_{t-1} - \eta(\nabla_{\boldsymbol{w}} J) \\ \boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \boldsymbol{m}_t \end{cases} \tag{5.12}$$

Here, $\boldsymbol{m}_t$ represents the momentum term, which is a decaying moving average of the gradients. The hyperparameter μ controls the decay rate of the momentum. The learning rate is denoted by η, and $\nabla_{\boldsymbol{w}} J$ denotes the gradient of the cost function with respect to the weights $\boldsymbol{w}$. The weights are updated by adding the momentum term, allowing the algorithm to take larger steps in directions where the gradients consistently have the same sign.

An extension of the momentum approach is the Nesterov momentum [149, 166]. Unlike the standard momentum, Nesterov momentum calculates the gradients at a projected position in the search space based on the momentum direction, rather than the current position. This helps the algorithm to have a better estimation of the true gradient and adjust the weights accordingly. By considering the decaying moving average of the gradients of projected positions, Nesterov momentum enables the algorithm to utilize the benefits of momentum's acceleration while slowing down as it approaches the optima, reducing the risk of overshooting or missing it.

The update equations for Nesterov momentum can be written as:

$$\begin{cases} \boldsymbol{m}_t = \mu \boldsymbol{m}_{t-1} - \eta(\nabla \boldsymbol{w} J) \\ \boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \boldsymbol{m}_t - \eta(\nabla_{\boldsymbol{w}} J) \end{cases} \tag{5.13}$$

In this case, the momentum term $\boldsymbol{m}_t$ is updated based on the gradient at the projected position, and then the weights are updated by subtracting the gradient term. This modification allows the algorithm to make more informed updates, especially when approaching optima, resulting in more accurate weight adjustments.

Both gradient descent with momentum and Nesterov momentum can significantly accelerate convergence compared to standard gradient descent. By leveraging the accumulated gradient information, they are more robust to noisy or oscillating gradients, allowing for faster and more stable optimization.

5.2.2 Adaptive gradient

In the context of gradient descent optimization, adaptive gradient (Adagrad) is an algorithm that introduces the concept of adaptive learning rates. Unlike standard gradient descent with a fixed learning rate, Adagrad dynamically adjusts the learning rate for each parameter based on the history of gradients accumulated during the optimization process. This adaptation is achieved by utilizing a cache, denoted by $\boldsymbol{\alpha}$, which accumulates the squared gradients.

The update step of Adagrad is defined as follows:

$$\begin{cases} \boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1} + \sum_{i=1}^{t-1} (\frac{\partial J}{\partial \boldsymbol{w}_{t-1}})^2 \\ \boldsymbol{w}_t = \boldsymbol{w}_{t-1} - \eta \frac{(\nabla_{\boldsymbol{w}} J)}{\sqrt{\boldsymbol{\alpha}_t + \epsilon}} \end{cases} \tag{5.14}$$

In this equation, $\boldsymbol{\alpha}_t$ represents the cache at iteration t, which accumulates the squared gradients up to that point. The parameter η denotes the learning rate, and $\nabla_{\boldsymbol{w}} J$ is the gradient of the cost function with respect to the weights $\boldsymbol{w}$. The denominator term $\sqrt{\boldsymbol{\alpha}_t + \epsilon}$ is used to normalize the learning rate.

The intuition behind Adagrad is that parameters with larger gradients will have their learning rates decreased more, while parameters with smaller gradients will have their learning rates increased. This adaptive adjustment ensures that the optimization process pays more attention to infrequent parameters with larger gradients and prevents the learning rates from becoming too large.

However, one limitation of Adagrad is that the accumulated sum of squared gradients in the cache continues to grow over time. As a result, the learning rates decrease rapidly, potentially hindering further progress and causing the algorithm to converge prematurely. This issue arises because the cache grows linearly without any normalization or scaling.

To address this problem, other optimization algorithms have been developed that aim to further enhance the learning process and overcome the limitations of Adagrad. These include algorithms such as RMSprop [**hinton2012neural**] and Adam [167], which introduce additional mechanisms to control the growth of the cache and provide better overall optimization performance.

5.2.3 Root mean square propagation

To address the issue of the cache growing too fast in Adagrad, an alternative optimization algorithm called Root Mean Square Propagation (RMSProp) was introduced [167]. RMSProp modifies the update rule to reduce the influence of previous gradients in the cache, preventing it from growing indefinitely.

The update step of RMSProp is defined as follows:

$$\begin{cases} \boldsymbol{\alpha}_t = \beta \boldsymbol{\alpha}_{t-1} + (1-\beta) \sum_{i=1}^{t-1} (\frac{\partial J}{\partial \boldsymbol{w}_{t-1}})^2 \\ \boldsymbol{w}_t = \boldsymbol{w}_{t-1} - \eta \frac{(\nabla_{\boldsymbol{w}} J)}{\sqrt{\boldsymbol{\alpha}_t + \epsilon}} \end{cases} \tag{5.15}$$

In this equation, $\boldsymbol{\alpha}_t$ represents the updated cache at iteration t. The term $(1-\beta) \sum_{i=1}^{t-1} (\frac{\partial J}{\partial \boldsymbol{w}_{t-1}})^2$ corresponds to the accumulation of squared gradients, similar to Adagrad. However, RMSProp introduces a decay rate β that controls the contribution of previous gradients. A higher value of β reduces the influence of older gradients in the cache.

By decreasing the influence of previous gradients, RMSProp allows the learning rates to adapt more dynamically to the current optimization landscape. It prevents the learning rates from becoming too small,

enabling the optimization process to continue making progress even in the later stages.

Typical values for the decay rate β in RMSProp are commonly set to values such as 0.99 or 0.999. These values ensure that the decay is slow, allowing the algorithm to retain useful information from previous iterations while still adjusting to the changing landscape of the optimization problem.

RMSProp is widely used in practice and has been shown to improve the convergence speed and overall performance of gradient-based optimization algorithms, especially in scenarios where Adagrad's rapidly decreasing learning rates may hinder progress.

5.2.4 Adaptive moment

One of the most widely used optimization algorithms in deep learning is Adaptive Moment Estimation (Adam) [167]. Adam combines the concepts of momentum and adaptive learning rates to efficiently update the model weights during training.

The updates in Adam are based on the estimates of both the first and second moments of the gradient. The first moment estimation is akin to momentum, while the second moment estimation adjusts the learning rate for each parameter based on the magnitude of past gradients.

The Adam algorithm can be defined as follows:

$$\begin{cases} \boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)(\nabla_{\boldsymbol{w}} J) \\ \boldsymbol{\alpha}_t = \beta_2 \boldsymbol{\alpha}_{t-1} + (1-\beta_2) \sum_{i=1}^{t-1} (\frac{\partial J}{\partial \boldsymbol{w}_{t-1}})^2 \\ \hat{\boldsymbol{m}}_t = \frac{\boldsymbol{m}_t}{1-\beta_1^t} \\ \hat{\boldsymbol{\alpha}}_t = \frac{\boldsymbol{\alpha}_t}{1-\beta_2^t} \\ \boldsymbol{w}_t = \boldsymbol{w}_{t-1} - \eta \frac{\hat{\boldsymbol{m}}_t}{\sqrt{\hat{\boldsymbol{\alpha}}_t + \epsilon}} \end{cases} \tag{5.16}$$

In this equation, $\boldsymbol{m}_t$ represents the first moment estimation (i.e., the moving average of gradients), $\boldsymbol{\alpha}_t$ represents the second moment estimation (i.e., the moving average of squared gradients), $\hat{\boldsymbol{m}}_t$ and $\hat{\boldsymbol{\alpha}}_t$ are the bias-corrected estimates, β_1 and β_2 are decay rates, and η is the learning rate. The term ϵ is a small constant added to the denominator for numerical stability.

The use of bias-correction terms ($1-\beta_1^t$ and $1-\beta_2^t$) ensures that the estimates are appropriately scaled during the early iterations of training when the moment and cache values are close to zero. This correction mitigates the initialization bias issue and helps Adam converge more reliably. Figure 5.9 provides a visual representation of the different optimization algorithms for training neural networks described earlier. Overall, Adam has demonstrated excellent performance in various deep learning applications and is widely adopted due to its ability to handle different types of data and architectures effectively.

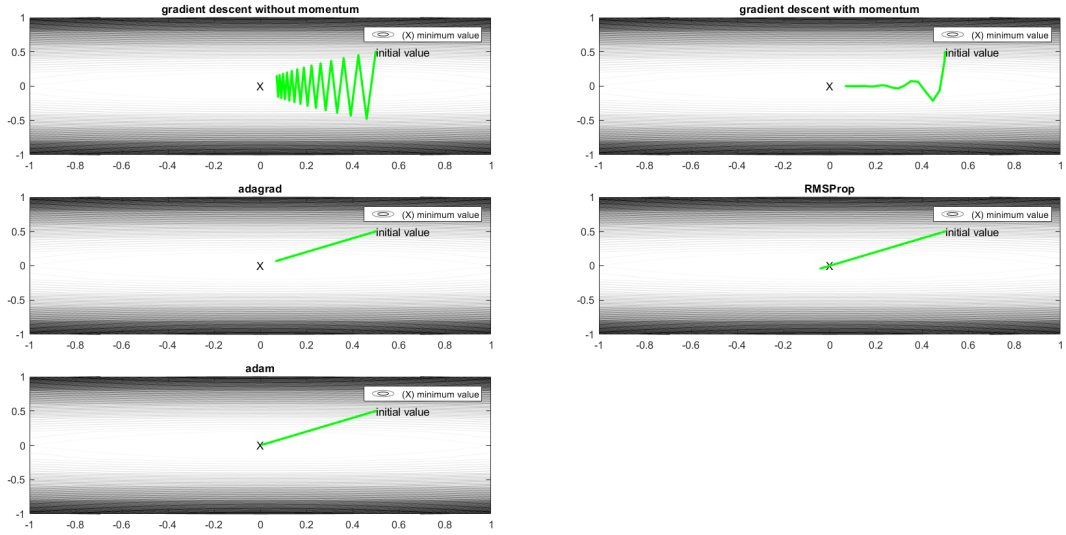


Figure 5.9: Visual representation of optimizer algorithms used for training a neural network after 25 iterations.

5.3 Loss functions

5.3.1 Mean square error loss function

The Mean Squared Error (MSE) is a commonly used loss function in various regression tasks. It measures the average squared difference between the predicted values of a model and the actual values in the training dataset. The MSE is widely employed when the output is a continuous real value.

Mathematically, for a dataset with m training examples, the MSE is calculated as:

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(y - \hat{y}_i)^2 \tag{5.17}$$

Here, y represents the true value of the target variable, $\hat{y}_i$ denotes the predicted value for the i-th example, and the summation is taken over all training examples. The squared difference between the predicted and true values is averaged over the dataset, providing a measure of the overall model performance.

The advantage of using MSE is that it emphasizes large errors due to the squaring operation, making it particularly effective at penalizing significant deviations between predictions and true values. This property is desirable in cases where it is crucial to avoid outlier predictions with substantial errors. However, a drawback of MSE is that it can heavily impact the overall loss if the model makes a single extremely poor prediction, as the squared term amplifies the error.

To address the outlier problem and provide alternative approaches, other loss functions have been developed. One such example is the mean absolute error (MAE) loss function, which ignores the squared term and calculates the average absolute difference between predictions and true values [168, 169]. The MAE is less sensitive to outliers and provides a more robust measure of error.

Another loss function that balances the handling of outliers is the Huber loss [170, 171]. The Huber loss offers a compromise between learning the outliers and ignoring them completely. It behaves like the squared loss for small errors but switches to the absolute loss for large errors. This makes the Huber loss more resistant to outliers while still providing meaningful gradients for efficient optimization.

In a probabilistic framework, assuming that the output follows a normal distribution, it can be shown that maximizing the log-likelihood is equivalent to minimizing the MSE. This connection highlights the relationship between the MSE loss and the maximum likelihood estimation, providing a statistical interpretation for its use in regression tasks.

Overall, the choice of the loss function depends on the specific requirements of the task and the characteristics of the data. While MSE is widely used and effective in many scenarios, alternative loss functions like MAE or Huber loss can be employed to handle outliers or accommodate different modeling assumptions.

5.3.2 Cross-entropy loss function

In addition to the Mean Squared Error (MSE) loss function, there are other loss functions commonly used in different scenarios. One such loss function is the cross-entropy loss, which arises when the output follows a specific distribution.

If we consider the output variable y to follow a binomial distribution, the negative log-likelihood of this distribution leads to the binary cross-entropy loss function [172]. This loss function is frequently used in binary classification tasks and is defined as:

$$\text{Binary Cross Entropy} = -\frac{1}{m}\sum_{i=1}^{m}(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)) \tag{5.18}$$

Here, y_i represents the true label (either 0 or 1) for the i-th example, and $\hat{y}_i$ denotes the predicted probability of the positive class. The binary cross-entropy loss measures the dissimilarity between the predicted probabilities and the true labels, taking into account both the positive and negative class predictions.

For multi-class classification problems with K classes, the output y can be assumed to follow a categorical distribution. In this case, the cross-entropy loss function for multi-class classification, often referred to as categorical cross entropy, is utilized. It is defined as:

$$\text{Categorical Cross Entropy} = -\frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{K} y_{ik} \log(\hat{y}_{ik}) \tag{5.19}$$

Here, y_{ik} represents the true label (0 or 1) for the i-th example and k-th class, and $\hat{y}_{ik}$ denotes the predicted probability for that class. The categorical cross-entropy loss calculates the dissimilarity between the predicted probabilities and the true labels across all classes.

Both the binary and categorical cross-entropy loss functions aim to minimize the difference between the predicted probabilities and the true labels. By penalizing incorrect predictions more heavily through the logarithmic terms, these loss functions effectively drive the model towards more accurate class probabilities.

It is important to note that the choice of the appropriate loss function depends on the nature of the task and the type of output variable. The MSE loss is commonly used in regression tasks, while the cross-entropy loss functions are prevalent in classification problems. By selecting the appropriate loss function, we can effectively train models that are capable of capturing the desired characteristics of the data and producing accurate predictions.

5.4 Convolutional neural network

Convolutional neural networks (CNNs) have emerged as a powerful approach for various computer vision tasks, including object detection and image classification. The development of CNNs was inspired by the remarkable findings of D. Hubel and T. Wiesel on the visual cortex [173, 174], where they observed that neurons in the visual cortex respond to specific regions of the visual field known as receptive fields. These receptive fields are local regions that collectively cover the entire visual field.

The neocognitron [175] was one of the first architectural designs that incorporated this biological insight into a neural network model. It laid the foundation for the development of the LeNet-5 architecture [176], which achieved significant success in handwritten digit classification.

While CNNs share similarities with traditional artificial neural networks (ANNs), such as the use of computational units that process input through a dot product and activation function, they also introduce unique characteristics tailored specifically for image data. One of the key challenges of traditional ANNs when dealing with image data is the computational complexity involved in processing images. Images need to be flattened into one-dimensional vectors before being fed into an ANN. For instance, an image of size 64x64x1 would be flattened into a 784x1 vector, which significantly increases the number of weights in just a single neuron of the first layer (e.g., 12288 weights).

CNNs address this issue by incorporating image-specific properties into their design, making them more suitable for image-focused tasks. They leverage a specialized layer called a convolutional layer, which applies filters to input images, capturing local spatial patterns. The convolutional layer allows the network to learn and extract features directly from the image, preserving the spatial structure. This is in contrast to ANNs, which treat input as a flat vector, disregarding the inherent spatial relationships present in images.

In addition to convolutional layers, CNNs often include other types of layers, such as pooling layers and fully connected layers. Pooling layers reduce the dimensionality of the feature maps, capturing the most important

information. Fully connected layers, similar to those in ANNs, connect every neuron from the previous layer to every neuron in the subsequent layer, enabling high-level abstraction and decision-making.

Training a CNN involves the same principles as ANNs, such as making predictions based on learnable weights and optimizing them using backpropagation and an error (loss) function. Techniques used to mitigate overfitting and address the vanishing/exploding gradient problem, such as regularization, dropout, and weight initialization, are also applicable to CNNs.

Overall, CNNs offer a powerful framework for analyzing and processing visual data. By leveraging the spatial structure of images and incorporating image-specific design principles, CNNs have revolutionized the field of computer vision and have become a cornerstone in various image-related applications.

5.4.1 CNN architecture

A CNN architecture is composed of three fundamental layers: convolutional layers, pooling layers, and fully connected layers, as described in section 5.1.4. These layers work together to extract meaningful features from the input data and make predictions.

The first type of layer in a CNN is the convolutional layer. This layer applies a set of learnable filters to the input data in order to detect various features or patterns. Each filter performs a convolution operation by sliding over the input and computing the dot product between the filter weights and the corresponding region of the input. This operation captures spatial relationships and local patterns in the data. Convolutional layers are typically followed by an activation function, such as ReLU, which introduces non-linearity into the network.

The second type of layer is the pooling layer, which is responsible for downsampling the feature maps produced by the convolutional layers. Pooling helps reduce the spatial dimensions of the data while retaining the most important features. The most common pooling operation is max pooling, which selects the maximum value within each pooling region. This operation helps to extract the most prominent features and improve the network's translation invariance.

The final layer of a CNN is the fully connected layer. This layer connects every neuron in the previous layer to the neurons in the current layer, similar to a traditional neural network. The fully connected layer takes the high-level features learned by the previous layers and combines them to make predictions. It performs a weighted sum of the inputs and applies an activation function to produce the final output. The number of neurons in this layer corresponds to the number of classes in a classification task.

By stacking these layers together, a CNN architecture can learn hierarchical representations of the input data. The convolutional layers extract low-level features such as edges and textures, which are then combined and abstracted by subsequent layers. This hierarchical representation allows the network to capture increasingly

complex patterns and make accurate predictions.

It is worth noting that CNN architectures can vary in terms of the number of layers, the size of the filters, the stride of the convolution operation, and other architectural choices. Different configurations can be more suitable for specific tasks or datasets. Additionally, advanced CNN architectures, such as VGGNet, ResNet, and InceptionNet, have been developed to improve performance on challenging visual tasks by introducing additional architectural components or modifications.

Convolutional layer

The convolutional layer is a fundamental building block of the CNN architecture, especially for processing images. It operates on the spatial dimensions (height and width) of the input while also considering the depth of the image. The key idea behind the convolutional layer is to utilize sparse connectivity and parameter sharing.

In a convolutional layer, each neuron is connected only to a small local region of the input called the receptive field. This means that neurons in the first convolutional layer are only connected to pixels in their receptive field, and neurons in subsequent layers receive inputs from previous layers. This local connectivity helps capture local patterns and spatial relationships within the input.

The weights of neurons in a convolutional layer are represented by small matrices, typically of sizes like 3x3, 5x5, or 7x7, known as filters or kernels. These filters are applied to the input using the convolution operation, resulting in a feature map that represents the extracted patterns in the image corresponding to that filter. The filter weights are learned during the training process to capture specific features or characteristics in the input.

To illustrate the convolution operation, consider an example shown in Figure 5.10. We have a 6x6 input matrix and a 3x3 filter. By convolving the filter over the input, we calculate the scalar product between the filter values and the corresponding input values, and then sum them up. This process is repeated for each location in the input, resulting in a 4x4 output feature map.

One limitation of the convolutional operation is that it can shrink the image too much, potentially causing information loss, especially in deep CNNs. To mitigate this, zero-padding can be applied to extend the image with zeros along the borders. The choice of padding depends on the desired output dimensions. "Valid" padding does not extend the image and may ignore some rows and columns at the bottom and right of the input image, depending on the stride. "Same" padding adjusts the amount of padding so that the input and output have the same dimensions.

The stride determines the number of pixels shifted across the input matrix when applying the filters. A stride of 1 means the filters move one pixel at a time, while a stride of 2 means they move two pixels at a time, and

so on.

In general, if the input image has a size of n x n, applying padding p, convolving the image with a filter of size f x f and a stride of s will result in an output image of size o x o, where o is calculated using Equation 5.11:

$$o = \left\lfloor \frac{n + 2p - f}{s} + 1 \right\rfloor \tag{5.20}$$

It's important to note that the size of the filter depends on the dimensions of the input. If the input image has a size of n x n x d, where d represents the depth, the filter will have a size of f x f x d. The convolution operation is performed across all depth dimensions, and Equation 5.11 still applies. The number of filters used in a layer corresponds to the depth when stacking the resulting feature maps from those filters. This allows the network to learn multiple sets of filters, each capturing different aspects or patterns of the input data.

5.4.2 Pooling layer

Pooling layers play a crucial role in reducing the computational cost and parameter count of the model by continuously decreasing the dimensionality of the representations. Similar to convolutional layers, pooling layers also have a local receptive field, where each neuron is connected to a small number of neurons in the preceding layer. This limited connectivity helps capture local patterns and reduce the spatial dimensionality of the input. Just like in convolutional layers, you can set the size, stride, and padding type of the pooling operation.

Pooling layers do not have any learnable weights, making them computationally lightweight. Their main function is to aggregate the inputs using a specific function, such as max pooling or mean pooling. Figure 5.11 provides a visual representation of how pooling layers work.

In max pooling, the maximum value within the receptive field is selected as the output. This helps retain the most prominent features or activations from the input. On the other hand, mean pooling calculates the average value within the receptive field, providing a more smoothed and generalized representation of the input.

The pooling operation helps to reduce the spatial dimensions of the representations, resulting in a compressed and abstracted version of the input. This dimensionality reduction not only decreases the computational cost but also helps in controlling overfitting by reducing the model's complexity.

Pooling layers can be inserted between convolutional layers to progressively downsample the representations. The downsampling process helps to capture hierarchical patterns and maintain the most relevant information while discarding unnecessary details. It also increases the receptive field of higher-level neurons, allowing

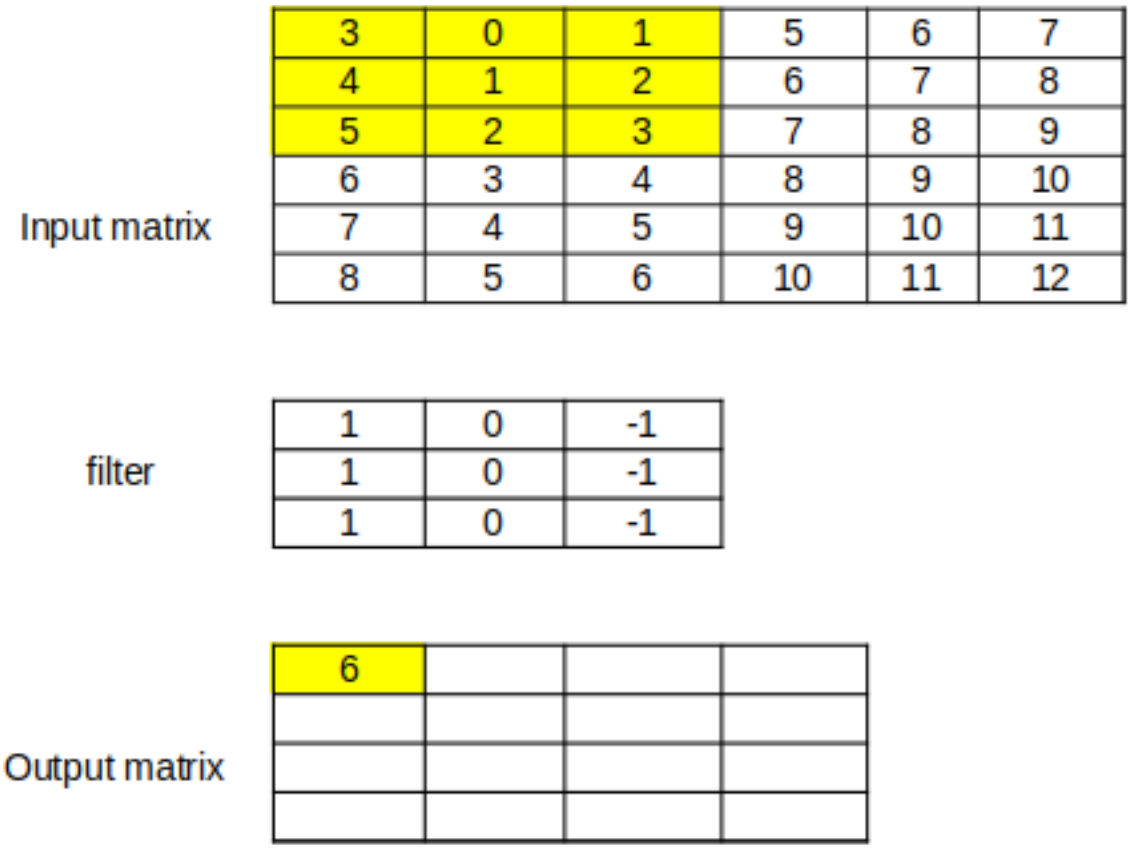


Figure 5.10: A visual representation of convolutional layer.

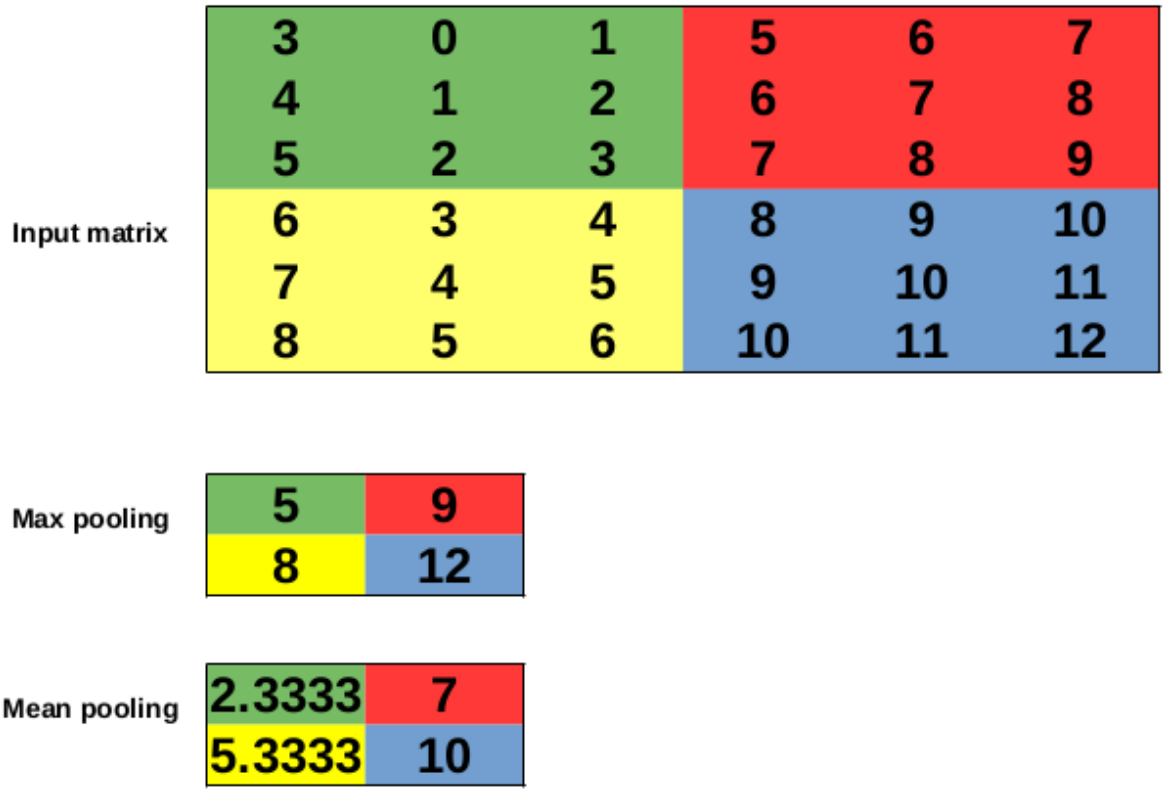


Figure 5.11: A visual representation of max pooling and mean pooling with a filter of size 2 and stride of 2.

them to capture larger contextual information. Overall, pooling layers are an essential component of CNN architectures, enabling efficient information extraction and reducing the model's computational burden.

5.5 Application of CNNs to EQN

Convolutional neural networks (CNNs) have proven to be highly effective in image classification tasks, leading researchers to explore their application in various domains, including earthquake early warning systems. Artificial intelligence and machine learning techniques have demonstrated their efficiency in numerous applications, particularly in the field of image classification.

In this thesis, our focus lies on CNNs as powerful feature extractors for earthquake-related data. This approach has been explored by Ahmed, Manat et al. in their work on recognizing spatial extreme dependence structures [177]. Mousavi, S. Mostafa et al. developed a CNN model for simultaneous earthquake detection and phase picking [178]. Similarly, Kuyuk, H. Serdar et al. employed CNNs for real-time earthquake classification [179], while Saad, Omar M et al. used CNNs for earthquake parameters classification in earthquake early warning systems [180].

In the context of the EQN project, a CNN approach is explored to address the challenges of false alarms and missed detections. This involves transforming the output of the EQN detection algorithm, as described in section 3.2.1, into an image representation. A CNN model is then designed to capture the observed patterns described in section 4.4.1 and classify between true and false earthquakes. However, before delving into the details of CNN architectures, it is important to understand how the input images for the CNN are generated.

5.5.1 Images generations

To ensure the robustness of the Convolutional Neural Network (CNN) and to eliminate sensitivity to the spatial structure of a specific city, we conducted simulations of various spatial structures generated from a multivariate Gaussian distribution. These structures were used as inputs for the EQN detection algorithm to simulate scenarios of both actual earthquakes and false detections.

For actual earthquakes, we considered several factors including the earthquake epicenter, depth, arrival time of seismic waves at smartphone locations (based on the earthquake propagation velocity), and the detectability of the earthquake by smartphones. We also took into account the possibility of smartphones detecting phenomena unrelated to earthquakes. More detailed information about these simulations can be found in Section 4.4. The output of the simulation is a list of triggering smartphones sorted by triggering time, which is then processed by the EQN detection algorithm. The algorithm stops when the detection condition is met, and an image is generated that represents the triggering smartphones and active smartphones (smartphones that are not triggers).

Specifically, for simulating actual earthquakes, we simulated both P-wave and S-wave earthquakes with respective velocities of 4.5 km/s and 7.8 km/s. Figure 5.12 shows an image from a simulated true earthquake with a P-wave velocity, while Figure 5.13 represents an image from a simulated true earthquake with an S-wave velocity.

To mimic false detections, we considered situations where the false detection could arise from a random event with no specific spatial pattern (refer to Section 4.3 for details). Additionally, we took into account cases where events unrelated to earthquakes could exhibit a clear propagation pattern, such as thunder. In this particular case, the simulations were similar to those for actual earthquakes, but without considering depth, and with a velocity proportional to the speed of light, which is approximately 0.3 km/s.

Figure 5.14 illustrates an example of a simulated false EQN detection related to a random event with no specific spatial pattern, while Figure 5.15 shows an example of a simulated false EQN detection related to a random event with a spatial pattern.

In general, we discriminate between four distinct classes: P-wave and S-wave, which represent actual earthquakes with different wave velocities, and random and thunder , which represent false earthquakes with random and spatial patterns, respectively.

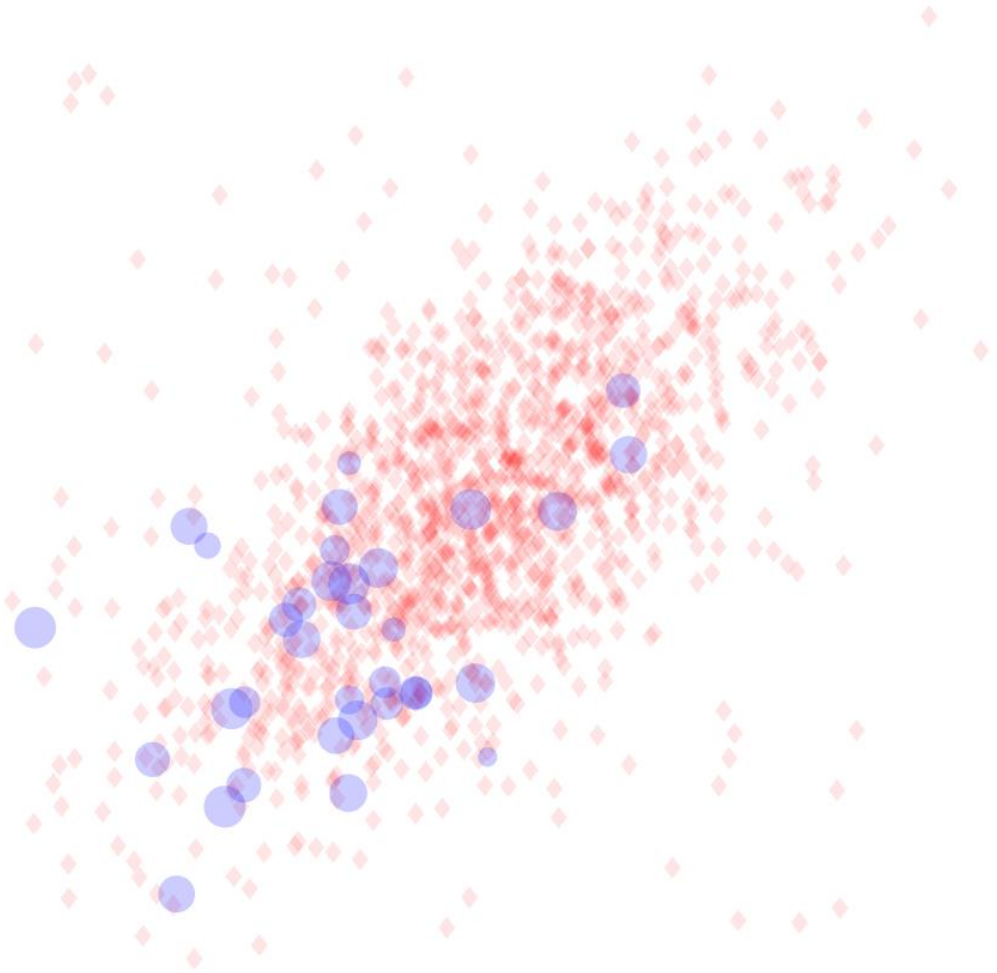


Figure 5.12: True simulated earthquake with a P-wave velocity based on the EQN simulated smartphone network. Smartphones that contributed to the earthquake detection are represented by circles with diameters proportional to the triggering time. Active smartphones are represented by diamond markers. The magenta triangular marker represents the simulated epicenter location.

5.5.2 Training and robustness of the CNNs model

The geometry of a spatio-temporal dataset, including the images generated from such a dataset, plays a crucial role in determining the effectiveness of a Convolutional Neural Network (CNN) for analyzing these images. Figure 5.16 shows a sample of images of Lima for the 4 classes

Make sure to include the subcaption package by adding subcaption in your document preamble. This should resolve any errors related to subfigures. If you encounter any other errors, please provide more details so I can assist further.

Our developed CNNs take as input an image, which is generated as described in section 5.5.1. The size of the

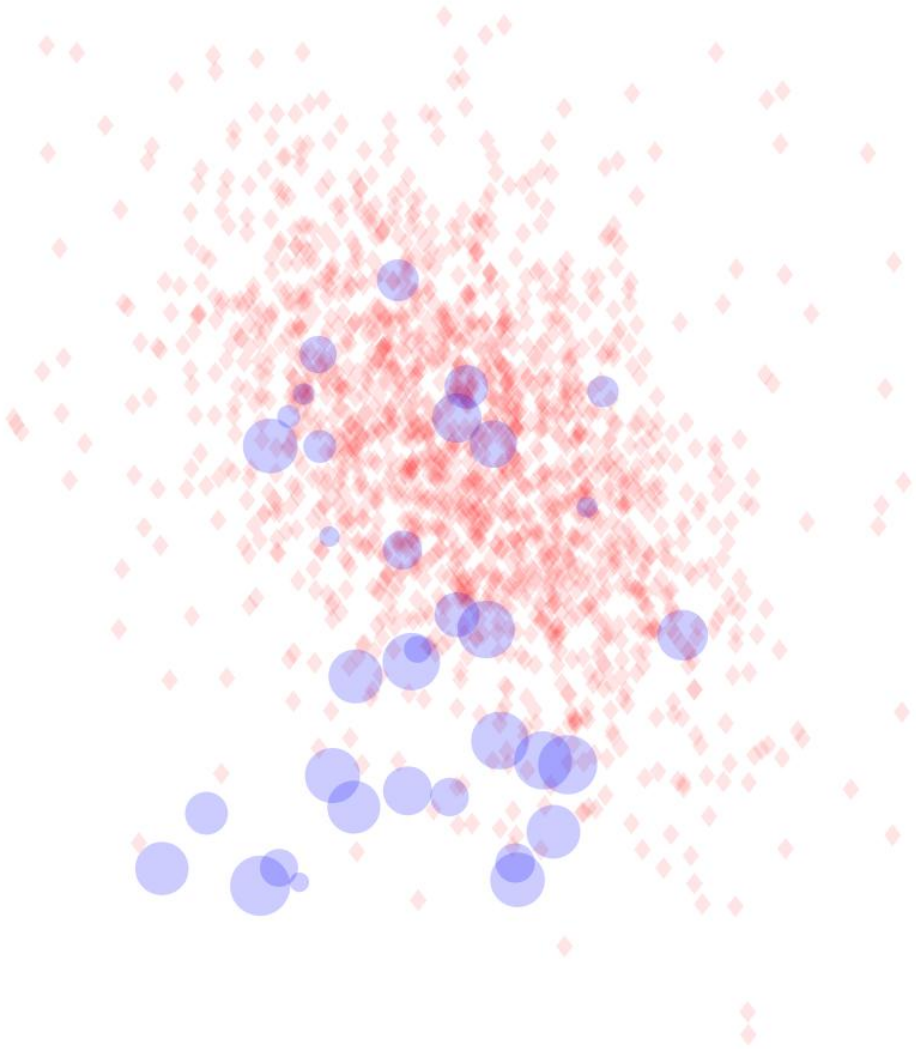


Figure 5.13: True simulated earthquake with an S-wave velocity based on the EQN simulated smartphone network. Smartphones that contributed to the earthquake detection are represented by circles with diameters proportional to the triggering time. Active smartphones are represented by diamond markers. The magenta triangular marker represents the simulated epicenter location.

generated image is not a determining factor, as the internal resizing of the image is performed based on the architecture used to address memory constraints. We have designed multiple CNN architectures, varying the number of layers and other hyperparameters, in order to search for the optimal model.

To train the CNN, we simulated 40,000 images as described in section 5.5.1, with 10,000 images representing different classes. The selection of the best architecture depends on the specific requirements of the client. For example, is it more important to minimize the probability of classifying a false earthquake as a positive event, or to minimize the probability of classifying a true earthquake as a false event? Generally, we anticipate that the CNN architectures may encounter challenges in discriminating between the inner-true classes (P-wave

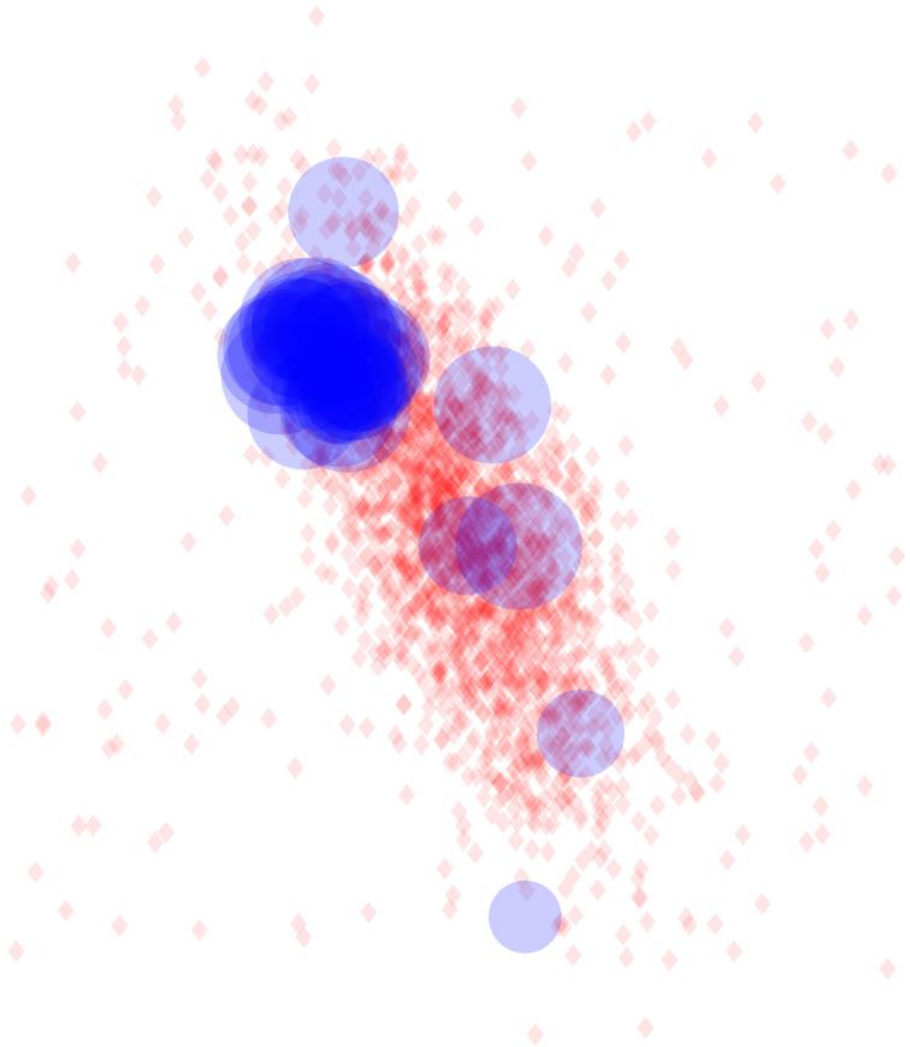


Figure 5.14: Simulated false detection (thunder) based on the EQN simulated smartphone network. Smartphones that contributed to the detection are represented by circles with diameters proportional to the triggering time. Active smartphones are represented by diamond markers.

and S-wave). While this factor may have minor significance in the overall warning alert system, it could potentially affect the accuracy of the CNN models provided. Tables 5.1 and 5.2 show a summary of the results obtained. for each architecture.

Architecture 1

The first architecture design, depicted in Figure 5.17, showcases the initial model used in our experiments. The model achieved an impressive accuracy of 81.25% on the training data, demonstrating its ability to effectively learn from the training set. Furthermore, it exhibited a validation accuracy of 79.46%, indicating its capability to generalize well to unseen data. The performance on the test data was also commendable,

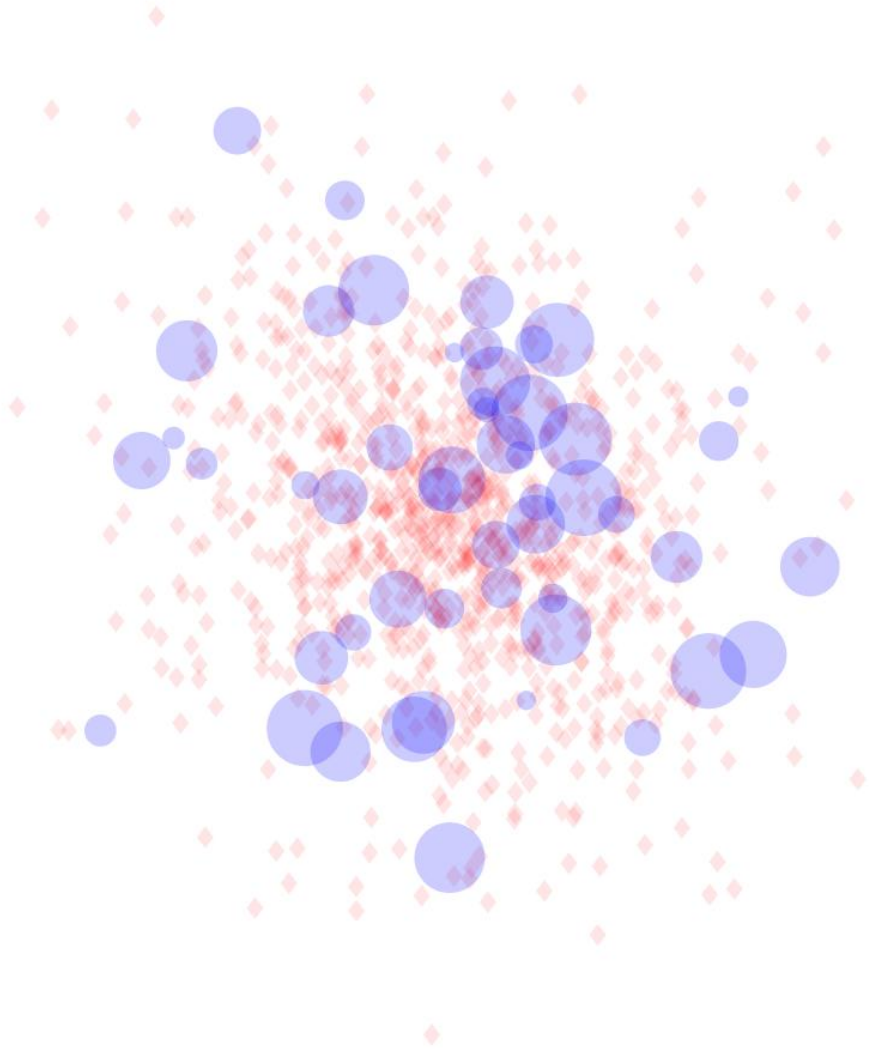
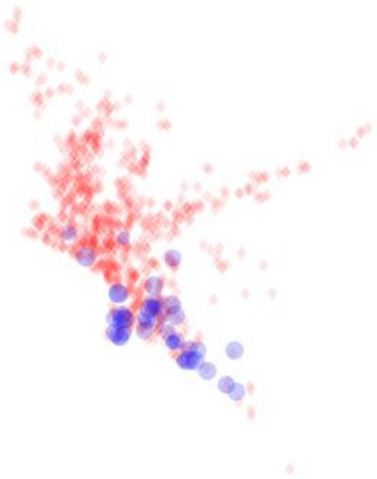


Figure 5.15: Simulated false detection (random) based on the EQN simulated smartphone network. Smartphones that contributed to the detection are represented by circles with diameters proportional to the triggering time. Active smartphones are represented by diamond markers.

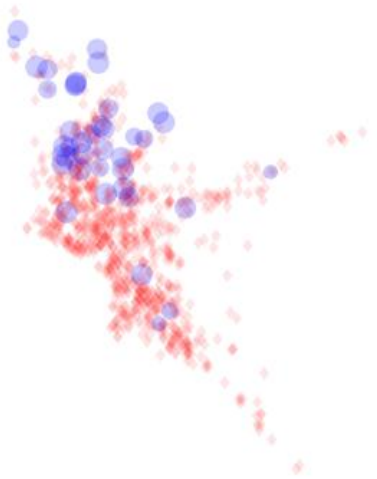
with an accuracy of 78.85%.

To gain further insights into the model's performance, we analysed the confusion matrices obtained from the test data. The confusion matrix of the test data reveals a false negative rate of 2.3% and a false positive rate of 0.6%. These metrics provide valuable information about the model's ability to correctly classify earthquakes and non-earthquake events. Additionally, we evaluated the model's performance on a real-world scenario by examining the confusion matrix obtained from the smartphone network in Lima, Peru. In this case, the model exhibited a false negative rate of 7.7% and a false positive rate of 0.5%.

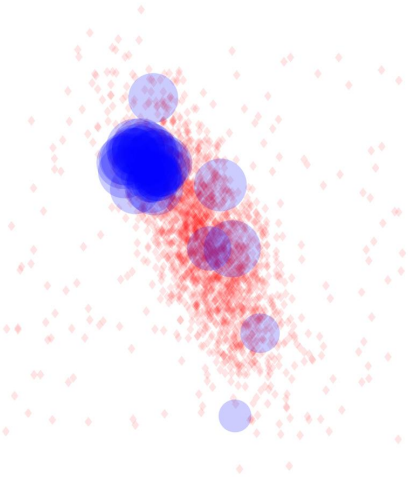
The options for training the model were set as follows: the Adam optimizer was used with an initial learning



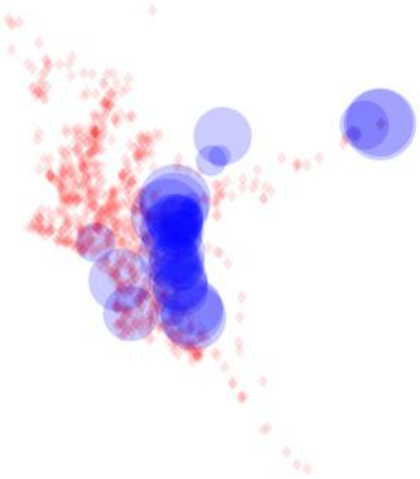
(a) Simulated earthquake with an P-wave velocity based on the city of Lima (Perù). Smartphones that contributed to the earthquake detection are represented by circles with diameters proportional to the triggering time. Active smartphones are represented by diamond markers. The magenta triangular marker represents the simulated epicenter location



(b) Simulated earthquake with an S-wave velocity based on the city of Lima (Perù). Smartphones that contributed to the earthquake detection are represented by circles with diameters proportional to the triggering time. Active smartphones are represented by diamond markers. The magenta triangular marker represents the simulated epicenter location



(c) Simulated false detection (random) based on the city of Lima (Perù). Smartphones that contributed to the detection are represented by circles with diameters proportional to the triggering time. Active smartphones are represented by diamond markers.



(d) Simulated false detection (thunder) based on the city of Lima (Perù). Smartphones that contributed to the detection are represented by circles with diameters proportional to the triggering time. Active smartphones are represented by diamond markers.

Figure 5.16: Sample of images representing the 4 scenario based on the city of Lima (Perù)

rate of 1×10^{-3}. The learning rate schedule was set to 'piecewise' with a drop factor of 0.1 and a drop period of 5 epochs. The maximum number of epochs was set to 15, and a mini-batch size of 256 was used. We use validation data, to perform a validation every 50 iterations. The data was shuffled every epoch, and verbose output was enabled to monitor the training progress. The training process was visualized using training plots, and early stopping was applied with a "patience" of 7 epochs.

Architecture 2

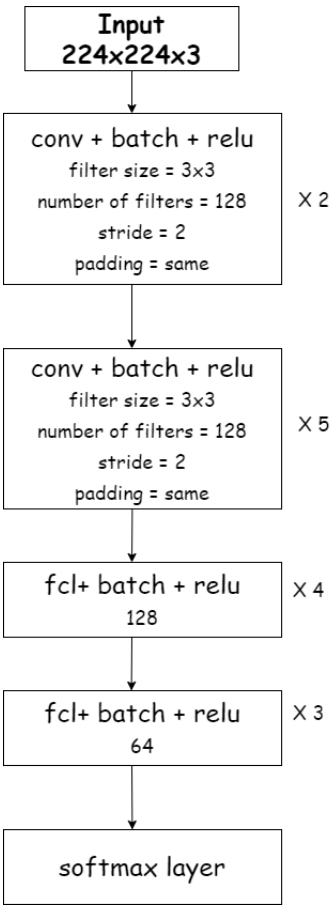
Figure 5.18 showcases the initial architecture design, which achieved remarkable accuracy on the different datasets. Specifically, the model demonstrated an impressive accuracy of 87.500% on the training data, indicating its ability to effectively learn from the training set. Furthermore, it exhibited strong performance on the validation data, achieving an accuracy of 80.47%, highlighting its ability to generalize well to unseen data. The commendable accuracy extended to the test data, where the model maintained an accuracy of 80.15%.

The corresponding confusion matrix obtained from the test data, also illustrated in Figure 5.18, provides valuable insights into the model's performance. It revealed a low false negative rate of 1.7% and an equally impressive false positive rate of 0.46%. This demonstrates the model's ability to accurately distinguish earthquakes from non-earthquake events. Additionally, the confusion matrix for the smartphone network in Lima, Peru, presented in Figure 5.18, showed a false negative rate of 8.7% and a false positive rate of 0.38%.

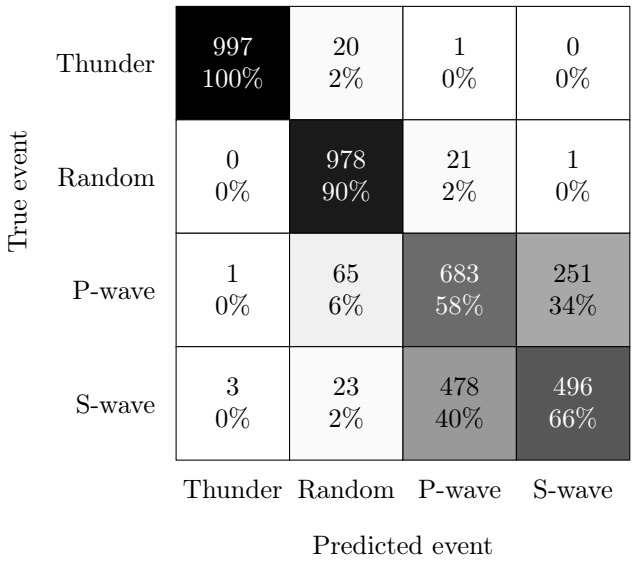
For training the model, we carefully selected specific options to achieve optimal results. The Adam optimizer was employed with an initial learning rate of 1×10^{-3}. To adaptively adjust the learning rate, we implemented a 'piecewise' learning rate schedule, incorporating a drop factor of 0.1 and a drop period of 5 epochs. The training process involved a maximum of 20 epochs, and a mini-batch size of 256 was utilized to enhance efficiency. Regular evaluations were performed using validation data every 25 iterations during training. To ensure training stability, the data was shuffled at the beginning of each epoch, and verbose output was enabled to closely monitor progress. The training process was visually represented through training plots, facilitating analysis and comprehension of the model's performance. Additionally, early stopping was implemented with a "patience" of 10 epochs to prevent overfitting, allowing the model to halt training if no significant improvements were observed. These thoughtfully chosen training options contributed to the model's robustness and overall success in the experiments.

Architecture 3

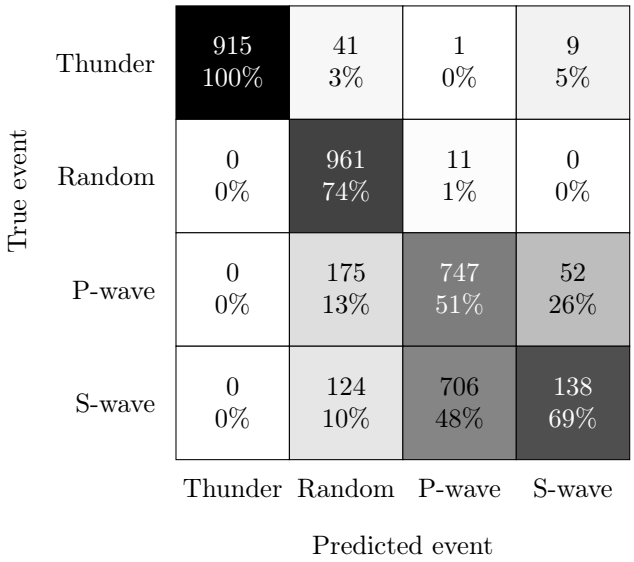
Figure 5.19 showcases the initial architecture design for our model, which achieved impressive accuracy levels on different datasets. Notably, the model attained a remarkable accuracy of 94% on the training data, demonstrating its capability to effectively learn from the training dataset. Furthermore, the model showcased solid performance on the validation data, achieving an accuracy of 79.47%, indicating its ability to generalize well to unseen data. Similarly, the model maintained a commendable accuracy of 80.13% on the test data, further validating its generalization capabilities.



(a) Diagram of architecture 1: conv represents a convolutional layer, fcl a fully connected layer, batch a batch-normalization layer.

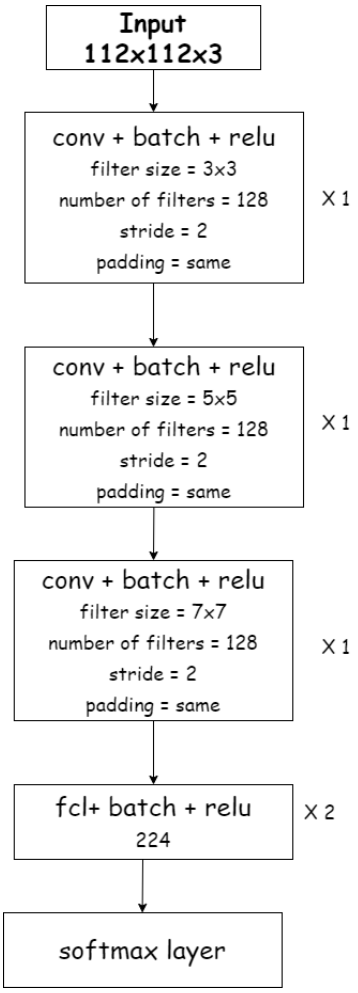


(b) Confusion matrix of the CNN model based on the test data.

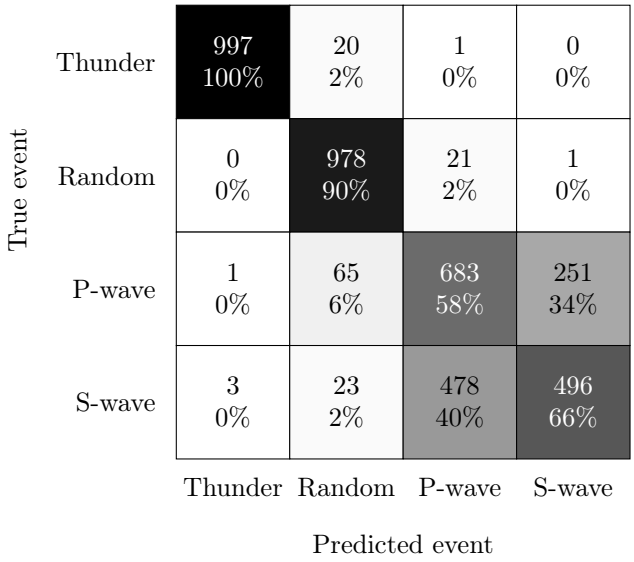


(c) Confusion matrix of the CNN model based on the smartphone network of Lima (Perù).

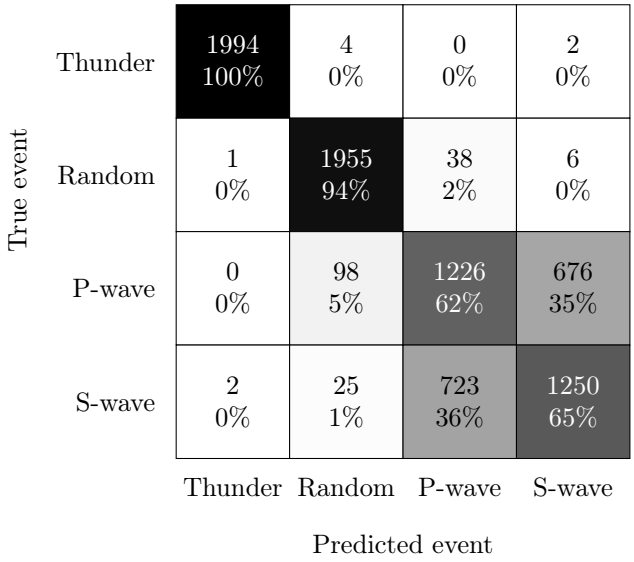
Figure 5.17: Training confusion matrix for the CNN based on architecture 1.



(a) Diagram of architecture 2: conv represents a convolutional layer, fcl a fully connected layer, batch a batch-normalization layer.



(b) Confusion matrix of the CNN model based on the test data.



(c) Confusion matrix of the CNN model based on the smartphone network of Lima (Perù).

Figure 5.18: Training confusion matrix for the CNN based on architecture 2.

To gain deeper insights into the model's performance, we analyzed the confusion matrix obtained from the test data, as illustrated in Figure 5.19. This matrix provided valuable information about the model's ability to correctly predict positive and negative outcomes. It revealed a low false negative rate of 1.6%, which represents instances where the model incorrectly predicted negative outcomes. Additionally, the model demonstrated an impressive ability to avoid false positive predictions, with a low false positive rate of 0.57%.

Figure 5.19 also presents the confusion matrix specific to the smartphone network in Lima, Peru. Here, we observed a false negative rate of 4.5%, indicating instances where the model incorrectly classified positive outcomes as negative. Furthermore, the model exhibited a false positive rate of 2.1%, representing cases where the model incorrectly predicted positive outcomes.

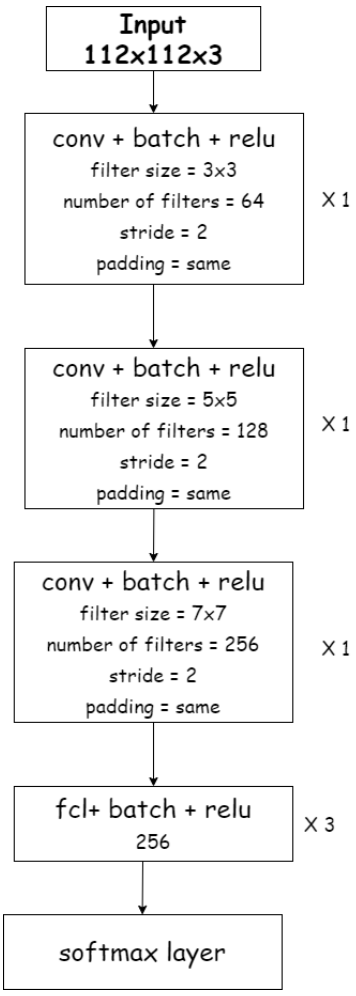
Throughout the training process, we employed a set of carefully chosen options to optimize the model's performance. The Adam optimizer served as the foundation, utilizing an initial learning rate of 1×10^{-3}. To fine-tune the learning rate during training, we implemented a 'piecewise' learning rate schedule, incorporating a drop factor of 0.1 and a drop period of 5 epochs. This adaptive learning rate schedule ensured that the model adjusted its learning rate appropriately as training progressed.

To strike a balance between convergence and efficiency, we set the maximum number of epochs to 15. Additionally, a mini-batch size of 256 was chosen to enable efficient processing of the training data. To assess the model's progress and evaluate its performance, we performed validations every 50 iterations using a separate validation dataset. This approach allowed us to closely monitor and analyze the model's behavior during training.

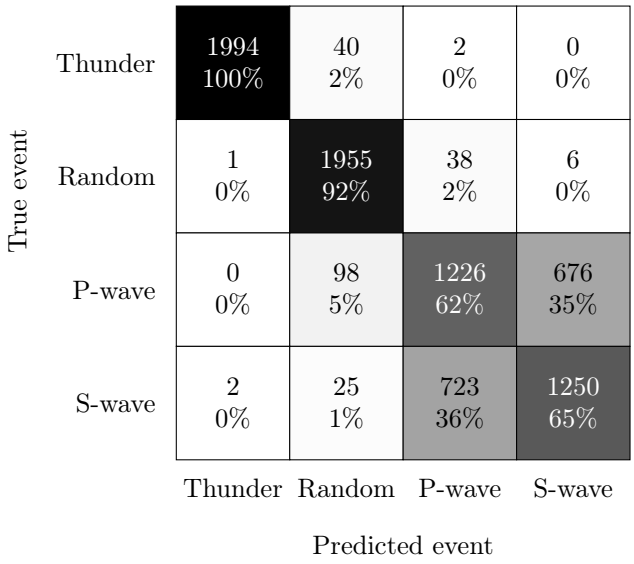
To enhance the stability and effectiveness of the training process, we shuffled the data at the beginning of each epoch. This prevented the model from being biased by the order of the training examples and ensured robust learning. Verbose output was enabled to provide detailed information about the training progress, facilitating comprehensive monitoring and analysis.

To safeguard against overfitting, we implemented early stopping with a patience of 7 epochs. This mechanism allowed the model to halt training if no significant improvements were observed, thereby preventing excessive reliance on the training data and promoting generalization.

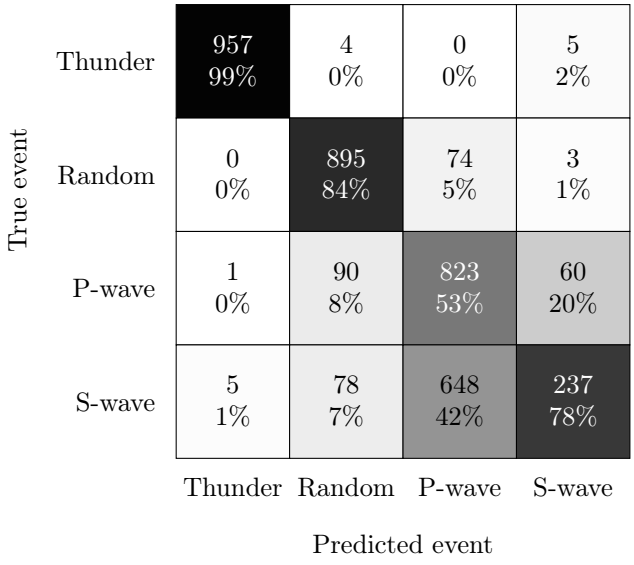
By visualizing the training process using training plots, we gained valuable insights into the model's behavior, performance, and convergence. These visualizations were instrumental in analyzing and understanding the training process, enabling us to make informed decisions and adjustments as needed.



(a) Diagram of architecture 3: conv represents a convolutional layer, fcl a fully connected layer, batch a batch-normalization layer.



(b) Confusion matrix of the CNN model based on the test data.



(c) Confusion matrix of the CNN model based on the smartphone network of Lima (Perù).

Figure 5.19: Training confusion matrix for the CNN based on architecture 3.

Architecture 4

In Figure 5.20, we present the initial architecture design of our model, showcasing its impressive performance on different datasets. After training, the model achieved an accuracy of 85.15% on the training data, indicating its ability to effectively learn from the training dataset. Furthermore, the model demonstrated strong performance on the validation data, achieving an accuracy of 81.85%. Moreover, it maintained a commendable accuracy of 81.76% on the test data, highlighting its capability to generalize well to unseen data.

To gain deeper insights into the model's performance, we analyzed the confusion matrix obtained from the test data, as shown in Figure 5.20. This matrix provided valuable insights into the model's ability to correctly predict positive and negative outcomes. Notably, we observed a low false negative rate of 1.4%, representing instances where the model incorrectly predicted negative outcomes. Additionally, the model exhibited an impressive ability to avoid false positive predictions, with a low false positive rate of 0.41%.

Figure 5.20 showcases the confusion matrix specifically for the smartphone network in Lima, Peru. In this case, we observed a false negative rate of 3.5%, indicating instances where the model incorrectly classified positive outcomes as negative. Furthermore, a false positive rate of 1.52% was recorded, representing cases where the model incorrectly predicted positive outcomes.

Throughout the model training process, we carefully selected options to optimize performance. We employed the Adam optimizer, utilizing an initial learning rate of 1×10^{-3}. To fine-tune the learning rate during training, we implemented a 'piecewise' learning rate schedule with a drop factor of 0.1 and a drop period of 5 epochs. This adaptive learning rate schedule ensured that the model adjusted its learning rate appropriately as training progressed.

To strike a balance between convergence and efficiency, we set the maximum number of epochs to 15. Additionally, we utilized a mini-batch size of 256, which facilitated efficient processing of the training data. We conducted evaluations using a separate validation dataset every 50 iterations, enabling us to closely monitor and assess the model's performance during training. To ensure the stability of the training process, we shuffled the data at the beginning of each epoch. This step prevented the model from being biased by the order of the training examples, promoting robust learning. Furthermore, we enabled verbose output to closely monitor the training progress and gather detailed information.

We visualized the training process using training plots, which aided in the analysis and understanding of the model's performance over time. This visualization provided valuable insights into the model's behavior, convergence, and potential areas for improvement. To prevent overfitting, we implemented early stopping with a patience of 7 epochs. This mechanism allowed the model to halt training if no significant improvements were observed during consecutive epochs, striking a balance between performance and generalization. Through the careful selection of training options and the use of early stopping, we aimed to ensure the model's robustness

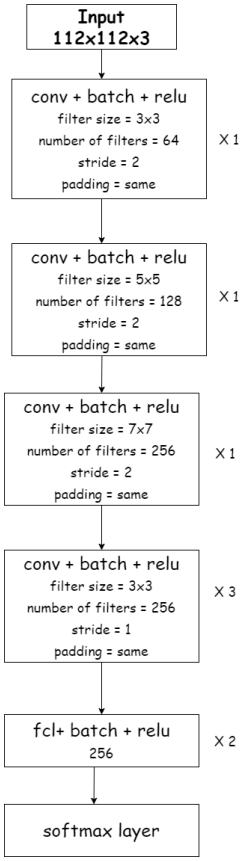
and achieve impressive results in our experiments.

Architecture 5

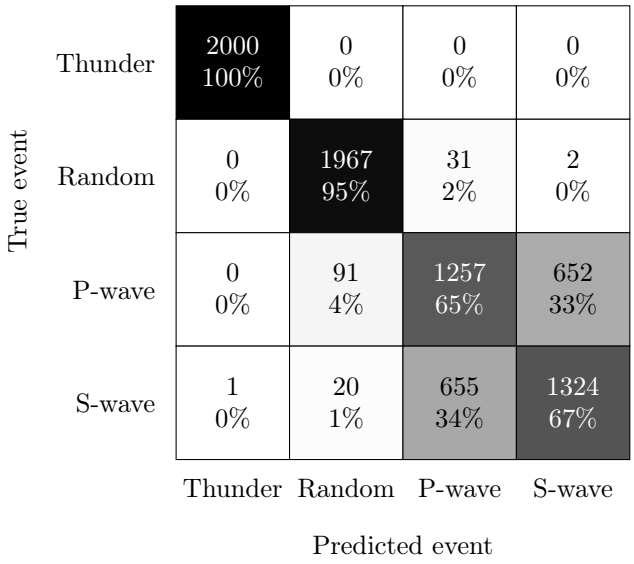
In Figure 5.21, we present the initial architecture design of our model, highlighting its performance on various datasets. After training, the model achieved an impressive accuracy of 83.59% on the training data, showcasing its capability to effectively learn from the training dataset. Moreover, the model demonstrated robust performance on the validation data, achieving an accuracy of 81.42%. Additionally, it maintained a commendable accuracy of 81.76% on the test data, demonstrating its ability to generalize well to new, unseen data. To gain a deeper understanding of the model's performance, we carefully analyzed the confusion matrix obtained from the test data, as shown in Figure 5.21. This matrix provided valuable insights into the model's ability to correctly predict positive and negative outcomes. Notably, we observed a low false negative rate of 1.08%, representing instances where the model incorrectly predicted negative outcomes. Additionally, the model exhibited a low false positive rate of 0.73%, indicating its ability to avoid false positive predictions.

Figure 5.21 provides insights into the model's performance on the smartphone network in Lima, Peru. In this scenario, we observed a false negative rate of 5.1%, which represents instances where the model incorrectly classified positive outcomes as negative. However, the model demonstrated an impressive ability to avoid false positive predictions, with a low false positive rate of 0.46%. Throughout the model training process, we carefully selected options to optimize performance. The Adam optimizer was utilized, with an initial learning rate of 1×10^{-3}. To fine-tune the learning rate, we implemented a 'piecewise' learning rate schedule, incorporating a drop factor of 0.1 and a drop period of 7 epochs. This adaptive learning rate adjustment ensured that the model effectively navigated the training process and efficiently updated its weights. To achieve a balance between convergence and efficiency, we set the maximum number of epochs to 30. Additionally, a mini-batch size of 256 was employed, enabling efficient processing of the training data. We regularly conducted evaluations using a separate validation dataset, performed every 50 iterations. This approach allowed us to closely monitor and assess the model's performance throughout the training process, ensuring its steady improvement.

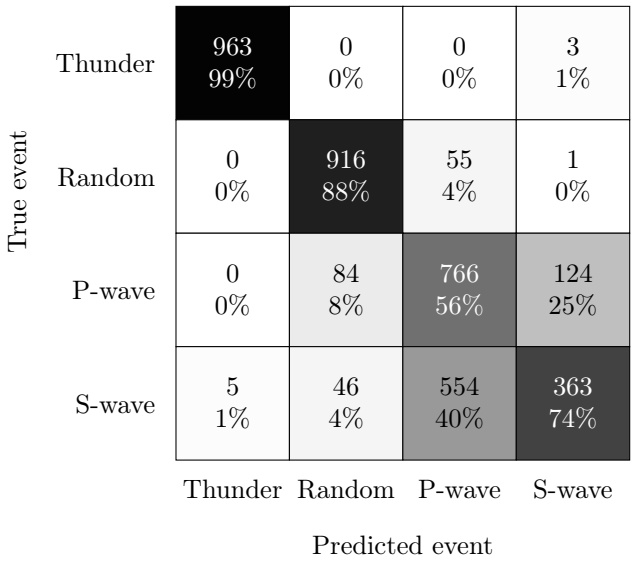
To enhance the stability of the training process, we shuffled the data at the beginning of each epoch. This step helped prevent any bias caused by the order of the training examples, ensuring that the model learned from diverse samples. Furthermore, we enabled verbose output to closely monitor the training progress and gather detailed information about the training dynamics. To gain a comprehensive understanding of the model's performance and behavior, we employed training plots to visualize the training process. These plots provided valuable insights into the model's progress, convergence, and potential areas for improvement, guiding us in making informed decisions during the training phase. To mitigate the risk of overfitting, we implemented early stopping with a "patience" of 7 epochs. This mechanism allowed the model to halt training if no significant improvements were observed over consecutive epochs. By applying early stopping, we ensured that the model achieved a balance between strong performance on the training data and generalization to new, unseen data, thereby enhancing its reliability and practicality.



(a) Diagram of architecture 4: conv represents a convolutional layer, fcl a fully connected layer, batch a batch-normalization layer.

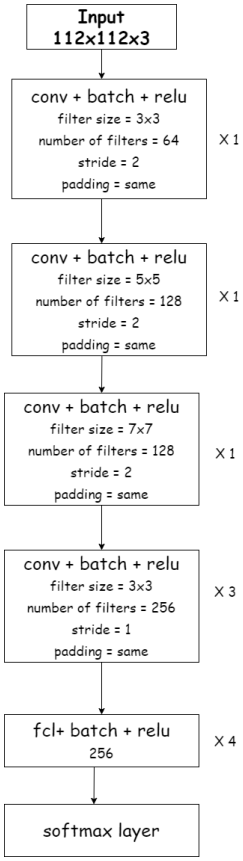


(b) Confusion matrix of the CNN model based on the test data.

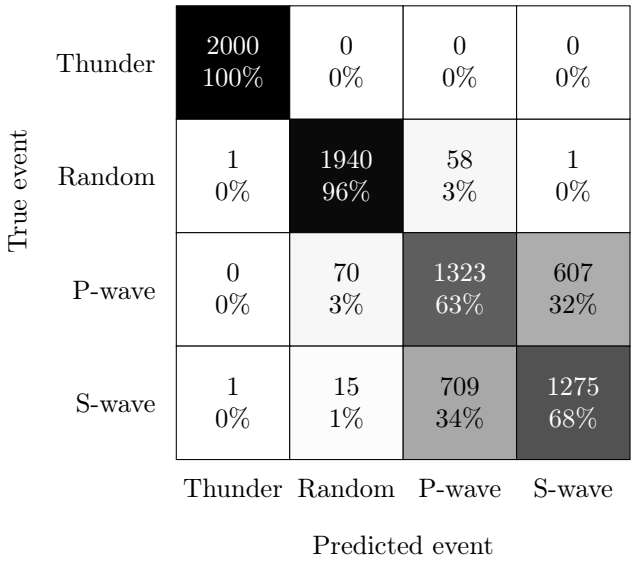


(c) Confusion matrix of the CNN model based on the smartphone network of Lima (Perù).

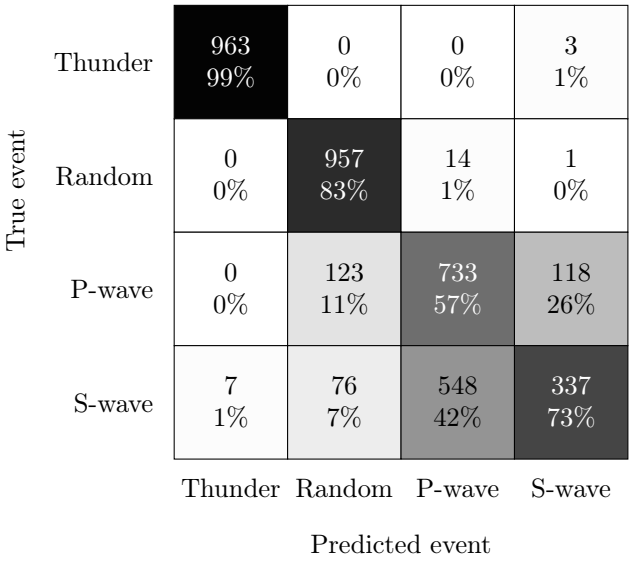
Figure 5.20: Training confusion matrix for the CNN based on architecture 4.



(a) Diagram of architecture 4: conv represents a convolutional layer, fcl a fully connected layer, batch a batch-normalization layer.



(b) Confusion matrix of the CNN model based on the test data.



(c) Confusion matrix of the CNN model based on the smartphone network of Lima (Perù).

Figure 5.21: Training confusion matrix for the CNN based on architecture 5.

Architecture 6

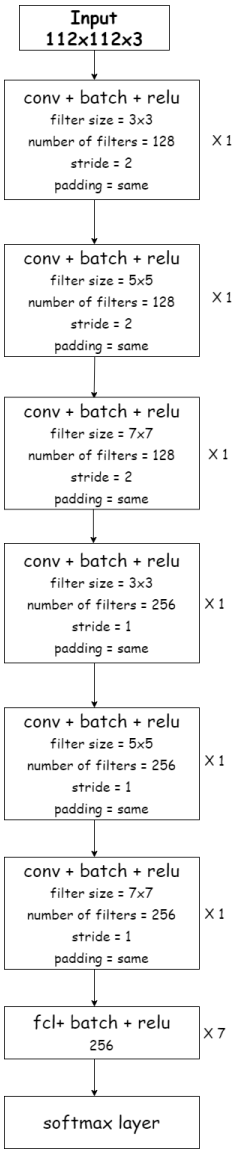
In Figure 5.22, we present the first architecture design of our model, which has undergone rigorous training and evaluation. The model achieved an impressive accuracy of 83.20% on the training data, demonstrating its effectiveness in learning from the training dataset. Additionally, the model showcased a strong performance on the validation data, achieving an accuracy of 81.46%. Furthermore, it maintained a commendable accuracy of 81.89% on the test data, highlighting its ability to generalize well to previously unseen data.

To delve deeper into the model's performance, we meticulously analyzed the confusion matrix obtained from the test data, as depicted in Figure 5.22. From this matrix, we observed a low false negative rate of 1%, which indicates instances where the model incorrectly predicted negative outcomes. Additionally, we noted a false positive rate of 0.6%, underscoring the model's capability to avoid false positive predictions. In Figure 5.22, we further evaluated the model's performance specifically on the smartphone network in Lima, Peru. Here, we observed a false negative rate of 1.6%, representing instances where the model incorrectly classified positive outcomes as negative. Additionally, a false positive rate of 3.2% was recorded, indicating cases where the model incorrectly predicted positive outcomes.

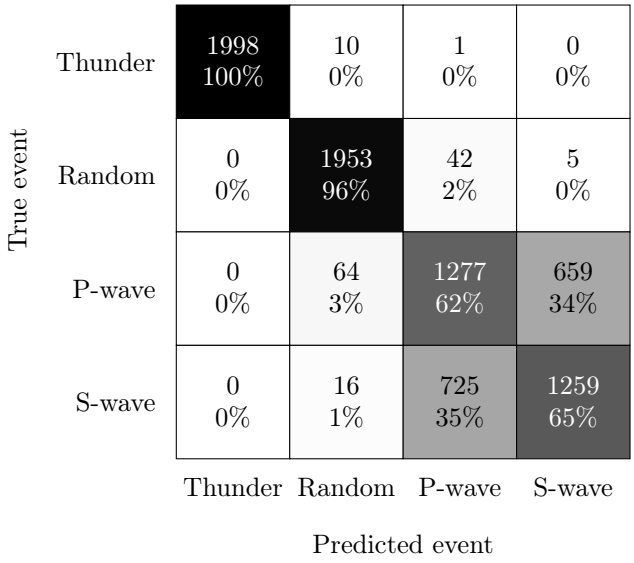
Throughout the model training process, we carefully selected various options to optimize its performance. The Adam optimizer, with an initial learning rate of 1×10^{-3}, served as a foundation for the training process. To fine-tune the learning rate, we adopted a 'piecewise' learning rate schedule, incorporating a drop factor of 0.1 and a drop period of 7 epochs. This adaptive learning rate adjustment proved crucial in steering the model effectively through the training process, enhancing its learning capabilities.

To ensure a balance between convergence and efficiency, we judiciously set the maximum number of epochs to 30. Additionally, we used a mini-batch size of 256, which facilitated efficient processing of the training data. Regular evaluations were conducted using a separate validation dataset, performed every 50 iterations. This approach allowed us to closely monitor and assess the model's performance throughout the training process. To enhance the stability of the training process, we consistently shuffled the data at the beginning of each epoch, thereby preventing any bias introduced by the order of the training examples. Enabling verbose output provided us with detailed information about the training progress, facilitating meticulous monitoring and data collection.

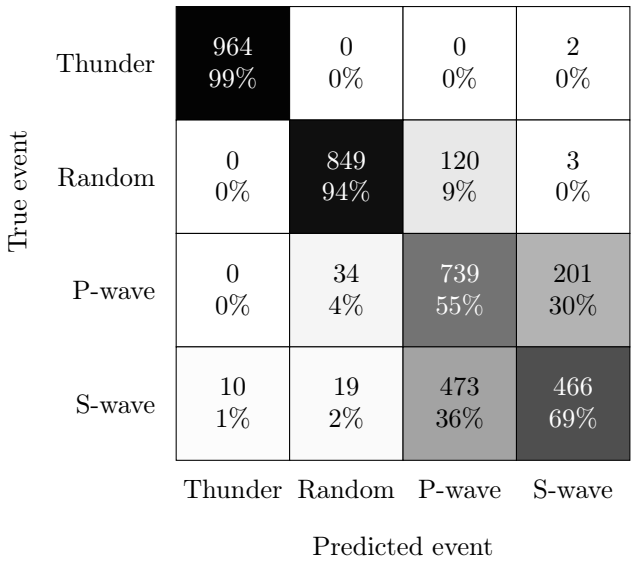
To gain a comprehensive understanding of the model's performance and behavior, we utilized training plots to visualize the training process. These plots proved invaluable in analyzing and interpreting the model's progress, convergence, and potential areas for improvement. To tackle overfitting, we wisely implemented early stopping with a patience of 7 epochs. This mechanism allowed the model to halt training if no significant improvements were observed over consecutive epochs, ensuring that the model attained a balance between strong performance on the training data and generalization to new, unseen data. This approach guaranteed the model's reliability and practicality in real-world scenarios.



(a) Diagram of architecture 4: conv represents a convolutional layer, fcl a fully connected layer, batch a batch-normalization layer.



(b) Confusion matrix of the CNN model based on the test data.



(c) Confusion matrix of the CNN model based on the smartphone network of Lima (Perù).

	Accuracy (%)	False positive (%)	False negative (%)
Architecture 1	78.85	2.30	0.60
Architecture 2	80.15	1.70	0.46
Architecture 3	80.31	1.60	0.57
Architecture 4	81.85	1.40	0.41
Architecture 5	81.73	1.08	0.73
Architecture 6	81.89	0.78	0.51

Table 5.1: Summary table of the result obtained on the test data.

	Accuracy (%)	False negative (%)	False positive (%)
Architecture 1	71.16	7.70	0.50
Architecture 2	74.20	8.70	0.38
Architecture 3	75.05	4.50	2.10
Architecture 4	77.53	3.50	1.52
Architecture 5	77.06	5.10	0.46
Architecture 6	76.47	1.50	3.58

Table 5.2: Summary table of the result obtained on the smartphone network of the city of Lima(Perù).

5.6 Comparing the different architectures

Performing a one-to-one comparison of various architectures can be a daunting task, given the exponential number of possible comparisons. Nevertheless, our objective is to present a comprehensive overview of the advantages and disadvantages of each architecture to facilitate the decision-making process.

Among the architectures, Architecture 2 stands out due to its smaller input image size (112x112x3), making it computationally efficient and requiring less memory during training and inference. Moreover, the reduced number of convolutional filters (64) in the initial layers, also present in architectures 3, 4, 5, and 6, contributes to faster training times, a valuable trait when computational resources are limited, or real-time processing is of utmost importance.

Conversely, Architecture 1 boasts a larger input image size (224x224x3), offering greater potential to process more detailed information. This characteristic enhances its capability for image classification tasks, potentially leading to higher accuracy and rendering it well-suited for applications that heavily rely on discerning fine-grained features.

Moving on, Architecture 6 distinguishes itself with an increased depth and multiple consecutive convolutional layers. This unique feature empowers the network to extract complex and abstract features from input images, fostering a profound understanding of intricate patterns and structures. Consequently, Architecture 6 becomes a compelling choice for tasks demanding a deep and nuanced comprehension of image content.

Furthermore, Architecture 6's integration of larger fully connected layers (256 units) in the latter stages further enhances its representation power for classification tasks. The ability to capture and learn intricate relationships between features provides a competitive edge when dealing with complex datasets, potentially leading to improved performance.

Selecting the optimal architecture depends on carefully considering the specific needs of the application and the available computational resources. Architecture 2's efficiency and real-time processing capabilities make it a compelling choice for certain scenarios, while Architecture 1 and Architecture 6 offer distinct advantages in capturing detailed information and tackling complex image classification tasks, respectively. By thoroughly evaluating these trade-offs, researchers and practitioners can make informed decisions aligning with their goals and requirements.

5.7 Real data example

The proposed methodology is applied to both true and false detections made by EQN in two different scenarios: Puerto Rico and Turkey.

In the case of Puerto Rico, we consider the earthquake that occurred on November 19, 2021, at 09:03:23 UTC. Figure 5.23 shows the geographical representation of the smartphone network in Puerto Rico, where the triggering smartphones (27) and the active smartphones (378) are visualized. The discrete pattern of wave propagation towards the west can be observed.

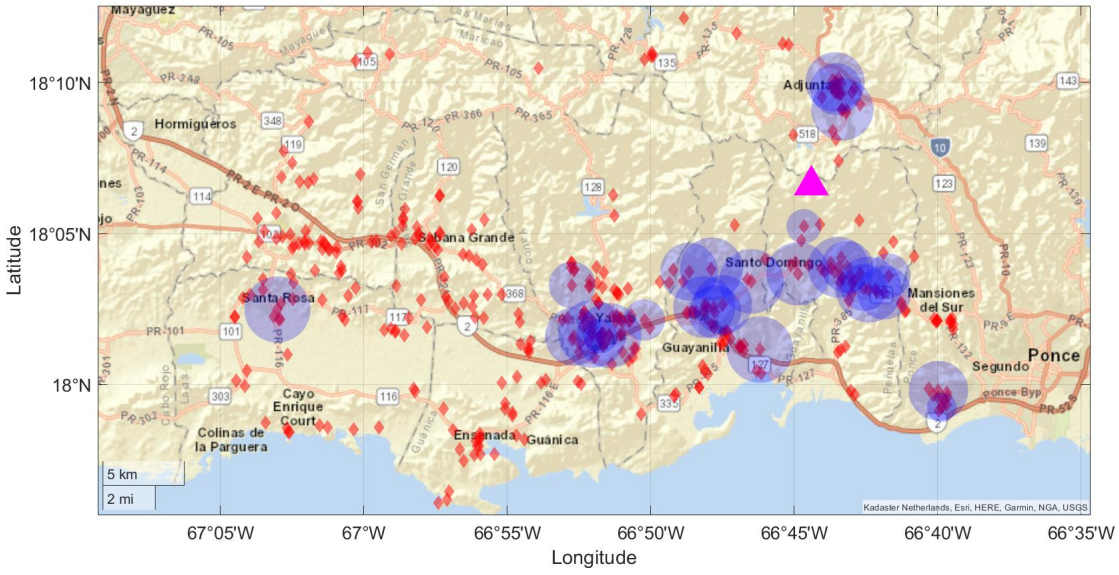


Figure 5.23: Geographical representation of the smartphone network of Puerto Rico. Smartphones that contributed to the earthquake detection are represented by circles with diameters proportional to the triggering time. Active smartphones are represented by red diamond markers. The magenta triangular marker represents the simulated epicenter location.

Table 5.3 provides a summary of the classification results and probabilities obtained for various architecture designs in the Puerto Rico scenario. Architecture 1 classifies the event as an S-wave with a probability of 0.68 and an execution time of 0.50 seconds. Similarly, architectures 2 and 3 also classify the event as an S-wave with probabilities of 0.95 and 1, respectively, and execution times of 0.26 and 0.30 seconds, respectively.

Architectures 4 and 5 classify the event as thunder with probabilities of 0.86 and 0.96, respectively, and execution times of 0.31 and 0.40 seconds, respectively. Architecture 6 classifies the event as random with a probability of 0.43 and an execution time of 0.63 seconds. Table 5.3 presents a summary table of the probabilities obtained during the classification of the smartphone network in Puerto Rico.

	Classification class	**Probability**	**Classification time (s)**
Architecture 1	S-wave	0.68	0.50
Architecture 2	S-wave	0.95	0.26
Architecture 3	S-wave	1.00	0.30
Architecture 4	Thunder	0.86	0.31
Architecture 5	Thunder	0.96	0.40
Architecture 6	Random	0.43	0.63

Table 5.3: Summary table of the results obtained on the smartphone network of Puerto Rico.

Moving on to the Turkey scenario, we analyze two earthquakes: the M7.8 earthquake on February 6th, 2023, and the M6.3 earthquake on February 20th, 2023. Figures 5.24 and 5.25 display the detection data in terms of active and triggering smartphones for each event.

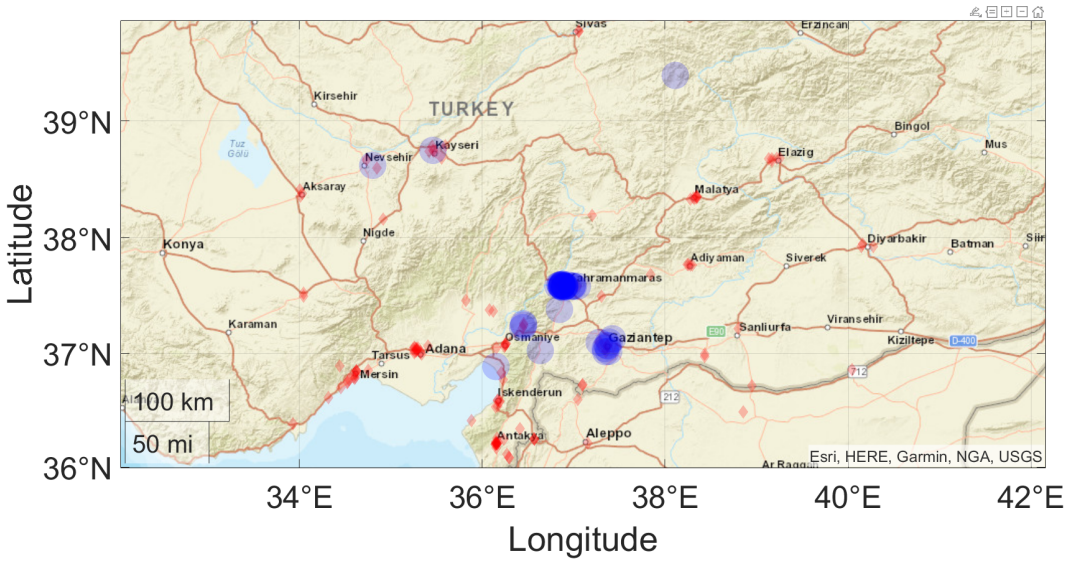


Figure 5.24: EQN detection data for the M7.8 event of February 6th, 2023, in Turkey. Blue circles represent triggering smartphones, while red diamonds represent active smartphones at the EQN detection time. The diameter of the circles is proportional to the triggering time.

The classification results for both earthquakes in the Turkey scenario are reported in Table 5.4 and 5.5. For both events, most of the CNN architectures classify the events as detections based on the S-wave with high probabilities. Although, due to the high magnitude, detections likely occurred on the P-wave as well, but at a short distance between the epicentre and detection location, P- and S-waves are nearly concurrent. The classification error is irrelevant in this case since both P- and S-waves are related to a true earthquake, and an alert was sent to citizens.

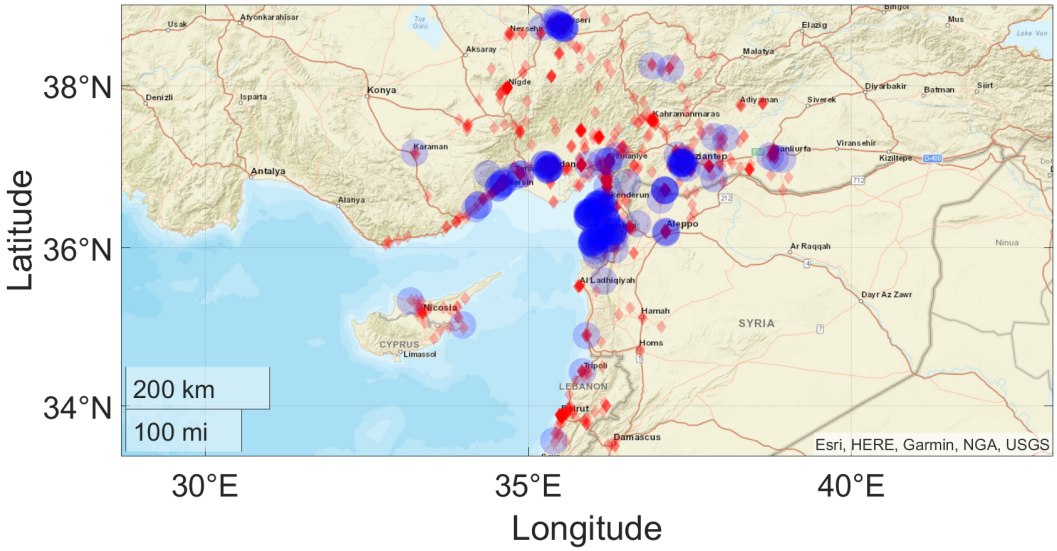


Figure 5.25: EQN detection data for the M6.3 event of February 20th, 2023, in Turkey. Blue circles represent triggering smartphones, while red diamonds represent active smartphones at the EQN detection time. The diameter of the circles is proportional to the triggering time.

	Classification	Probability	Classification time (s)
Architecture 1	P-wave	0.60	0.50
Architecture 2	Thunder	0.99	0.26
Architecture 3	S-wave	0.56	0.30
Architecture 4	S-wave	0.69	0.31
Architecture 5	S-wave	0.66	0.40
Architecture 6	S-wave	0.64	0.60

Table 5.4: Classification results for the EQN detection related to the M7.8 earthquake of February 6th, 2023, with an epicenter in Turkey.

	Classification	Probability	Classification time (s)
Architecture 1	P-wave	0.73	0.50
Architecture 2	S-wave	0.84	0.26
Architecture 3	S-wave	0.62	0.30
Architecture 4	S-wave	0.61	0.31
Architecture 5	S-wave	0.65	0.40
Architecture 6	S-wave	0.56	0.60

Table 5.5: Classification results for the EQN detection related to the M6.3 earthquake of February 20th, 2023, with an epicenter in Turkey.

Chapter 6

Comparison of the two methods

After exploring two distinct approaches to address a common problem, it is natural to conduct a thorough comparison in order to determine the most suitable approach. Throughout our investigations, we found that the statistical approach yielded quite satisfactory results. Notably, we were able to estimate the epicenter parameters with a high degree of accuracy while achieving a low probability of committing type-1 errors (false positives) at 1% and type-2 errors (false negatives) at 0.8%. The execution time of approximately 1.02 seconds was also reasonable. However, it is essential to acknowledge that as the number of smartphones increases, the execution time also shows an upward trend, raising concerns about scalability.

Given the growing popularity of deep-learning techniques, particularly the use of Convolutional Neural Networks (CNNs) in traditional earthquake early warning systems, we ventured into exploring this alternative approach within a smartphone network-based system. Our analyses on simulated data, as described in section 5.5.1, revealed that the CNN approach demonstrated results that were comparable to the statistical method, with the added advantage of a relatively improved execution time, averaging around 0.4 seconds. This represents a significant reduction, nearly halving the time required compared to the statistical method, regardless of the number of smartphones involved.

However, it is crucial to consider the trade-off encountered in generalizing the CNN approach to smartphone networks with diverse geometries. This trade-off revolves around the acceptable probability of errors, particularly false positives and false negatives. Our observations showed that even the best-performing architecture, architecture 6, presented a trade-off with 0.78% false positives and 0.51% false negatives on the test data, and 1.5% false positives and 3.58% false negatives on the Lima data. This trade-off was evident across all the explored architectures.

Additionally, when it came to estimating epicenter parameters, the CNN approach did not yield satisfactory results. Training CNN architectures on simulated smartphone networks allowed us to avoid the need to create a specific architecture for each possible smartphone network, which would have been less realistic and practical. Nevertheless, the results obtained through this CNN approach fell short of expectations, and thus

far, the statistical approach remains the most effective, despite not utilizing all available data.

In conclusion, our comparative analysis indicates that the statistical approach currently outperforms the CNN approach in terms of epicentre parameter estimation and the probability of error. However, the CNN approach shows promise with its faster execution time and potential for handling smartphone networks of varying geometries. It is evident that both approaches have their strengths and limitations, and further research and development are necessary to optimize and enhance their respective performances. As we continue to explore innovative solutions in earthquake early warning systems, the ultimate goal remains to maximize accuracy, efficiency, and the utilization of available data, ensuring the safety and well-being of communities worldwide.

Chapter 7

Conclusions

This thesis represents a significant contribution to the field of earthquake classification within smartphone-based earthquake early warning systems. The methodology developed here encompasses two distinct and complementary approaches: a statistical method and a deep-learning method.

The statistical methodology employed in this research draws on fundamental principles such as maximum likelihood estimation, hypothesis testing, and Monte Carlo simulations. Its elegant simplicity and computational efficiency enable real-time classification and estimation of essential earthquake parameters, making it well-suited for practical implementation in operational systems. Despite its merits, this approach has certain limitations, particularly in its exclusive focus on the triggering smartphones while neglecting potentially valuable information from active non-triggering smartphones. By harnessing this untapped data, we could potentially achieve more accurate estimates of epicenter and depth, thus further enhancing the overall performance of the statistical model. As a result, future endeavors will delve into exploiting this unexplored data to refine and optimize the statistical framework.

In contrast, the deep-learning method leverages the full dataset, incorporating both trigger and active data, and adopts an image generation-based approach for earthquake classification. While promising outcomes were observed in simulated datasets using various convolutional neural network (CNN) architectures, real dataset testing has exposed challenges in balancing false positives and false negatives. Addressing this inherent trade-off and achieving better generalization to real-world datasets is the focus of future research. By simulating a diverse range of network structures and expanding the training datasets, we aim to enhance the deep-learning model's robustness and adaptability to real-world scenarios.

This thesis is a moderate step towards refining earthquake early warning systems, as it combines both statistical and deep-learning approaches, thereby offering a comprehensive and multi-faceted solution to earthquake classification challenges. The insights gained from this research hold great promise for real-time earthquake detection and parameter estimation. Furthermore, the continual exploration of parameter estimation techniques and earthquake magnitude estimation, as well as the ongoing fine-tuning of deep-

learning models for real-world datasets, will undoubtedly propel the effectiveness and reliability of earthquake early warning systems to new heights. As a result, communities worldwide will benefit from enhanced disaster preparedness, reduced risks, and more timely and accurate alerts in the face of seismic events. This work, therefore, serves as a moderate stepping stone in the advancement of earthquake monitoring and response strategies, contributing to the broader goals of seismic hazard mitigation and public safety.

Bibliography

[1] C. Malalgoda, D. Amaratunga, and R. Haigh. "Challenges in creating a disaster resilient built environment". In: *Procedia Economics and Finance* 18 (2014), pp. 736–744.

[2] P. Gasparini, G. Manfredi, J. Zschau, et al. *Earthquake early warning systems*. Springer, 2007.

[3] R. M. Allen and H. Kanamori. "The potential for earthquake early warning in southern California". In: *Science* 300.5620 (2003), pp. 786–789.

[4] H. Kanamori. "Real-time seismology and earthquake damage mitigation". In: *Annual Review of Earth and Planetary Sciences* 33 (2005), pp. 195–214.

[5] Y.-M. Wu and H. Kanamori. "Experiment on an onsite early warning method for the Taiwan early warning system". In: *Bulletin of the Seismological Society of America* 95.1 (2005), pp. 347–353.

[6] Y.-M. Wu and H. Kanamori. "Rapid assessment of damage potential of earthquakes in Taiwan from the beginning of P waves". In: *Bulletin of the Seismological Society of America* 95.3 (2005), pp. 1181–1185.

[7] M. Böse et al. "Real-time testing of the on-site warning algorithm in southern California and its performance during the July 29 2008 Mw5. 4 Chino Hills earthquake". In: *Geophysical Research Letters* 36.5 (2009).

[8] A. Zollo et al. "A threshold-based earthquake early warning using dense accelerometer networks". In: *Geophysical Journal International* 183.2 (2010), pp. 963–974.

[9] M. Picozzi. "An attempt of real-time structural response assessment by an interferometric approach: A tailor-made earthquake early warning for buildings". In: *Soil Dynamics and Earthquake Engineering* 38 (2012), pp. 109–118.

[10] P. Brondi et al. "Predicting the macroseismic intensity from early radiated P wave energy for on-site earthquake early warning in Italy". In: *Journal of Geophysical Research: Solid Earth* 120.10 (2015), pp. 7174–7189.

[11] A. Caruso et al. "An on-site alert level early warning system for Italy". In: *Journal of Geophysical Research: Solid Earth* 122.3 (2017), pp. 2106–2118.

[12] F. Finazzi. "The earthquake network project: Toward a crowdsourced smartphone-based earthquake early warning system". In: *Bulletin of the Seismological Society of America* 106.3 (2016), pp. 1088–1099.

[13] F. Finazzi. "The earthquake network project: A platform for earthquake early warning, rapid impact assessment, and search and rescue". In: *Frontiers in Earth Science* 8 (2020), p. 243.

[14] R. M. Allen and M. Stogaitis. "Global growth of earthquake early warning". In: *Science* 375.6582 (2022), pp. 717–718.

[15] M. S. Abdalzaher et al. "Employing Machine Learning and IoT for Earthquake Early Warning System in Smart Cities". In: *Energies* 16.1 (2023), p. 495.

[16] S. M. Mousavi and G. C. Beroza. "Machine Learning in Earthquake Seismology". In: *Annual Review of Earth and Planetary Sciences* 51 (2023).

[17] M. Carratù et al. "A deep learning approach for the development of an Early Earthquake Warning system". In: *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. IEEE. 2022, pp. 1–6.

[18] N. Hast. "The state of stress in the upper part of the earth's crust". In: *Tectonophysics* 8.3 (1969), pp. 169–211.

[19] S. R. Taylor. "Abundance of chemical elements in the continental crust: a new table". In: *Geochimica et cosmochimica acta* 28.8 (1964), pp. 1273–1285.

[20] R. L. Rudnick and D. M. Fountain. "Nature and composition of the continental crust: a lower crustal perspective". In: *Reviews of geophysics* 33.3 (1995), pp. 267–309.

[21] A. H. Lachenbruch. "Crustal temperature and heat production: Implications of the linear heat-flow relation". In: *Journal of Geophysical Research* 75.17 (1970), pp. 3291–3300.

[22] C. M. Jarchow and G. A. Thompson. "The nature of the Mohorovicic discontinuity". In: *Annual Review of Earth and Planetary Sciences* 17.1 (1989), pp. 475–506.

[23] L. D. Hale and G. A. Thompson. "The seismic reflection character of the continental Mohorovicic discontinuity". In: *Journal of Geophysical Research: Solid Earth* 87.B6 (1982), pp. 4625–4635.

[24] F. A. Cook et al. "How the crust meets the mantle: Lithoprobe perspectives on the Mohorovičić discontinuity and crust–mantle transitionThis article is one of a series of papers published in this Special Issue on the theme Lithoprobe—parameters, processes, and the evolution of a continent." In: *Canadian Journal of Earth Sciences* 47.4 (2010), pp. 315–351.

[25] Y. Kanazawa and M. Kamitani. "Rare earth minerals and resources in the world". In: *Journal of alloys and compounds* 408 (2006), pp. 1339–1343.

[26] A. Levinson. "A system of nomenclature for rare-earth minerals". In: *American Mineralogist: Journal of Earth and Planetary Materials* 51.1-2 (1966), pp. 152–158.

[27] M. Moser and L. Völgyesi. "The inner structure of the Earth". In: *Periodica Polytechnica Chemical Engineering* 26.3-4 (1982), pp. 155–204.

[28] R. Walcott. "Isostatic response to loading of the crust in Canada". In: *Canadian Journal of Earth Sciences* 7.2 (1970), pp. 716–727.

[29] A. W. Hofmann. "Chemical differentiation of the Earth: the relationship between mantle, continental crust, and oceanic crust". In: *Earth and Planetary Science Letters* 90.3 (1988), pp. 297–314.

[30] L. Völgyesi and M. Moser. "The inner structure of the Earth". In: *Periodica Polytechnica Chemical Engineering* 26.3 (1982).

[31] N. I. Christensen and M. H. Salisbury. "Structure and constitution of the lower oceanic crust". In: *Reviews of Geophysics* 13.1 (1975), pp. 57–86.

[32] P. J. Fox, E. Schreiber, and J. Peterson. "The geology of the oceanic crust: compressional wave velocities of oceanic rocks". In: *Journal of Geophysical Research* 78.23 (1973), pp. 5155–5172.

[33] S. M. Peacock, T. Rushmer, and A. B. Thompson. "Partial melting of subducting oceanic crust". In: *Earth and planetary science letters* 121.1-2 (1994), pp. 227–244.

[34] G. Kimura and J. Ludden. "Peeling oceanic crust in subduction zones". In: *Geology* 23.3 (1995), pp. 217–220.

[35] M. W. Schmidt and S. Poli. "Generation of mobile components during subduction of oceanic crust". In: *Treatise on geochemistry* 3 (2003), p. 659.

[36] E. Klein. "Geochemistry of the igneous oceanic crust". In: *Treatise on geochemistry* 3 (2003), p. 659.

[37] R. L. Rudnick. "Making continental crust". In: *Nature* 378.6557 (1995), pp. 571–578.

[38] K. H. Wedepohl. "The composition of the continental crust". In: *Geochimica et cosmochimica Acta* 59.7 (1995), pp. 1217–1232.

[39] S. R. Taylor and S. M. McLennan. "The continental crust: its composition and evolution". In: (1985).

[40] B. GUTEl'iBERG. "Uber Erdbebenwellen. VII. A. Beobachtungen an Registrierungen von Fernbeben in Gottingen und Folderungen tiber die Konstitution des Erdkorpers. Kaehrichten K". In: *Gesell. Wiss. Gottingen. Math.-Physik Klasse* (1914), pp. 166–218.

[41] C. AH. *Phvsics of the Earth and Planets*. London, 1973.

[42] A. E. Ringwood. *Origin of the Earth and Moon*. Springer Science & Business Media, 2012.

[43] F. Birch. "The Earth's mantle". In: *Eos, Transactions American Geophysical Union* 35.1 (1954), pp. 79–85.

[44] D. L. Anderson. "Chemical composition of the mantle". In: *Journal of Geophysical Research: Solid Earth* 88.S01 (1983), B41–B52.

[45] F. Birch. "Density and composition of mantle and core". In: *Journal of geophysical research* 69.20 (1964), pp. 4377–4388.

[46] Y.-S. Zhang and T. Tanimoto. "High-resolution global upper mantle structure and plate tectonics". In: *Journal of Geophysical Research: Solid Earth* 98.B6 (1993), pp. 9793–9823.

[47] P. M. Shearer. "Seismic imaging of upper-mantle structure with new evidence for a 520-km discontinuity". In: *Nature* 344 (1990), pp. 121–126.

[48] D. Pearson and G. Nowell. "The continental lithospheric mantle: characteristics and significance as a mantle reservoir". In: *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 360.1800 (2002), pp. 2383–2410.

[49] S. Y. O'Reilly et al. "Are lithospheres forever? Tracking changes in subcontinental lithospheric mantle through time". In: *GSA today* 11.4 (2001), pp. 4–10.

[50] C. P. Conrad and C. Lithgow-Bertelloni. "Influence of continental roots and asthenosphere on plate-mantle coupling". In: *Geophysical Research Letters* 33.5 (2006).

[51] J. Ita and L. Stixrude. "Petrology, elasticity, and composition of the mantle transition zone". In: *Journal of Geophysical Research: Solid Earth* 97.B5 (1992), pp. 6849–6866.

[52] E. Ito and T. Katsura. "A temperature profile of the mantle transition zone". In: *Geophysical Research Letters* 16.5 (1989), pp. 425–428.

[53] E. J. Garnero and A. K. McNamara. "Structure and dynamics of Earth's lower mantle". In: *science* 320.5876 (2008), pp. 626–628.

[54] T. Masters. "Structure of the earth-Mantle and core". In: *Reviews of Geophysics* 29 (1991), pp. 671–679.

[55] D. Stevenson. "Models of the Earth's core". In: *Science* 214.4521 (1981), pp. 611–619.

[56] R. Jeanloz. "The nature of the Earth's core". In: *Annual Review of Earth and Planetary Sciences* 18.1 (1990), pp. 357–386.

[57] D. Rubie, F. Nimmo, and H. Melosh. "Formation of Earth's core". In: *Evolution of the Earth* 9 (2007), pp. 51–90.

[58] J. A. Jacobs. *The Earth's core*. Academic Press, 1987.

[59] J. Jacobs. "The Earth's inner core". In: *Nature* 172 (1953), pp. 297–298.

[60] L. Stixrude, E. Wasserman, and R. E. Cohen. "Composition and temperature of Earth's inner core". In: *Journal of Geophysical Research: Solid Earth* 102.B11 (1997), pp. 24729–24739.

[61] A. Hallam. "Alfred Wegener and the hypothesis of continental drift". In: *Scientific American* 232.2 (1975), pp. 88–97.

[62] M. T. Greene. *Alfred Wegener: Science, exploration, and the theory of continental drift*. JHU Press, 2015.

[63] W. R. Dickinson. "Plate tectonics and sedimentation". In: (1974).

[64] D. A. Falvey. "The development of continental margins in plate tectonic theory". In: *The APPEA Journal* 14.1 (1974), pp. 95–106.

[65] J. F. Luhr et al. "Active rifting in southwestern Mexico: Manifestations of an incipient eastward spreading-ridge jump". In: *Geology* 13.1 (1985), pp. 54–57.

[66] T. J. Wright et al. "Geophysical constraints on the dynamics of spreading centres from rifting episodes on land". In: *Nature Geoscience* 5.4 (2012), pp. 242–250.

[67] J. M. Stock and J. Lee. "Do microplates in subduction zones leave a geological record?" In: *Tectonics* 13.6 (1994), pp. 1472–1487.

[68] T. Atwater and J. Stock. "Pacific-North America plate tectonics of the Neogene southwestern United States: an update". In: *International Geology Review* 40.5 (1998), pp. 375–402.

[69] R. W. Day. *Geotechnical earthquake engineering handbook: with the 2012 International building code*. McGraw-Hill Education, 2012.

[70] C. Childs, J. Watterson, and J. Walsh. "A model for the structure and development of fault zones". In: *Journal of the Geological Society* 153.3 (1996), pp. 337–340.

[71] C. Childs, J. Walsh, and J. Watterson. "Complexity in fault zone structure and implications for fault seal prediction". In: *Norwegian petroleum society special publications*. Vol. 7. Elsevier, 1997, pp. 61–72.

[72] D. A. Cole Jr and P. V. Lade. "Influence zones in alluvium over dip-slip faults". In: *Journal of Geotechnical Engineering* 110.5 (1984), pp. 599–615.

[73] P. V. Lade, D. A. Cole Jr, and D. Cummings. "Multiple failure surfaces over dip-slip faults". In: *Journal of Geotechnical Engineering* 110.5 (1984), pp. 616–627.

[74] H. Kanamori and E. E. Brodsky. "The physics of earthquakes". In: *Physics today* 54.6 (2001), pp. 34–40.

[75] H. Kanamori and E. E. Brodsky. "The physics of earthquakes". In: *Reports on progress in physics* 67.8 (2004), p. 1429.

[76] A. Barka and K. Kadinsky-Cade. "Strike-slip fault geometry in Turkey and its influence on earthquake activity". In: *Tectonics* 7.3 (1988), pp. 663–684.

[77] D. Needham, G. Yielding, and B. Freeman. "Analysis of fault geometry and displacement patterns". In: *Geological Society, London, Special Publications* 99.1 (1996), pp. 189–199.

[78] J. Dewey and A. C. ŞENGÖR. "Aegean and surrounding regions: complex multiplate and continuum tectonics in a convergent zone". In: *Geological Society of America Bulletin* 90.1 (1979), pp. 84–92.

[79] J. Havskov and L. Ottemoller. *Routine data processing in earthquake seismology: with sample data, exercises and software*. Springer Science & Business Media, 2010.

[80] F. E. Richart, J. R. Hall, and R. D. Woods. "Vibrations of soils and foundations". In: (1970).

[81] R. A. Uhrhammer and E. R. Collins. "Synthesis of Wood-Anderson seismograms from broadband digital records". In: *Bulletin of the Seismological Society of America* 80.3 (1990), pp. 702–716.

[82] M. Trifunac and D. Hudson. "Analysis of the Station No. 2 seismoscope record—1966, Parkfield, California, earthquake". In: *Bulletin of the Seismological Society of America* 60.3 (1970), pp. 785–794.

[83] B. Gutenberg and C. F. Richter. "Earthquake magnitude, intensity, energy, and acceleration". In: *Bulletin of the Seismological society of America* 32.3 (1942), pp. 163–191.

[84] C. F. Richter. "An instrumental earthquake magnitude scale". In: *Bulletin of the seismological society of America* 25.1 (1935), pp. 1–32.

[85] B. Gutenberg. "Amplitudes of surface waves and magnitudes of shallow earthquakes". In: *Bulletin of the Seismological Society of America* 35.1 (1945), pp. 3–12.

[86] B. Gutenberg and C. F. Richter. "Earthquake magnitude, intensity, energy, and acceleration: (Second paper)". In: *Bulletin of the seismological society of America* 46.2 (1956), pp. 105–145.

[87] K. Aki. "Analysis of the seismic coda of local earthquakes as scattered waves". In: *Journal of geophysical research* 74.2 (1969), pp. 615–631.

[88] C. R. Real and T.-L. Teng. "Local Richter magnitude and total signal duration in southern California". In: *Bulletin of the Seismological Society of America* 63.5 (1973), pp. 1809–1827.

[89] H. Kanamori. "Magnitude scale and quantification of earthquakes". In: *Tectonophysics* 93.3-4 (1983), pp. 185–199.

[90] I. L. Cifuentes. "The 1960 chilean earthquakes". In: *Journal of Geophysical Research: Solid Earth* 94.B1 (1989), pp. 665–680.

[91] S. E. Barrientos and S. N. Ward. "The 1960 Chile earthquake: inversion for slip distribution from surface deformation". In: *Geophysical Journal International* 103.3 (1990), pp. 589–598.

[92] Y. Nakamura. "On the urgent earthquake detection and alarm system (UrEDAS)". In: *Proc. of the 9th World Conference on Earthquake Engineering*. Vol. 7. 1988, pp. 673–678.

[93] J. M. Espinosa Aranda et al. "Mexico City seismic alert system". In: *Seismological research letters* 66 (1995), pp. 42–53.

[94] O. Velazquez et al. "A review of the technical and socio-organizational components of earthquake early warning systems". In: *Frontiers in Earth Science* (2020), p. 445.

[95] M. Moghtaderi et al. "LETTER TO EDITOR: SKIN TEST REACTIVITY TO FUNGAL AEROALLERGENS IN ASTHMATIC CHILDREN IN SOUTHERN IRAN". In: (2010).

[96] R. M. Allen et al. "The status of earthquake early warning around the world: An introductory overview". In: *Seismological research letters* 80.5 (2009), pp. 682–693.

[97] A. Zollo, M. Lancieri, and S. Nielsen. "Earthquake magnitude estimation from peak amplitudes of very early seismic signals on strong motion records". In: *Geophysical Research Letters* 33.23 (2006).

[98] Y.-M. Wu and H. Kanamori. "Development of an earthquake early warning system using real-time strong motion signals". In: *Sensors* 8.1 (2008), pp. 1–9.

[99] Y. Nakamura and J. Saita. "UrEDAS, the earthquake warning system: Today and tomorrow". In: *Earthquake early warning systems*. Springer, 2007, pp. 249–281.

[100] Y. Wu et al. "Experiment using the tau-c and Pd method for earthquake early warning in southern California". In: *Geophysical Journal International* 170 (2007), pp. 711–717.

[101] A. Overeem et al. "Crowdsourcing urban air temperatures from smartphone battery temperatures". In: *Geophysical Research Letters* 40.15 (2013), pp. 4081–4085.

[102] N. D. Lane et al. "A survey of mobile phone sensing". In: *IEEE Communications magazine* 48.9 (2010), pp. 140–150.

[103] F. Snik et al. "Mapping atmospheric aerosols with a citizen science network of smartphone spectropolarimeters". In: *Geophysical Research Letters* 41.20 (2014), pp. 7351–7358.

[104] S. E. Minson et al. "Crowdsourced earthquake early warning". In: *Science advances* 1.3 (2015), e1500036.

[105] E. S. Cochran et al. "Comparison between low-cost and traditional MEMS accelerometers: a case study from the M7. 1 Darfield, New Zealand, aftershock deployment". In: *Ann. Geophys* 54.6 (2011), pp. 728–737.

[106] A. D'Alessandro and G. D'Anna. "Suitability of low-cost three-axis MEMS accelerometers in strong-motion seismology: Tests on the LIS331DLH (iPhone) accelerometer". In: *Bulletin of the Seismological Society of America* 103.5 (2013), pp. 2906–2913.

[107] E. S. Cochran et al. "The quake-catcher network: Citizen science expanding seismic horizons". In: *Seismological Research Letters* 80.1 (2009), pp. 26–30.

[108] R. Clayton et al. "Community seismic network". In: *Annals of Geophysics* 54.6 (2011), pp. 738–747.

[109] F. Finazzi and A. Fassò. "A statistical approach to crowdsourced smartphone-based earthquake early warning systems". In: *Stochastic environmental research and risk assessment* 31.7 (2017), pp. 1649–1658.

[110] F. Finazzi et al. "A Probabilistic Framework for Modeling the Detection Capability of Smartphone Networks in Earthquake Early Warning". In: *Seismological Research Letters* (2022).

[111] M. Lancieri and A. Zollo. "A Bayesian approach to the real-time estimation of magnitude from the early P and S wave displacement peaks". In: *Journal of Geophysical Research: Solid Earth* 113.B12 (2008).

[112] C. Satriano et al. "Earthquake early warning: Concepts, methods and physical grounds". In: *Soil Dynamics and Earthquake Engineering* 31.2 (2011), pp. 106–118.

[113] R. Bossu et al. ""Shaking in 5 Seconds!"—Performance and User Appreciation Assessment of the Earthquake Network Smartphone-Based Public Earthquake Early Warning System". In: *Seismological Society of America* 93.1 (2022), pp. 137–148.

[114] M. D. Kohler et al. "Earthquake early warning ShakeAlert system: West coast wide production prototype". In: *Seismological Research Letters* 89.1 (2018), pp. 99–107.

[115] D. D. Given et al. *Revised technical implementation plan for the ShakeAlert system—An earthquake early warning system for the West Coast of the United States*. Tech. rep. US Geological Survey, 2018.

[116] G. M. Atkinson, C. B. Worden, and D. J. Wald. "Intensity prediction equations for North America". In: *Bulletin of the Seismological Society of America* 104.6 (2014), pp. 3084–3093.

[117] P. Tosi et al. "Macroseismic intensity assessment method for web questionnaires". In: *Seismological Research Letters* 86.3 (2015), pp. 985–990.

[118] T. I. Allen, D. J. Wald, and C. B. Worden. "Intensity attenuation for active crustal regions". In: *Journal of seismology* 16 (2012), pp. 409–433.

[119] R. Bossu et al. "Rapid public information and situational awareness after the November 26, 2019, Albania earthquake: lessons learned from the LastQuake system". In: *Frontiers in Earth Science* (2020), p. 235.

[120] R. Steed et al. "Analysis of detections by the Earthquake Network app between 2017-12-15 and 2020-01-31". In: (2021).

[121] L. Fallou et al. "A Questionnaire Survey of the Earthquake Network App's Users in Peru Following an M8 Earthquake in 2019". In: (2021).

[122] R. Durrett. *Stochastic calculus: a practical introduction*. Vol. 6. CRC press, 1996.

[123] J. V. Candy. *Bayesian signal processing: classical, modern, and particle filtering methods*. Vol. 54. John Wiley & Sons, 2016.

[124] M. J. Fisher and A. P. Marshall. "Understanding descriptive statistics". In: *Australian critical care* 22.2 (2009), pp. 93–97.

[125] P. Kaur, J. Stoltzfus, V. Yellapu, et al. "Descriptive statistics". In: *International Journal of Academic Medicine* 4.1 (2018), p. 60.

[126] P. Diaconis. "Application of the method of moments in probability and statistics". In: *Moments in mathematics* 37 (1987), pp. 125–142.

[127] X. Wang and Z. Peng. "Method of moments for estimating uncertainty distributions". In: *Journal of Uncertainty Analysis and Applications* 2.1 (2014), pp. 1–10.

[128] R. A. Fisher. "On the mathematical foundations of theoretical statistics". In: *Philosophical transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character* 222.594-604 (1922), pp. 309–368.

[129] H. Cramér. *Mathematical methods of statistics*. Vol. 26. Princeton university press, 1999.

[130] A. O'hagan and T. Leonard. "Bayes estimation subject to uncertainty about parameter constraints". In: *Biometrika* 63.1 (1976), pp. 201–203.

[131] P. Diaconis and D. Freedman. "On the consistency of Bayes estimates". In: *The Annals of Statistics* (1986), pp. 1–26.

[132] D. L. Donoho and I. M. Johnstone. "Minimax estimation via wavelet shrinkage". In: *The annals of Statistics* 26.3 (1998), pp. 879–921.

[133] J. Jiao et al. "Minimax estimation of functionals of discrete distributions". In: *IEEE Transactions on Information Theory* 61.5 (2015), pp. 2835–2885.

[134] G. Casella and R. L. Berger. *Statistical inference*. Cengage Learning, 2021.

[135] R. E. Walpole and R. H. Myers. *Probability and statistics for engineering and scientists*. MacMillan, 1985.

[136] M. A. Tanner. *Tools for statistical inference*. Vol. 3. Springer, 1993.

[137] B. W. Lindgren. *Statistical theory*. Routledge, 2017.

[138] D. C. Montgomery and G. C. Runger. *Applied statistics and probability for engineers*. John wiley & sons, 2010.

[139] W. L. Martinez and A. R. Martinez. *Computational statistics handbook with MATLAB*. Chapman and Hall/CRC, 2001.

[140] B. L. Kennett, E. Engdahl, and R. Buland. "Constraints on seismic velocities in the Earth from traveltimes". In: *Geophysical Journal International* 122.1 (1995), pp. 108–124.

[141] W. S. McCulloch and W. Pitts. "A logical calculus of the ideas immanent in nervous activity". In: *The bulletin of mathematical biophysics* 5 (1943), pp. 115–133.

[142] F. Rosenblatt. “The perceptron: a probabilistic model for information storage and organization in the brain.” In: *Psychological review* 65.6 (1958), p. 386.

[143] H.-D. Block. “The perceptron: A model for brain functioning. i”. In: *Reviews of Modern Physics* 34.1 (1962), p. 123.

[144] D. O. Hebb. *The organization of behavior: A neuropsychological theory*. Psychology Press, 2005.

[145] A. Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. " O'Reilly Media, Inc.", 2022.

[146] M. L. Minsky and S. A. Papert. *Perceptrons: expanded edition*. 1988.

[147] A. B. Novikoff. *On convergence proofs for perceptrons*. Tech. rep. STANFORD RESEARCH INST MENLO PARK CA, 1963.

[148] Z. Yanling, D. Bimin, and W. Zhanrong. “Analysis and study of perceptron to solve XOR problem”. In: *The 2nd International Workshop on Autonomous Decentralized System, 2002.* IEEE. 2002, pp. 168–173.

[149] S. Ruder. “An overview of gradient descent optimization algorithms”. In: *arXiv preprint arXiv:1609.04747* (2016).

[150] L. Bottou. “Stochastic gradient descent tricks”. In: *Neural networks: Tricks of the trade*. Springer, 2012, pp. 421–436.

[151] S. Hochreiter. “The vanishing gradient problem during learning recurrent neural nets and problem solutions”. In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6.02 (1998), pp. 107–116.

[152] Y. Hu et al. “Overcoming the vanishing gradient problem in plain recurrent networks”. In: *arXiv preprint arXiv:1801.06105* (2018).

[153] R. A. Dunne and N. A. Campbell. “On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function”. In: *Proc. 8th Aust. Conf. on the Neural Networks, Melbourne*. Vol. 181. Citeseer. 1997, p. 185.

[154] S. Sharma, S. Sharma, and A. Athaiya. “Activation functions in neural networks”. In: *towards data science* 6.12 (2017), pp. 310–316.

[155] D.-A. Clevert, T. Unterthiner, and S. Hochreiter. “Fast and accurate deep network learning by exponential linear units (elus)”. In: *arXiv preprint arXiv:1511.07289* (2015).

[156] B. Xu et al. “Empirical evaluation of rectified activations in convolutional network”. In: *arXiv preprint arXiv:1505.00853* (2015).

[157] X. Glorot and Y. Bengio. “Understanding the difficulty of training deep feedforward neural networks”. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings. 2010, pp. 249–256.

[158] S. K. Kumar. “On weight initialization in deep neural networks”. In: *arXiv preprint arXiv:1704.08863* (2017).

[159] S. Ioffe and C. Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *International conference on machine learning*. PMLR. 2015, pp. 448–456.

[160] C. F. G. D. Santos and J. P. Papa. "Avoiding overfitting: A survey on regularization methods for convolutional neural networks". In: *ACM Computing Surveys (CSUR)* 54.10s (2022), pp. 1–25.

[161] C. Garbin, X. Zhu, and O. Marques. "Dropout vs. batch normalization: an empirical study of their impact to deep learning". In: *Multimedia Tools and Applications* 79.19 (2020), pp. 12777–12815.

[162] P. J. Werbos. "Backpropagation through time: what it does and how to do it". In: *Proceedings of the IEEE* 78.10 (1990), pp. 1550–1560.

[163] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning representations by back-propagating errors". In: *nature* 323.6088 (1986), pp. 533–536.

[164] Y. LeCun et al. "Backpropagation applied to handwritten zip code recognition". In: *Neural computation* 1.4 (1989), pp. 541–551.

[165] N. Qian. "On the momentum term in gradient descent learning algorithms". In: *Neural networks* 12.1 (1999), pp. 145–151.

[166] Y. Nesterov. "A method for unconstrained convex minimization problem with the rate of convergence O (1/kˆ 2)". In: *Doklady an ussr*. Vol. 269. 1983, pp. 543–547.

[167] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[168] T. Chai and R. R. Draxler. "Root mean square error (RMSE) or mean absolute error (MAE)?–Arguments against avoiding RMSE in the literature". In: *Geoscientific model development* 7.3 (2014), pp. 1247–1250.

[169] C. J. Willmott and K. Matsuura. "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance". In: *Climate research* 30.1 (2005), pp. 79–82.

[170] T. Hastie et al. *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. Springer, 2009.

[171] G. P. Meyer. "An alternative probabilistic interpretation of the huber loss". In: *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*. 2021, pp. 5261–5269.

[172] K. P. Murphy. *Probabilistic Machine Learning: An introduction*. MIT Press, 2022. URL: `probml.ai`.

[173] D. H. Hubel. "Single unit activity in striate cortex of unrestrained cats". In: *The Journal of physiology* 147.2 (1959), p. 226.

[174] D. H. Hubel and T. N. Wiesel. "Receptive fields of single neurones in the cat's striate cortex". In: *The Journal of physiology* 148.3 (1959), p. 574.

[175] K. Fukushima and S. Miyake. "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition". In: *Competition and cooperation in neural nets*. Springer, 1982, pp. 267–285.

[176] Y. LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[177] M. Ahmed, V. Maume-Deschamps, and P. Ribereau. “Recognizing a spatial extreme dependence structure: A deep learning approach”. In: *Environmetrics* 33.4 (2022), e2714.

[178] S. M. Mousavi et al. “Earthquake transformer—an attentive deep-learning model for simultaneous earthquake detection and phase picking”. In: *Nature communications* 11.1 (2020), pp. 1–12.

[179] H. S. Kuyuk and O. Susumu. “Real-time classification of earthquake using deep learning”. In: *Procedia Computer Science* 140 (2018), pp. 298–305.

[180] O. M. Saad, A. G. Hafez, and M. S. Soliman. “Deep learning approach for earthquake parameters classification in earthquake early warning system”. In: *IEEE Geoscience and Remote Sensing Letters* 18.7 (2020), pp. 1293–1297.