Machine learning for fast and accurate assessment of earthquake source parameters

Implications for rupture predictability and early warning

DISSERTATION

zur Erlangung des akademischen Grades

doctor rerum naturalium
im Fach Informatik

eingereicht an der
Mathematisch-Naturwissenschaftlichen Fakultät
der Humboldt-Universität zu Berlin

von
M.Sc. Jannes Münchmeyer

Kommissarischer Präsident der Humboldt-Universität zu Berlin:
Prof. Dr. Peter Frensch

Dekan der Mathematisch-Naturwissenschaftlichen Fakultät:
Prof. Dr. Elmar Kulke

Gutachter:

1. Prof. Dr. Ulf Leser, Humboldt-Universität zu Berlin
2. Prof. Dr. Frederik Tilmann, Freie Universität Berlin
3. Prof. Dr. Gregory Beroza, Stanford University

Datum der Disputation: 16.08.2022

Abstract

Earthquakes are among the largest and most destructive natural hazards known to humankind. While records of earthquakes date back millennia, and systematic studies of earthquakes have been conducted for over a century, many questions about their nature remain open. One particularly interesting question is termed *rupture predictability*: to what extent is it possible to foresee the final size of an earthquake while it is still ongoing? This question is integral to earthquake early warning systems trying to provide information about ongoing earthquakes to places where shaking has not yet arrived, thereby allowing for last moment preparatory action. Still, research on this question so far has reached contradictory conclusions.

In recent years, the advent of big data and big data analysis techniques opened up novel opportunities for investigating rupture predictability. The amount of data available for earthquake research has grown exponentially during the last decades, as for many other scientific domains, reaching now tera- to petabyte scale, with future growth to be expected. This wealth of data, while making manual inspection infeasible, allows for data-driven analysis and complex models with high numbers of parameters. One class of these models are machine learning methods, in particular, deep learning methods. Deep learning has gained overwhelming interest across domains in the last decade, driven by new developments in the field. In seismology, it already led to considerable improvements upon previous methods for many analysis tasks. Nonetheless, the application of deep learning methods to seismological observables is still in its infancy.

In this thesis, we develop machine learning methods for the study of rupture predictability and the closely related task of earthquake early warning. We first study the calibration of a high-confidence magnitude scale in a post hoc scenario. For this, we develop a hybrid approach, based on mathematical optimisation and machine learning. Subsequently, we focus on real-time estimation models based on deep learning. We develop the transformer earthquake alerting model (TEAM), a method for earthquake early warning, estimating ground motion parameters directly from seismic waveforms. TEAM outperforms traditional early warning methods in terms of warning times and the relation between true, false and missed alerts. Based on TEAM, we develop TEAM-LM, a model for real-time location and magnitude estimation. TEAM-LM outperforms both classical approaches and previous deep learning approaches. Using TEAM-LM, we study the advantages and shortcomings of deep learning for earthquake assessment. While showing excellent average performance, deep learning models exhibits systematic mispredictions in face of data sparsity. In particular, large magnitudes are systematically underestimated. We discuss and evaluate strategies for mitigating this issue.

In the last step, we use TEAM-LM and the insights gained through its analysis to study rupture predictability. For this, we collate a dataset of teleseismic P wave arrivals, encompassing events and stations worldwide. We complement this analysis with results obtained from a deep learning model based on moment rate functions. Our analysis shows that earthquake ruptures are not predictable early on, but only once their peak moment release has been reached, after approximately half of their duration. Even then, potential further asperities can not be foreseen. While this thesis finds no rupture predictability, the methods developed within this work demonstrate how deep learning methods make a high-quality real-time assessment of earthquakes practically feasible. We hope that these results will allow improving future earthquake early warning systems, and thereby help to reduce the harm caused by earthquakes.

Zusammenfassung

Erdbeben gehören zu den größten und zerstörerischsten Naturgefahren auf diesem Planeten. Obwohl das Auftreten von Erdbeben seit Jahrtausenden dokumentiert ist und auch systematische Studien seit mehr als einem Jahrhundert durchgeführt werden, bleiben viele Fragen zu Erdbeben unbeantwortet. Eine besonders interessante Frage ist die *Vorhersagbarkeit von Brüchen*: Inwieweit ist es möglich, die endgültige Größe eines Bebens zu bestimmen, bevor der zugrundeliegende Bruchprozess endet? Diese Frage ist zentral für Frühwarnsysteme. Diese Systeme messen die ersten Erschütterungen des Bebens und senden Warnungen an Orte, an denen starke Erschütterungen zu erwarten sind, um kurzfristige Schutzmaßnahmen zu ermöglichen. Die bisherigen Forschungsergebnisse zur Vorhersagbarkeit von Brüchen sind widersprüchlich.

big data und Methoden zum Analysieren dieser großen Datenmengen in den vergangenen Jahren haben neue Möglichkeiten zum Studium der Vorhersagbarkeit von Brüchen eröffnet. Die Menge an verfügbaren Daten für Erdbebenforschung wächst exponentiell und hat den Tera- bis Petabyte-Bereich erreicht. Während viele klassische Methoden, basierend auf manuellen Datenauswertungen, hier ihre Grenzen erreichen, ermöglichen diese Datenmengen den Einsatz hochparametrischer Modelle und datengetriebener Analysen. Eine Art dieser Modelle sind Methoden des maschinellen Lernens, insbesonderes des *deep learning*. Gestützt durch methodische Durchbrüche hat *deep learning* in einer Vielzahl von Anwendungsfeldern große Bedeutung gefunden. Auch in Seismologie erzielen *deep learning* Ansätze deutliche Verbesserungen gegenüber klassischen Methoden. Allerdings sind viele Möglichkeiten der Anwendung von *deep learning* in Seismologie noch unerforscht.

Diese Doktorarbeit befasst sich mit der Entwicklung von Methoden des maschinellen Lernens zur Untersuchung der Vorhersagbarkeit von Brüchen und der Frühwarnung vor Erdbeben. Wir untersuchen zuerst die Kalibrierung einer hochpräzisen Magnitudenskala in einem post hoc Scenario. Hierfür entwickeln wir einen hybriden Ansatz, basierend auf mathematischer Optimierung und maschinellem Lernen. Nachfolgend befassen wir uns mit Echtzeitanalyse von Erdbeben mittels *deep learning*. Wir präsentieren TEAM, eine Methode zur Frühwarnung. TEAM schätzt direkt aus den seismischen Wellenformen die zu erwartende Stärke von Bodenbewegungen. TEAM ermöglicht längere Warnzeiten als traditionelle Ansätze bei einem besseren Verhältniss von korrekten, falschen und verpassten Warnungen. Auf TEAM aufbauend entwickeln wir TEAM-LM zur Echtzeitschätzung von Lokation und Magnitude eines Erdbebens. TEAM-LM verbessert die Magnituden- und Lokationsschätzungen im Vergleich zu klassichen Modellen und vorangegangenen *deep learning* Ansätzen. Anhand von TEAM-LM analysieren wir die Stärken und Schwächen von *deep learning* Modellen zur Erdbebenanalyse. Im Gegensatz zur ausgezeichneten Durschnittsqualität zeigt das Modell systematische Fehler für Beispiele mit unzureichenden Trainingsdaten. Wir diskutieren und evaluieren mögliche Lösungsstrategien für dieses Problem.

Im letzten Schritt untersuchen wir die Vorhersagbarkeit von Brüchen mittels TEAM-LM anhand eines Datensatzes von teleseismischen P-Wellen-Ankünften. Dieser Analyse stellen wir eine Untersuchung von Quellfunktionen großer Erdbeben gegenüber. Unsere Untersuchung zeigt, dass die Brüche großer Beben erst vorhersagbar sind, nachdem die Hälfte des Bebens vergangen ist. Selbst dann können weitere Subbrüche nicht vorhergesagt werden. Nichtsdestotrotz zeigen die hier entwickelten Methoden, dass *deep learning* die Echtzeitanalyse von Erdbeben wesentlich verbessert. Wir hoffen, dass diese Ergebnisse Frühwarnsysteme verbessern werden und helfen, Schäden durch Erdbeben zu reduzieren.

Acknowledgements

As much as a PhD thesis is the work of one person, it is only possible through the support of several people, who I owe my gratitude. First, I want to thank my supervisors Frederik Tilmann and Ulf Leser for their advice and mentorship throughout the PhD. I also want to thank Dino Bindi and Fabrice Cotton for providing another perspective on my work and on my personal development. Being positioned between two institutes, I enjoyed insightful discussions and social interactions with colleagues from both, the Seismology section at the GFZ Potsdam and the WBI group at Humboldt-Universität zu Berlin. For this, I want to thank all of my colleagues there.

While not key part of this thesis, several side projects helped to shape my views and thereby this thesis. I learned much from the collaboration with Jack Woollam, Andreas Rietbrock and Dietrich Lange within the REPORT-DL project. I'd like to thank the two students whose theses I supervised, Thomas Bornstein and Viola Hauffe, for the inspiration given and for their questions that forced me to reevaluate some of my beliefs on machine learning in seismology. The work with Tabea Rettelbach, Niklas Kitzmann and William Foster in palaeontology showed me the connections between different fields of geoscience, which I am grateful for, even though still slightly disappointed about the lack of dinosaurs in the project. Furthermore, I want to thank Leon Weber for including me in countless projects, enabling me to keep in touch with "pure" machine learning research. In particular, including me on the HunFlair project together with Mario Sänger was very instructive, and I drew lots of inspiration for SeisBench from this project.

This PhD was funded by the Helmholtz Einstein International Berlin Research School in Data Science (HEIBRiDS). I would like to thank HEIBRiDS for this funding, but I also want to thank the HEIBRiDS team and the fellow HEIBRiDS students for their commitment, providing me with both motivation and potential for personal development. Along the same lines, I want to thank everyone involved in Geo.X and the Geo.X Young Academy for all that I have learned through the community.

Just as integral as the support within the project itself, was the support outside work. For this, I want to thank my parents and siblings. Furthermore, I want to thank my friends who have supported me throughout this thesis, in particular Fabian, Birte, Chantal, Julian, Johanna, علاء, Alex, Alice, and Jonas. Lastly, I'd like to thank my Ultimate Frisbee team *Parkscheibe* for giving me an offset to my work and at the same time other interesting challenges.

Contents

1 Introduction

Every day, countless earthquakes occur throughout many regions worldwide, caused by ruptures of seismic faults [Stein and Wysession, 2003, Shearer, 2009]. While most of these earthquakes cannot be felt by humans, but are only recorded with sensitive instruments, some of them cause noticeable ground shaking. Every year, several earthquakes cause significant damage, and every decade some earthquakes cause widespread devastation and loss of human life. Some recent examples are the 2015 Gorkha earthquake (Nepal), the 2010 Haiti earthquake, and the 2004 Sumatra-Andaman earthquake. This renders earthquakes among the most destructive natural hazards.

Given the threat posed by earthquakes, it is essential to build a deep understanding of these events. Observations of earthquakes have been documented for several millennia [Marcellinus, around 390]. Within the last century, seismic instruments have been deployed worldwide, allowing for quantitative recordings of ground shaking [Richter, 1935]. This has led to a very good understanding of the propagation of the seismic waves emitted by earthquakes. However, many questions about the earthquakes themselves remain open. For example, while it is possible to assess the likelihood of earthquake occurrence in a certain region in a time frame of years to tens of years, specific earthquakes can not be predicted, i.e., it is not possible to pinpoint the time, location and size of a future earthquake [Jordan et al., 2011].

Given the impossibility of earthquake prediction, a common strategy for reducing the impact of large earthquakes is earthquake early warning [Allen and Melgar, 2019]. The goal of early warning is to detect earthquakes as early as possible after their nucleation and to provide warnings to affected locations. As the damaging seismic waves usually require seconds to tens of seconds to travel from the earthquake source to vulnerable targets, this time can be used to take preparatory action [Allen and Melgar, 2019]. For successful early warning, it is essential to correctly assess the impact of an earthquake early on, usually by determining its size in terms of its magnitude. Large earthquakes have rupture durations of seconds to tens of seconds, the same or sometimes even longer than the travel time of the seismic waves from the source to the target. To assess the maximum potential warning times achievable with early warning, it is, therefore, necessary to understand how well the size of an earthquake can be constrained while its rupture is still ongoing. This question is known as rupture predictability [Allen and Melgar, 2019]. One one hand, the size might already be defined at the event onset. On the other hand, the rupture might be driven by a stochastic process and therefore the final size might be unclear until the event arrests. Which of these scenarios occurs has a major impact on the warning times and thereby defines fundamental limitations on the effectiveness of early warning systems.

Traditionally, rupture predictability is studied through model- or hypothesis-driven research. In a model-driven approach, a physical or empirical model for the rupture process, in particular, for its initiation, is proposed. Based on the model, certain observables can be predicted. By measuring the agreement or disagreement between predicted and observed data, the model's validity can be assessed. In hypothesis-driven research, a theory is postulated, for example, a connection between a certain measurable parameter X and the size of an earthquake. This connection can, again, be verified or falsified using statistical inference over the observational data. In recent years, a novel approach has gained popularity: data-driven research. Using data mining techniques, patterns are extracted from large collections of data. These patterns can then be studied to gain insights into the underlying question, in this case, rupture predictability. This transition towards data-driven methods has been described for solid Earth geoscience [Bergen et al., 2019],

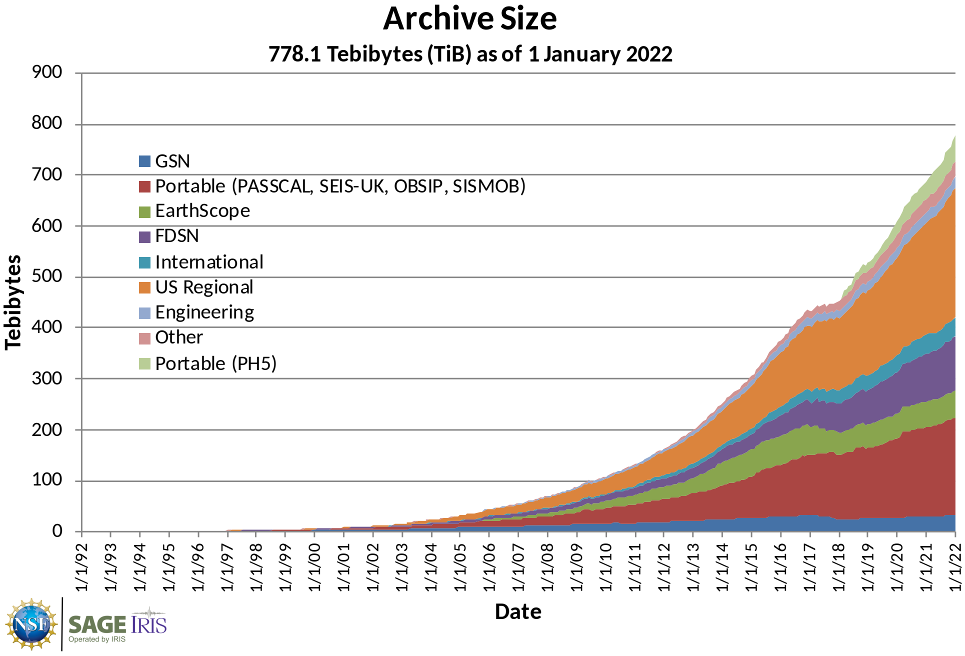


Figure 1.1: Size of the IRIS (Incorporated Research Institutions for Seismology) archive for seismic waveforms over time. Figure adapted from IRIS, originally available at `https://ds.iris.edu/files/stats/data/archive/Archive_Growth.jpg`, last accessed 16th February 2022.

but also across other disciplines [e.g. Carrol and Goodstein, 2009, Shih and Chai, 2016]. In contrast to hypothesis- or model-driven research, data-driven research can model by far more complex relationships, even though often at the drawback of lower interpretability of the findings.

The advent of data-driven research can be attributed to three drivers: the availability of data, the development of novel methods, and the availability of computational capacities. These factors have led to breakthroughs throughout scientific disciplines [Krizhevsky et al., 2012, Stokes et al., 2020, Jumper et al., 2021]. While compute capabilities and methods are mostly independent of the field of application, data availability is an aspect specific to a field. Seismology is a data-rich field. Given the large number of both permanently and temporarily deployed seismic stations, the amount of seismic data in archives has grown exponentially over recent decades (Figure 1.1). In addition, not only is the amount of continuous waveform data large but records have also been associated with tens of millions of earthquakes, even though mostly with small events.

Most computational methods underlying the major data-driven breakthroughs in recent years used the principles of neural networks and deep learning [LeCun et al., 2015]. Neural networks, developed 80 years ago, were designed to mimic the structure of the human brain [McCulloch and Pitts, 1943]. Like all machine learning models, neural networks are trained by fitting them to example data. Through several methodological advances, it recently has become possible to build and train very large neural networks, a discipline now called deep learning [LeCun et al., 2015]. Deep learning models have been proven to be particularly effective when applied to high dimensional data, as is the case for most seismological observables. In addition, deep learning methods benefit strongly from being trained on very large collections of examples. As outlined in the previous paragraph, such data is available in seismology. These factors make data-driven approaches, in particular

Table 1.1: Overview of the main characteristics for the three methods introduced in this thesis. The methods are a method for magnitude scale calibration, the transformer earthquake alerting model (TEAM), and a TEAM adaptation for location and magnitude estimation (TEAM-LM).

	Calibration	TEAM	TEAM-LM
Time scale	post hoc	real-time	real-time
Input	event catalog, waveforms	waveforms	waveforms
Features	hand designed	automatic (CNN)	automatic (CNN)
Modelling	hybrid (physics motivated corrections, gradient boosting)	deep learning	deep learning
Output	magnitude	ground shaking (probabilistic)	magnitude, location (probabilistic)
Reference	Chapter 3 [Münchmeyer et al., 2020]	Chapter 4 [Münchmeyer et al., 2021b]	Chapter 5 [Münchmeyer et al., 2021a]

deep learning, a prime candidate for studying rupture predictability. This might enable us to identify complex indicators of rupture predictability that could not be derived with a model- or hypothesis-driven approach.

1.1 Goals and contributions

In this thesis, we[1] study earthquake rupture predictability through real-time assessment of the earthquake source parameters. To this end, we use two observations about the relation between practical methods and physical limitations of real-time earthquake assessment. First, any practical method is a lower bound on the physical limitations, i.e., building a method that achieves a certain precision and timeliness implies the physical feasibility of the same. Second, by analysing the limitations of high-quality methods and comparing them to physics-based models, we can infer hypotheses on rupture predictability and gather evidence for potential physical limitations. Until recently, existing real-time methods were insufficient to conduct this type of study, lacking either precision or timeliness. In this thesis, we show how the application of deep learning, combined with the exponential growth of seismic data, allows building accurate real-time methods for the study of rupture predictability. As this thesis combines machine learning method development with an underlying geophysical question, it takes an interdisciplinary standpoint, positioned between computer science and seismology.

In conjunction with the central goal of this thesis, we pursue two further, related aims. First, the limitations of earthquake assessment with machine learning in a post hoc scenario remain yet unexplored. Exploring these limitations contributes towards

[1] Throughout this thesis, we use the 1^{st} person *plural*, highlighting that the results were obtained in a collaboration of me and all coauthors of the underlying publications presented. The specific contributions of each author to the publications are described in Chapter 1.3. Where appropriate, further details on the contributions are given in the footnotes.

the main goal, as it provides high-quality reference data and a reference frame for the performance of real-time methods. Post hoc earthquake assessment is therefore part of this thesis. Second, real-time assessment methods, besides their relevance for studying rupture predictability, are essential for earthquake early warning systems. Therefore, this thesis also studies the application of the developed methods to early warning.

The scientific contributions of this thesis can be split into three categories: method contributions, theoretical contributions, and seismological insights. On the methods side, this thesis introduces three new ideas:

- a method for the post hoc calibration of high confidence magnitude scales
- the transformer earthquake alerting model (TEAM), a deep learning based earthquake early warning method using real-time waveforms
- TEAM for location and magnitude estimation (TEAM-LM), a deep learning model for real-time earthquake source parameter estimation

In addition, we introduce an adaptation of TEAM-LM to teleseismic waveforms. Each of these methods is accompanied by an extensive study of its performance and properties. An overview of the main characteristics of the different methods is provided in Table 1.1.

On the theoretical side, we introduce a general, yet comprehensive, formulation of rupture predictability in terms of stochastic processes and conditional distributions. Using this model we argue that rupture predictability is an inherently probabilistic question, and identify shortcomings of the deterministic view on rupture predictability. We show how the conditional probabilities in our formulation can be estimated from data using variational inference. This enables probabilistic analyses of rupture predictability, which we conduct for two sets of observables.

On the seismological side, the first contribution are high confidence magnitude values for the Northern Chile earthquake catalog by Sippl et al. [2018]. The second, and main, contribution is the study of earthquake rupture predictability based on moment rate functions and teleseismic P arrival waveforms. We show that no signs of early rupture predictability exists in these observables, not even in a probabilistic sense.

1.2 Outline

This remaining part of this thesis is composed of the scientific background, four chapters presenting the main contributions, and a conclusion with an outlook on future research questions. The main text is supplemented by an appendix, providing additional figures and tables, as well as technical method details. The appendix structure mirrors the structure of the four main chapters.

Chapter 2 provides the scientific background to the thesis. To account for the interdisciplinary nature of this thesis, the background is constituted of both a part on earthquakes and seismic waves and a part on machine learning and deep learning. The chapter closes with an overview of applications and developments of machine learning and deep learning in seismology. Throughout the chapter, we aim to provide an intuition of each concept introduced that will aid the understanding of the later main chapters. Therefore, we derive most concepts from the underlying mathematical principles, rather than providing a hands-on guide.

Figure 1.2 provides an overview of the four main chapters in the context of an earthquake. The earthquakes emits seismic waves. These can be recorded as seismic waveforms, or more generally speaking observables O_t until time t. These waveforms are the main

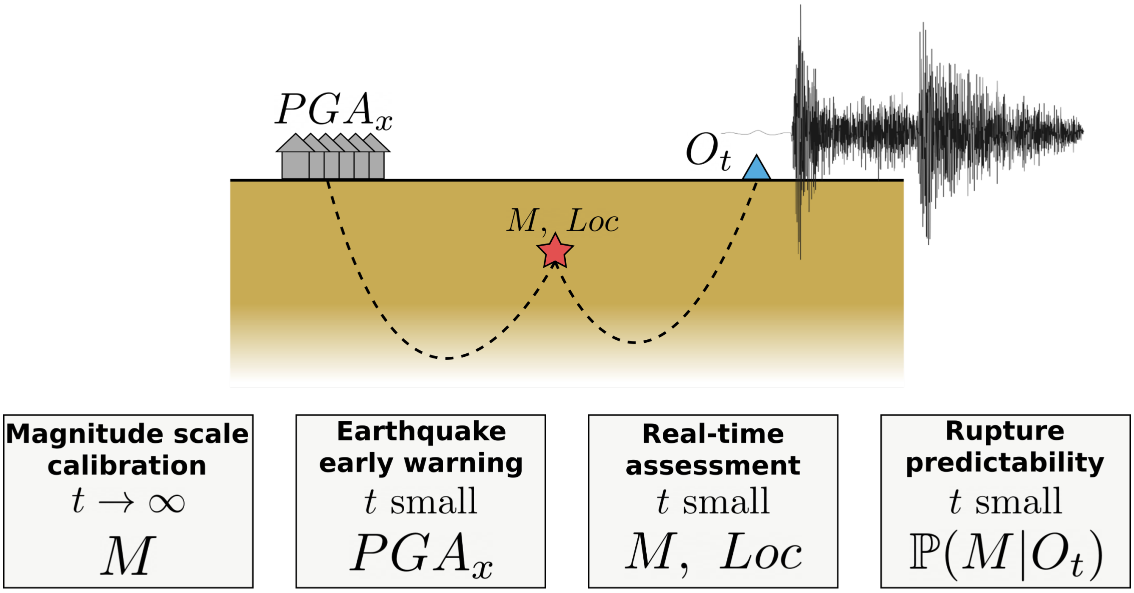


Figure 1.2: Overview of the four main chapters of this thesis in the context of an earthquake. The red star visualises an earthquake, characterised by its magnitude M and its (hypocentral) location Loc. The dashed lines show travel paths of seismics waves. To the right, they are recorded at a seismometer (blue triangle) as a seismogramm, the observables O_t. To the left, they hit a city, causing shaking characterised by the peak ground acceleration PGA_x. The four bottom panels show Chapters 3 to 6 (left to right). Each panel provides an abbreviated chapter title, the time scale, and the modelled property of the event.

type of input data we study throughout this thesis. Chapter 3 discusses how to infer the earthquake source parameters, in particular the magnitude M, from these observables in a post hoc scenario. To this end, we present a novel method for the calibration of a high confidence magnitude scale and its application to Northern Chile. Our method is a hybrid approach consisting of two parts: physics-based attenuation correction terms, derived using mathematical optimisation, and a combination of waveform features using gradient boosted trees. We study the properties and characteristics of the method, as well as the obtained magnitude scales and attenuation functions. Lastly, we apply the method to augment the highly complete earthquake catalog for Northern Chile from Sippl et al. [2018] with high confidence magnitude values.

Seismic waves can cause damaging shaking at targets x, often characterised through ground motion parameters, e.g., the peak ground acceleration PGA_x. To reduce the resulting damage, earthquake early warning methods can be used. Therefore, in chapter 4 we present the transformer earthquake alerting model (TEAM), an earthquake early warning method based on deep learning. TEAM is a hybrid method between source estimation based and propagation based earthquake early warning methods. For evaluation, we use two datasets from highly seismically active regions, Italy and Japan. We put special emphasis on the performance for large events and show how transfer learning techniques can considerably improve performance for these cases.

Chapter 5 introduces an adaptation of the TEAM method called TEAM-LM. TEAM-LM is a method with close architectural similarity to TEAM, but for the estimation of earthquake source parameters, namely magnitude and location, instead of ground motion. This way, TEAM-LM combines the methodology and time scale of Chapter 4 with the target of Chapter 3, magnitude estimation. In addition to the datasets from Japan and Italy used for the evaluation of TEAM in Chapter 4, we also evaluate TEAM-LM on the dataset from the Northern Chile with the high confidence magnitude values from Chapter 3. Due to the large number of events and the low uncertainties in both location and magnitude estimates, this catalog serves as a gold standard for several analyses. We compare TEAM-LM both to classical baselines and deep learning baselines. Furthermore, we study the impact of training schemes, such as transfer learning or multitask learning, on TEAM-LM performance. As a key issue, we identify significant performance degradation in low data scenarios, manifesting, for example, in a systematic underestimation of large magnitude events. We show that this effect overshadows potential signs of limited rupture predictability. We conclude that the currently available methods and datasets are insufficient to draw conclusions on rupture predictability.

Chapter 6 presents a principled analysis of rupture predictability for large earthquakes, a geophysical question inherent to real-time magnitude assessment. We first introduce a probabilistic formulation of rupture predictability, posing the question in terms of conditional distributions. We show how to estimate these distributions from data by using neural networks and variational inference. To address the issue of insufficient data that we identified in Chapter 5, we introduce two new observables: moment rate functions and teleseismic waveforms. Due to their global scope, both provide more examples of large events than the datasets in Chapter 5, mitigating the data sparsity issue. Furthermore, moment rate functions incorporate physics-derived information, making the magnitude estimation task easier. For both observables, we do not find any indication of early magnitude predictability. Magnitudes can only be predicted after the peak of the moment rate function, usually around half of the rupture duration. Even then, it is impossible to foresee future asperities. This hints at a universal initiation behaviour of earthquakes independent of their size.

Chapter 7 summarises the findings from Chapters 3 to 6 and discusses future research direction. We evaluate the potentials and limitations of the presented approaches, both from a technical standpoint and concerning the potential underlying physical mechanism.

1.3 Own prior publications

Some parts of this thesis are based on work that has been published in peer-reviewed publications. The magnitude calibration method and its evaluation presented in Chapter 3 have been published as:

> J. Münchmeyer, D. Bindi, C. Sippl, U. Leser, and F. Tilmann. Low uncertainty multifeature magnitude estimation with 3-D corrections and boosting tree regression: Application to North Chile. *Geophysical Journal International*, 220(1):142–159, Jan. 2020. ISSN 0956-540X. doi: 10.1093/gji/ggz416

The TEAM method and its evaluation presented in Chapter 4 have been published as:

> J. Münchmeyer, D. Bindi, U. Leser, and F. Tilmann. The transformer earthquake alerting model: A new versatile approach to earthquake early warning. *Geophysical Journal International*, 225(1):646–656, 2021b. ISSN 0956-540X. doi: 10.1093/gji/ggaa609

The TEAM-LM method and its evaluation presented in Chapter 5 have been published as:

> J. Münchmeyer, D. Bindi, U. Leser, and F. Tilmann. Earthquake magnitude and location estimation from real time seismic waveforms with a transformer network. *Geophysical Journal International*, 226(2):1086–1104, 2021a. ISSN 0956-540X. doi: 10.1093/gji/ggab139

The study on rupture predictability in Chapter 6 is currently under review.

For all these publications, Jannes Münchmeyer designed, conducted and evaluated the experiments. Jannes Münchmeyer wrote the manuscripts for all publications. Frederik Tilmann, Ulf Leser and Dino Bindi contributed to the study design and the manuscript preparation for Münchmeyer et al. [2020, 2021a,b]. Frederik Tilmann and Ulf Leser contributed to the study design and the manuscript preparation for the manuscript forming the basis of Chapter 6. For Münchmeyer et al. [2020], Christian Sippl provided phase picks and moment magnitude values, along with a supplementary text describing the determination procedure for the moment magnitude values.

The SeisBench framework and the pick benchmark summarised in Chapter 7.1 have been published in Woollam et al. [2022] and Münchmeyer et al. [2022]. SeisBench was developed by Jannes Münchmeyer and Jack Woollam. The benchmark code and evaluation were implemented by Jannes Münchmeyer. Jack Woollam wrote the SeisBench manuscript in collaboration with all coauthors. Jannes Münchmeyer wrote the benchmark manuscript in collaboration with all coauthors.

In a few places, this thesis refers to the master's thesis from Hauffe [2021]. Jannes Münchmeyer suggested the study design and supervised the thesis. Viola Hauffe developed the study design further, conducted and evaluated the experiments, and documented the results.

2 Background

2.1 Seismic wave propagation

The central objects of study in this thesis are *earthquakes*. More specifically, this thesis will analyse earthquakes using seismic observation, i.e., observations of ground motion. Therefore, this section will present the fundamentals of seismic wave propagation, with the following sections discussing earthquakes, their observation and characterisation. While these concepts will be familiar to seismologists, these sections account for the interdisciplinary nature of the thesis. Most of the material presented is based on the textbooks by Shearer [2009] and Stein and Wysession [2003], as well as the New Manual of Seismological Observatory Practice [Bormann, 2012].

The theory of seismic wave propagation can in large parts be derived from Newton's second law $F = ma$, stating that the force F on a body equals the product of its mass m and the acceleration a acting on it. This law gives rise to the wave equation, a partial differential equation (PDE), which for a homogeneous medium in a single dimension is given as

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2} \tag{2.1}$$

with the displacement field $u(t, x) : \mathbb{R}^2 \to \mathbb{R}$, depending on time t and position x. In the one dimensional case, there exists one fundamental solution to this PDE, a wave propagating with a velocity of c.[2] Note that for now, we only look at the homogeneous case, i.e., without any outside forces acting on the system.

For the 3D case, a similar wave equation can be derived. The full formula and a formal derivation is given, for example, in [Shearer, 2009, Chapters 2, 3] or [Stein and Wysession, 2003, Chapter 2]. For now, we assume a homogeneous, isotropic medium, i.e., a translation and rotation invariant medium, without any outside forces. The 3D seismic wave equation has two fundamental solutions: *P* and *S waves* (Figure 2.1). P waves, primary or also pressure waves, are longitudinal waves, i.e., they oscillate in the direction of travel. P waves can travel in both solid and liquid media.[3] S waves, secondary or also shear waves, are transversal waves, i.e., they oscillate in a direction orthogonal to the direction of travel. S waves can travel only in solid media, but not in liquids. The propagation velocities of P (v_p) and S (v_s) waves are properties of the medium. P waves travel faster than S waves, with typical velocity ratios v_p/v_s around 1.7, even though variations between different media are considerable.

The Earth is not homogeneous, with seismic velocity varying laterally and, to an even larger extent, with depth, impacting seismic wave propagation. To discuss wave propagation in inhomogeneous media, it is useful to model seismic waves as rays pointing in the direction of travel of a plane wave. When a seismic wave hits an interface between two media, it can be refracted or reflected, depending on the velocities in the two media. For a horizontal interface, a wave passing from a medium with higher velocity into a medium with lower velocity is refracted towards a steeper incidence angle. Furthermore, interfaces can introduce conversions between different phases of seismic waves: P waves can be converted to S waves and vice versa.

[2] Technically, there is a second solution, the wave with a velocity of $-c$. This solution only differs by the direction of propagation. As we will disregard the direction of propagation when analysing the solutions to the 3D wave equation in the following, we also regard the c and $-c$ solutions as identical here.

[3] P waves can also travel in gases as acoustic waves, however, this is rarely relevant in traditional seismology. Nonetheless, P waves in the air caused by earthquakes can under certain circumstances be recorded on infrasound sensors.

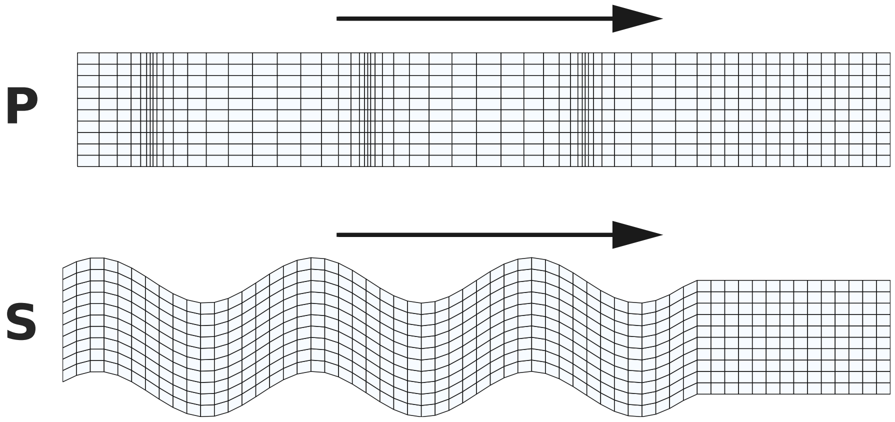


Figure 2.1: P and S waves in a homogeneous medium travelling from left to right. The P waves (top) travel through a sequence of compressions and dilations. The S waves travel through shear displacement. In the visualisation, the polarisation of the S waves is vertical. For both wave types, a segment of yet undisturbed medium is shown at the right end. Note that the visualisation omits the third dimension, pointing orthogonal to the page.

Average seismic velocities in the Earth's interior can be described by a collection of vertically stacked, homogeneous layers [Dziewonski and Anderson, 1981]. Within the crust and mantle, velocities increase with depth. At the core-mantle boundary (≈ 2900 km depth) seismic velocities drop. As the outer core is liquid, only P waves, but no S waves can propagate through it. Given the velocity model, we can describe the ray travel paths of seismic waves inside the Earth. Figure 2.2 shows P wave travel paths in a regional scenario, assuming a laterally uniform velocity model. The figure shows a source at 300 km depth and three recording stations at 500 km, 1750 km and 3000 km horizontal distance. For the closest stations, there is only a single travel path, going directly upwards. For both of the stations further away, there are multiple travel paths. Following Fermat's principle of extremal travel times, all rays depart downwards, as velocities in the lower layers are higher. The travel paths consist of curved segments within layers with a velocity gradient, refractions at interfaces, and in some cases reflections at interfaces. The waves from these different travel paths can be identified as separate phase arrivals in seismic recordings [Storchak et al., 2003].

A special case of interactions with interfaces are interactions of seismic waves with free surfaces, as these produce *surface waves*. The amplitude of these waves decreases with the distance to the free surface, giving rise to the name surface wave. This stands in contrast to P and S waves that do not require a free surface and have constant amplitude along a wavefront. P and S waves are therefore also called body waves. The wavefront of a body wave from a point source in a homogeneous, isotropic medium is a sphere at any time. Body wave energy density, therefore, decays with a factor r^{-2} of the distance r from the source, proportional to the surface of this sphere. In contrast, as surface wavefronts only occur along a circle on the surface, their energy density only decays with a factor of r, proportional to the circumference of this circle. Therefore, in far-field observations of a seismic event (above several thousand kilometres), surface waves are usually the waves with the highest amplitudes.

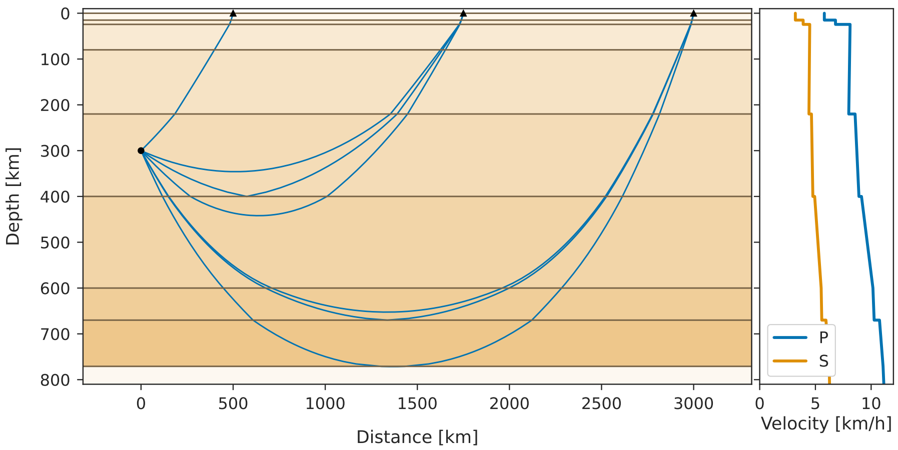


Figure 2.2: Preliminary Reference Earth Model [PREM, Dziewonski and Anderson, 1981] for P and S wave velocities (right) and ray paths for P waves at regional distances (left). The source at 300 km depth is shown by a black dot, the stations at the surface by black triangles. The different rays show different possible travel paths, including paths with interface reflections.

2.2 Earthquakes and their observation

An *earthquake* is the sudden release of seismic energy caused by two blocks of the Earth suddenly slipping past each other. The interface between these blocks is called the *fault* or *fault plane* (Figure 2.3). Earthquakes emit seismic waves that can be recorded and, for sufficiently large events, also felt. Earthquakes can be caused by tectonic loading, i.e., the long-term movement of tectonic plates, but also by other factors such as volcanism or hydraulic stimulation. The largest earthquakes occur along plate boundaries, as visible in the map of global seismicity in Figure 2.4. In this section, we describe fundamental properties of earthquakes that we will refer to throughout this thesis.

2.2.1 Types of faulting and focal mechanisms

Earthquakes exhibit different types of *faulting*, i.e., the type of motion of the sliding blocks relative to each other [Shearer, 2009, Chapter 9], which can be represented as focal mechanisms. The three main types of faulting, strike-slip, normal and reverse/thrust, are depicted in Figure 2.3. In a strike-slip event, blocks move horizontally along each other with a vertical fault and no vertical displacement. Normal and reverse faulting occurs on sloped interfaces. An event is called normal if the upper block moves downwards along the slope, and reverse if it moves upwards. Reverse events with a shallow slope are called thrust faulting. In practice, events sometimes exhibit a mixture of faulting types, for example slipping both along the fault in horizontal direction (strike-slip) and along the slope of the fault (normal/reverse).

Faulting types can be visualised through their *focal mechanisms* using so-called beach balls (bottom row of Figure 2.3). Beach balls show the radiation pattern of the P wavefield: in the black areas the waves are extensional, the first motion of the P wave is outward from the source, in the white areas the waves are compressional, the first P motion is towards the source. Along the planes connecting black and white areas, the so-called nodal planes, no P waves are emitted. Note that in practice, in particular at higher frequencies, shaking from P waves can be observed in the direction of the nodal planes

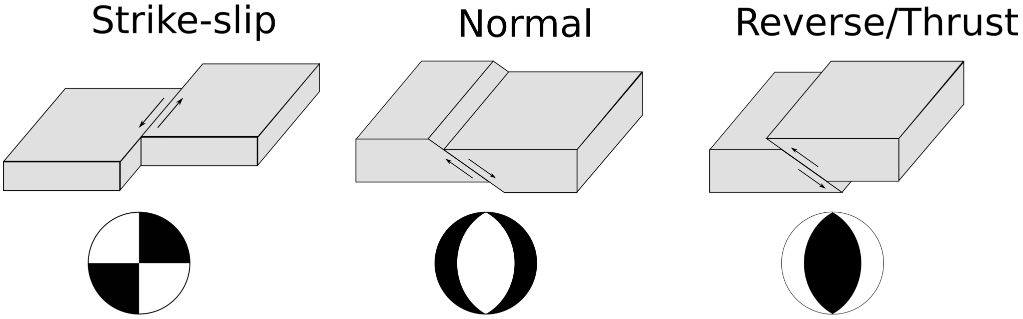


Figure 2.3: Faulting types and focal mechanisms. The left column shows left-lateral strike-slip faulting, the middle column normal faulting, and the right column a reverse faulting. Reverse faults are called thrust faults if their dip is sufficiently small (roughly $< 45°$). Arrows indicate the directions of motion. The bottom row shows beach balls, visualisations of the faulting mechanisms, assuming a North-South orientation of the depicted faults.

as well, for example, due to scattering. Each focal mechanism has two nodal planes, the actual faulting plane and a so-called auxiliary plane with orthogonal orientation to each other. Slip along the faulting plane produces the same radiation pattern as slip in opposite direction along the auxiliary plane. The radiation pattern, therefore, does not uniquely determine the faulting plane and mechanism. To identify along which plane a rupture occurred, further information is required, such as context regarding the tectonic setting, geodetic measurements, or, for large events, the distribution of aftershocks.

Notably, focal mechanisms describe the earthquake using a *point source*. While this introduces a simplification, this assumption is sufficient for most analysis, in particular for smaller earthquakes or for observations at large distances or low frequencies.[4] Specific effects of this point source assumption will be discussed in Chapters 4 and 6.

2.2.2 Magnitudes and source scaling relations

Earthquakes can be characterised using their *magnitude*. Informally speaking, magnitudes measure the size of earthquakes. However, there are a plethora of magnitude scales, measuring different properties of an earthquake [Bormann et al., 2013b]. An important, early instrumental magnitude scale is the *local magnitude* M_L [Richter, 1935]. Richter [1935] defined it using the peak displacement $A_{\max}$ on a Wood-Anderson seismometer, a specific instrument, at 100 km distance to the earthquake source as

$$M_L = \log_{10} \frac{A_{\max}}{A_0}. \tag{2.2}$$

Here A_0 is a normalising value. By varying A_0 appropriately, accounting for the attenuation with distance, the local magnitude can also be computed from recordings at distances other than 100 km.

While well-established, the local magnitude is not based on a physical model of the earthquake source, but empirically based on observations. As such, it is not per se a measure of a property of the earthquake source, but rather of observable properties. The

[4] Not all effects resulting from the spatial extent of the rupture decay with distance. For example, deviations from the radiation pattern of the focal mechanism due to rupture directivity effects are mostly insensitive to distance. As these aspects are not relevant to this thesis, we refrain from a detailed discussion.

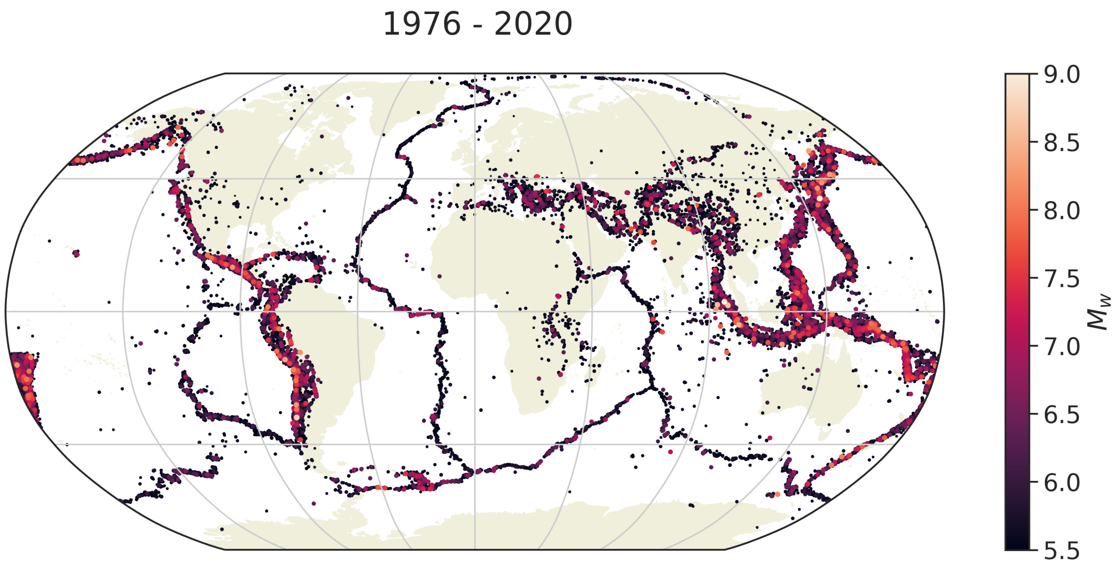


Figure 2.4: Global seismicity, represented through the events in the Global Centroid Moment Tensor (GCMT) catalog from 1976 to 2020 [Ekström et al., 2012]. Each dot represents an event, colours encode the moment magnitude M_w. Large events are plotted on top of smaller ones and with slightly increased size for better visibility.

most common magnitude scale based on source properties is the *moment magnitude* M_w [Hanks and Kanamori, 1979]. The moment magnitude M_w is defined as

$$M_w = \frac{2}{3}(\log_{10} M_0 - 9.1) \tag{2.3}$$

$$M_0 = \mu \bar{D} A \tag{2.4}$$

where M_0 is the seismic moment of the event in Nm. It is derived from the shear modulus μ, a property of the material, the average displacement $\bar{D}$, and the slip area A. In contrast to the definition of M_L, this definition does not immediately imply how to derive M_w from observables. Instead, estimating M_w requires modelling the earthquake source based on observations.

Depending on their magnitude, earthquakes release seismic energy at different frequencies. This frequency dependence needs to be taken into account, for example, when evaluating seismic hazard or analysing seismic waveforms. The distribution of energy release across different frequencies is called the *source spectrum* of an earthquake [Bormann et al., 2013b, Chapter 3.1.2.3]. Figure 2.5 shows prototypical spectra in ground displacement amplitude (left) and ground velocity amplitude (right). The model shows a flat spectrum followed by a f^{-2} decay in displacement amplitude, and consequently an f increase followed by a f^{-1} decrease in velocity amplitude. The frequency with peak amplitude in the (smoothed) velocity spectrum is called the corner frequency f_c. The corner frequency is lower the larger an event is, i.e., large events are depleted in high frequencies. This prototypical model is simplified: while the f^{-2} decay is observed in real earthquakes, decay rates can range from -1 to -3. Furthermore, actual source spectra are considerably less smooth than the presented prototypical ones. Lastly, the spectrum also depends on other parameters, such as the rupture velocity, or the (static) stress drop, the average difference of stress on the fault before and after an earthquake [Shearer, 2009, Chapter 9.5].

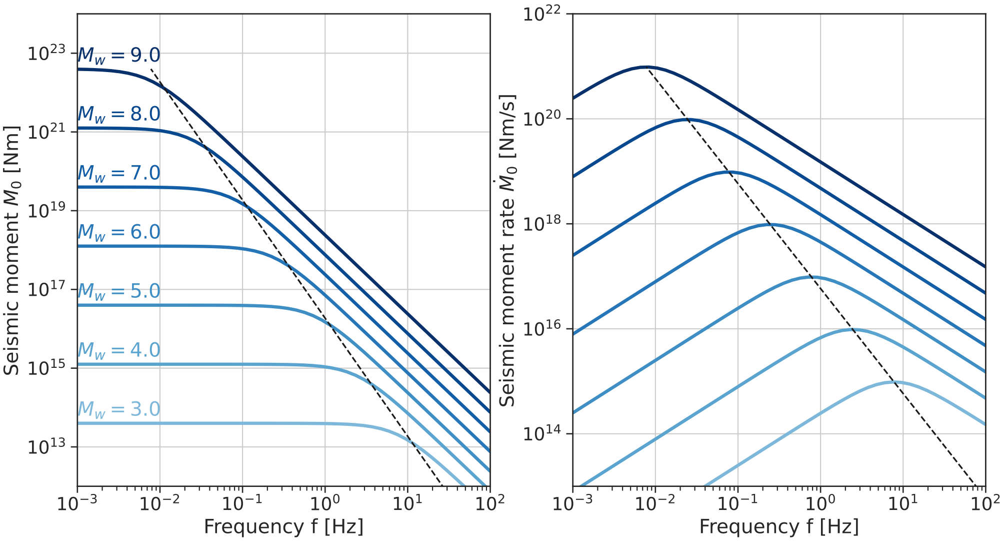


Figure 2.5: Prototypical earthquakes source spectra for different magnitudes M_w in ground displacement amplitude (left) and ground velocity amplitudes (right). Amplitudes have been rescaled to seismic moment and seismic moment rate. The diagonal line indicates the corner frequencies f_c. Source spectra were calculated assuming a f^{-2} decay and a constant stress drop $\Delta\sigma = 3$ MPa. Figure modelled after [Bormann et al., 2009, Fig. 1 and associated text].

Similar to the scaling of the source spectra with magnitude, event duration and rupture extent scale with magnitude [Gomberg et al., 2016]. To discuss these scaling relations we deviate from the point source assumption used above and instead consider a rectangular fault with width W and length L. The seismic moment M_0, as defined in (2.4), can then be modelled as

$$M_0 \sim W^2 L \ . \tag{2.5}$$

For unbounded growth, both the width and length grow with similar rupture velocity v_r, consequently yielding $M_0 \sim L^3$. However, in practice the width of the seismogenic zone is limited, leading to an upper bound on W at the order of tens of kilometres, or for subduction megathrust events sometimes even above 100 km. Once this bound is reached, the scaling changes to $M_0 \sim L$. Assuming a constant rupture velocity $v_r = L/T$, these relations can be used to derive scaling laws for the rupture duration T in the unbounded and bounded case, using the definition of M_w from M_0 (2.3). In the unbounded case, we get $M \sim 2\log_{10} T$, in the bounded case $M \sim 2/3 \log_{10} T$. Typical event durations are around 3 s at $M_w = 6$, 10 s at $M_w = 7$, 30 s at $M_w = 8$, and > 100 s at $M_w = 9$. Typical rupture lengths L range from several kilometres ($M_w = 6$) up to hundreds of kilometres ($M_w = 9$). These duration and length scaling relationships will be essential for the discussion of rupture predictability in Chapters 5 and 6.

2.2.3 Earthquake occurrence patterns

So far, we discussed the sources and characteristics of single earthquakes, but not the occurrence patterns of earthquakes. Extensive knowledge on the distribution of earthquakes has been obtained in the field of statistical seismology [Rhoades et al., 2019], of which we are only going to highlight two aspects: the Gutenberg-Richter distribution of magnitudes and the seismic cycle.

The *Gutenberg-Richter law* describes the distribution of magnitude values [Shearer, 2009, Chapter 9.7.1]. For a magnitude threshold M, the number of events N with at least magnitude M is described by a power law

$$\log_{10} N \approx a - bM, \tag{2.6}$$

where a is the total number of earthquakes and b, called the b-value, describes the relative number of small to large events. The Gutenberg-Richter law holds for a wide range of regions with typical b-values between 0.8 and 1.2. It also holds globally, at least for events with $M_w > 5.5$, with a b-value close to 1. This means, that globally the number of earthquakes observed above a certain magnitude decreases by a factor of 10 with every increase of one magnitude unit. Consequently, observations of very large events are rare. For example, only 5 events with $M_w \geq 9.0$ have been observed worldwide in the era of instrumental seismology (since roughly 80 years). This low sample size poses a difficulty when conducting quantitative research on very large earthquakes. This effect will also appear as training data sparsity throughout the main chapters of this thesis.

The occurrence of very large earthquakes and the seismicity on a fault is assumed to follow a long-term pattern, the so-called *seismic cycle*, based on the elastic rebound theory [Scholz, 2012, Chapter 5]. Within this thesis, we need to take the seismic cycle into account when designing our evaluation procedues, in particular, dataset splits, in Chapters 4 and 5. The seismic cycle consists of four phases: the inter-, pre-, co-, and postseismic phases. During the interseismic phase, a fault is loaded, stress is building up. This usually happens due to tectonic forces and the associated movement of tectonic plates. During the interseismic phases, there is a low level of seismicity. Seismicity increases during the preseismic phase, leading up to a major earthquake. These events are called foreshocks. There are open discussions about whether a preseismic phase needs to occur in each seismic cycle and which characteristics it shows. At some point, a major earthquake happens, known as the mainshock. The short time during the earthquake is called the coseismic period. In the postseismic phase, following the major earthquake, seismicity levels are strongly elevated but decay towards the background rate over weeks to years. These earthquakes are called aftershocks. Once the seismic activity decayed to the background rate, the interseismic phase is reached again and the seismic cycle restarts. Notably, only in hindsight it is possible to identify which event was the mainshock, i.e., it is usually not possible to foresee if a larger event is still imminent. While this model describes the first-order behaviour of most active faults, it leaves several phenomena unaccounted for, such as earthquake swarms, interactions between fault zones, earthquake triggering, or aseismic release of stress. All of these are questions of active research [Ide et al., 2007, Roland and McGuire, 2009, Brodsky and van der Elst, 2014].

2.2.4 The seismic analysis workflow

Within this thesis, we propose novel methods for the analysis of seismic events. As we will contrast our methods to standard approaches in the four main chapters (Chapters 3 to 6), in this section we give an overview of a typical seismic analysis workflow. The workflow shows the analysis steps from raw seismic waveforms to source characterisation (Figure 2.6).

Ground motion is recorded using *seismometers*, which record either displacement, velocity or acceleration [Shearer, 2009, Chapter 11.1]. Modern seismometers typically

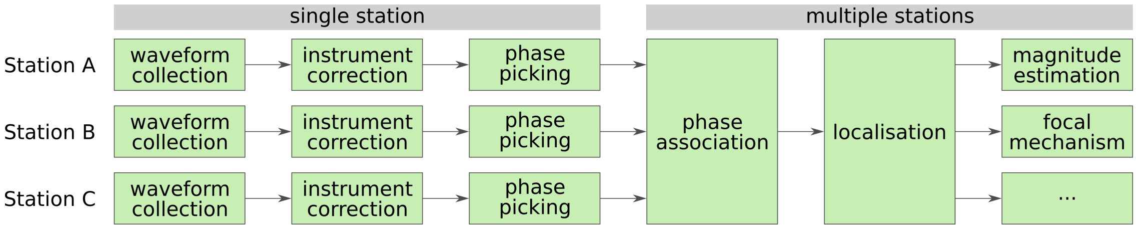


Figure 2.6: Overview of a typical seismic analysis workflow. The first three steps, waveform collection, instrument correct and phase picking are usually conducted separately for each seismic station. The subsequent steps, in this case, phase association, localisation, and source characterisation, require combining observations from multiple seismic stations.

record ground motion along three orthogonal axes, one vertical axis and two horizontal axes. The signal A recorded by a seismometer, called a waveform, is not directly the ground motion U. Instead, the signal is composed of the ground motion and the so-called *instrument response* R. The signal can be written as a function of the frequency ω as

$$A(\omega) = R(\omega)U(\omega) \tag{2.7}$$

where we apply a complex multiplication in the Fourier domain, equivalent to a convolution in the time domain. The instrument response describes, for each frequency the factor between input and output amplitude, called sensitivity, and the phase shift between input and output. To recover the original signal U from A, the instrument response R needs to be restituted, a step known as instrument correction. However, this step is numerically unstable at frequencies with low sensitivity. Figure 2.7 shows the response R of a modern broadband seismometer. Notably, both sensitivity and phase are mostly flat in a frequency range from roughly 0.01 Hz to 10 Hz. Consequently, any analysis only concerned with this frequency range can often be performed without removing the response, but only correcting for the average sensitivity in the flat part. This alleviates the numerical instabilities.

After correcting for the instrument response, the next step in a typical analysis workflow is identifying events within the waveforms. For this, potential candidates of seismic phase arrivals are identified [Bormann et al., 2013a]. Figure 2.8 shows a waveform containing the P and S arrivals from a seismic event at regional distance. To identify phase arrivals a wide range of algorithms has been presented, ranging from simple greedy rules on the signal variance [Trnkoczy, 2009], over sophisticated classical pickers [Baer and Kradolfer, 1987], to deep learning algorithms [Ross et al., 2018a].

Often the waveform from a single seismic station is insufficient to verify whether a pick corresponds to an actual phase arrival or is a false pick on noise. For this reason, picks from multiple seismic stations at different locations are aggregated. As an earthquake should yield picks at multiple stations, it can be verified whether a set of picks can be associated with a consistent origin time and location. This consistent origin time and location identify an earthquake. The process of identifying earthquakes based on phase picks is called *phase association.*

Earthquakes can be located using the travel time differences between phase arrivals at different stations [Shearer, 2009, Chapter 5.7].[5] Taking a set of associated picks, i.e., picks

[5] Technically, earthquake location is also possible using single station recordings. However, given the large uncertainties in these methods and the abundance of seismic data, single station localisation methods

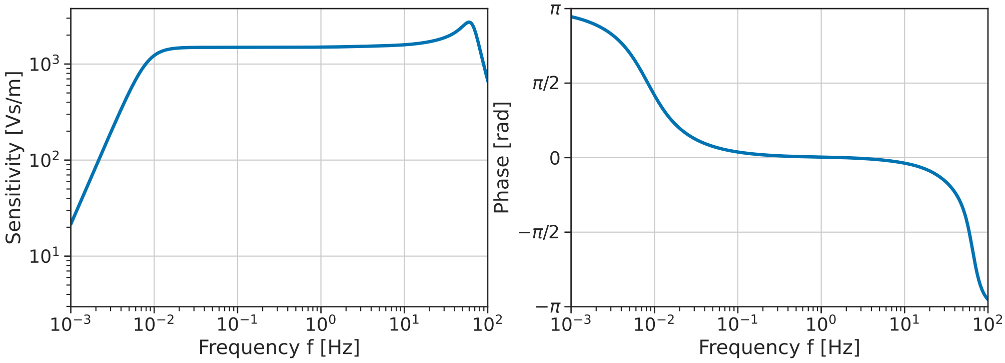


Figure 2.7: Instrument response of an STS-2 (generation 3) broadband seismometer with a period of 120 s and a sensitivity of 1500 Vs/m. The left plot shows the sensitivity/amplitude response, the factor between input and output. The right plot shows the phase response. The amplitude response is flat between frequencies of roughly 0.01 Hz and 10 Hz. Similarly, the phase response is almost flat in a slightly smaller frequency window. Response information was obtained from `https://ds.iris.edu/NRL/sensors/streckeisen/streckeisen_sts2_sensors.htm`, last accessed 1st February 2022.

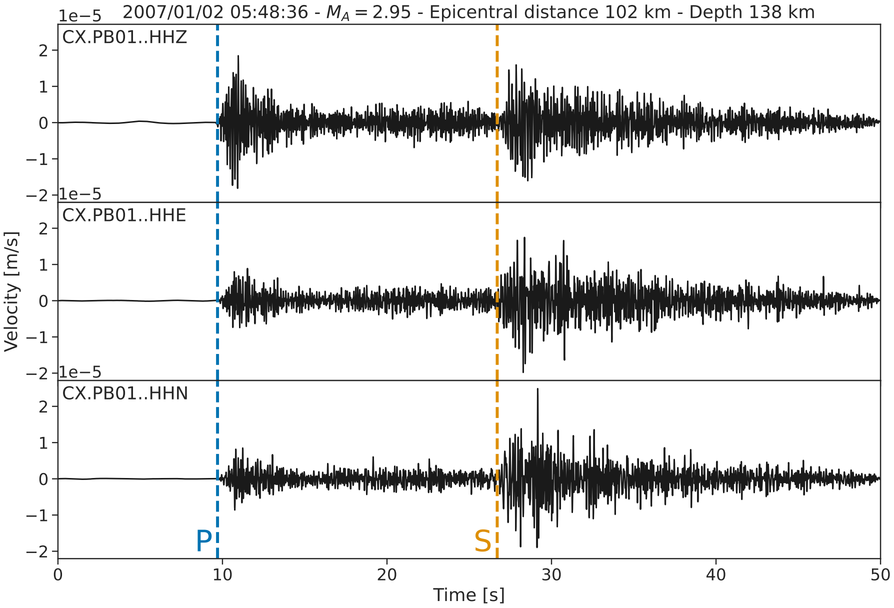


Figure 2.8: Example three-component waveform from an event in Northern Chile at regional distance. The waveform was recorded using a broadband instrument at the seismic station CX.PB01. Event information from Sippl et al. [2018], magnitude information from Münchmeyer et al. [2020]. The instrument response has been restituted and the waveforms were bandpass filtered between 0.2 Hz and 20 Hz. The P and S arrivals are annotated by vertical dashed lines.

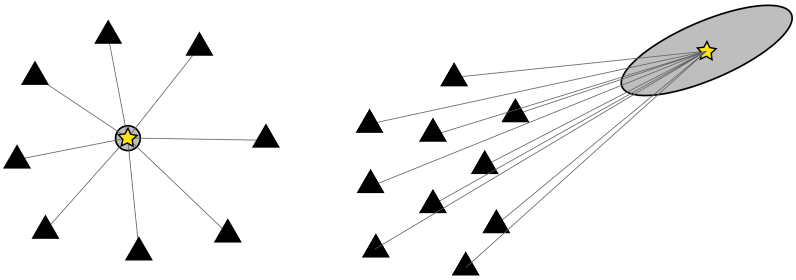


Figure 2.9: Earthquake localisation with good (left) and poor (right) azimuthal coverage in a 2D scenario with homogeneous velocity structure. Black triangles mark the seismic stations, the yellow star the maximum likelihood event location, the grey ellipse the uncertainty, straight lines the travel paths. While the uncertainties are small and the uncertainty ellipsis is rotation symmetric in the case with good azimuthal coverage, the uncertainty ellipse is large and elongated away from the network in the case with poor azimuthal coverage.

belonging to the same event, and a model for the seismic velocities, for every location it can be evaluated whether the observed arrival times match the predicted ones. Once the event location has been determined, subsequent analysis steps, such as the determination of the magnitude or of the focal mechanism, can be performed.

The inferred earthquake origins incur uncertainties from several factors, for example, errors in the pick times or uncertainties in the underlying velocity model. As it will be of interest for the discussion in Chapter 5, here we highlight the strong influence of the *azimuthal coverage*, i.e., whether recordings to each side of the event are available, on the location uncertainties. Figure 2.9 visualises the uncertainties in the case of good (left) and poor (right) azimuthal coverage. Notably, as the origin time of the event is unknown, only the travel times differences between the stations can be used for the localisation. If picks from stations to all sides of the event are available, the location uncertainties are small, as a change in location would increase the travel time at some stations while decreasing it at other stations, leading to a clear change in the differential travel times. In contrast, poor azimuthal coverage leads to an elongated uncertainty ellipse. A change in the location of the event, in particular along the axis pointing towards the network, will affect all the pick times at all stations similarly, i.e., only produce minor variations in the differential travel times.

Two terms are used to define the location of an earthquake: the *hypocenter* and the *epicenter*. The hypocenter is the origin of an earthquake within the Earth. The epicenter is the projection of the hypocenter onto the Earth's surface. Correctly constraining the depth of earthquakes is often challenging, as the station distribution is a special case of bad azimuthal coverage. For most events, the majority of travel paths from the source to the stations depart downward from the source (see Chapter 2.1), often all with similar angles. As a consequence, there is a trade-off between depth and origin time: a slightly earlier/later origin time with a slightly shallower/deeper hypocenter would lead to nearly the same arrivals. This makes constraining depth difficult.

While the examples in Figures 2.8 and 2.9 showed earthquakes at regional distances, the same analysis steps can also be used to analyse earthquakes at so-called *teleseismic* distances, i.e, several thousand kilometers away from the source. This is possible because

are nowadays rarely used. Notable exceptions are studies on extraterrestrial bodies, such as the InSight mission on Mars [Banerdt et al., 2020].

the seismic waves emitted from a large earthquake (roughly $M_w > 6$) can be observed worldwide [von Rebeur-Paschwitz, 1889]. Recordings of these waves are called teleseismic waveforms. Teleseismic waveforms usually have a lower frequency content than regional observations, as high frequencies are attenuated along the travel path. Nonetheless, the analysis workflow of phase picking, phase association, and event localisation can be performed similarly to the regional case. Once the events have been identified, their source characteristics, for example, magnitude and focal mechanism, can be determined from the teleseismic records as well. Teleseismic recordings allow monitoring of large events in remote regions without good instrumentation, such as mid-oceanic ridges. These teleseismic analyses are regularly performed by global monitoring services [e.g. U.S. Geological Survey, 2017, Quinteros et al., 2021].

2.3 Seismic hazard and risk

In seismology, two terms need to be distinguished: *hazard* and *risk* [Wang, 2009]. Hazard describes the probability of an earthquake happening. It can, for example, be quantified by estimating the distribution of events expected within a certain time frame. In contrast, risk describes the probability of harm created by an earthquake. This means risk is the combination of hazard and vulnerability. While seismic hazard can not be avoided, seismic risk can be influenced by reducing the vulnerability, for example, through appropriate construction guidelines for buildings and infrastructure [Hall et al., 1995].

While shaking is the most prominent hazard caused by an earthquake, there are several related hazards. Even though these are not subjects of this thesis, we give a quick overview. Through the shaking, earthquakes can cause soil liquefaction, decreasing the stability of the foundations of buildings and infrastructure. Relatedly, ground shaking can trigger landslides. Building and infrastructure damage can lead to post-earthquake fires, which are particularly damaging in industrial or urban areas [Mousavi et al., 2008]. Certain earthquakes, in particular large, thrust events, can trigger tsunamis, large displacements of seawater that can lead to severe flooding and damage in coastal areas. Large tsunamis, such as the ones caused by the Sumatra-Andamanen earthquake 2004 [Lay et al., 2005] or the Tohoku earthquake 2011 [Mori et al., 2011] can impact large regions, even spanning multiple continents.

In the following sections, we discuss several aspects of assessing and reducing seismic risk: quantifying ground motion (Chapter 2.3.1), the principles of earthquake early warning (Chapter 2.3.2), and the underlying question of rupture predictability (Chapter 2.3.3).

2.3.1 Quantifying ground motion

Quantifying the ground motion at vulnerable targets is fundamental to assess the impact and damage caused by an earthquake [Baker, 2013]. Ground motion can be assessed through measurable quantities, called ground motion parameters, such as *peak ground acceleration* or *velocity*, or the *spectral acceleration*, i.e., the acceleration at a specific frequency. The relation between ground motion parameters and earthquake source parameters is described by *ground motion prediction equations (GMPEs).* Simple GMPEs account for the scaling of ground motion with magnitude and the attenuation with distance through a linear model, while modern, more complex GMPEs incorporate more parameters and nonlinear interactions [Abrahamson et al., 2016]. GMPEs are usually built by defining a functional form and then fitting the free parameters to observed ground motion values.

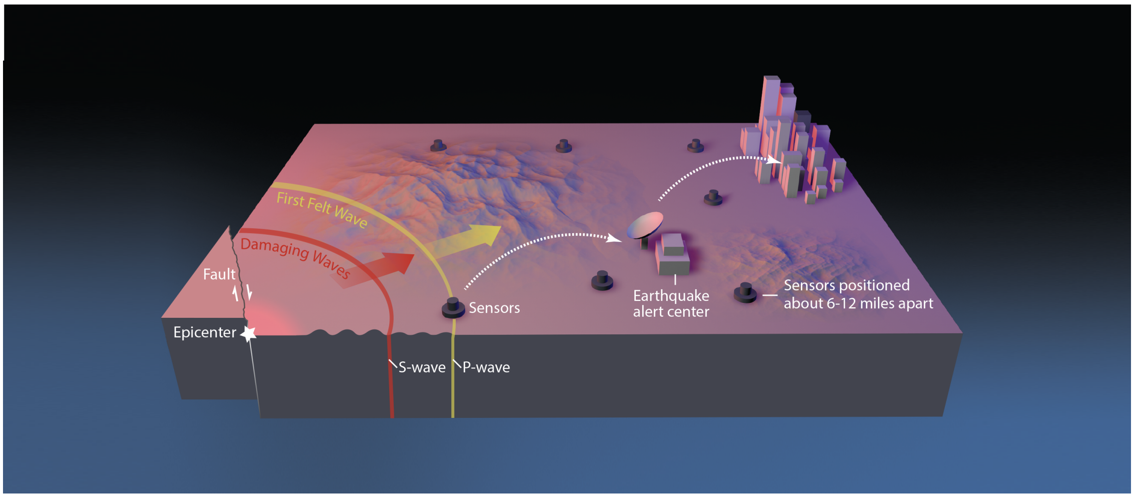


Figure 2.10: Schematic visualisation of the network based early warning system ShakeAlert. Image courtesy of the U.S. Geological Survey, published in Public Domain at `https://www.usgs.gov/media/images/shakealert-earthquake-early-warning-system-us-western-states`, last accessed 13th January 2022.

An alternative for quantifying earthquake ground motion is through its *intensity*, for example using the Modified Mercalli intensity (MMI) scale [Shearer, 2009, Chapter 9.7.2]. In contrast to the directly measurable parameters, the MMI is defined through a list of criteria, such as the level of damage to buildings or the fraction of people noticing the shaking, that classify the shaking into twelve categories. There exist several empirical relationships between the intensity and measurable quantities [e.g., Tselentis and Danciu, 2008]. In combination with GMPEs, this allows estimating the intensity at a given location from the earthquake source parameters.

2.3.2 Earthquake early warning

One option for reducing seismic risk is *earthquake early warning* [Allen and Melgar, 2019]. The idea of early warning is to detect and assess an ongoing earthquake in the time between its onset and the arrival of damaging waves at a target, and to use this lead time to distribute warnings of incoming shaking. These warnings can either initiate automated actions, e.g., slowing down trains, or can provide information for people to seek shelter. The first operational early warning system was installed by the Japanese railway in the 1960s, but only in the last three decades has early warning found more widespread application [Allen et al., 2009]. Crucially, early warning only can provide alerts once an earthquake has started, and not before its initiation. It is therefore not a form of earthquake prediction, which is currently not possible and will likely not become possible anytime in the foreseeable future [Jordan et al., 2011]. In the following, we discuss different early warning approaches. Our discussion focuses on the underlying algorithms for assessing ground motion or earthquake source parameters. We do not discuss implementation aspects, such as telemetry, real-time computation, or alert dissemination. For this, see for example the survey by [Allen et al., 2009] and reference therein.

Among early warning methods, two general concepts can be separated: *on-site warning* and *network based warning* [Allen and Melgar, 2019]. In on-site warning, the location of the measuring instrument is identical to the target. The lead time is achieved by the

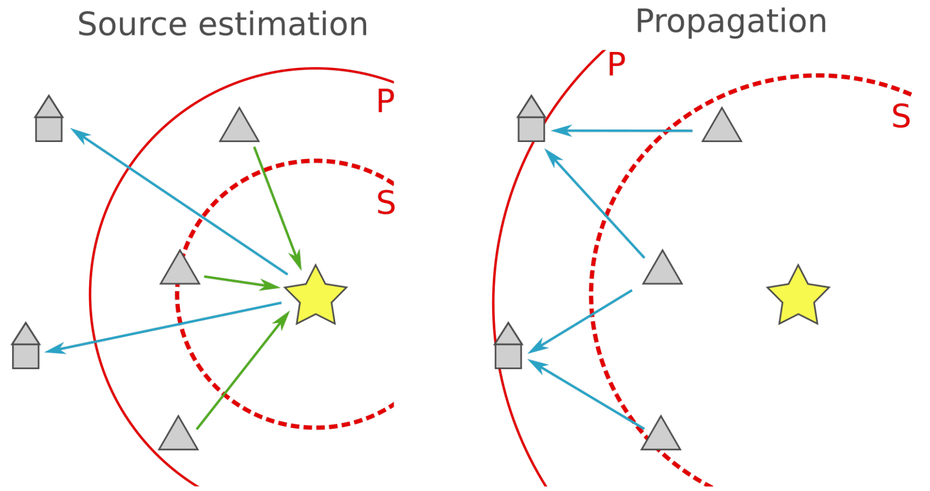


Figure 2.11: Source estimation based and propagation based warning methods. In the source estimation based approach (left), warnings can be issued once the P wave (solid red line) has been recorded. Using the recordings at several stations (triangles), the source characteristics (yellow star) are estimated (green arrows). Based on the source characteristics, the shaking at the targets is estimated (blue arrows). In the propagation based approach, warnings can only be issued once strong shaking, usually from the S wave (dashed red line), has been recorded. Warnings are directly propagated from stations to nearby targets (blue arrows).

difference in travel time between the P wave and the S and surface waves. While the P wave travels fast, most damage is caused by later phases. On-site warning is relatively easy to implement, as it does not require a distributed infrastructure. On the downside, it usually exhibits low precision. In contrast, network based warning uses a collection of several seismic stations to detect and assess an earthquake and distributes warnings to other targets in the area. Network based approaches can provide longer warning times than on-site methods, as their lead times do not only depend on the time difference between P and S waves, but can be prolonged if instruments are located between the earthquake source and the target. A visualisation of the network based ShakeAlert system is provided in Figure 2.10. In addition to longer warning times, network bases methods can usually constrain the expected levels of ground shaking better than on-site methods, as they can use more comprehensive observations of the event. On the downside, network based methods are considerably more complex than on-site approaches, both regarding their algorithms, due to the higher amount and complexity of available recordings, and regarding their implementation, as they require, for example, real-time telemetry.

Network based early warning algorithms follow either a *source estimation based* approach to assess ground shaking, or a *propagation based* approach (Figure 2.11). Source estimation based approaches aim to characterise the source and then infer the ground shaking at the target sites from the source properties. The source can either be characterised as a point source, usually through its magnitude and hypocentral location, or as a finite fault, taking into account the spatial extent of the source. Ground motion is then calculated using ground motion prediction equations (GMPEs) using the obtained source characterisation. Source estimation based methods can achieve long warning times, as first estimates can be obtained as soon as the first P recordings at any station are available. At the same time, these approaches have rather high uncertainties, as they include two modelling steps, source estimation and the GMPE, both making simplifying assumptions. In particular, the uncertainties can not go below the inherent uncertainties in the underlying GMPE. Well known source based algorithms include EPIC [point source, Allen, 2007], FINDER [finite fault, Böse et al., 2012] and PRESTo [point source, Satriano et al., 2011].

In contrast to source based approaches, propagation based approaches do not model the source characteristics but rather infer the expected shaking at a target from recordings at surrounding stations. The most common propagation based algorithm is PLUM [Kodera et al., 2018]. It issues a warning for a certain level of shaking in a region once this level of shaking has been observed at any surrounding stations, i.e., it expects locally undamped propagation of shaking. In particular for large events, this approach leads to good estimates of shaking, as it inherently incorporates aspects such as the radiation pattern, the frequency content or the regional-scale site conditions. On the other hand, warning times for propagation based algorithms are considerably shorter than for source estimation based approaches, as warnings can only be issued once strong shaking has been observed at a station in close proximity to the target.

The performance of early warning algorithms is commonly assessed using the number of correct alerts, false alerts, missed alerts, and the achieved warning times [Meier, 2017, Meier et al., 2020, Minson et al., 2018, 2019]. All of these metrics are typically calculated at different levels of shaking. Most algorithms can be tuned to achieve different trade-offs between those parameters. For example, when reducing the threshold above which warnings are issued, the number of missed alerts will usually decrease, while the number of correct alerts and the warning time will increase. On the other hand, this comes at the cost of an increased number of false alerts. How these metrics should be balanced depends on the target of interest, in particular the costs associated with missed alerts and with false alerts. If the cost for missed alerts is considerably higher than the cost of false alerts, a rather liberal warning strategy is advisable, while vice versa a rather conservative approach should be used if false alerts are similarly expensive as missed alerts. Notably, for no currently existing early warning algorithm perfect performance is expected, i.e., there will always be false or missed alerts [Minson et al., 2019]. This results from the apparent aleatoric uncertainties of the underlying models, i.e., the fact that the observations do not give full information about the events and that the algorithms can not model all aspects of the earthquakes.

2.3.3 Rupture predictability

As discussed above, the warning time is a critical metric for the effectiveness of an early warning system. Typical warning times range from seconds to tens of seconds, depending on the relative location of the earthquake origin, the stations and the vulnerable targets. However, large earthquakes have rupture durations from a few seconds (M_w 6 to 7) to several minutes ($M_w > 9$). For large earthquakes, the rupture duration often surpasses the possible warning time. When the warning is issued, the rupture has not yet been fully observed. This opens up the question, to which extent the size of an earthquake, and consequently the resulting ground shaking, can be constrained before the rupture has terminated. This question is known as *rupture predictability* and has been subject of active research in the last decades [Allen and Melgar, 2019].

We point out that rupture predictability needs to be distinguished from the discussion of a *preparatory phase*. Rupture predictability discusses the possibility to assess the size of an earthquake once it nucleated, in particular from its initiation. In contrast, a potential preparatory phase would happen before the nucleation of an earthquake and lead up to

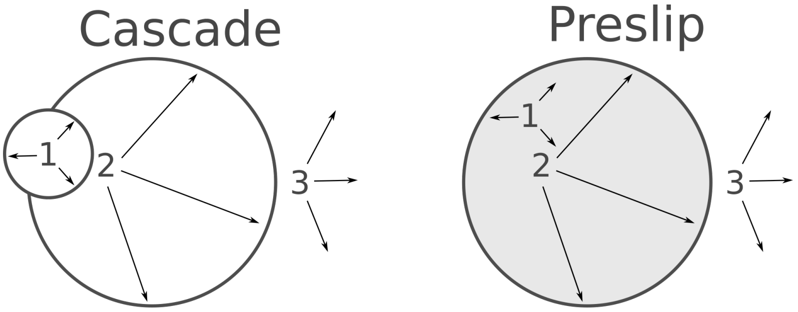


Figure 2.12: Schematic visualisation of the cascade and preslip model for earthquake nucleation. In the cascade model, a sequence of events (1, 2) triggers the breakaway of the large event (3). In the preslip model, aseismic slip occurs within a nucleation zone of confined extent (1, 2, grey area). Once this zone reaches a critical size, the slip accelerates and propagates at high velocity as an earthquake (3). Figure designed after [Ellsworth and Beroza, 1995, Fig.1].

this nucleation. Within this thesis, we will put the focus on rupture predictability (in particular in Chapter 6). We will return to the relation of rupture predictability and a potential preparatory phase when discussing future work in Chapter 7.4.

There are two basic models for the initiation of a large earthquake: the *cascade model* and the *preslip model* [Ellsworth and Beroza, 1995]. Both models are visualised in Figure 2.12. The cascade model proposes that large earthquakes start with a cascade of sequentially growing events triggering each other: a small failure triggers a bigger failure, which in turn triggers an even bigger failure, eventually leading up to the main event. This process is stochastic, and the final size of the event can not be assessed during the rupture. In contrast, the preslip model suggests that large events start with a period of aseismic slip in a so-called nucleation zone. Once this slip patch reaches a critical size, the stable aseismic slip becomes unstable and breaks away at high rupture velocities as an earthquake. The properties of the nucleation zone, in particular its size and the amount of accumulated slip before, might then be indicative of the final earthquake size. This would imply rupture predictability shortly after the onset of the event, at least if the dimensions of the nucleation zone are sufficient to actually observe it.

Both approaches on rupture predictability have been supported with observational evidence. In support of a preslip model, studies found differences between small and large events in the onsets of waveform [Ellsworth and Beroza, 1995], moment rate functions [Danré et al., 2019], ground motion parameters [Colombelli et al., 2020], or geodetic signals [Melgar and Hayes, 2017]. On the other hand, similar evidence, in some cases using even the same observation parameters, has been brought forward for the cascade model. These studies found an universal initiation behaviour, implying that small and large events can not be distinguished early one, in moment rate functions [Meier et al., 2017], waveform onsets [Ide, 2019], or peak displacement [Trugman et al., 2019]. Furthermore, studies of rupture predictability are susceptible to artefacts introduced through the observation process or the analysis, that might be mistaken for signs of predictability. For example, Scherbaum and Bouin [1997] highlighted that finite impulse response filters, commonly used for anti-aliasing in digitisers, can introduce apparent precursory signals, and Meier et al. [2021] pointed out how an apparent predictability in an earlier study results solely from a sampling bias in the analysis.

All studies mentioned above analysed rupture predictability using observations from real world earthquakes. Another line of studies investigates earthquakes in laboratory experiments. Laboratory setups allow for a high degree of control on the experiment, good repeatability, and high-quality near-source recordings. Possible experiments include saw cut samples [Latour et al., 2013, McLaskey and Lockner, 2014], ring shear experiments [Chang et al., 2012], and block sliders [McLaskey, 2019]. As for real world earthquakes, observations from laboratory events are contradictory. However, given the much better possibility for instrumentation, nucleation phases are observed more often. Besides, it is yet unclear to which degree laboratory results can be transferred to natural faults, given that natural conditions can not be fully reproduced in aspects such as pressure, temperature, or complexity of the fault zone.

While the cascade and the preslip model are both well established, they are prototypical models, describing either very high or very low levels of predictability. There are approaches combining aspects of both models. For example, McLaskey [2019] suggested a rate-dependent cascade model, backed by laboratory evidence. In this model, an aseismically slipping patch creates a cascade of small events, of which one then triggers the failure of the large event. Notably, the small events do not trigger each other, but are all triggered by the aseismic slip, which stands in contrast to the cascade model. At the same time, the large event is not triggered by an acceleration and breakaway of the aseismic slip, as in the preslip model, but rather by a small event. While this is only one example of an intermediate model between preslip and cascade, it illustrates that the question of rupture predictability is more nuanced than the end member models. We will discuss these aspects, in particular their inherent probabilistic nature, further in Chapter 6.

2.4 Machine learning

Most analyses in this thesis use *machine learning* (ML), in particular, supervised learning methods. This section introduces the fundamentals of supervised ML, with a focus on the aspects required within this thesis. As such, rather than giving a general overview, the selection of topics might seem biased, e.g., while most introductions to ML focus on classification, we will primarily discuss regression. For a more grounded introduction see, for example, Goodfellow et al. [2016].

A machine learning algorithm is an algorithm that is able to learn from data [Goodfellow et al., 2016]. A supervised ML algorithm is an algorithm that learns a mapping from input data to output data using examples. More formally, let $\mathscr{X}$ be a space of possible input data and $\mathscr{Y}$ a space of possible output data. The output data are often termed *labels*. ML then aims to learn a mapping $f : \mathscr{X} \to \mathscr{Y}$ according to a set of example pairs $(X, y) \in \mathscr{X} \times \mathscr{Y}$, called samples. If $\mathscr{Y}$ is discrete, the task is called *classification*, if it is continuous, it is called *regression*.

The term *learn* is still vague. To this end, let $\mathscr{F} \subseteq \{f \mid f : \mathscr{X} \to \mathscr{Y}\}$ be a set of candidate functions. We will describe possible sets of functions in the next sections. To learn a function f based on samples (X, y), means to select a function from $\mathscr{F}$. This choice is usually made by minimising a *loss function* L, a process also-called *training* the ML algorithm. We will discuss the properties and examples of loss functions in the next section.

To illustrate the concepts above, we introduce a simple example: linear regression for ground motion estimation. Given the magnitude of an earthquake and the distance to a receiver, we want to estimate the peak ground acceleration in log units. We model $\mathscr{X} = \mathbb{R} \times \mathbb{R}^+$, pairs of magnitude m and distance d, and $\mathscr{Y} = \mathbb{R}$, the log peak ground acceleration pga. For linear regression, our candidate functions are $\mathscr{F} = \{(m, d) \mapsto$

$am + bd + c \mid a, b, c \in \mathbb{R}\}$, i.e., all linear functions from magnitude and distance to peak acceleration. Given a set of samples $\{((m_0, d_0), pga_0), \ldots, (m_{n-1}, d_{n-1}), pga_{n-1})\}$, we now aim to select a function $\hat{f} \in \mathscr{F}$. As the loss function, we use the L2 loss defined as $L(y, y') = (y - y')^2$. Therefore, training the algorithm is equal to solving the following optimisation problem:

$$\hat{f} = \text{argmin}_{f \in \mathscr{F}} \frac{1}{n} \sum_{i=0}^{n-1} L(y_i, f(m_i, d_i)) \tag{2.8}$$

$$= \text{argmin}_{f \in \mathscr{F}} \frac{1}{n} \sum_{i=0}^{n-1} (f(m_i, d_i) - y_i)^2 \tag{2.9}$$

While the example is clearly oversimplified from a seismological standpoint, it illustrates all key concepts of supervised machine learning: input and output sets, candidate functions, and the loss function.

2.4.1 Loss functions and scoring

The choice of loss functions is of key importance for ML algorithms as they define towards which objective the algorithm is optimised. We present loss functions in the context of scoring probabilistic forecasts [Gneiting and Raftery, 2007].[6] To reduce the mathematical overhead, in the following we refrain from discussing all required mathematical conditions, such as the σ-algebras or integrability of functions, and rather present the general concepts. A full technical discussion is given by Gneiting and Raftery [2007].

Let $\mathscr{P}$ be a set of probability measures on $\mathscr{Y}$. A choice of probability measure $P \in \mathscr{P}$ is called a probabilistic forecast. A loss function is a function $S : \mathscr{P} \times \mathscr{Y} \to \bar{\mathbb{R}}$, where $\bar{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$. If a forecast P is given and $y \in \mathscr{Y}$ materialises, the loss is $L(P, y)$. For a true distribution $Q \in \mathscr{P}$, we write the expected loss $L(P, Q) = \mathbb{E}_{y \sim Q} L(P, y)$. A loss function is called *proper* relative to $\mathscr{P}$ if

$$L(Q, Q) \leq L(P, Q) \mid \forall P, Q \in \mathscr{P}. \tag{2.10}$$

It is called *strictly proper*, if equality holds only for $P = Q$. In other words, for a (strictly) proper loss function, the minimal loss is achieved if (and only if) the forecast matches the true distribution.

An example of a strictly proper loss function is the negative log-likelihood, defined by $L(P, y) = -\log P(y)$. It is commonly used for classification tasks, where it is also referred to as cross-entropy loss [Goodfellow et al., 2016, Chapter 6.2]. Notably, also the the common L2 loss used in regression can be regarded as a log-likelihood. To this end, we interpret a deterministic prediction $\hat{y}$ as a normal distribution $P = \mathscr{N}(\hat{y}, \sigma^2)$ with mean $\hat{y}$ and fixed standard deviation σ. The resulting negative log-likelihood can be written as:

$$-\log P(y) = -\log \left(\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{y - \hat{y}}{\sigma})^2} \right) \tag{2.11}$$

$$= -\log \left(\frac{1}{\sigma\sqrt{2\pi}} \right) + \frac{1}{2}(\frac{y - \hat{y}}{\sigma})^2 \tag{2.12}$$

As σ is constant, the first term is a constant. Disregarding the equally constant factor of $\frac{1}{2\sigma^2}$ in the second term, we reproduce the simple squared loss $(y - \hat{y})^2$, but now with

[6] Note that we discuss loss functions, while Gneiting and Raftery [2007] discuss scoring rules. However, these concepts only differ with regard to the sign convention: loss functions are minimised, while scoring rules are maximised.

an interpretation in terms of loss functions. We use this loss in Chapter 3 and in slightly modified forms in Chapters 4 and 5.

A particular loss function for forecasts on $\mathbb{R}$ that will be used in Chapter 6 is the *continuous ranked probability score* [CRPS, Matheson and Winkler, 1976]. The key reason for developing the CRPS is the inability of other scores, such as the negative log-likelihood, to take the distance between prediction and true value into account. For example, a prediction with point mass 1 at a location y' will always be scored with ∞ under the negative log-likelihood, as long as $y' \neq y$, with a materialising value of y. However, in many scenarios a forecast with y' close to y should still be regarded as considerably better than one with y' further from y, which is not reflected in the negative log-likelihood. In contrast, the CRPS incorporates this aspect of distance. It is defined as

$$CRPS(P, y) = \int_{-\infty}^{\infty} (F(t) - \mathbb{1}_{\{t \geq y\}})^2 dt \tag{2.13}$$

with F defined as the cumulative distribution function of the predicted distribution P and $\mathbb{1}_{\{t \geq y\}}$ the indicator function taking value 1 if $t \geq y$ and 0 otherwise. The CRPS is a strictly proper scoring rule. The CRPS can also be computed as

$$CRPS(P, y) = -\frac{1}{2}\mathbb{E}_{Y,Y' \sim P}|Y - Y'| + \mathbb{E}_{Y \sim P}|Y - y| \tag{2.14}$$

where Y, Y' are two independent random variables distributed according to P [Gneiting and Raftery, 2007]. The identity (2.14) allows to compute the CRPS analytically for many types of distributions and, furthermore, to evaluate the CRPS using Monte Carlo sampling for others.

To discuss how loss functions for probabilistic forecasts relate to supervised ML, we need to introduce a slight generalisation of our definition for supervised ML. So far, we discussed functions $f : \mathscr{X} \to \mathscr{Y}$, i.e., *deterministic* predictors. We extend this notion by introducing *probabilistic* predictors. Let again $\mathscr{P}$ be a set of probability measures on $\mathscr{Y}$. Learning probabilistic predictor now means selecting a function $f : \mathscr{X} \to \mathscr{P}$, with the goal that $f(x) \approx \mathbb{P}(y|x)$, i.e., a function describing the distribution of labels y given the data x. Often deterministic predictions are implicitly interpreted as a probabilistic forecast: in the example above, showing that the L2 loss is equivalent to the log-likelihood of a Gaussian, the deterministic point prediction was interpreted as a Gaussian with fixed standard deviation. We will return to this point in more detail in Chapter 2.5.7.

To assess the quality of the fit $f(x) \approx \mathbb{P}(y|x)$, loss functions can be used, i.e., the goal of the ML algorithm is minimising $L(f(x), \mathbb{P}(y|x))$. Clearly, this term does not only need to be minimised for a fixed x but in expectation over all x, giving:

$$\begin{aligned} \mathbb{E}_X[L(f(x), \mathbb{P}(y|x))] &= \mathbb{E}_X[\mathbb{E}_{Y|X}[L(f(x), y)]] &(2.15)\\ &= \mathbb{E}_{X,Y}[L(f(x), y)] &(2.16) \end{aligned}$$

The first equality is simply the notational convention defined above. The second equality results from the law of total expectation. We use the subscripts of the expectation to indicate the random variable that is integrated over: $\mathbb{E}_X$ is the expectation with respect to X, $\mathbb{E}_{Y|X}$ the conditional expectation, and $\mathbb{E}_{X,Y}$ the expectation with respect to X and Y. The last term can be approximated using a set of samples $\{(x_0, y_0), \dots (x_{n-1}, y_{n-1})\} \subset \mathscr{X} \times \mathscr{Y}$.

$$\frac{1}{n}\sum_{i=0}^{n-1} L(f(x_i), y_i) \xrightarrow{n \to \infty} \mathbb{E}_{X,Y}[L(f(x), y)] \tag{2.17}$$

Notably, this estimation is possible even without having samples from $\mathbb{P}(y|x)$ available for all possible x.

While the discussion of proper loss functions allows to derive theoretical guarantees on the optimality of models, these guarantees rely on several critical assumptions: (i) a sufficiently rich class of models, (ii) identically, independently distributed data and (iii) infinitely many samples. Assumption (i) refers to the class of probability measures $\mathscr{P}$ and mappings $\mathscr{F}$. So far, we assumed that the true distribution $Q \in \mathscr{P}$. However, this will often not be the case, for example, when looking at the simple Gaussian error model introduced to derive the L2 loss. Still, loss functions can be used to derive an element of $\hat{Q} \in \mathscr{P}$ that is close to Q, i.e., that has a low loss. The difference between $\hat{Q}$ and Q is called the *approximation error*. However, we will later show (Chapter 2.5.1) that using deep learning, models $\mathscr{P}$ can be chosen sufficiently expressive to effectively disregard this error.

Assumption (ii), independently, identically distributed data, will in most cases not be fully satisfied in real world data. First, the data generation process will often be non-stationary, i.e., the generating distribution might change over time. Second, the independence of samples is often not given. For example, in a dataset of earthquakes, the independence would be violated by fore- and aftershock sequences, or on a larger scale even by the seismic cycle. Therefore evaluating a model requires careful assessment to which extent the assumptions of independence and identical distribution are satisfied.

Assumption (iii), infinitely many samples, is relevant as the approximation in equation (2.17) only converges in the limit. With a finite amount of data, instead of selecting the optimal predictor $\hat{Q} \in \mathscr{P}$, the loss will be minimised for Q^*. The difference between $\hat{Q}$ and Q^* is called *estimation error*. The magnitude of this error depends critically on the number of samples used. This does not only mean the total number of samples overall but also samples for certain ranges of x or y. We will encounter the impact of this limitation throughout this thesis (Chapters 3 to 6) when discussing degraded estimation performance in low data scenarios.

2.4.2 Training and evaluating ML algorithms

As shown in the previous section, several aspects need to be taken into account when using loss functions for training ML algorithms. Therefore, in this section we discuss how to train and evaluate ML algorithms given these considerations [Goodfellow et al., 2016, Chapter 5].

We will start of at limitation (iii), finite data. Given a finite collection of samples

$$\{(x_0, y_0), \ldots, (x_{n-1}, y_{n-1})\} \sim Q, \tag{2.18}$$

the most likely distribution generating this sample, at least when making no further assumptions about the generating process, is an equally weighted collection of point masses $\hat{Q} = \frac{1}{n} \sum_{i=0}^{n-1} \delta_{(x_i, y_i)}$. Consequently, for the conditional distributions we get $\mathbb{P}(y|x) = \frac{1}{n_x} \sum_{i|x=x_i} \delta_{y_i}$. Notably, this distribution is not defined for any x for which we do not have any sample. In addition, this model, while fitting perfectly the empirical distribution of our samples, might not actually fit the underlying generating distribution Q, which usually differs from $\hat{Q}$. This concept, a model fitting the empirical distribution well but not the underlying generating distribution, is known as *overfitting*. In practice, this would mean predictions on previously unseen data would likely be incorrect. Equivalently, one says that the model shows poor *generalisation* ability from the training data [Goodfellow et al., 2016, Chapter 5.2].

The key idea for addressing overfitting is to assume some notion of smoothness in the conditional distributions. For x and x' close, the conditional distributions $\mathbb{P}(y|x)$ and $\mathbb{P}(y|x')$ will likely be close as well. This smoothness is commonly enforced through *regularisation.* One way of regularisation is introducing an additional penalty term $R : \mathscr{F} \to \mathbb{R}$ on the model that penalises complex models. This stands in agreement with Occam's razor [Duignan, 2021], favouring simple models. Instead of optimising the loss alone, one optimises the sum of the loss and the penalty term:

$$\operatorname{argmin}_{f \in \mathscr{F}} L(f(x), \hat{Q}) + \lambda R(f) \tag{2.19}$$

The parameter λ can be used to adjust the strength of regularisation.

While regularisation can reduce or even eliminate overfitting using a single collection of samples, we actually cannot easily measure whether our model overfits or shows appropriate fit. Therefore, a common practice in ML is to split the available samples into a *training set* and a *test set.* We furthermore introduce a third set, disjoint from the other two, called the *development set*, which we will justify below.[7] The ML algorithm is now trained on the training set and then evaluated on the test set. If the test set is an independent sample from Q, it is not affected by the overfitting and the performance on the test set gives a reliable estimate of the model performance on Q.

In most cases, the ML algorithm has parameters controlling its behaviour that are not part of the optimisation, such as the weight term λ of the regularisation in (2.19). These parameters are called *hyperparameters* and their optimisation is called hyperparameter tuning. The usual way for tuning hyperparameters is training multiple models with different hyperparameters and comparing their performance. If one would use the test set for this performance comparison, one becomes susceptible to hyperparameter overfitting, i.e., one might choose hyperparameters that fit the test set particularly well. In other words, by selecting between models based on their performance on the test set, the estimate of their performance becomes unreliable. To this end, the development set is used instead. One selects the best model using the performance on the development set and then evaluates it on the test set, thereby obtaining a reliable performance estimate.

The reliability of the performance estimate depends on limitation (ii), the independence and identical distribution of the data set. If training, development and test data are not independent, potential overfitting on the training set might still give a (positively) biased performance assessment on the development and test set. It the obtained model is then applied to actual independent data, the performance will likely be worse.[8] In practice, it will often be impossible to achieve actual independence and identical distribution between training and test set, for example, because the underlying generating distribution is non-stationary. Therefore, it is required to choose splits to resemble the envisioned application scenario. For this reason, we will explicitly discuss the choice of splits and how the splitting could potentially affect our results in the main chapters (Chapters 3 to 6).

[7] While we adapt the terminology training/development/test set, the sets are sometimes also referred to as training/validation/test. We decided on the first option to avoid confusion, as the term validation set is sometimes also used to refer to the test set.

[8] We highlight that *independence* here needs to be interpreted in a purely mathematical sense, i.e., as independence of random variables. The samples should be independent between training and test set in a stochastic sense. In contrast, they should result from the same generating distribution and, in particular, follow the same conditional distributions $\mathbb{P}(Y|X)$. In this sense, one might commonly call these samples *not independent*, as they expose the same governing laws. However, this is not the notion of independence used here.

So far, we discussed evaluating model performance using loss functions. However, often the quality of a model in practical applications is not fully reflected by the loss but rather in terms of a downstream metric directly relevant for the application. For example, when training a model for fast ground motion estimation in the context of early warning, the evaluated metric will usually depend on the true, false, and missed warnings, as well as the warning time, instead of the loss itself. ML models are usually trained using loss functions, as these are easier to optimise than these downstream metrics, e.g., because of smoothness and differentiability properties. In this sense the loss function serves as a surrogate for the metric. When selecting models or model hyperparameters however, an optimisation task that is commonly performed by comparing separately trained models and where optimisation requirements as for the loss functions do not apply, the downstream metric should be used as selection criterion rather than the loss.

2.4.3 Calibration

Many commonly used metrics for evaluating machine learning models hide the probabilistic aspects of the prediction. For example, classification tasks are often evaluated using accuracy, regression tasks using the root mean squared error. In both cases, it is disregarded that the model might give an assessment of its confidence or vice versa its uncertainty. However, high quality uncertainty estimates are essential for making informed decision, for example, to ensure that predictions with high uncertainties are not relied on. Recent studies [Guo et al., 2017, Snoek et al., 2019] showed that while developments in deep learning (see Chapter 2.5) considerably improved, for example, the accuracy of models, the quality of their uncertainty estimates decreased. The models are overconfident, i.e., overestimate their confidence and underestimate their uncertainty. This can be interpreted as overfitting in the probability domain. As within this thesis, we are taking a probabilistic viewpoint on the machine learning methods and underlying seismological questions, we also need to analyse the quality of the uncertainty estimates (Chapters 4 to 6).

One way to analyse the quality of uncertainty estimates is through their *calibration*. For illustration purposes we first discuss calibration for a simple binary classification task between classes -1 and 1. In this task, the prediction from a machine learning model f is simply the probability $p \in [0, 1]$ of a sample belonging to class 1. Let $S_{[p,p+\varepsilon]}$ be the set of samples with a predicted probability between p and $p + \varepsilon$.[9] A model f is now well calibrated if for all p

$$\frac{1}{|S_{[p,p+\varepsilon]}|} \sum_{(x,y)\in S_{[p,p+\varepsilon]}} f(x) \approx \frac{|\{x \in S_{[p,p+\varepsilon]} \mid y = 1\}|}{|S_{[p,p+\varepsilon]}|} \tag{2.20}$$

where $|.|$ is the cardinality of a set. We note that the left term, by definition of $S_{[p,p+\varepsilon]}$, will fall between p and $p + \varepsilon$. Rephrasing equation (2.20), a model is *well calibrated* if among all predictions with probability p the fraction of samples actually belonging to class 1 is p as well. If this condition is not met, a model is called *miscalibrated*. Note that calibration alone is not a sufficient condition for a useful classifier. For example, the marginal distribution $\mathbb{P}(y)$ is by definition a perfectly calibrated model, at the same time it is not a useful model.

For continuous predictions, such as in regression tasks, calibration needs to be analysed differently, as predictions are not associated with probability mass but rather densities.

[9] This is simply an approximation of the predictions with probability p required due to the finite sample size. For a practical discussion on how to choose ε, see Guo et al. [2017] or Snoek et al. [2019].

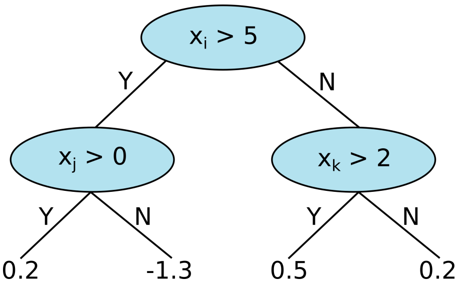


Figure 2.13: Schematic visualisation of a decision tree of depth two. Given an input vector $x \in \mathbb{R}^d$ the tree uses the entries i, j and k to determined the value to assign. At each node the tree checks whether the provided condition is fulfilled (Y) or not (N). The tree leafs indicate the assigned values w.

We only discuss calibration analysis for single dimensional predictions. For a sample (x_i, y_i), let F_{x_i} be the cumulative distribution function predicted by the model using sample x_i. We now calculate the quantile q_i at which the correct label y_i occurs as $q_i = F_{x_i}(y_i)$. For a well calibrated model, following the definition of the cumulative distribution function, q_i must be distributed according to a uniform distribution $\mathscr{U}([0,1])$. This can be verified using the samples $q_0, \dots, q_{n-1}$, for example, through a Kolmogorov–Smirnov test [Kolmogorov, 1933, Smirnov, 1948].

2.4.4 Decision tree ensembles and gradient boosted trees

After introducing the fundamentals of machine learning, we now present different machine learning methods. We start with a classical approach, decision tree ensembles, before presenting deep learning in the subsequent Chapter 2.5.

Decision tree ensembles are machine learning models applicable to both regression and classification [Chen and Guestrin, 2016]. We will use decision tree ensembles for magnitude scale calibration in Chapter 3. A decision tree ensemble is a collection of decision trees $f_0, \dots, f_{m-1}$. Each decision tree f_j partitions the input space $\mathscr{X}$ into pairwise disjoint subspaces $\mathscr{X}_0 \dot\cup \dots \dot\cup \mathscr{X}_{p-1} = \mathscr{X}$ and assigns a label $w_0, \dots, w_{p-1}$ to each subspace. The function f_j is a piecewise constant function:

$$f_j(x) = \Big\{ w_i \quad | \, x \in \mathscr{X}_i \tag{2.21}$$

Partitioning is performed along thresholds of single features, yielding a tree-like structure as visualised in Figure 2.13, and giving rise to the name decision tree.

Based on these single decision trees, an ensemble function f, the decision tree ensemble, is defined as the sum of the ensemble members:

$$f(x) = \sum_{i=0}^{m-1} f_i(x) \tag{2.22}$$

Notably, as every decision tree is a piecewise constant function, decision tree ensembles are piecewise constant as well. This implies, that their output space is discrete rather than continuous, which stands in contrast to other regression methods, for example, linear regression. However, in practice the number of outputs can be very high, given a sufficient number of trees and sufficient tree depth. Therefore, the discretisation error will usually be far below the remaining uncertainties in the model.

There are different ways to train a decision tree ensemble. Here we present *gradient boosting*, as we use this approach in Chapter 3. For a detailed mathematical derivation of gradient boosting, see [Chen and Guestrin, 2016]. Gradient boosting iteratively adds trees to the ensemble, starting from a constant tree $f_0 \equiv c$. Each new tree models the residual between the current prediction and the actual labels. To train this new tree, two aspects need to be optimised: the partitions and the value for each partition.

We first discuss how to choose the values for each partition, given fixed partitions, before discussing how to determine the partitions. For this, we take a single sample (x_0, y_0) with $x \in \mathscr{X}_j$ of the new tree f_k. The sample is assigned value w_j in the new tree and we aim to find the optimal value w_j. The loss for the corresponding sample can be expressed as:

$$\begin{aligned} L\left(y_0, \sum_{i=0}^{k} f_i(x_0)\right) &= L\left(y_0, \sum_{i=0}^{k-1} f_i(x_0) + f_k(x_0)\right) && (2.23)\\ &= L\left(y_0, \sum_{i=0}^{k-1} f_i(x_0) + w_j\right) && (2.24)\\ &\approx L\left(y_0, \sum_{i=0}^{k-1} f_i(x_0)\right) + w_j g + w_j^2 h && (2.25) \end{aligned}$$

The approximation in equation (2.25) is the second order Taylor approximation, with g the gradient and h the Hessian with respect to the second component of the loss:

$$\begin{aligned} g &= \partial_2 L\left(y_0, \sum_{i=0}^{k-1} f_i(x_0)\right) && (2.26)\\ h &= \partial_2^2 L\left(y_0, \sum_{i=0}^{k-1} f_i(x_0)\right) && (2.27) \end{aligned}$$

Here we write ∂_2 to indicate the partial derivative with respect to the second component of L. The Taylor approximation in equation (2.25) is a second order function of the weight w_j and can be minimised analytically. The same holds true when adding up the loss for all training samples. Consequently, assuming a given partition, an approximation of the optimal weights can be chosen efficiently.

In addition to providing the (approximate) optimal weights, the analytic expression allows to quantify the improvement achieved by adding a new tree with a given partition. This property can be used to find suitable partitions by simply iterating over possible partitions. As enumerating all possible partitions is combinatorically intractable, the loss is optimised greedily level by level. In every level, a one-dimensional feature is selected and the optimal threshold in terms of the loss reduction is calculated. This process is iterated until the maximum number of levels is reached or the addition of a new level does not reduce the loss any further.

In addition to providing a computationally efficient way of training decision tree ensembles, gradient boosting gives a way for interpreting the importance of each features. Each time a feature is used for splitting partitions, a reduction in the loss is associated with this split. The average loss reduction for each feature, therefore, describes how relevant each feature is towards the task. A feature with high loss reduction is considered more relevant than a feature with lower loss reduction. We use this approach for interpreting feature importance in Chapter 3.

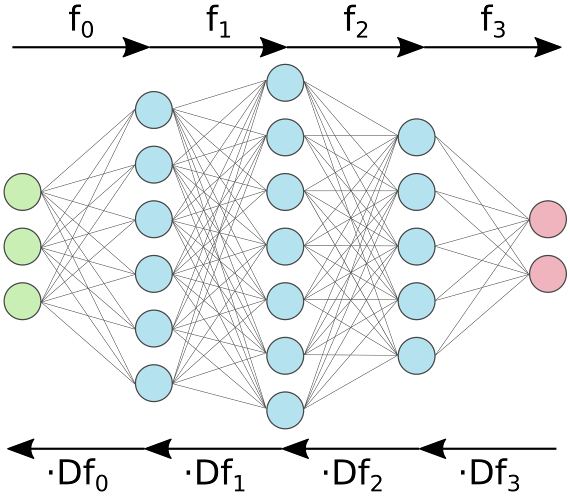


Figure 2.14: Schematic visualisation of a multi-layer perceptron, a simple type of neural network. The input layer is shown in green, hidden layers in blue and the output layer in red. Each value in each layer is called a neuron, visualised by a circle. The connections indicate linear combinations. Bias terms and non-linearities are not shown. The arrows on the top show the forward pass, the sequential application of the layers. The bottom arrows show the backward pass, a series of multiplications of the derivatives of the layers.

2.5 Deep learning

A subdiscipline of machine learning that has gained widespread attention in the last decade is *deep learning* [LeCun et al., 2015]. At the basis of deep learning are *artificial neural networks* (ANNs or simply NNs). Neural networks are parameterised functions $f^\theta : \mathbb{R}^{d_x} \to \mathbb{R}^{d_y}$, mapping inputs x to outputs y. The parameters θ are called *weights*. Neural networks are constituted of several simple subfunctions $f_0^{\theta_0} : \mathbb{R}^{d_x} \to \mathbb{R}^{d_1}, \dots, f_{n-1}^{\theta_{n-1}} : \mathbb{R}^{d_{n-1}} \to \mathbb{R}^{d_y}$, also-called *layers*, such that $f = f_{n-1}^{\theta_{n-1}} \circ \dots \circ f_0^{\theta_0}$. The number of layers n is called the *depth* of the network. A neural network is called deep if it consists of several layers, giving rise to the term deep learning.[10] The maximal number of dimensions $\max_i d_i$ is called the *width* of the network. A simple example network is shown in Figure 2.14.

Each layer outputs an intermediate representation $z_{i+1} = f_i^{\theta_i}(z_i)$, where $z_0 = x$. As for the functions, the intermediate representations are sometimes also referred to as layers: the *input layer* $x = z_0$, the *hidden layers* $z_1, \dots, z_{n-1}$, and the *output layer* $y = z_n$. By transforming between the different representations, the neural network aims to find a representation in which the targeted estimation problem is easily solvable, e.g., where classification can be achieved with a linear classifier. For this reason, neural network algorithms belong to the *representation learning* methods.

A key difference between deep learning algorithms and most classical approaches is the role of features. Classical machine learning algorithms typically need to be provided with hand-crafted features. Features are representations of the input data, for example, seismic waveforms could be described by their mean, standard deviation or certain quantiles. Feature engineering, the process of designing appropriate features, is laborious and requires expert knowledge. In contrast, through the representation learning approach, neural networks can usually be fed with raw data that can be very high dimensional or

[10] There is no universal agreement on the number of layers required to qualify as deep, with some advocating that already a NN with depth 2 is deep. As this is only a question of terminology, we do not take a stand and stick to the informal "several layers".

even of structured nature, such as graphs [Battaglia et al., 2018]. Neural networks then learn to extract features themselves by finding representations useful for the task at hand.

Several architectures, i.e., choices of layers $f_i^{\theta_i}$, exist. The simplest form of a neural network layer is the fully connected layer [Goodfellow et al., 2016, Chapter 6]. The function $f_i : \mathbb{R}^{d_{i-1}} \to \mathbb{R}^{d_i}$ for a fully connected layer is given as

$$f_i^{\theta_i}(x) = \sigma(Ax + b) \tag{2.28}$$

where $A \in \mathbb{R}^{d_i \times d_{i-1}}$ is a matrix, $b \in \mathbb{R}^{d_i}$ a bias vector, and $\sigma : \mathbb{R} \to \mathbb{R}$ a non-linearity, that is applied pointwise to each entry of the vector. The matrix A and the bias vector b constitute the parameters θ_i. f_i is the combination of an affine transformation with a non-linearity σ. Common non-linearities σ are the hyperbolic tangent or the ReLu function [Nair and Hinton, 2010], given by $\sigma(x) = \max(0, x)$. A stack of multiple fully connected layers is called a multilayer perceptron (MLP). A visualisation can be found in Figure 2.14.

2.5.1 The universal approximation theorem

While simple in structure, MLPs are very versatile, as shown by the different variants of the *universal approximation theorem* [Goodfellow et al., 2016, Chapter 6.4.1]. Informally speaking, the universal approximation theorem states that for every continuous function $g : K \to \mathbb{R}^m$ there exists an MLP that approximates it arbitrarily well. More formally, for any compact subset $K \subseteq \mathbb{R}^n$ and $\varepsilon > 0$ there exists a MLP f with width at most $m + n + 2$, such that

$$\sup_{x \in K} ||g(x) - f(x)|| < \varepsilon. \tag{2.29}$$

A similar result holds for two-layer MLPs with unbounded width of the hidden layer. Both results can be proven under mild assumptions on the non-linearity σ. In other words, neural networks are dense in the space of continuous functions $C(K, \mathbb{R}^m)$ with respect to the L_∞ norm. By choosing a sufficiently large neural network, the approximation error (introduced in Chapter 2.4.1) can be made arbitrarily small.

While the universal approximation theorems only hold in the limiting case of infinitely sized neural networks, in practice the error ε already becomes neglectable small for most applications when using MLPs with practically feasible numbers of layers and neurons. The additional assumption of a compact subset K rather than the full space $\mathbb{R}^n$ is equally unimportant for practical applications, as the input data is usually bounded and thereby lies within a compact subset. This also explains why assumption (i) introduced in Chapter 2.4.1, the requirement of a sufficiently rich class of functions, is usually not a concern in deep learning. The universal approximation property of neural networks makes them particularly suitable for our analyses in Chapters 4 to 6.

2.5.2 Training neural networks

While the universal approximation theorem showed that neural networks can represent arbitrary functions, we did not yet discuss how to find the network weights, i.e., how to train neural networks. Training a given neural networks architecture means selecting good parameters θ. Let f^θ be the neural network, $(x_0, y_0), \dots (x_{n-1}, y_{n-1}) \in \mathscr{X} \times \mathscr{Y}$ the training examples and L the loss function. The parameter selection can be represented as:

$$\hat{\theta} = \operatorname{argmin}_\theta L(\theta) \tag{2.30}$$

$$L(\theta) = \frac{1}{n}\sum_{i=0}^{n-1} L(f^\theta(x_i), y_i) \tag{2.31}$$

We use $L(\theta)$ as a shorthand notation for the loss given the parameter choice θ. As θ usually has a very high number of dimensions, this optimisation problem can rarely be solved exactly. Instead, the term is typically minimised using a *gradient descent* algorithm [Goodfellow et al., 2016, Chapter 8].

For gradient descent, first, starting parameters θ are initialised. Different schemes for initialisation exist, which can have a major impact on the final network performance [Glorot and Bengio, 2010, Frankle and Carbin, 2018]. Second, gradient update steps are performed. For this, the parameters θ are updated using a rule of the form

$$\theta := \theta - \eta \nabla L(\theta) \tag{2.32}$$

with ∇ representing the gradient with respect to θ and η a constant, called the *learning rate*.[11] As long as the gradient is non-zero and the learning rate η is sufficiently small, the new parameters θ will have a lower loss than the old ones.

In (2.32) the gradient is calculated across the full training dataset. In practice, evaluating the gradient on the full training dataset is computationally expensive and often even infeasible. For this reason, neural networks are usually trained using *minibatch gradient descent*, also known as *stochastic gradient descent* (SGD).[12] Here, instead of evaluating the gradient on the full training set, one estimates the gradient on a subset $(x_{i_0}, y_{i_0}), \ldots, (x_{i_{m-1}}, y_{i_{m-1}})$ of the training dataset. The number of examples m is called the *batch size*. The resulting update rule and gradient approximation are:

$$\theta := \theta - \frac{m}{n}\eta \nabla L'(\theta) \tag{2.33}$$

$$\nabla L'(\theta) = \frac{1}{m}\sum_{j=0}^{m-1} \nabla L(f^\theta(x_{i_j}), y_{i_j}) \approx \nabla L(\theta) \tag{2.34}$$

Stochastic gradient descent usually converges significantly faster than batch gradient descent in terms of the number of evaluations of L on each sample. Furthermore, SGD has a regularising effect, often leading to models with better generalisation ability [Wilson and Martinez, 2003].

The rules presented in (2.32) and (2.33) are very simplistic examples of update rules. In practice, often further terms are added, for example, momentum [Qian, 1999] or adaptive learning rate terms [Kingma and Ba, 2014]. These lead to better convergence behaviour than vanilla SGD which has been verified both theoretically and empirically. For a comprehensive overview of gradient descent strategies for deep learning see Ruder [2016].

[11] The vast majority of algorithms used for optimising neural networks only use first order derivatives and can be expressed using a similar update rule as provided in (2.32). Methods using higher order derivatives, such as the L-BFGS algorithm, are usually not computationally feasible for neural network optimisation [Goodfellow et al., 2016, Chapter 8].

[12] Gradient descent using the gradient of the full training set is sometimes referred to as batch gradient descent. As this term might lead to confusion with minibatch/stochastic gradient descent, we abstain from using it. The confusion is particularly problematic, as the term "batch" is commonly used to refer to a single "minibatch" in minibatch gradient descent.

To apply gradient descent methods efficiently to neural networks, one needs to calculate the gradient of the loss function with respect to the parameters θ efficiently. This can be done using backpropagation [Goodfellow et al., 2016, Chapter 6.5]. Backpropagation relies on the structure of a neural network $f = f_{n-1} \circ \cdots \circ f_0$ and the chain rule $D_x(u \circ v) = D_{u(x)} v \cdot D_x u$, where $D_x u$ denotes the total derivative of u at x. We introduce the shorthand notation $f^{:k} = f_{k-1} \circ \cdots \circ f_0$ for the first k layers in the network. The derivative of f can be computed as:

$$\begin{aligned} D_{(x,\theta)} f &= D_{(x,\theta)}(f_{n-1} \circ f^{:n-1}) && (2.35)\\ &= D_{(f^{:n-1}(x),\theta)} f_{n-1} \cdot D_{(x,\theta)} f^{:n-1} && (2.36)\\ &= D_{(f^{:n-1}(x),\theta)} f_{n-1} \cdot D_{(f^{:n-2}(x),\theta)} f_{n-2} \cdot D_{(x,\theta)} f^{:n-2} && (2.37) \end{aligned}$$

Similarly, applying the chain rule to the loss term we obtain:

$$D_{(x,\theta)} L(f(.), y) = D_{(f(x),\theta)} L(., y) \cdot D_{(x,\theta)} f \qquad (2.38)$$

This means that the total differential, and thereby the gradient with respect to θ, can be calculated iteratively by calculating the derivative of each layer. This process is called backpropagation or backward pass: the neural network is “applied backward”, i.e., gradients of each layer are calculated sequentially backwards from the loss towards the inputs (see also Figure 2.14). The only requirement is that all layers f_i need to be differentiable. For all layer types presented in this thesis, differentiability is given.

2.5.3 Stabilising training for deep networks

While in principle very deep networks can easily be built by stacking many layers, in practice deeper networks turn out to be significantly harder to train than shallow ones [Bengio et al., 1994, Glorot and Bengio, 2010, He et al., 2016]. One cause for this behaviour is *vanishing/exploding gradients* [Bengio et al., 1994, Glorot and Bengio, 2010], i.e., gradients with either very small or very large norms. Vanishing/exploding gradients can be explained with the backpropagation algorithm. With each step of backpropagation, i.e., each multiplication with the derivative of a layer, the norm of the gradients changes. If the norm of the derivative of the layers constantly is larger than 1, the norm of the gradients will explode for the early layers. Similarly, if the norm is constantly below zero, the norm of the gradients will vanish. In both cases, it is impossible to choose an appropriate learning rate for all layers jointly, as the gradients can differ by orders of magnitude between layers. Therefore, gradient descent will effectively only train some layers while leaving others untouched (vanishing gradients) or making significantly too large steps (exploding gradients).

Vanishing/exploding gradients can be mitigated by choosing an appropriate parameter initialisation [Glorot and Bengio, 2010], and by adding intermediate normalisation layers. As appropriate parameter initialisation is nowadays usually handled automatically by most deep learning frameworks, we only present two normalisation layers: *batch normalisation* and *layer normalisation.*

Batch normalisation recenters and rescales a hidden representation using statistics of the current minibatch used in training [Ioffe and Szegedy, 2015]. Let $z^0, \dots, z^{m-1} \in \mathbb{R}^d$ be the hidden representations at a fixed layer corresponding to inputs $x_0, \dots, x_{m-1}$. Batch normalisation first calculates the mean vector μ and variance vector σ^2 over z.

$$\mu = \frac{1}{m}\sum_{i=0}^{m-1} z^i \tag{2.39}$$

$$\sigma^2 = \frac{1}{m}\sum_{i=0}^{m-1} (z^i - \mu)^2 \tag{2.40}$$

Using these summary statistics, the outputs $\hat{z}^i$ are calculated as

$$\hat{z}^i = \gamma \frac{z^i - \mu}{\sqrt{\sigma^2 + \varepsilon}} + \beta. \tag{2.41}$$

$\varepsilon > 0$ ensures the stability of the division and the derivative of the square root at 0. $\gamma, \beta \in \mathbb{R}^d$ are learnable parameters, representing the new mean and standard deviation of the data. During the evaluation phase, in contrast to during the training, calculating batch statistics is not desirable, as predictions for each sample should be independent of the batching. Therefore, μ and σ are not derived from the evaluation samples, but instead, the population mean and standard deviation of the training set are used.

While abundant empirical evidence has proven the effectiveness of batch normalisation, its theoretical underpinning is still purely understood [Santurkar et al., 2018]. Initial publications suggested that batch normalisation reduces the so-called internal covariate shift, describing that the means and standard deviations of the inputs of layers change over time during training [Ioffe and Szegedy, 2015]. However, later research suggested that internal covariate shift does not explain the success of batch normalisation. Instead, Santurkar et al. [2018] argue that batch normalisation makes the optimisation landscape more smooth and thereby makes the gradient estimates more stable. Note that, as batch normalisation depends on minibatch statistics, it introduces a further dependency of the learning behaviour on the batch size.

An alternative to batch normalisation that does not rely on batch statistics in layer normalisation. Layer normalisation can be interpreted as a transposed batch normalisation: the statistics for normalisation are calculated along the feature dimension of each sample individually instead of along the batch dimension. Given a hidden representation $z \in \mathbb{R}^d$, mean and variance are calculated as

$$\mu = \frac{1}{d}\sum_{i=1}^{d} z_i \tag{2.42}$$

$$\sigma^2 = \frac{1}{d}\sum_{i=1}^{d} (z_i - \mu)^2 \tag{2.43}$$

where z_i indicates the ith element in the vector z. In contrast to batch normalisation, here μ and σ are not vectors, but scalar values. The inputs are then rescaled using the same formula as for batch normalisation (2.41), just with the alternative definition of μ and σ. As the mean and standard deviation for the layer normalisation can be derived from a single sample, the same normalisation can be applied in both training and evaluation. Similar to batch normalisation, layer normalisation has become commonly used in many architectures [Ba et al., 2016, Vaswani et al., 2017].

Even with appropriate initialisation and intermediate normalisation layers, sufficiently deep networks still show degraded performance in comparison to shallower ones [He et al., 2016]. This is surprising, as a network with more layers is in principle more expressive, i.e., it can in principle model a larger class of functions. This follows directly from the fact that any added layer might simply learn an identity function, thereby reproducing a network with fewer layers. In practice, however, learning this identity function is difficult to achieve [He et al., 2016], leading to the degraded performance for deeper networks. To mitigate this, *residual connections*, also termed *skip connections* were introduced. In a skip connection, the input x to a layer f is added to its output, giving a new output $f'(x) = f(x) + x$. The function f now only needs to model the residual to an identity function. In particular, this makes learning the identify function itself easily feasible, as it only requires learning the constant function $f \equiv 0$. Residual connections can not only be established around single layers but also around stacks of layers, as long as input and output shapes agree. Using residual connections, deeper networks can be trained that outperform shallow networks considerably. Consequently, they have found widespread application in state of the art models [Ronneberger et al., 2015, He et al., 2016, Vaswani et al., 2017].

2.5.4 Overparametrisation and equivariance

After discussing training strategies and methods for stabilising the training, we now turn towards specific neural network architectures. To motivate the need for these architectures, we first introduce the problem of *overparametrisation.* The universal approximation theorem guarantees that any smooth function can be approximated arbitrarily well with an MLP. However, it gives no indication of the amount of training data required to actually achieve a certain quality of estimation. In practice, neural networks are usually massively overparametrised. This means that the optimal weights can not be properly constrained from the training data alone [Goodfellow et al., 2016, Chapter 7], i.e., there are different sets of weights explaining the training data similarly well. It is however unclear, which choice of weights fits the underlying generating distribution best and would yields the best model. To mitigate this issue, models are given a so-called *inductive bias*, either through regularisation, as discussed in general terms before, or through particular network architectures, which we are going to discuss below.

Neural network architectures can be interpreted through their symmetries using the concept of *equivariance* [Bronstein et al., 2021]. Simply speaking, a function is *equivariant* with respect to a transformation, if applying the transformation to the input only changes the output of the function by a similar transformation. For example, a function is shift-equivariant if applying a shift to the input data leads to an identical shift to the output data. To use equivariance in neural network design, one needs to identify the symmetries in the data. For example, when analysing waveform data from a collection of seismic stations, the result should be independent of the order of stations in the input, i.e., there is a permutation invariance. A neural network for this task should be designed to be permutation-equivariant. This greatly reduces the number of parameters in comparison to a neural network modelling all possible functions and thereby can be optimised using fewer samples.

For a formal definition of equivariance, let $f : X \to Y$ be a function and G be a group with (left) group actions on both X and Y. f is called *equivariant* with respect to G if for all $x \in X$ and $g \in G$ the condition $f(g \cdot x) = g \cdot f(x)$ is fulfilled. For example, let $f : \mathbb{R}^{n \times d_1} \to \mathbb{R}^{n \times d_2}$ be a function and S_n be the permutation group of n elements. The function f is called equivariant with respect to S_n, if for all $(x_0, \dots, x_{n-1}) \in \mathbb{R}^{n \times d_1}$ and $\pi \in S_n$ the equality $f(\pi(x_0, \dots, x_{n-1})) = \pi(f(x_0, \dots, x_{n-1}))$ holds. In other words,

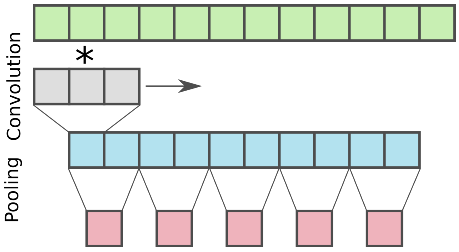


Figure 2.15: Schematic visualisation of a 1D convolution and a pooling layer. The input sequence is visualised in green, the convolution kernel in grey, the sequence after the convolution in blue, and the sequence after pooling in red. For visualisation purposes, only a single channel is shown. In practice, the input can have multiple channels and the convolutional kernel will be of size *kernel width* × *input channels* × *output channels*. The output sequence of the convolution is shorter than the input sequence as no padding was applied to the sides. The shown pooling layer has both a stride of 2 and a pooling width of 2.

the function is equivariant if a permutation of the inputs leads to the same permutation of the outputs, but no change in the values of the outputs. A neural network f^θ is called equivariant if equivariance holds for all θ. Notably, the concatenation $f \circ g$ of two equivariant functions is again equivariant. Therefore, if all layers of a neural network are equivariant, the network is equivariant as well. If the group action of G on Y is trivial, i.e., $g \cdot y = y$ for all $g \in G$ and $y \in Y$, the function is called *invariant* to G. A simple example of a permutation invariant function would be the mean of a set of real numbers.

We note that often it is sufficient to find approximate symmetries in the data or design network architectures that are nearly equivariant to gain the advantages regarding overparametrisation. Such only approximate symmetries emerge commonly because symmetry transformations become invalid at the boundaries of the data, for example, at the beginning and the end of a time series.

2.5.5 Convolutional neural networks

A symmetry commonly observed in data is translation equivariance. For example, in image processing, a key task is to discover certain structures, like edges or corners. Similarly, in seismology, certain local waveform characteristics need to be identified. These features are translation invariant, i.e., they have the same characteristics irrespective of the position within the image or time series. An architecture incorporating this translation equivariance are *convolutional neural networks* (CNNs) [Goodfellow et al., 2016, Chapter 9]. In this thesis, we use CNNs for feature extraction from waveforms (Chapters 4 to 6).

The main components of CNNs are convolutional layers, performing a convolution of the input data with a kernel. As an example, we take the 1D case.[13] Let $x = (x_0, \ldots, x_{n-1}) \in \mathbb{R}^{n \times c}$ be an input with n samples along the time axis and c channels. A convolutional kernel[14] K with kernel size $2s + 1$ and c' output channels is defined as a matrix of shape $K = (K_{-s}, \ldots K_s) \in \mathbb{R}^{(2s+1) \times c' \times c}$. The convolution $K \star x$ can be written

[13] 1D here refers to the spatial/temporal axis, i.e., the axis along which translation equivariance is given. We do not make any assumptions about the additional channel dimension.

[14] For mathematical simplicity and because this is also commonly used in practice, we only discuss kernels with odd kernel size.

as

$$(K \star x)_t = \sum_{i=-s}^{s} K_i x_{t-i}. \tag{2.44}$$

The same convolutional kernel is applied locally at each position of the input trace. This is visualised in Figure 2.15. In contrast to a fully connected layer, a convolutional layer has considerably fewer parameters, because through the convolution operation it shares the parameters between all positions in the input trace. In addition to the convolution itself, a convolutional layer f usually incorporates a bias term $b \in \mathbb{R}^{c'}$ and a pointwise non-linearity $\sigma : \mathbb{R} \to \mathbb{R}$, giving as full formula

$$f(x) = \sigma(K \star x + b). \tag{2.45}$$

We note that the addition of the bias b needs to be interpreted as pointwise addition. As all operations in a convolution are either applied locally or pointwise, the same convolutional layer can be applied to inputs of different lengths. This flexibility regarding the input length is not present for fully connected layers.

Another layer type commonly used in CNNs are pooling layers. Pooling layers aggregate features over multiple samples. This reduces the dimensionality of the data and redundancy commonly observed between neighbouring samples. We again discuss the 1D case with input $x = (x_0, \dots, x_{n-1}) \in \mathbb{R}^{n \times c}$. A pooling layer f with stride s, pooling size p and aggregation function $\rho : \mathbb{R}^p \to \mathbb{R}$ is given as

$$f(x)_t = \rho(x_{t*s}, \dots, x_{t*s+p-1}), \tag{2.46}$$

where the aggregation function ρ is applied separately to each channel. Note that the output of a pooling layer will only be of length n/s. Common pooling functions are either the maximum or the mean function, with the resulting pooling layers called mean or maximum pooling layers. An example pooling with a stride of 2 and pooling size of 2 is shown in Figure 2.15.

Two aspects need to be mentioned regarding the translation equivariance of CNNs: border effects and alias effects. Border effects occur because the trace is not of infinite length. In (2.44) the terms $x_{-s}, \dots, x_{-1}$ and $x_n, \dots, x_{n+s-1}$ occur, which are not defined. To mitigate this, either the output can be truncated to $n - 2s$ samples or sensible values can be chosen for the undefined terms. Both strategies are used in practice, but in both cases, translation equivariance is violated at the trace borders. The second issue, alias effects, arises from the stride s in pooling layers. Given this stride, pooling operations are only translation equivariant (up to border effects) for translations by a multiple of s. Other translations might show alias effects. Anti-alias techniques exist, but so far have not found widespread adoption [Zhang, 2019].

2.5.6 Transformers

As the last example of a NN architecture, we introduce the *transformer*, designed for data with a permutation symmetry [Vaswani et al., 2017]. The specific transformer architecture presented here was originally termed *transformer encoder* [Vaswani et al., 2017], but referring to it simply as *transformer* has quickly become customary [Devlin et al., 2018]. Inputs to a transformer f are $x = (x_0, \dots, x_{n-1}) \in \mathbb{R}^{n \times c}$ and outputs are $y = (y_0, \dots, y_{n-1}) \in \mathbb{R}^{n \times c}$. For a permutation π, it holds that $f(\pi(x)) = \pi(f(x))$, i.e., permuting the input vectors will permute the output vectors in the same way, but will

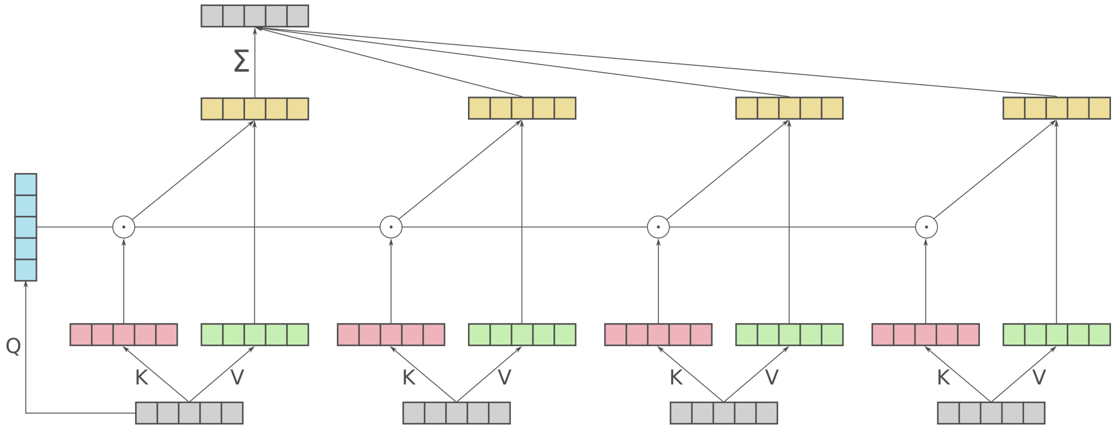


Figure 2.16: Schematic visualisation of the self-attention mechanism calculating the output corresponding to the first input. The inputs are shown at the bottom in grey. Each input is projected to a key (red) and a value (green) vector, using linear transformations K and V. Furthermore, the query (blue) from the first input, computed using a linear transformation Q, is shown. The query is matched with every key using a dot product, yielding a score. Scores are normalised to sum to 1 using a softmax function, yielding the attention weights (not shown). The weight for each input is multiplied with its value, yielding a set of weighted values (yellow). The sum of the weighted values gives the output (grey, on top) corresponding to the first input. While only the computation for the first output is shown, the same procedure is applied to obtain an output for each input. Queries for this are generated from the other input vectors using the same projection Q.

not affect their values in any way. This is particularly useful for inputs without a natural ordering on them, for example, measurements from different instruments. Furthermore, the number of parameters in a transformer is independent of the number of inputs n. Therefore, transformers can be applied to inputs with a variable number of input vectors. We use transformers for combining information across seismic stations in Chapters 4 to 6.

Transformers consist of two main components: *multi-head attention* layers and *point-wise feed-forward* layers. Multi-head attention is a way to combine information from a set of input vectors in a learnable way, i.e., it allows the transformer to identify meaningful connections between the inputs. It is based on the key-value attention mechanism, which we are going to explain first, before introducing its application within the multi-head attention. Key-value attention can be understood as a differentiable lookup table. Given a query vector $q \in \mathbb{R}^c$, a set of key vectors $k = (k_0, \ldots, k_{n-1}) \in \mathbb{R}^{n \times c}$, and a set of value vectors $v = (v_0, \ldots, v_{n-1}) \in \mathbb{R}^{n \times c}$, key-value attention calculates an output vector that represents a combination of the values weighted by the similarity of the keys to the query. This combination is given as:

$$attention(q, k, v) = \sum_{i=0}^{n-1} w_i v \tag{2.47}$$

$$w_i = \frac{e^{s_i}}{\sum_{j=0}^{n-1} e^{s_j}} \tag{2.48}$$

$$s_i = \frac{q \cdot k_i}{\sqrt{c}} \tag{2.49}$$

Here, $q \cdot k_i$ denotes the scalar product of the query and key vector. The weights w_i are called attention weights, calculated from the similarity scores s_i, and describe how much relevance is given to each input.

Based on this attention mechanism, we define the self-attention layer. Given inputs $x = (x_0, \dots, x_{n-1}) \in \mathbb{R}^{n \times c}$, the self-attention layer calculates a set of queries $q = (q_0, \dots, q_{n-1}) \in \mathbb{R}^{n \times c'}$, a set of keys $k = (k_0, \dots, k_{n-1}) \in \mathbb{R}^{n \times c'}$, and a set of values $v = (v_0, \dots, v_{n-1}) \in \mathbb{R}^{n \times c'}$. These are usually computed through linear projections $Q, K, V \in \mathbb{R}^{c' \times c}$, such that $q_i = Qx_i$ and analogously for keys and values. These projection matrices are the learnable parameters of a self-attention layer and define the way the inputs are recombined. The output y_i corresponding to the input x_i of a self-attention layer is given by

$$y_i = attention(q_i, k, v). \tag{2.50}$$

In other words, the self-attention layer calculates its output for each query by calculating the key-value attention between the query and all keys and values. A self-attention layer is visualised in Figure 2.16.

Transformers use a slight extension of self-attention, the multi-head attention. For this, several self-attention layers, called heads, with different projection matrices are applied to the same input and their outputs are concatenated. Another linear projection is applied to the concatenated output. Usually one chooses the output dimension c' and the number of heads h such that $c = c'h$, giving identical input and output dimensions. In contrast to simple self-attention, multi-head attention can jointly attend to information from different subspaces. This allows to model different relationships between the inputs at once.

The second main component of a transformer are pointwise feed-forward layers. These are simple MLPs, usually with one hidden layer and ReLU activation. The MLPs are applied individually to each input vector x_i and are hence called pointwise. The most common feed-forward layer is given by

$$FFN(x_i) = W_2 \; max(0, W_1 x + b_1) + b_2, \tag{2.51}$$

where $W_1 \in \mathbb{R}^{d' \times d}, W_2 \in \mathbb{R}^{d \times d'}$ are the weights and $b_1 \in \mathbb{R}^{d'}, b_2 \in \mathbb{R}^{d}$ are the bias vectors. Usually one chooses $d' \gg d$, for example $d' = 2d$, as this has shown to yield good performance.

A transformer consists of multiple transformer layers. Each transformer layer consists of one multi-head attention, one pointwise feed-forward layer, a set of residual connections and two layer normalisations. A transformer layer is visualised in Figure 2.17. It can be written as:

$$TransformerLayer(x) = LayerNorm_2(x' + FFN(x')) \tag{2.52}$$
$$x' = LayerNorm_1(x + MHA(x)) \tag{2.53}$$

where FFN denotes the feed-forward layer and MHA denotes the multi-head attention. A full transformer consists of several transformer layers that are applied sequentially.

All components of a transformer layer are permutation equivariant, consequently making the full transformer permutation equivariant as well. While this reduces the overparametrisation of the neural network, it disregards potential additional input structure. For example, if $x = (x_0, \dots, x_{n-1})$ is a sequence, i.e, if there exists a natural ordering of the inputs, this information is not accessible to the transformer. To solve this issue, Vaswani et al. [2017] introduced positional encodings. Let $p = (p_0, \dots, p_{n-1}) \in \mathscr{P}$ be metadata associated with the inputs x, called positions. In the sequence example above,

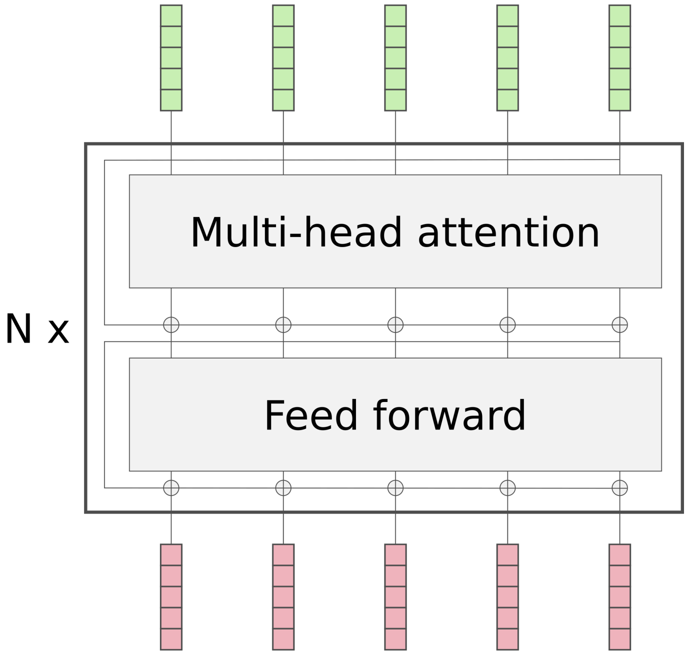


Figure 2.17: Schematic visualisation of a transformer. Inputs are shown in green, outputs in red. The main box shows one transformer layer, consisting of a multi-head attention layer, a feed-forward layer, and residual connections around both of these layers. The residual connections are coupled with a layer norm, indicated by the circles with the pluses.

p is simply $(0, \dots, n-1)$, denoting the position in the sequence. A position encoding is a mapping $Pos : \mathscr{P} \to \mathbb{R}^d$, mapping each position to a vector. These positions are then added elementwise to the input of the transformer. Instead of providing x directly to the transformer, we use x' as input with

$$x'_i = x_i + Pos(p_i). \tag{2.54}$$

This way, the transformer can incorporate the position information associated with the inputs into its computations. Position encodings can either be predefined functions [e.g. Vaswani et al., 2017], or can be parametrised as well and optimised jointly with the remaining neural network weights [e.g. Devlin et al., 2018].

2.5.7 Variational approximations and mixture density networks

An important question for neural networks is how their output is interpreted. This question is tightly coupled to the definition of the loss function. For example, a regression neural network might have as the last layer a fully connected layer with a single output $\hat{y}$. When using the L2 loss for training, there are several options to interpret this output. First, one can interpret it as the mean estimator, i.e., the mean of the posterior distribution $\mathbb{P}(y|x)$, as it can be shown that this mean estimator minimises the L2 loss. Second, one can interpret the output as a Gaussian distribution $\mathscr{N}(\hat{y}, \sigma^2)$ with mean $\hat{y}$ and fixed variance σ^2.[15] The L2 loss is identical to the log-likelihood of this distribution

[15] As shown in Chapter 2.4.1, the particular choice of the value σ only accounts for a constant offset in the loss value. We can therefore assume any σ. It is however crucial that we assume the same σ for each output, as the network does not provide an estimate of σ.

(see Chapter 2.4.1). $\mathcal{N}(\hat{y}, \sigma^2)$ is called a *variational approximation* to $\mathbb{P}(y|x)$, described by its parameter $\hat{y}$ [Blei et al., 2017].

However, in many cases, this model will be overly simplistic. First, the problem might exhibit heteroskedasticity, i.e., the uncertainties σ might depend on the inputs x. Second, the distribution $\mathbb{P}(y|x)$ might not be Gaussian. Therefore, the concept of variational approximation can be generalised. A variational approximation $\mathbb{Q}_{\theta(x)}(y)$ to $\mathbb{P}(y|x)$ is a distribution that is described by its parameters θ. In the example above, $\mathbb{Q}$ was a normal distribution with $\theta = \{\hat{y}\}$.

One option to parametrise $\mathbb{Q}_{\theta(x)}(y)$ are mixture densities, where the density $f_{\mathbb{Q}}$ is given as the weighted average of the densities $f_0, \ldots, f_{n-1}$ of n simple base distributions:

$$f_{\mathbb{Q}_\theta}(y) = \sum_{i=0}^{n-1} \alpha_i f_i^{\theta_i}(y) \tag{2.55}$$

The parameters $\theta = \{\alpha_i, \theta_i \mid i = 0, \ldots, n-1\}$ consist of the mixture weights α_i and the parameters of the base distributions θ_i. To ensure $f_{\mathbb{Q}_\theta}$ is a density function, we require that $\alpha_i \in [0,1]$ for all i and $\sum_{i=0}^{n-1} \alpha_i = 1$. The parameters $\theta(x)$ can then be estimated with a neural network, called a *mixture density network* [Bishop, 1994]. The network can be trained with any proper loss function as introduced in 2.4.1. To this end, the loss value needs to be derived given the true value and the parametrised distribution, i.e., one needs to derive the loss as a function of the distribution's parameters. Throughout this thesis we use closed-form representations of the loss functions but some losses can also be calculated through other means, such as Monte Carlo sampling.

One example of a mixture density network is the Gaussian mixture network. Here the base distributions are Gaussians, parameterised by their means μ_i and standard deviations σ_i. The resulting density is

$$f(y) = \sum_{i=0}^{n-1} \alpha_i \sigma_i^{-1} \varphi(\frac{y - \mu_i}{\sigma_i}), \tag{2.56}$$

where φ is the density of a standard normal distribution. In the limiting case with infinitely many mixture components, Gaussian mixtures have a universal approximation property [Bengio et al., 2017], i.e., under mild smoothness assumptions on the target distribution, any distribution can be approximated arbitrarily well by a Gaussian mixture. A Gaussian mixture network outputs the mixture weights α_i, the means μ_i and the standard deviations σ_i. Note that this parameterisation can be used for both one dimensional and higher dimensional Gaussian distributions. In the latter case, rather than a scalar σ_i, the network needs to output a covariance matrix Σ_i. Care needs to be exercised to ensure positive definiteness of Σ_i. A simple, even though restrictive option, is assuming independence of the different components, i.e., a diagonal matrix Σ_i with positive entries on the diagonal. We use Gaussian mixture density networks to calculate probabilistic estimates in Chapters 4 to 6. We use multi-dimensional Gaussian mixtures in Chapter 5.

2.6 Machine learning in seismology

After presenting the fundamentals of both seismology and machine learning, we now discuss applications of machine learning in seismology, focusing on recent developments. For a general overview, see, for example, the surveys by Bergen et al. [2019] or Kong et al. [2019]. Research on and application of machine learning in seismology has greatly increased in the last years. Consequently, several key contributions were developed concurrently to the work presented in this thesis, which was conducted from 2018 to 2022. In this chapter, we present an overview of key developments, including recent publications. Additionally, we will point out relevant concurrent developments in later chapters where appropriate.

Machine learning approaches for seismic data have been developed since the early 90s, with early works primarily focusing on event detection and phase picking [Dowla et al., 1990, Dysart and Pulli, 1990, Dai and MacBeth, 1995]. Subsequent research tackled further tasks, such as source parameter estimation for earthquake early warning [Böse et al., 2008]. However, none of these approaches was competitive to classical approaches. Driven by the growth in available data, computing capabilities, and the development in data mining and machine learning in the last years, deep learning approaches evolved. Again, first approaches covered event detection and phase picking [Yoon et al., 2015, Mousavi et al., 2019b, Perol et al., 2018, Ross et al., 2018b, Zhu and Beroza, 2019]. Closely related methods were developed for polarity picking [Ross et al., 2018a] and pairwise phase association [Ross et al., 2019]. The detection and picking methods offer a considerable improvement above previous, non-ML approaches, as evidenced by the recently published "deep catalogs" [Tan et al., 2021, Park et al., 2021, Jiang et al., 2022]. These "deep catalogs" are earthquake catalogs with a low magnitude of completeness and high precision locations that were obtained using deep learning methods.

A range of further tasks has been addressed with machine learning recently. Several methods have been published for the assessment of earthquake source parameters, such as magnitude or location [Perol et al., 2018, Lomax et al., 2019, Kriegerowski et al., 2019, Mousavi and Beroza, 2020b,b, van den Ende and Ampuero, 2020, Zhang et al., 2021]. In the context of early warning, neural networks for ground motion estimation were proposed [Jozinović et al., 2020, Otake et al., 2020]. However, none of these methods has yet found widespread application. We will discuss the reasons for this missing adoption in Chapters 4, 5 and 7. Besides the assessment of earthquakes, machine learning methods were presented for seismic signal processing, such as denoising [Zhu et al., 2019], seismic tomography [Earp and Curtis, 2020, Zhao et al., 2022], and the unsupervised exploration of seismic waveform data [Seydoux et al., 2020]. Given their recent development, it is yet unclear how their performance compares to traditional methods.

To foster future development and comparability, several benchmark datasets were compiled, such as STEAD [Mousavi et al., 2019a], INSTANCE [Michelini et al., 2021], or LenDB [Magrini et al., 2020]. However, just as the recent wave of models for diverse seismological tasks, these benchmark datasets have only been published lately. Consequently, there exist no large scale comparison of the performance of different models using such datasets. The only exception known to us is our recent benchmark for seismic picking and detection models [Münchmeyer et al., 2022]. Many publications presenting new models do not compare their results to previous methods. It is therefore practically impossible to identify state-of-the-art methods for most tasks. We provide a more in-depth discussion of standardisation and benchmarking needs and efforts in Chapter 7.1.

Machine learning has not only been applied to natural seismic recordings, but also to laboratory data. As laboratory data is recorded under more controlled environments than real data, applications to this type of data can serve as prototypes for later real-world models. Rouet-Leduc et al. [2017] and Hulbert et al. [2019] showed that machine learning can predict the timing and duration of large stick-slip events in laboratory double direct shear experiments. Through interpretation of their models, they identified key frequency features for the prediction and related them to acoustic emission activity. They transferred their results to a real-world scenario, showing that noise variations in the seismic signal allow predicting geodetic displacement related to aseismic processes in the Cascadia region [Rouet-Leduc et al., 2019]. Corbi et al. [2019] successfully used machine learning on laboratory geodetic data to predict the timing of large failures. Johnson et al. [2021] conducted a challenge for predicting failure in laboratory experiments on the data science competition platform Kaggle, attracting more than 4,500 teams. The challenge was won by a team consisting exclusively of members without an Earth science background. While the results did not provide novel insights into the underlying physical processes, they showcased how proper data engineering can lead to significant improvements in applying models to seismological tasks.

Most of the approaches in the previous paragraphs are so-called black-box approaches. This means, that the models are exclusively derived from the training data, but do not incorporate further knowledge, such as physical constraints. An alternative approach are models incorporating such constraints derived from physics. Smith et al. [2020] developed EikoNet, a neural network solving the eikonal equation that describes the travel times of seismic waves. Once trained, EikoNet provides estimates of seismic travel times at significantly lower computational cost than traditional methods, while achieving competitive accuracy. In addition, the derivatives of the travel time with respect to source and target location can be calculated through the neural network structure. Using this property, Smith et al. [2022] built HypoSVI, a method for seismic event localisation using variational inference. Along similar lines, Gao et al. [2021] used EikoNet to develop a probabilistic, physics informed seismic travel time tomography method. Yang et al. [2021] and Song et al. [2022] presented different approaches for solving the seismic wave equation with deep learning with the goal of developing a fast and exact method for the forward modelling of seismic waveforms. It is yet unclear, how the performance of these approaches compares to traditional methods. We discuss the potential of physics informed networks for real-time assessment further in Chapter 7.3.

In addition to the research on machine learning methods in seismology, these methods are adopted in routine seismological operations. The U.S. Geological Survey National Earthquake Information Center (NEIC) use CNN models in their global seismic monitoring to improve the quality of their phase pick, generate phase labels and estimate distances [Yeck et al., 2021]. The GEOFON group [Quinteros et al., 2021] is currently testing CNN methods for pick refinement.[16] While quantitative evaluations are still outstanding, early results suggest a considerable improvement upon the previously employed classical method. Both the number of picks at a low signal to noise ratio and the temporal precision of the picks are improved using the DL methods. In addition to the monitoring for catalog generation, machine learning systems start being deployed for real-time application. Within the ShakeAlert early warning system on the US West Coast, deep learning methods are used to reduce the number of false positive triggers [Kong, 2021]. In addition, the suitability of the TEAM-LM method (Chapter 5) for use within ShakeAlert is currently evaluated.[17]

[16] Jannes Münchmeyer is contributing to this project.

[17] Jeff McGuire, personal communications, February 2022

3 Post hoc calibration of a high confidence magnitude scale

In the first main chapter, we discuss magnitude calibration in a post hoc scenario, i.e., once all observations about an event are available.[18] More specifically, we will discuss magnitude calibration based on seismic waveforms and a high quality earthquake catalog. Magnitudes are key metrics for characterising earthquakes, required, for example, in statistical seismology or to assess the impact of earthquakes in disaster response. These tasks require high precision of the magnitude scale as well as quantified uncertainties. In this chapter, we calibrate interpretable magnitude scales with low uncertainties. Within this thesis, this post hoc analysis fulfils two goals. First, by quantifying the level of uncertainty on the magnitude values in a post hoc scenario, we get an lower bound on the possible uncertainties in a real-time scenario. The post hoc analysis thereby helps interpreting the performance of the real-time methods. Second, using the method developed in this chapter, we obtain a seismicity catalog with high precision magnitude values. This catalog will serve as a gold standard for the detailed analysis of real-time magnitude estimation methods in Chapter 5.

In this chapter, we focus on magnitudes derived from simple waveform features. The first magnitude of this kind was the local magnitude M_L, as defined by Richter [1935], which was based on the peak horizontal displacement recorded with a particular instrument, the Wood-Anderson seismometer. While many different magnitude scales exist for a multitude of use cases, local magnitude remains popular, in particular, for regional scenarios and small events. An extensive overview of magnitude scales is provided by Bormann [2012]. The local magnitude M_L has the advantage of a simple definition, allowing for fast and robust determination. On the downside, M_L and similar magnitude scales require distance-dependent attenuation correction functions, which need to be calibrated for each region to take the local earth structure into account.[19]

Attenuation functions are typically expressed as non-parametric models [Brillinger and Preisler, 1984]. Savage and Anderson [1995] proposed a simple 1D model with a linear interpolation that can be fit using quadratic optimisation. For this, magnitude values and attenuation functions are jointly calculated using a seismicity catalog and peak displacement measurements at several seismic stations. For the study area used in this Chapter, Northern Chile, Bindi et al. [2014] calibrated a magnitude scale using 106 events from the Iquique sequence 2014.

While these attenuation terms are well established, they have several deficiencies, leading to inconsistency and uncertainty in the magnitude scale. First, while modelling attenuation as a function of hypocentral distance is a good approximation for crustal events, for which the scale was originally developed [Richter, 1935], the approximation is less suitable for subduction zones. In these zones, where both crustal and interface seismicity is present, the events experience different attenuation for the same hypocentral distance, given the different travel paths and therefore different anelastic and geometric focusing effects experienced for crustal and deep events. Second, the attenuation terms do not take into account the spatial variation of attenuation within a region. This is equivalent to the ergodicity assumption in ground motion estimation, which has lately been shown to be a limiting factor to the model performance [Kotha et al., 2016]. Third,

[18] This chapter has been published as [Münchmeyer et al., 2020]. Compared to the publication, the Introduction and Conclusion of this chapter have been modified to highlight the context of the chapter within this thesis. Minor modifications were introduced to the remaining text and figures.

[19] Within this chapter, when using the term *attenuation*, we refer to the combined effects of (physical) attenuation and geometric spreading. Both will usually reduce the recorded amplitudes with the distance travelled by the waves.

M_L is based on a single feature, the peak displacement on the horizontal components. This feature is noisy, i.e., it includes a stochastic component, leading to uncertainties in the magnitude estimate. To reduce the uncertainties from these limitations, usually measurements from multiple seismic stations are averaged. However, the uncertainties of the average still depend linearly on the single-station uncertainties, i.e., any reduction in the single-station uncertainties will translate into lower uncertainties of the network average.

In this chapter, we develop a novel method for magnitude scale calibration that addresses the issues discussed above. For this, we focus on reducing the uncertainties of the single-station magnitude estimates, leading to lower uncertainties for the magnitude values averaged across seismic stations. We develop a three-step approach. In a first step, we define 110 physically motivated features that can be easily derived from the single station waveform. For selecting these features, we took inspiration from features proposed in the context of early assessment [Zollo et al., 2006, Festa et al., 2008, Lancieri and Zollo, 2008, Picozzi et al., 2018, Spallarossa et al., 2019].[20] In a second step, we model the attenuation using a 2D grid function, together with a station-specific, adaptive 3D source correction function to account for the complex subduction zone geometry. The 2D attenuation accounts for the depth dependence of attenuation, the 3D correction for spatial variations. Finally, we add a third step where we combine the single station features using boosting tree regression to obtain more precise magnitude estimates. By combining multiple features, we address the uncertainty from the noisy single features. We apply our method to the IPOC catalog for Northern Chile by Sippl et al. [2018], featuring a high number of earthquakes and high quality location estimates. As a result, we get high-confidence magnitude values with quantified uncertainties for the approach.

Our approach in this chapter is multi-stepped based on hand-crafted, physics-inspired features. This stands in contrast to end-to-end machine learning, where machine learning is used to directly model the relation of the waveforms to the magnitude. For the post hoc analysis, this has several advantages. First, the resulting scales are more interpretable, as the different steps can be analysed individually. This includes analysis of the correction functions, comparison of the scales to each other and interpretation of the key features for the boosting tree regression. Second, true magnitudes are not known or even necessarily well defined, as most magnitude scales, except, e.g., M_W and M_E, are defined through measured features rather than through independent physical source properties. Therefore, our approach uses bootstrapping by first creating high-quality single feature network-average magnitudes using extended correction functions, and then applying boosting tree regression with these magnitudes as labels and the corrected measurements as features. In contrast to this chapter, we will focus on end-to-end methods in the subsequent Chapters 4 to 6, where the multi-step approach becomes infeasible due to the real-time requirements.

3.1 Data and Methods

3.1.1 Earthquake catalog and stations

Our analysis is based on the earthquake catalog of Sippl et al. [2018]. The catalog covers the region of the Northern Chile forearc and contains 101,601 events. The events were extracted from 8 years of continuous seismic data between 2007 and 2014 using automatic

[20] The quantitative results in these studies were obtained in the context of early warning. Consequently, they can not be directly compared to the results of this study. Nonetheless the methods share the idea of choosing appropriate features to minimise single station uncertainties. We will study real-time earthquake assessment in the subsequent Chapters 4 to 6.

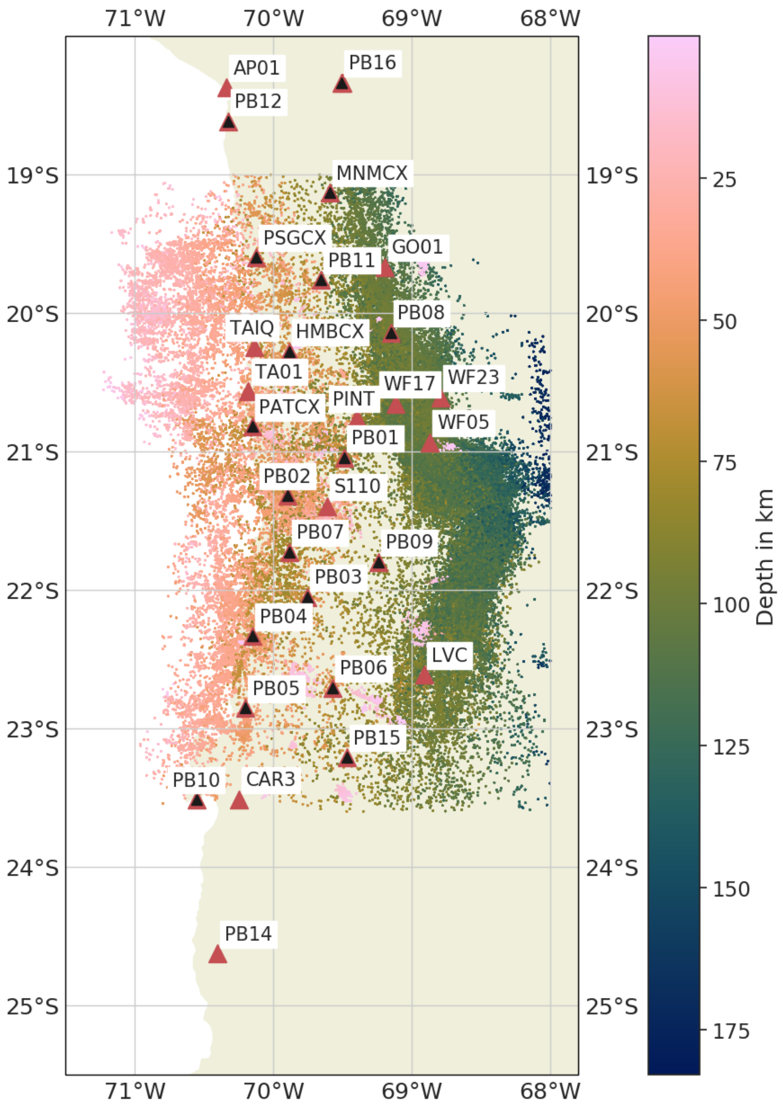


Figure 3.1: Event distribution (from Sippl et al. [2018]) and broadband station locations. Stations with additional strong motion instruments are denoted by a black triangle. The sharp boundaries on the North, East and South side of the study area are due to the event selection criteria in the original catalog.

event detection and phase picking routines. The magnitudes range from < 2 up to 7.7 and the estimated magnitude of completeness for M_L is ~ 2.8 [Sippl et al., 2018]. All event hypocenters were double-difference relocated. The catalog is based on data from the IPOC network [CX, GFZ German Research Centre For Geosciences and Institut Des Sciences De L'Univers-Centre National De La Recherche CNRS-INSU, 2006]. Additional seismic data were obtained from the GEOFON [GE, GEOFON Data Centre, 1993], CSN [C, C1, Universidad de Chile, 2013], WestFissure [8F, Wigger et al., 2016], Iquique [IQ, Cesca et al., 2009] and Minas [5E, Asch et al., 2011] networks. A full map showing the detected events and the stations used can be found in Figure 3.1.

Sippl et al. [2018] use this catalog to analyse the double seismic zone of the Northern Chile forearc. The catalog events are classified into upper plate, plate interface, upper plane and lower plane, based on their location. In addition, the authors identify an

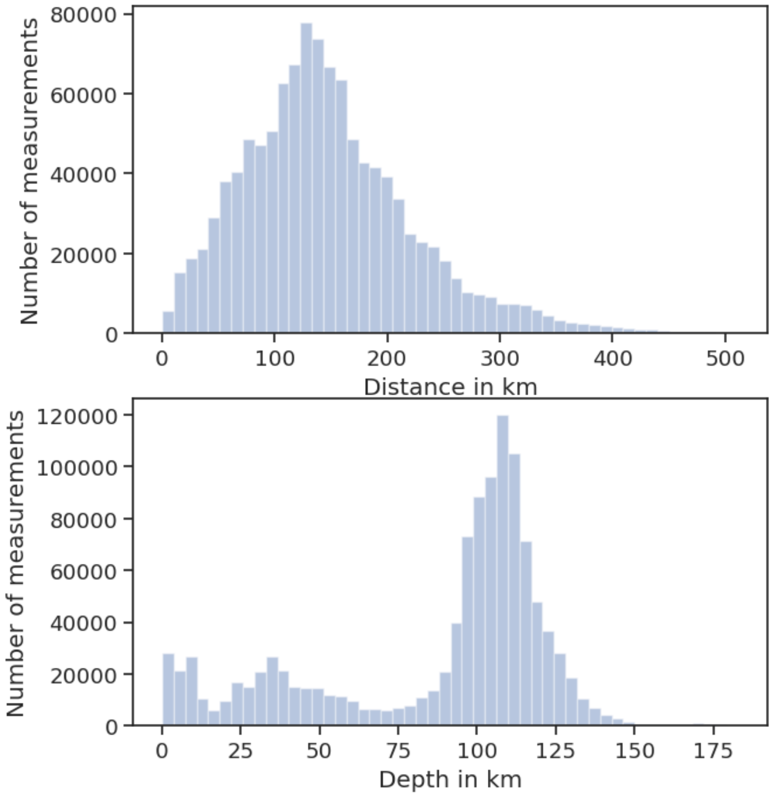


Figure 3.2: Distribution of the measurements over epicentral distance and event depth.

intermediate depth cluster, which is assigned a separate class. The catalog features some events belonging to none of the classes mentioned, mostly events at the border of the study area. We removed these events from our analysis as they are expected to have higher location uncertainties, resulting in a total number of 96,185 events included in this study. For further information on the classification, catalog and study region, we refer to Sippl et al. [2018].

We use the catalog to evaluate our method, as it is both consistent and challenging, while offering a large amount of data. The consistency is achieved by the low temporal variability in the station coverage, a consistent tool chain and double difference relocated hypocenters. This consistency is a prerequisite for the consistent and low uncertainty calibration of magnitude scales. The catalog is challenging, due to the wide range of magnitudes and the different types of seismicity present in the subduction zone.

For our analysis, we use the same seismic stations as Sippl et al. [2018], but also incorporate data from strong motion instruments. A list of all 31 stations can be found in the Table B.4. In total, we use $\sim 1,100,000$ P picks and $\sim 650,000$ S picks from the catalog. We predicted a further 450, 000 S picks using the 1D velocity model of Graeber and Asch [1999].

Figure 3.2 shows the distribution of measurements across distance and depth. Nearly all measurements were taken at distances below 400 km and depths shallower than 150 km, while few additional measurements exist up to 500 km distance and 200 km depth. We observe multiple peaks in the depth distribution, with two smaller peaks around 5 km and 30 km and one large peak around 110 km. These are caused by the different types of seismicity present, namely crustal events and events in the intermediate depth cluster.

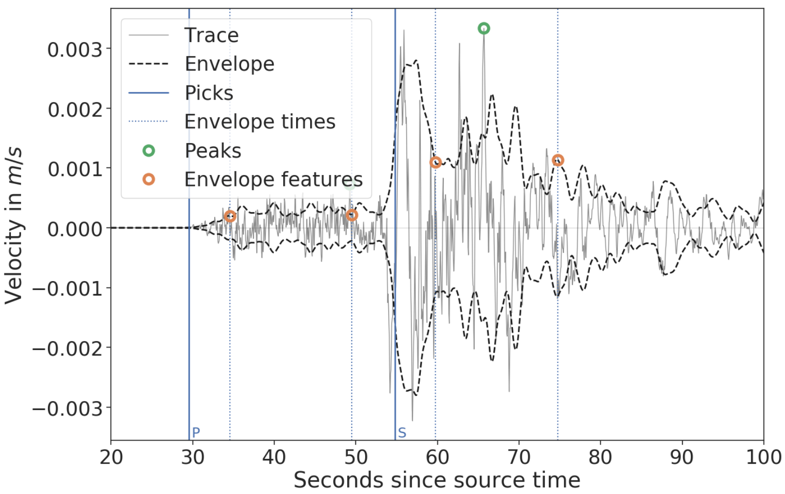


Figure 3.3: Example trace with extracted features denoted by circles. The trace shows the vertical component from station PB01 for a $M_w = 6.3$ event at a depth of 21km and an epicentral distance of 193km.

For further details on the catalog and seismicity in Northern Chile, we refer to the original publication of the catalog by Sippl et al. [2018].

3.1.2 Feature extraction

The feature extraction process encompasses some common preprocessing steps and the actual feature generation. A schematic overview of the workflow for each waveform is shown in Figure B.5. For each event we generate the features for all stations, for which the catalog contained at least one phase pick.

We generally use broadband records. Only for clipped traces, we use strong motion records instead. We assume a clipped trace if its peak value exceeds 80% of the maximum output of the digitiser, as estimated from its bit count [Cauzzi et al., 2016]. If no strong motion data is available, the record is discarded. We also discard traces with gaps.

We remove the instrument response using the inventories provided by GEOFON. We apply a cosine taper in the frequency domain with corner frequency parameters 0.005 Hz, 0.01 Hz, 30 Hz, 35 Hz before the deconvolution. Whenever strong motion data is used, the data is integrated to obtain velocity traces.

As the recorded signal is often below the noise level in the broadband records, we high-pass filter the data to increase the signal-to-noise ratio (SNR), while retaining as much of the low frequency information as is possible. To this end we use a greedy strategy: for a set of frequency intervals $f_{\text{low}}, f_{\text{high}}$, we check whether the mean spectral amplitude increases at least by a factor of 4 between the 30 s before the P pick and the 30 s after. The lowest frequency interval from a pre-defined set of candidates (Table B.1) fulfilling these conditions is used. The data is then high-pass filtered with the corner frequency f_{low}. The frequency f_{high} is only used for frequency selection, but is discarded for the filtering of the actual data. This strategy is applied as we observed sufficient SNRs for high frequencies for all events. Therefore we only need to identify the lowest band, where the SNR is still sufficient for the following steps. More details on the applied filtering and the distribution of selected frequencies can be found in Appendix B.1. As a final step of the pre-processing, we detrend the filtered data using the best linear fit in a 300 s window around the event.

The resulting velocity trace is differentiated to obtain the acceleration trace and integrated to obtain the displacement trace for the vertical (Z), radial (R) and transverse component (T). We use the absolute value of all horizontal components (NE) as well as the absolute value of all components (ZNE) as additional traces where we compute the absolute value from the vectorial sum of the single components.

From each trace we export six values, as shown in Figure 3.3. We extract the peak values of the P and S wave. For the P wave peak we use the waveform between the P pick and the S pick minus a safety margin in order to avoid interference from the S waves. As the safety margin we use 5% of the measured or estimated S wave travel time, but always at least 0.5 s. For the S wave peak, we restrict the search window to the first 30 s after the S pick to minimise the possibility of overlapping events.

In addition, we extract values from the P and S wave envelopes. We calculate the signal envelope and low-pass filter it at 0.5 Hz. We then export the values at 5 and 20 s after the P and S picks. We do not report the envelope values for the P wave in case the time difference between the P and S pick is less than the lag time. We include the envelope values as we expect them to be less influenced by the radiation pattern and distance uncertainties. We chose delays of 5 and 20 s because the 5 s envelope value should be representative of the energy in the direct arrival for moderately sized events but less variable than the peak, while the 20 s value represents a compromise between accessing, for most event-station pairs, the late coda where the wavefield is fully diffusive but still retaining signal levels well above the noise level for practically all events. For further details on the choice of envelope times we refer to appendix B.2.

In addition to the features from the displacement, velocity and acceleration traces, we export the energy, and the peak value of a simulated Wood-Anderson instrument. We calculate the energy as the integral of the squared velocity trace. We export both the integral over the time between P and S pick and the integral over the first 30 s after the S pick. For the Wood-Anderson instrument, we report the peak values from the P and S waves as before. All resulting feature values are logarithmised with base 10.

We rescale the resulting energy features by a factor of 2/3. Following the analysis by Deichmann [2018b], the factor of 2/3 is the theoretically derived scaling factor between M_L and $\log E$. The different scaling of energy E compared to the displacement scale is further discussed in Section 3.2.

In total we extract 110 features, 22 from each component or combination thereof. Of the 22 features half are from the P wave and half from the complete waveform. The features are energy and the simulated Wood-Anderson peak as well as the peak, 5 s envelope and 20 s envelope values from displacement, velocity and acceleration (see Table 3.1).

In our dataset, features might be incomplete due to missing waveform data for single components or because the P envelope values are later than the S arrival. All features are present in at least 98.8% of the measurements. The only exceptions are the 5 s P wave envelope value with only 97.7% availability and the P envelope value at 20 s with only 21.4%. This lower availability is expected, as the value can only be measured at a significant distance to the event.

3.1.3 Correction terms and normalisation

To correct the measurements for the source-receiver distance, station bias and source conditions, we employ a set of non-parametric correction functions. The classical approach of Richter [1935] uses a table of hypocentral distance correction values. We extend this method by using a non-parametric 2D correction function incorporating source-receiver distance and source depth, as well as by adding a station correction and a station-specific source correction term. The latter will be mostly affected by propagation effects related to three-dimensional heterogeneity, but could theoretically also incorporate radiation pattern effects if certain mechanisms are dominant in some area.

Let E be the set of events and S be the set of stations. Let $E_s \subseteq E$ be the subset of events measured at station $s \in S$. For station $s \in S$ and event $e \in E_s$ we model the difference between the measured feature Y_s^e and the corresponding event magnitude M^e through an attenuation function Γ, a station specific source correction term L_s, and a station correction B_s. With an error term ε_s^e, we obtain

$$Y_s^e - M^e = \Gamma(r_s^e, d^e) + L_s(p^e) + B_s + \varepsilon_s^e \tag{3.1}$$

where r_s^e is the epicentral distance between event and station, d^e the event depth, and p^e the hypocenter. We formulate a quadratic minimisation problem on the squared error objective function

$$Obj_\varepsilon = \frac{1}{n} \sum_{s \in S} \sum_{e \in E_s} (\varepsilon_s^e)^2 \ . \tag{3.2}$$

Here, n denotes the number of error terms or equivalently the number of measurements for the feature. We now describe the definitions of the different correction terms, as well as their normalisation and regularisation. This will also lead to an extension of the objective function for the quadratic optimisation.

The attenuation function Γ is defined as a two dimensional non-parametric function on a grid of epicentral distances and depth values. We use a grid G with 50 linearly spaced distance values between 20 and 500 km and 20 linearly spaced depth values between 10 and 200 km. Values between the grid points are interpolated bilinearly between the four adjacent values.

We enforce smoothing of the attenuation function by introducing a penalty term derived from the 2nd order finite difference approximation of the Laplacian with a regularisation term

$$R_\Gamma = \frac{1}{|G|} \sum_{(r,d) \in G} \lambda_r \left(\frac{\partial^2 \Gamma}{\partial r^2} \right)^2 + \lambda_d \left(\frac{\partial^2 \Gamma}{\partial d^2} \right)^2 \ . \tag{3.3}$$

For clarity reasons we write the continuous version of the Laplacian here, rather than its finite difference approximation. The factors λ_r and λ_d are model hyperparameters describing the level of smoothing. We use $|G|$ to denote the cardinality of the set G, i.e. the number of grid points.

To account for source location specific systematic errors, we introduce a source specific correction function L_s for each station s. We randomly sample a set of events $\bar{E}_s \subset E_s$ and assign to each of the events $e \in \bar{E}_s$ a correction term l_s^e. The correction for a single event is defined through the correction terms of the k nearest neighbours:

$$L_s(e) = \frac{1}{k} \sum_{e' \in \mathrm{kNN}(e, \bar{E}_s)} l_s^{e'} \tag{3.4}$$

Here $\mathrm{kNN}(e, \bar{E}_s)$ is the set of the k nearest neighbours of e in $\bar{E}_s$. For our experiments we chose $k = 10$ and $\bar{E}_s$ such that $|\bar{E}_s|/|E_s| \approx 0.1$. As distance metric for the determination of the nearest neighbours we use the euclidean distance between the hypocenters, but scale the depth difference with a factor of 3, to account for the high importance of the depth.

We use the average over the set of neighbours to obtain a smoothly varying function of position. As the density of events is not uniform over the region, the nearest neighbour based function can represent higher variability in regions with many events, while being smoother in regions where a high resolution function would not be well constrained. The subsampling $\bar{E}_s$ from E_s is necessary for performance reasons, as each element in $\bar{E}_s$ introduces an additional free parameter. We choose one subset $\bar{E}_s \subseteq E_s$ for each feature and station.

The location correction is normed and regularised by:

$$\begin{aligned} R_L =& \lambda_L \frac{1}{|S|} \sum_{s \in S} \frac{1}{|\bar{E}_s|} \sum_{e \in \bar{E}_s} l_s^{e2} && (3.5) \\ \forall s \in S :& \sum_{e \in E_s} L_s(e) = 0 && (3.6) \end{aligned}$$

The factor λ_L is a hyperparameter to adjust the level of regularisation.

For each station s, we add a station bias B_s to account for site effects. We constrain the biases of all stations to sum up to zero:

$$\sum_{s \in S} B_s = 0 \tag{3.7}$$

The magnitude scale needs to be anchored, i.e., aligned to a reference, as the system would otherwise be underdetermined. Specifically attenuation with depth can not be extracted from the data, as the depth is only event but not station specific as opposed to the distance. The Richter definition resolves attenuation with depth by using hypocentral distance. Due to the separation of depth and distance in our approach, the standard Richter definition of assigning magnitude 3.0 to a 1 mm displacement at a distance of 100 km is not applicable. Therefore we calibrate our scale against M_w, which also includes information on the attenuation in depth direction.

We obtain a total of 155 M_w values from the Global CMT Project [Dziewonski et al., 1981, Ekström et al., 2012]. As we do not expect a linear scaling between M_w and our magnitude scales for the full range of magnitudes covered by Global CMT, we only used the 114 events with magnitudes between 5.0 and 6.0 in the calibration. To incorporate the information into our model, let E_{M_w} denote the events for which a moment tensor solution is available. We then define an objective by:

$$Obj_{M_w} = \lambda_{M_w} \frac{1}{|E_{M_w}|} \sum_{e \in E_{M_w}} (M^e - M_w^e)^2 \tag{3.8}$$

The factor λ_{M_w} controls the trade-off between fitting to M_w and smoothness of the correction functions. For our analysis we use $\lambda_{M_w} = 0.1$.

We use a weak connection between M^e and M_w^e instead of setting $M^e = M_w^e$ for multiple reasons. First, we do not expect the features to correspond 1:1 with M_w as they also depend on parameters other than the seismic moment, e.g. the stress drop, which influences the high frequency content in particular. We investigate this scaling in more detail in section 3.2.1. Second, we only have values for M_w for a small subset of the dataset available. In conclusion, enforcing equality to M_w would introduce perturbations into the correction functions. The weak connection resolves the underdetermination of our system, while minimising the perturbation on the correction functions.

All correction functions and bias terms are optimised concurrently using quadratic optimisation on the full objective:

$$\min(Obj_{\varepsilon} + Obj_{M_w} + R_{\Gamma} + R_L) \tag{3.9}$$

It consists of the primary objective, the calibration against M_w and the regularisation terms for Γ and L. It is additionally constrained by the relations (3.6) and (3.7). The free parameters are the event magnitudes M^e, the values of the grid G, the values of the correction function Γ, the correction terms $\{l_s^e\}_{s\in S, e\in \bar{E}_s}$, and the station biases $\{B_s\}_{s\in S}$.

While the source-path correction term could in principle incorporate the whole attenuation function, we still decided to split the attenuation into the distance-depth, the source-path and the station term for multiple reasons. First, the source-path term is station-specific, while the distance-depth term is universal for all stations. This enables a by far better calibration of the attenuation with distance and depth, especially for stations and ranges with only few measurements. Second, we can formulate a sensible regularisation more easily: the distance-depth correction is forced to be smooth, whereas the source-path correction is damped towards zero to ensure deviations from the generic distance-depth correction are only introduced where required by the data. Thus, the correction functions are easier to interpret, as the station-specific and the mean attenuation effects are separated. For details on the interpretation see Section 3.3.3.

3.1.4 Multi-feature magnitude estimation

The methods proposed so far only use each feature separately, but do not leverage combinations of features. As a framework for combining multiple features and obtaining a joint magnitude estimate, we state a regression problem: given all features of a *single station* estimate a chosen target *network* magnitude. We use the term *network* magnitude to refer to the average across all *single station* magnitude estimates. We want to emphasise that the key point is estimating *network-wide* information from *single-station* features. The target magnitude scale can be chosen arbitrarily among the scales derived using the calibration functions, for example, the network magnitude from peak displacement on the horizontal components. Due to the limited amount of data, we do not use M_w as a target magnitude.

The regression problem has a canonical baseline: the station magnitude estimated from the feature corresponding to the target magnitude. If for example, the target magnitude is the peak displacement magnitude averaged over all stations, the baseline prediction from a single station would be its peak displacement magnitude estimate. The error level of this baseline is exactly the error from the modelling obtained in the previous step. The task of the regression problem is to estimate network-wide information from the combined features of a single station.

We use boosting trees [Friedman, 2002] for regression, training one common model for all stations. Boosting trees are a special class of gradient boosting models and use decision trees as the underlying classifiers. They are a popular regression technique for non-linear problems. We use a non-linear approach to model complex dependencies between the features. We show by quantitative comparison to linear regression that those complex dependencies are indeed present.

Boosting trees are better suited for our problem than other non-linear approaches like support vector machines or neural networks. Support vector machines suffer from long training times for our problem size and are therefore intractable. Neural networks are harder to train in the presence of missing values, as they represent smooth functions.

We tried multiple imputation techniques to alleviate this problem but were not able to achieve the performance level of boosting trees using neural networks. Boosting trees can handle this problem by learning a default action for splits at missing data points (for details see Chen and Guestrin [2016], algorithm 3).

An additional upside of boosting trees is their interpretability regarding feature importance. We can analyse the information gain through splits at specific features to get a view of their internal workings. This is in strong contrast to neural networks, where such an interpretation is not easily possible.

As boosting trees rely on decision trees, their value range is discrete. While this poses a theoretical limitation, the number of values inside the range is high enough that the discretisation is barely observable. The residuals in the regression predictions are still by far higher than those added by the discretisation. This effect is only causing higher approximation errors for events with high magnitudes, as their number in the training set is limited.

3.1.5 Evaluation

We split our data into a training, a development, and a test set with the ratios 60:10:30. All measurements for one event are guaranteed to be in the same split. The sets contain $\sim 670,000$, $\sim 110,000$ and $\sim 330,000$ measurements and $\sim 58,000$, $\sim 9,600$ and $\sim 29,000$ events. We split randomly between the events, but keep the splits fixed for all evaluation steps and across all features. All models are trained only on the training set. This includes the correction functions as well as the boosting tree for feature combination. We use the development set for hyperparameter selection and report the scores on the test set. An overview of hyperparameter values can be found in Tables B.2 and B.3. We discuss the choice of hyperparameters and advise on the adaptation to other datasets in appendix B.2.

To evaluate the uncertainty of our models we are using the root mean square error (RMSE) between station magnitude and event magnitude. Using the definitions from Section 3.1.3 we define the RMSE as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{s\in S}\sum_{e\in E}{\varepsilon_s^e}^2} \tag{3.10}$$

To compare the uncertainty of different scales with each other we need to normalise the scales. This is necessary to ensure a fair comparison, as the different scales show different slopes. We normalise by dividing the RMSE by the difference between the 25^{th} and 75^{th} percentile of the predicted magnitudes. We rescale all magnitudes by multiplying by the 25^{th} and 75^{th} interquantile distance of the Wood-Anderson magnitude on the horizontal components. Thereby we obtain RMSE values that approximately resemble local magnitude units. We chose these quantile values as all scales show a relatively linear dependency with each other between those values. We only use scaling to compare the scales with each other. For our experiments on the combination of multiple features, we use the plain RMSE, as we only compare the uncertainty between scales with the same value range.

For the multi-feature regression, we always report the RMSE between the predictions from the single station and the network magnitude from the target feature. We do not optimise for the mean of all multi-feature predictions, as this would trivially converge against a constant.

Table 3.1: Normalised RMSE (in local magnitude units) for all analysed features and components on the test set. The second column indicates whether the features are extracted from the full wave (P+S) or the P wave (P) only. The third column indicates if peak or envelope values are used. The best combination of peak or envelope, wave and component for each feature class is highlighted in bold. The columns denote the components, where NE is the absolute value of all horizontal components and ZNE is the absolute value of all components. The rightmost column indicates the average normalisation factor applied. The norm factor does not vary significantly between different components of the same feature and is therefore only given as average across the components. We note that measurements for the 20 s envelope value on the P wave are only possible for the ∼ 30% of the event-station pairs with sufficiently large distances to achieve at least 20 s separation between P- and S arrivals. They are thus skewed towards larger magnitudes.

			Z	R	T	NE	ZNE	∅ Norm factor
Displacement	Full	Peak	0.191	0.195	0.195	0.194	**0.190**	1.01
		Env 5 s	0.241	0.243	0.243	0.232	0.225	1.01
		Env 20 s	0.279	0.266	0.264	0.248	0.241	1.28
	P	Peak	0.270	0.285	0.302	0.297	0.290	1.39
		Env 5 s	0.322	0.328	0.347	0.330	0.320	1.47
		Env 20 s	0.263	0.259	0.295	0.254	0.252	1.48
Velocity	Full	Peak	0.147	0.164	0.162	0.163	0.155	0.94
		Env 5 s	0.172	0.199	0.201	0.195	0.184	0.91
		Env 20 s	0.132	0.138	0.138	0.129	**0.120**	1.02
	P	Peak	0.183	0.191	0.194	0.194	0.191	1.03
		Env 5 s	0.141	0.162	0.168	0.158	0.144	1.06
		Env 20 s	0.143	0.149	0.155	0.144	0.138	1.13
Acceleration	Full	Peak	0.160	0.171	0.170	0.172	0.165	0.98
		Env 5 s	0.169	0.193	0.196	0.190	0.179	0.94
		Env 20 s	0.128	0.132	0.133	0.125	**0.117**	1.03
	P	Peak	0.181	0.187	0.187	0.189	0.187	1.01
		Env 5 s	0.137	0.146	0.150	0.142	0.132	1.01
		Env 20 s	0.119	0.124	0.125	0.120	0.116	1.02
Wood-Anderson	Full	Peak	0.195	0.195	0.197	0.193	**0.188**	1.02
	P	Peak	0.292	0.308	0.332	0.320	0.301	1.58
Energy	Full		0.124	0.134	0.134	0.132	**0.122**	0.75
	P		0.144	0.160	0.165	0.160	0.147	0.81

Our feature extraction is based on Obspy [Beyreuther et al., 2010]. The extraction is parallelised event-wise and conducted on a compute cluster. As no dependencies between events exist, parallelisation can easily be scaled to clusters of arbitrary size. To optimise our models, we used the Gurobi optimiser [Gurobi Optimization LLC, 2018] using a free academic license. Optimization took ∼ 3 hours per model using 64 threads on four Intel Xeon E7-4870 CPUs and required ∼ 120 GB of main memory. All boosting tree experiments were conducted using XGBoost [Chen and Guestrin, 2016] on four Intel Xeon E7-4870 CPUs. Each training process took less than 30 minutes.

3.2 Results

We report the average RMSE for all extracted features and components in Table 3.1. Differences in RMSE depend more on the feature than on the component. Nonetheless, we see differences between the components as well. For the peaks of the P wave, the average normalised RMSE over all feature classes (i.e., displacement, velocity and acceleration) is 0.216 on the Z component and 0.239 on the T component. In between are the ZNE, R and NE components (in this order). This matches the characteristics of the P wave as a longitudinal wave, which is expected to have smaller amplitudes on the transverse than on the vertical and radial components. The effect is also observable for the envelope values, although the combinations of components tend to perform similarly or even better in this case.

For the peak amplitude measurements of the full waveform, all components achieve nearly identical RMSE values. For the envelope values, the differences are more pronounced, especially for the 20 s envelope values. The best scoring component at 20 s is ZNE with 0.162 normalised RMSE and the worst is Z with 0.183 normalised RMSE. We suspect that taking the absolute value of all components effectively reduces noise and thereby improves envelope performance.

We see major differences regarding the normalised RMSE between the different feature classes. For the peaks of the full trace, the lowest average RMSE across all components occurs for velocity (0.163), followed by acceleration (0.172), Wood-Anderson (0.197) and displacement (0.198). Energy (0.134) achieves a better score than all peak values.

Envelope values behave differently for different features. For displacement, the envelope derived scales show considerably higher RMSE on most components. In contrast, the best 20 s velocity and acceleration envelope values have a 23% and 29% lower RMSE than the respective best peak scales. Scales derived from features on the P wave show a higher normalised RMSE in all cases. The increase is up to 69% for the Wood-Anderson instrument compared to the full wave.

The lowest normalised RMSE values overall are the 20 s envelope values of acceleration and velocity on the ZNE component (0.120 and 0.117). The best peak derived feature is velocity on the Z component with 0.147. The best combination of peak or envelope, wave, and component for each feature class is highlighted in Table 3.1.

3.2.1 Relations between the scales

We now compare the scales obtained from the peak values of the ZNE components for the different feature classes and energy (Fig. 3.4). As a reference scale we use peak displacement, as this scale shows no saturation effects. We denote the scale by M_A as proposed by Deichmann [2018a]. The scatter visible in the plot reflects both systematic effects of earthquake physics and the uncertainties of both M_A and the other scale under consideration.

As all scales are tied to M_w between 5.0 and 6.0, they match between those values. They deviate outside this range. The Wood-Anderson based magnitude scales 1:1 with the displacement magnitude for magnitudes below 6.0 and slowly saturates above. Unsurprisingly, it shows the lowest variance in comparison with the displacement scale. The velocity magnitude scales 1:1 with displacement for small magnitudes and increases more slowly for $M_A > 5.0$. The acceleration shows a similar behaviour as the velocity, but nearly completely saturates for $M_A > 6.0$. The saturation effects for velocity and acceleration have previously been observed and are due to the shifted frequency spectra [e.g., Katsumata, 2001]. Interestingly, the variance of the acceleration magnitude is highest among the scales, suggesting a high variability of peak acceleration compared to peak displacement.

The energy magnitude grows slightly stronger than M_A below 4.0 and scales nearly

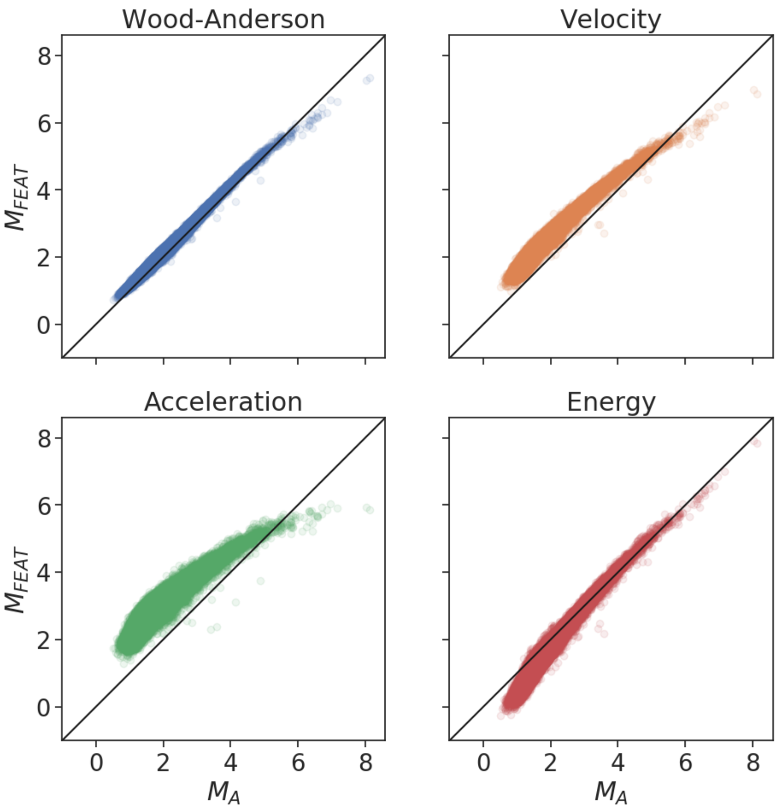


Figure 3.4: Comparison of the different magnitude scales, all relative to the scale based on displacement (M_A). All scales use the ZNE component of the full waveform. Wood-Anderson, velocity and acceleration scales use the peak values, the energy scale uses the integrated square velocity. The identity line has been added in black for comparison.

1:1 above, exhibiting scatter similar to the velocity magnitude. Below 2.0, the energy magnitude compared to M_L approximately follows a 4:3 scaling. Combined with the factor of 2:3 in the definition of our energy features, this scaling provides empirical evidence for the 2:1 scaling between M_E and M_L derived by Deichmann [2018b]. For large magnitudes, this scaling only holds compared to M_L, caused by the Wood-Anderson response, but not for M_A.

We compare the magnitude scales to M_w using 155 moment tensor solutions from the Global CMT project and 507 further solutions we determined using regional moment tensor inversion (see Appendix B.3). Figure 3.5 shows the relation between M_w and the scales generated from different features. Due to the calibration used, all scales match M_w fairly well between 5.0 and 6.0. Strong differences can be seen outside this range, especially for larger events. Saturation effects cause velocity and acceleration magnitudes both to underestimate large events. The saturation effect also causes them to overestimate smaller events, as the saturation already affects the calibration magnitude range M5 to M6. The Wood-Anderson magnitude shows a saturation effect only above M $\sim$ 6.5.

The trends of M_A and energy magnitude match M_W approximately over the whole range of magnitudes. The energy magnitude exhibits more scatter, possibly related to varying source properties, e.g. stress drop, although ambient noise could also affect the measurements. Deichmann [2018a] proposed M_A as a non-saturating alternative to M_L. Our empirical results support this proposal.

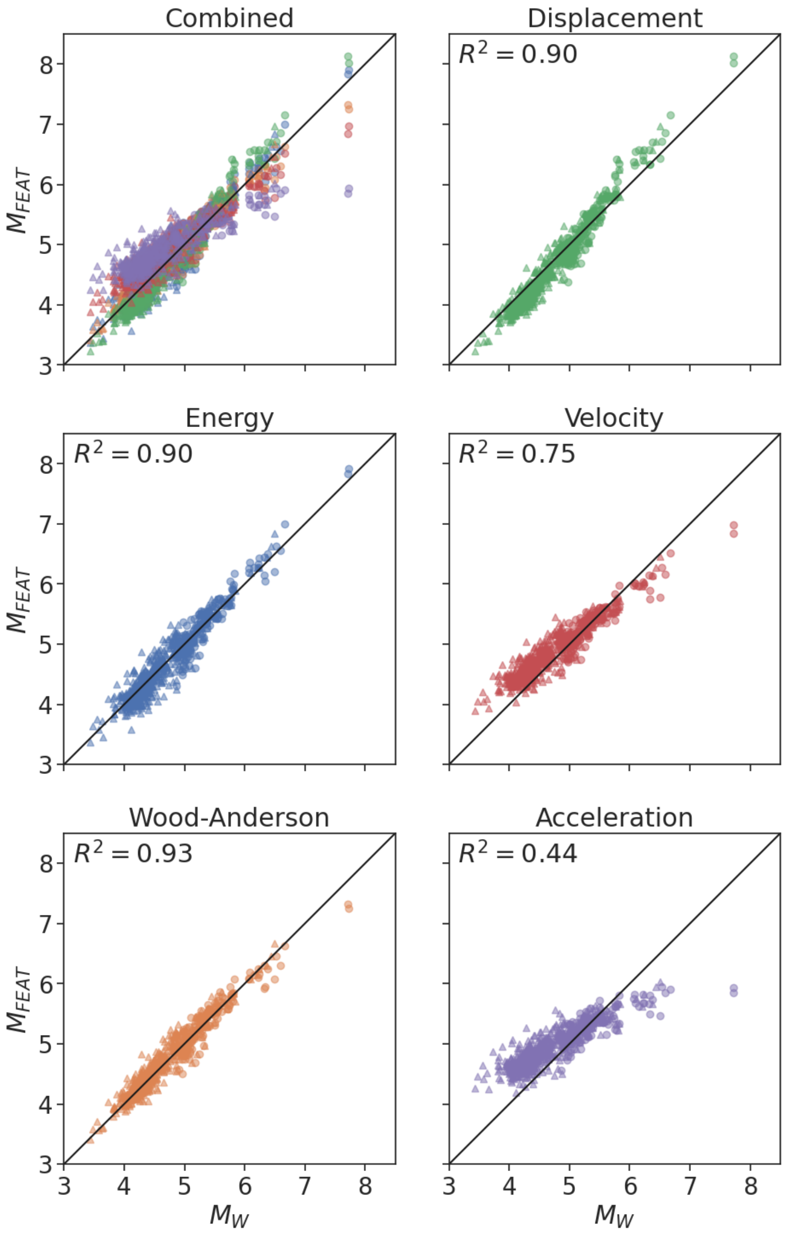


Figure 3.5: Estimated magnitudes from different features in comparison to M_w from Global CMT and additional solutions. Comparisons to Global CMT are shown as circles, comparisons to our moment tensor solutions as triangles. All magnitudes were determined from the peak values of the ZNE component of the full waveform (except energy). The identity line has been added in black for comparison. We report R^2 scores as a further orientation.

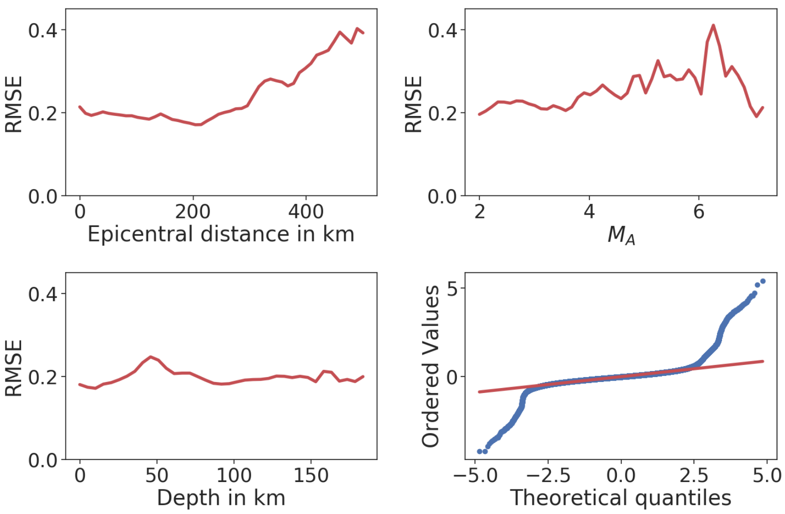


Figure 3.6: Residual analysis for displacement magnitudes on the NE components. Three plots show the dependency of RMSE on depth, M_A and epicentral distance. M_A refers to the peak displacement magnitude on the horizontal components. All traces represent running square means. The averaging window widths are 20 km (epicentral distance), 10 km (depth) and 0.2 m.u. (M_A). The bottom-right plot shows the distribution of the residuals in comparison to a normal distribution as a Q-Q plot.

3.2.2 Residual distribution

We analyse the residuals as a function of depth, M_A, hypo- and epicentral distance (Fig. 3.6). While residuals do not depend strongly on depth, we observe a near doubling of residuals with distance and presumably a weak increase with M_A. The increase with distance can be easily understood, as the SNR decreases with distance.

Varying residuals could also be caused by implicit frequency dependencies of the correction terms. The increased RMSE for larger magnitude values could be caused by the lower dominant frequency of large events compared to the small events composing the majority of the training events. Lower frequency waves might encounter less physical attenuation and scattering, therefore experiencing reduced amplitude decay with larger distances. As site response might be frequency-dependent, we expect a weak distance dependence in the station term. On the other hand, this effect is offset by the source-path correction, as it is station-specific. While frequency effects can not be accounted for by the linear single-feature model, we expect the boosting tree regression to mitigate them, as it has access to spectral information through the combined use of displacement, velocity and acceleration features.

We observe different RMSE values for different types of seismicity. We use the classification from Sippl et al. [2018] to classify events into upper plate, plate interface, upper plane, lower plane and intermediate-depth cluster events. The lowest RMSE for peak displacement on the combined horizontal components occurs for crustal and intermediate-depth events, a 0.02 higher RMSE for lower plane and plate interface events and another 0.01 for upper plane events. Results are similar for other features.

The Q-Q plot in Figure 3.6 shows that the residual distribution deviates from a normal distribution, as it exhibits heavy tails. Those likely indicate measurement errors, caused by overlapping events, instrument issues or wrong frequency selection, causing low SNR. We observe a general positive skewness of outlier residuals, i.e., magnitudes are more

Table 3.2: Test set RMSE for models based on the combinations of multiple features. Separate columns represent different feature sets. A plus sign indicates that further information was added, a minus sign indicates that all features of the respective type were removed. Unadjusted refers to the plain features, on which no correction terms have been applied. Please note that the RSME values are not normalised, therefore comparisons of absolute values are only valid inside each column, but not between the columns.

Features	Displacement NE	Acceleration Z
Single (Baseline)	0.196	0.159
All + Timing	0.103	0.097
All	0.105	0.099
All - Env	0.113	0.108
All - P wave	0.110	0.103
All - Env - P Wave	0.121	0.112
Only Z component	0.111	0.101
Only velocity	0.122	0.115
Unadjusted + Timing	0.162	0.162
Unadjusted	0.177	0.176
Unadjusted P-wave	0.203	0.199

likely to be grossly overestimated than underestimated. This holds for all stations except AP01, LVC, PINT and S100 which exhibit a negative skewness. In the appendix, we give a further analysis of the residuals for each station (Figure B.6), possible time dependency (Figure B.7) and the effect of different SNR thresholds (Appendix B.4).

3.2.3 Multi-feature magnitude estimation

For the experiments with multi-feature magnitude estimation, we use the peak horizontal displacement and the peak vertical acceleration as target scales. We choose horizontal displacement because of its similarity to the standard M_L for smaller magnitudes and no observed saturation effects and we choose vertical acceleration as a challenging benchmark, as it already has a low RMSE.

A joint boosting tree predictor is trained on the multi-feature sets for all stations simultaneously (Table 3.2). We achieve the best results for both target scales using the full feature set with additional features measuring temporal information, i.e. the difference between P and S pick time and the time at which each feature was extracted relative to the P pick. For horizontal displacement, we reduce the RMSE by 47%; for vertical acceleration the reduction is 39%. The smaller improvement for acceleration is likely caused by the already lower RMSE of this feature. To elucidate the effect of different features on prediction quality, we removed certain features from the full feature set. The RMSE still improves significantly compared to the single feature for all tested combinations, although of course, the prediction accuracy decreases somewhat (see the top part of Table 3.2). To evaluate the benefit of combining features from velocity, displacement and acceleration, we conducted an experiment solely on velocity features. The resulting RMSE is 16% higher than for the full feature set. The information gain from including features from the displacement, velocity and acceleration traces can be explained with the different frequency bands effectively covered by the features. While acceleration covers the higher frequency ranges, displacement covers mostly the lower

Table 3.3: RMSE for different subsets of the correction functions. Full refers to the complete correction function, as described in section 3.1.3. Distance-Depth only contains the Γ term and the station corrections, but not the source corrections. Distance, in addition, reduces the Γ function to a 1D function using hypocentral distance.

Corrections	Displacement NE	Acceleration Z
Full	0.196	0.159
Distance-Depth	0.227	0.202
Distance	0.237	0.221

frequencies. Hanks and McGuire [1981] discuss the relations between acceleration, velocity and displacement and argue that their values are affected differently by attenuation and that their peaks are expected to occur at different times in the waveform. This gives a further explanation for the information gains from incorporating all three feature classes.

We additionally trained a boosting tree on the plain features from step one, without applying the correction functions from step two. In addition, we removed all features based on the radial and transverse components, as they can only be obtained if the epicenter location is known. It, therefore, uses no information about the location or time of the event, but only information gained from the single station. We experiment with both, a feature set with and without temporal information. In particular, the S-P arrival time difference represents a strong constraint on the hypocentral distance, which controls the dominant term of the correction function. For horizontal displacement, both reduced feature sets still clearly outperform the (corrected) single feature baseline. The reduction in RMSE is 17% with timing and 10% without timing (Table 3.2, bottom part). For acceleration, the RMSE is nearly identical with timing and 11% higher without. We conclude that, when properly combined, the uncorrected features are already competitive with the single corrected features. As is natural, boosting tree regression on the corrected features outperforms the uncorrected features.

To employ our method in an early warning context, the system needs to deliver its estimate rapidly. Therefore we run an additional experiment using only the uncorrected data from the P wave. This information is available at the time of the S arrival. For the displacement magnitude, the RMSE is only 4% worse than the single feature after applying corrections. For acceleration, the RMSE is 25% higher.

We want to emphasise that the complete feature set can be made available only 30 s after the S arrival. All P wave features are already available at the moment of the S arrival. While this is interesting for fast magnitude estimates, its applicability to early warning is limited. This is caused by the catalog consisting mostly of small, non-hazardous events and the relatively far source station distance of up to 500 km. Applicability to early warning would need to be assessed on an appropriate catalog.

3.3 Discussion

3.3.1 Influence of different correction functions

We conduct an ablation study to quantify the impact of different correction terms on the residuals. We compare the full model to a model without source correction, and a model without source correction and only a 1D hypocentral distance correction. Similarly to Section 3.2.3, we conduct the analysis for horizontal displacement and vertical acceleration. The results are shown in Table 3.3.

Both features incur an improvement from both the 2D correction as well as the source

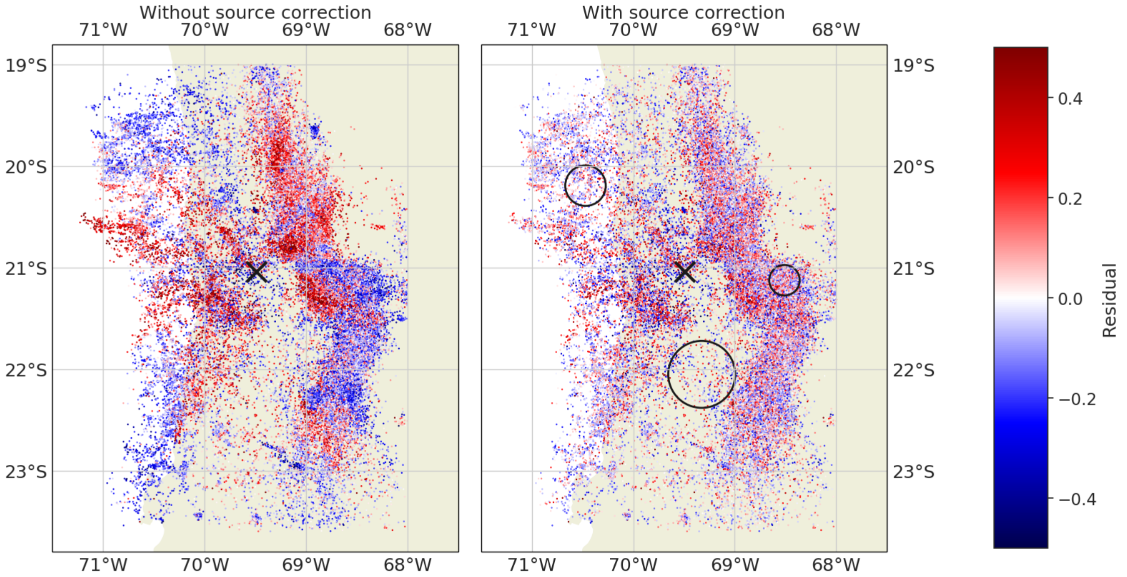


Figure 3.7: Spatial distribution of the residuals at station PB01 for displacement magnitude on the horizontal components with and without source correction. The location of the station is denoted by a cross. The circles indicate the distance to the 10^{th} nearest neighbour with a correction term to visualise the adaptive window size. While the source correction uses the 3D location, we reduced the picture to 2D for simplicity.

correction. The improvement from the 2D source correction is around 4.2% for displacement and 8.6% for acceleration. The effect of the source correction is by far greater, with an additional improvement of 14% for displacement and 21% for acceleration. The combined improvement is 17% for displacement and 28% for acceleration compared to a classic distance-only correction function.

We next analyse the spatial distribution of the residuals with and without source correction. Figure 3.7 shows the residuals for station PB01. Without source correction, they show clear spatial biases, while with source correction there is no bias visible. Without source correction, there are strong azimuthal dependencies. This suggests that the residuals are dominated by path effects, which are similar across a wide distance range with the same azimuth while being less affected by the properties of the physical source such as radiation pattern. While residuals with source correction show no spatial bias, they still appear heteroskedastic, i.e., their variance shows a spatial variation.

Changing the correction terms alters the resulting magnitude calibration. While removing the source correction only has a minor impact, switching from a 2D to a 1D correction introduces a depth-dependent offset between the scales. Unlike distance and source corrections, the depth is an inherent property of the event not improved by averaging. Therefore, the magnitude calibration is performed essentially for each depth level. For depth levels without events in the training set, magnitudes can only be determined by interpolation or extrapolation. This also implies that the calibration of the 2D correction could be more fragile, requiring careful testing of the performance with the test and validation sets (see also the section 3.3.2).

Figure 3.8 compares the distance and depth correction functions obtained with and without the source correction. The correction function without the source correction is significantly rougher than the one with the source correction. This suggests that the depth and distance correction function derived without a source correction term represents a

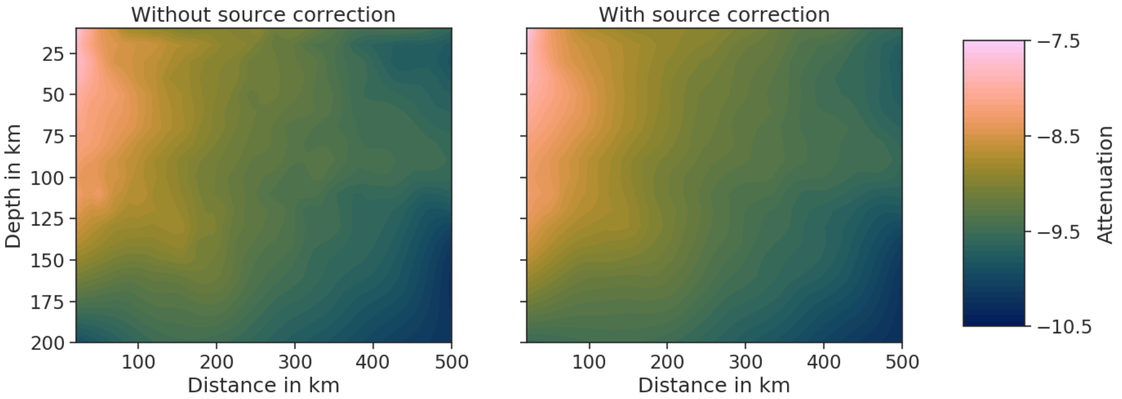


Figure 3.8: Comparison of correction functions with and without source correction terms for peak horizontal displacement.

biased estimate influenced by the particularities of the event distribution. Concurrent optimisation of source correction and distance and depth correction, therefore, does not only yield a good source correction but also improves the smoothness of the distance and depth correction. This suggests that it captures the actual average attenuation in the study region rather than the specifics of the dataset.

3.3.2 Stability of the correction functions

Due to the high number of parameters in our model, it might be susceptible to overfitting. We estimate the level of overfitting in our model by comparing the RMSE on the training and test sets. A high level of overfitting suggests that the model is not well constrained and poses an issue to interpretability. On average the RMSE on the test set is 2.7% larger than on the train set. This increase is fairly consistent across the different features, varying from 1.0% to 3.9%. The only exceptions are 20 s P wave envelope values. Their RMSE on the test set is on average 8.3% higher than on the training set, probably because there are far fewer measurements. To assess the significance of the increases in RMSE, we evaluated the uncertainty of the RMSE values under the assumption that errors are uncorrelated and identically distributed. In this case, the standard deviation of the RMSE is simply the RMSE divided by the square root of the number of samples, which comes out at around 0.1% of the RMSE. Therefore all differences in the RMSE discussed here are significant. While these results show that the source correction functions are slightly underdetermined, the ablation study in section 3.3.1 shows that this does not negatively impact their predictive performance.

We investigate how well the parameters of the correction functions are constrained by the given measurements. To this end, we partition the set of events randomly into ten equal-sized, disjoint subsets and calibrate a model for each of those. Due to the source correction, the changed numbers of events and measurements also change the number of model parameters. To ensure that the model differences are not dominated by changing the subset of events used for calibration with M_W, we always add the events with M_w to the subsets. We analyse models for peak horizontal displacement.

Results show that the station bias terms are robust. For stations with more than $2,000$ measurements in the complete dataset, the standard deviation between the ten sets is below 0.01. For stations with few measurements ($< 2,000$), we observe standard deviations up to 0.036. We emphasise that these deviations apply between the sliced sets containing less than 200 measurements each for these stations. On the full set, this implies that we expect the uncertainty of the station biases to be below 0.01 for all stations. Biases, uncertainties and number of measurements for each station are shown in Figure B.8.

The distance and depth correction is also very robust, albeit with a higher level of uncertainty than the station biases (see Figure B.9). At distances below 250 km and depths shallower than 100 km the standard deviation is always below 0.05. Higher standard deviations occur at large distances and depths, as data are very sparse there. Standard deviations of more than 0.1 solely occur for distances above 400 km and depths below 175 km. We want to emphasise that uncertainties on the final model are likely to be even smaller by a factor around $\sqrt{10}$, as it has been trained on ten times the data.

To assess the stability of the source correction, we evaluate the standard deviation between the ten subsets for $100,000$ randomly chosen measurements. The mean standard deviation is 0.027. The 90^{th} percentile is 0.039. The parameter uncertainties in the model are clearly below the random effects in the measurements.

To analyse the stability of the boosting tree scales, we split the dataset event-wise into three equal-sized partitions A, B and C. We train one boosting tree on A and another one on B, both using the full feature set including timing. We compare the predictions of the boosting trees on C and also compare them to the non-boosting predictions on C. The target scale is again the horizontal peak displacement of the full wave. The event magnitude, averaged across all stations, differs between the non-boosting predictions and the boosting trees by 0.062 (0.063 for tree B) in quadratic mean. The two boosting tree scales only differ by 0.015 in this measure, even though they are trained on completely disjoint sets. The significantly smaller difference between the boosting scales suggests that the boosting trees are indeed reducing estimation errors on the event magnitude. This does not hold for the largest events (>6.0), as only relatively few of these events occurred in the observational period. We experience higher differences between boosting and non-boosting scales for those events, which are likely caused by sparse training data. Boosting tree scales should therefore not be used for the largest events.

3.3.3 Analysis of the correction functions

To analyse the different correction functions, we first need to emphasise the interconnections between them. Without regularisation, the full distance correction could be incorporated into the source correction function, only requiring an offset to calibrate the correction function. We chose our regularisation constants in a way that penalises putting distance corrections into the source correction function by regularising this function towards zero (see equations (3.1) and (3.5)). Nonetheless, both interact, and separation of the effects is not fully possible. In addition, our stations and events are not uniformly distributed. Therefore, the distance correction function, being a mean across all stations and events, incorporates effects from the average paths, which do not necessarily reflect the average ground structure.

We compare the absolute values of the distance and depth correction functions between the different displacement, velocity and acceleration features. Due to different units, absolute values are not comparable between displacement, velocity and acceleration. Absolute differences in the correction function represent differences in the magnitude of the signal. For peaks from the full wave, the signal level is similar for the R and T components, but around 0.15 orders of magnitude smaller on the Z component. For the P wave the signals are strongest on the Z and R component and about 0.1 orders of magnitude smaller on

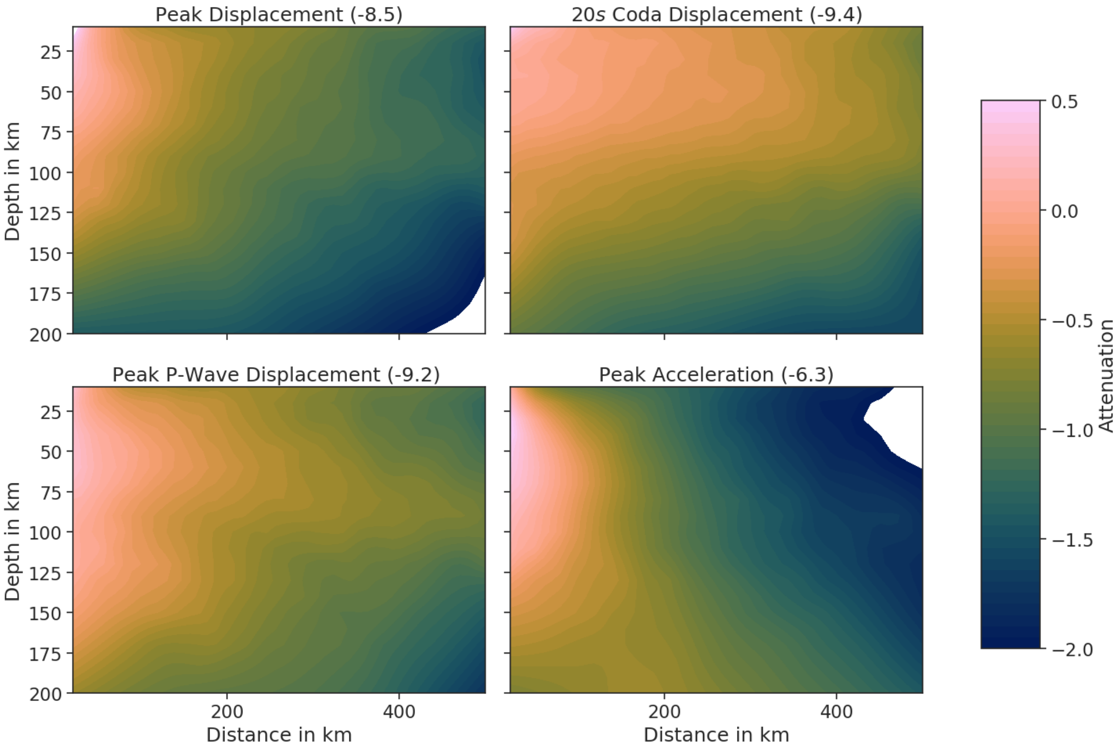


Figure 3.9: Distance and depth correction functions for selected features on the Z component. All corrections are shifted to 0 at a distance of 50km and a depth of 30km, to better visualise the relative differences. The shifts are denoted in brackets after the title. White areas indicated attenuation values below the minimum value in the colour scale. They only occur in regions where the attenuation is poorly constrained.

the T component. The envelope derived values are on average 0.4 orders of magnitude smaller after 5 s and 0.5 after 20 s.

Figure 3.9 shows the comparison of four selected normalised correction functions. We focus on different features rather than components, as we observed no major differences between the different components. Comparing the peak displacement of the complete waveform with the peak P wave displacement, we see that the peak displacement shows a stronger attenuation with both distance and depth. The peak acceleration shows the strongest decay with distance while being only weakly dependent on depth at near offsets. For far offsets (> 250 km), deeper events are less attenuated than shallower events (opposite the pattern for displacement). This effect could arise from the dominant importance of physical attenuation over geometric spreading.

In contrast, the 20 s envelope displacement amplitudes only show a relatively weak dependence on distance. This lower attenuation stems probably from the fact that the envelope is made of scattered waves; the theory of coda normalisation predicts that energy will be distributed equally through all space and degrees of freedom after an asymptotically long time after the event [Sato et al., 2012]. The remaining decay with distance stems most likely from the fairly short time window of 20 s used. This window is necessary to account for the many small events in the catalog, for which the envelope values tend to fall below the noise quickly.

Figure 3.10 shows sections through the source correction terms of the stations WF05 and WF23 at 20.8°S. The two stations are both located approximately 150 km from the

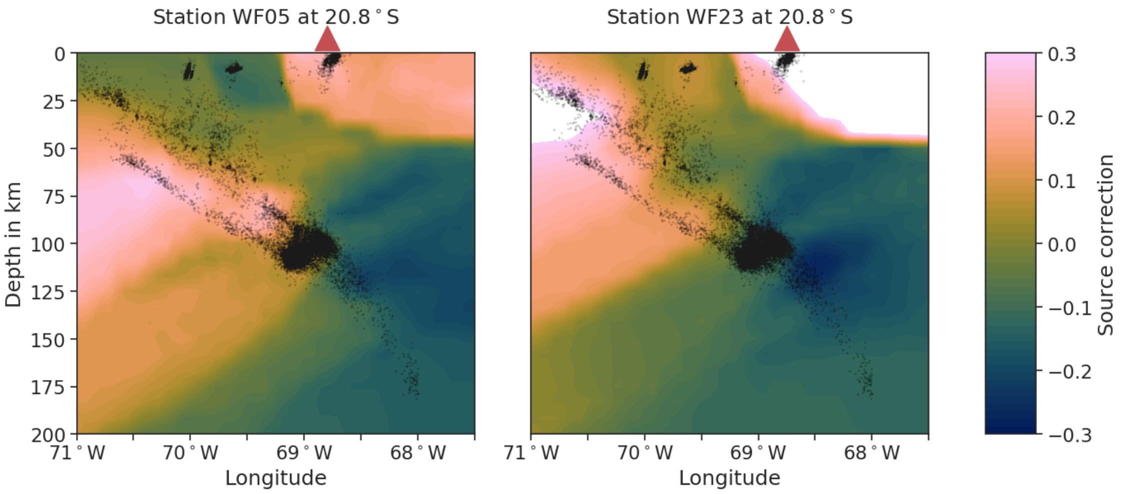


Figure 3.10: Sections through the source correction terms for peak horizontal displacement of the stations WF05 and WF23 at 20.8°S, the latitude in the middle of the two stations. The positions of the stations are marked by red triangles. For orientation, events inside 20.8 ± 0.2°S are shown by black dots. Note that in areas without any seismicity, the correction term will be effectively controlled by the nearest seismicity, even if that is far from the point under consideration. While the source-specific correction term in those areas is likely to be biased, this normally does not matter, because hardly any seismicity occurs in these poorly constrained areas in any case. Out of scale values are plotted in white instead of clipped to avoid suggesting constant correction values in these areas.

coastline, with a distance of only about 40 km between each other. The source correction terms for the two stations are quite similar, which is consistent with the explanation that the source correction terms indeed capture large-scale path effects. The source correction terms also exhibit tectonic features. The most prominent is the sudden change for shallow earthquakes around 69°W. In addition, the source corrections resolve the slab, which can be seen as a diagonal boundary in the corrections, approximately matching the slab determined by Sippl et al. [2018].

By comparing sections at different latitudes, we observed that the resolution of structural features becomes worse for sections further away from the station. As the source correction measures both source and path effects, for large distances it is dominated by aggregated path effects. Therefore the resolution of structural features gets worse. In contrast, the similarity between the corrections for nearby stations stays similar, as the paths get even more similar. We inspected several sections through the source correction volumes of different stations. All sections showed the clear change for shallow earthquakes around 69°W and an imprint of the slab geometry. In general, the sections for stations located close to each other were mostly very similar.

3.3.4 Insights into the multi-feature estimation

As boosting trees use decision trees as their base classifiers, they inherently lead to a ranking of features regarding their feature importance. Feature importance is derived from the information gain of the splits using this feature. We analyse the feature importance for the two target scales used in section 3.2.3 (Table 3.4).

While we are not able to state the reason for the importance of these features with certainty, we provide some intuition. P wave features on the vertical and radial compo-

Table 3.4: Top 10 features in the boosting regression ordered by importance. The columns denote whether the feature is from peak (no annotation) or envelope, whether the feature is from the P wave, the trace it was exported from, and the component. NE refers to the horizontal components, ZNE to the combination of all components. We abbreviate displacement (DISP), velocity (VEL) and acceleration (ACC).

DISP NE				ACC Z			
	P	VEL	Z		P	VEL	Z
	P	VEL	R		P	VEL	R
5 s	P	DISP	Z	5 s	P	DISP	Z
	P	VEL	T		P	VEL	T
5 s	P	ACC	R	5 s	P	ACC	R
5 s		ACC	T		P	ACC	Z
5 s	P	DISP	R	5 s		ACC	T
	P	ACC	Z	5 s	P	DISP	R
5 s	P	DISP	T	5 s	P	DISP	T
		ACC	Z			VEL	Z

nents are least affected by local site conditions. As shown before, the vertical component features from P waves have the lowest RMSE values among the P wave components. The envelope values are less affected by the radiation pattern as well as uncertainties in the location or correction functions. Both P wave features and envelope values have worse signal-to-noise ratios than features from the full waveform, making them less precise scales using only single features, while the combination of those features can likely be used to better separate signal from noise. We attribute the dominance of velocity features to two factors. In contrast to acceleration features, velocity features show less saturation, as discussed earlier. In addition, as our underlying data are velocity traces, velocity is not affected by artefacts from integration that occur for displacement.

To verify the presence of complex interactions, we compare the boosting tree to a simple linear regression. We use the full parameter set without timing information and the same target scales. Similar to boosting trees, linear feature combination significantly reduces RMSE. For displacement, the RMSE is 0.133 (0.103 for the boosting tree) and for acceleration, it is 0.120 (0.097). This highlights that, although parts of the gain can be achieved with linear regression, a significant part of the improvement from the boosting tree is due to its capability to model non-linear relationships and complex interactions between multiple parameters.

3.3.5 Magnitudes for the IPOC catalog

Following our analysis, we provide well-calibrated magnitude values for the IPOC catalog. For each event, we provide magnitude estimates from both the Wood-Anderson instrument and the peak displacement on the horizontal components. The former is chosen for its close resemblance to the standard local magnitude M_L, while the second offers a non-saturating alternative, which we refer to as M_A as proposed by Deichmann [2018a].

We additionally report uncertainty values for the magnitude estimates. We derive those uncertainties from the residuals between the stations. The detailed procedure for uncertainty estimation is described in Appendix B.5.

We apply multiple steps to further increase the quality of the published scales. After calibrating and applying the correction functions, we remove all outliers. Outliers are defined as measurements with a residual of at least twice the global RMSE. We recalibrate the correction functions on the set without outliers. Due to overfitting, we can not use a global boosting tree for the full dataset. We therefore randomly split the dataset event-wise into three equal-sized sets A, B and C. We train one boosting tree on each pair of these sets and use it to produce predictions on the last set. The analysis in section 3.3.2 suggests that these predictions are consistent between the different boosting trees. This is especially the case, as, contrary to section 3.3.2, the training sets of the trees are not disjoint. Following the results from section 3.3.2, we use the non-boosting estimates for events with magnitude > 6.0. For events with magnitude < 5.5, we use the boosting tree scales. We interpolate linearly for events of magnitude between 5.5 and 6.0 to obtain continuously defined scales.

3.4 Conclusion

In this chapter, we developed a method to calibrate high confidence magnitude scales using mathematical optimisation and machine learning. Our method consists of three steps: feature extraction, physically-motivated attenuation functions, and machine learning for the combination of different waveform features. We showed that our method reduces uncertainties on the magnitude values by up to $\sim 57\%$, of which $\sim 23\%$ can be attributed to the improved attenuation functions and the remainder to the usage of multiple waveform features. In conclusion of our analysis, we provide calibrated magnitude values M_A and peak Wood-Anderson based magnitude values, similar to standard M_L but with a richer calibration function, and their estimated uncertainties for the catalog of Sippl et al. [2018].

For our method, we did not explicitly consider frequency dependencies, but rather investigated effects on a broad frequency band. We pursued this approach to capture the wide magnitude range present in the catalog. We acknowledge that attenuation functions are frequency-dependent, as shown for example by Dawood and Rodriguez-Marek [2013] for the Japan subduction zone. This possibly is the cause of the increased RMSE values for larger magnitudes in our estimates, which will be based on longer period data less affected by physical attenuation. Incorporating frequency dependency into the model could also open up a perspective for applying the model to ground motion prediction.

While we applied the method to a catalog of $\sim 100,000$ events, our analysis in Section 3.3.2 suggests that our method could also be applied to significantly smaller datasets. All correction terms are already well defined with 10,000 events and we expect the boosting tree to work as well. For catalogs with more measurements per event, we even expect a far lower number of events required.

The results from multi-feature estimation, especially the results from the experiments with uncorrected features, give a hint at the wealth of information contained in a single trace. This information is of major interest for reliable magnitude estimation in the context of early warning. However, the method developed in this chapter requires high-quality location estimates and full waveform recordings and is, therefore, not applicable to early warning. In contrast, convolutional neural networks (CNNs) might be a promising tool for real-time assessment. CNNs have recently been shown to be beneficial for several seismological tasks, including earthquake localisation [Kriegerowski et al., 2019], phase picking and polarity determination [Ross et al., 2018a], or magnitude estimation [Lomax et al., 2019]. We will focus on CNN models for earthquake assessment in the next chapters, focusing on early warning in Chapter 4, on real-time magnitude and location estimation in Chapter 5, and on rupture predictability in Chapter 6.

Resource availability

As a result of this study, we publish the catalog with the calibrated magnitude values. In addition, to enable in-depth analysis, we provide the full set of extracted features and magnitude predictions on the station level in CSV format. For convenience, we also provide the calibrated correction functions for each feature in the dataset. The catalog with magnitude values, and the additional data are available at `https://doi.org/10.5880/GFZ.2.4.2019.004`. We provide our code to calibrate correction functions and train boosting tree models at `https://github.com/yetinam/magnitude-calibration`.

4 End-to-end early warning with machine learning

After analysing magnitude estimation in a post hoc scenario, we now turn towards earthquake early warning.[21] Compared to the previous chapter, this changes our scope in three ways. First, instead of a post hoc scenario, we discuss real-time methods, as such methods are required for early warning. Second, instead of estimating magnitude, we directly estimate ground motion from seismic waveforms. This approach is beneficial for early warning, as ground shaking is a proxy for the expected damage at a target, while the magnitude only describes the source. Third, instead of feature-based, parametric approaches, we now develop deep learning approaches. This is necessary, as feature-based approaches are difficult to define for the real-time application and deep learning considerably outperforms these approaches in this scenario.

The concept of earthquake early warning has been around for over a century, but the necessary instrumentation and methodologies have only been developed in the last three decades [Allen et al., 2009, Allen and Melgar, 2019]. Early warning systems aim to raise alerts if shaking levels likely to cause damage are going to occur. Existing methods are split into two main classes: source estimation based and propagation based. The former, like EPIC [Chung et al., 2019] or FINDER [Böse et al., 2018], estimate the source properties of an event, i.e., its location or fault extent and magnitude, and then use a ground motion prediction equation (GMPE) to infer shaking at target sites. They provide long warning times but incur a large apparent aleatoric uncertainty due to simplified assumptions in the source estimation and in the GMPE [Kodera et al., 2018]. Propagation based methods, like PLUM [Kodera et al., 2018], infer the shaking at a given location from measurements at nearby seismic stations. Predictions are more accurate, but warning times are reduced, as warnings require measurements of strong shaking at nearby stations [Meier et al., 2020].

Recently, machine learning methods, particularly deep learning methods, have emerged as a tool for the fast assessment of earthquakes. They led to improvements in various tasks, e.g., estimation of magnitude [Lomax et al., 2019, Mousavi and Beroza, 2020b], location [Kriegerowski et al., 2019, Mousavi and Beroza, 2020a] or peak ground acceleration (PGA) [Jozinović et al., 2020]. Nonetheless, no existing method applies to early warning because they lack real-time capabilities and instead require fixed waveform windows after the P arrival. Furthermore, the existing methods are restricted in terms of their input stations, as they use either a single seismic station as input [Lomax et al., 2019, Mousavi and Beroza, 2020b] or a fixed set of seismic stations, that needs to be defined at training time [Kriegerowski et al., 2019, Jozinović et al., 2020]. While single station approaches miss out on a considerable amount of information obtainable from combining waveforms from different sources, fixed stations approaches can not adapt to changes in the underlying seismic network. However, such changes in the underlying network occur regularly in practical applications, as, for example, for large, dense networks the stations of interest, i.e., the stations closest to an event, will change on a per-event basis. Finally, existing methods systematically underestimate the strongest shaking and the highest magnitudes, as these are rare and therefore underrepresented in the training data (Fig. 6, 8 in Jozinović et al. [2020], Fig. 3, 4 in Mousavi and Beroza [2020b]). However, early warning systems must also be able to provide reliable warnings for earthquakes larger than any

[21] This chapter has been published as [Münchmeyer et al., 2021b]. Compared to the publication, the Introduction and Conclusion of this chapter have been modified to highlight the context of the chapter within this thesis. Furthermore, we moved several figures from the supplementary material into the main text. Minor modifications were introduced to the remaining text and figures.

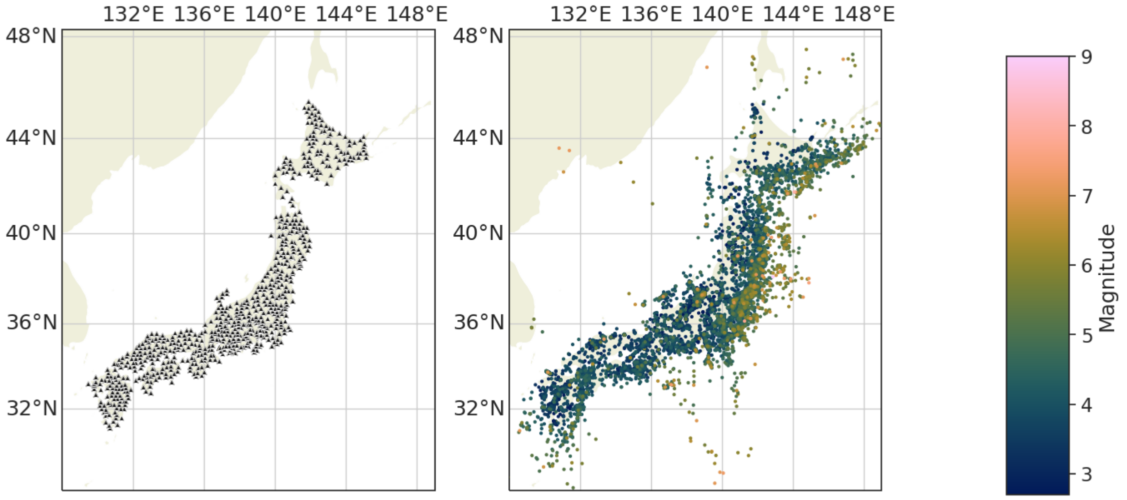


Figure 4.1: Map of the station (left) and event (right) distribution in the Japan dataset. Stations are shown as black triangles, events as dots. The event colour encodes the event magnitude. There are ∼20 additional events far offshore, which are outside the displayed map region in the catalog. The magnitude values are using the M_{JMA} scale.

previously seen in a region.

In this chapter, we present the transformer earthquake alerting model (TEAM), a deep learning method for early warning, combining the advantages of both classical early warning strategies while avoiding the deficiencies of prior deep learning approaches. We evaluate TEAM on two datasets from regions with high seismic hazard, namely Japan and Italy. Due to their complementary seismicity, this allows evaluating the capabilities of TEAM across scenarios. We do not use the Northern Chile catalog from Chapter 3 for the study of TEAM. Due to the small number of very large events and the high distance between seismic stations, the catalog only contains very few instances of strong shaking, and is therefore not applicable to early warning. We will, however, use this catalog for studying real-time magnitude and location estimation in Chapter 5. We compare TEAM to two state-of-the-art warning methods, of which one is prototypical for source based warning and one for propagation based warning. TEAM will furthermore serve as a basis for TEAM-LM, the magnitude and location estimation model introduced in Chapter 5 and applied in Chapter 6.

4.1 Data and Methods

4.1.1 Datasets

For our study, we use two nation-scale datasets from highly seismically active regions with dense seismic networks, namely Japan (13,512 events, years 1997-2018, Figure 4.1) and Italy (7,055 events, years 2008-2019, Figure 4.2). Their seismicity is complementary, with predominantly subduction plate interface or Wadati-Benioff zone events for Japan, many of them offshore, and shallow, crustal events for Italy. We split both datasets into training, development and test sets with ratios of 60:10:30. We employ an event-wise split, i.e., all records for a particular event will be assigned to the same subset. We use the training set for model training, the development set for model selection, and the test set only for the final evaluation. We split the Japan dataset chronologically, yielding the events between August 2013 and December 2018 as test set. For Italy, we test on

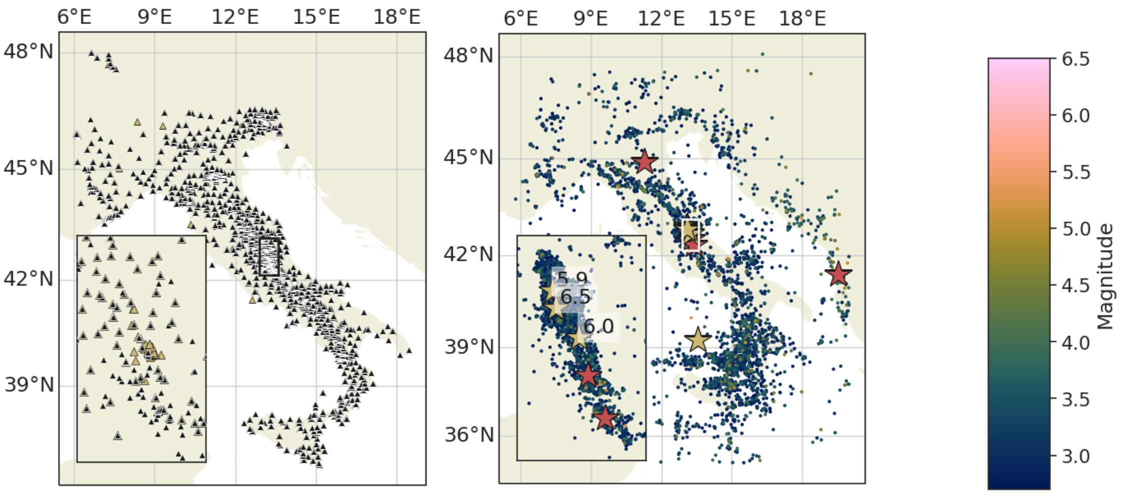


Figure 4.2: Map of the station (left) and event (right) distribution in the Italy dataset. Stations present in the training set are shown as black triangles, while stations only present in the test set are shown as yellow triangles. Events are shown as dots with the colour encoding the event magnitude. All events with magnitudes above 5.5 are shown as stars. The red stars indicate large training events, while the yellow stars indicate large test events. The inset shows the central Italy region with intense seismicity and high station density in the test set. Moment magnitudes for the largest test events are given in the inset. The magnitude values are either M_L ($> 90\%$ of the events), m_B ($< 1\%$) or M_w ($< 10\%$), as provided in the INGV catalog.

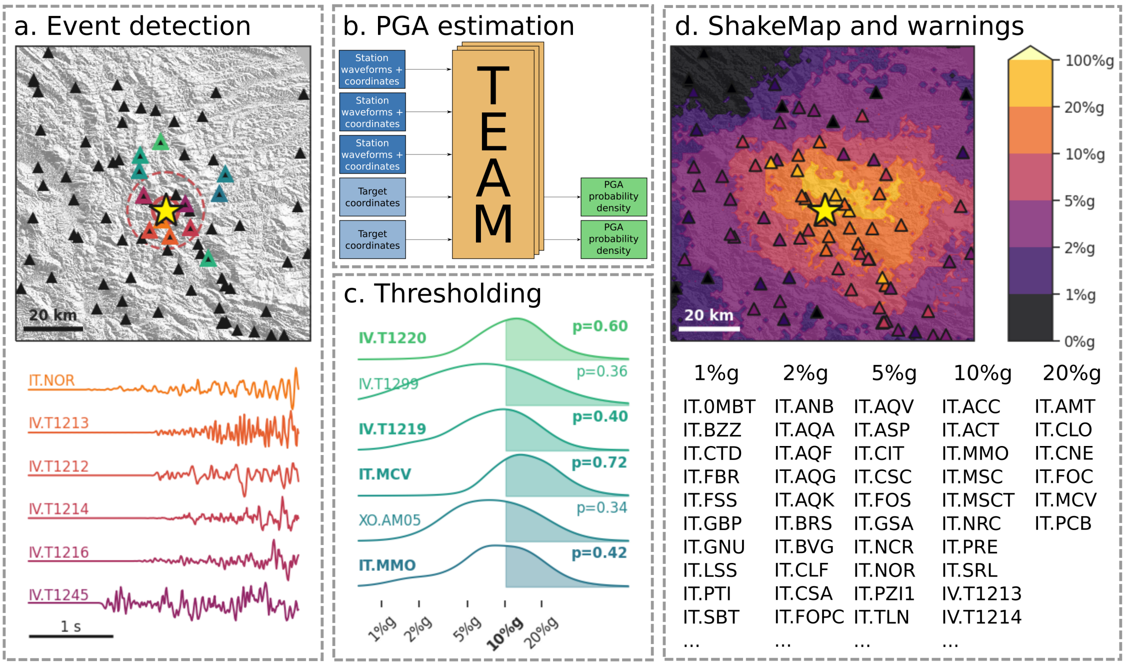
all events in 2016, as these are of particular interest, encompassing most of the Central Italy sequence with the $M_w = 6.2$ and $M_w = 6.5$ Norcia events [Dolce and Di Bucci, 2018]. Especially the latter event is notably larger than any in the training set ($M_w = 6.1$ L'Aquila event in 2007), thereby challenging the extrapolation capabilities of TEAM. We do not explicitly split station-wise but, due to temporary deployments, there are a few stations in the test set which have no records in the training set (Figure 4.2).

Both datasets consist of strong motion waveforms. For Japan, each station comprises two sensors, one at the surface and one borehole sensor, while for Italy only surface recordings are available. As the instrument response in the frequency band of interest is flat, we do not restitute the waveforms but only apply a gain correction. This has the advantage that it can trivially be done in real-time. The data and preprocessing are further described in Appendix C.1.

4.1.2 The transformer earthquake alerting model

The early warning workflow with TEAM encompasses three separate steps (Figure 4.3): event detection, PGA estimation and thresholding. We do not further consider the event detection task here, as it forms the base of all methods discussed and affects them similarly. The PGA estimation, resulting in PGA probability densities for a given set of target locations, is the heart of TEAM and is described in detail below. In the last step, thresholding, TEAM issues warnings for each target location where the predicted exceedance probability p for fixed PGA thresholds surpasses a predefined probability α.

TEAM conducts end-to-end PGA estimation: its inputs are raw waveforms, its output predicted PGA probability densities. There are no intermediate representations in TEAM that warrant an immediate geophysical interpretation. The PGA assessment can



1%g	2%g	5%g	10%g	20%g
IT.0MBT	IT.ANB	IT.AQV	IT.ACC	IT.AMT
IT.BZZ	IT.AQA	IT.ASP	IT.ACT	IT.CLO
IT.CTD	IT.AQF	IT.CIT	IT.MMO	IT.CNE
IT.FBR	IT.AQG	IT.CSC	IT.MSC	IT.FOC
IT.FSS	IT.AQK	IT.FOS	IT.MSCT	IT.MCV
IT.GBP	IT.BRS	IT.GSA	IT.NRC	IT.PCB
IT.GNU	IT.BVG	IT.NCR	IT.PRE	
IT.LSS	IT.CLF	IT.NOR	IT.SRL	
IT.PTI	IT.CSA	IT.PZI1	IV.T1213	
IT.SBT	IT.FOPC	IT.TLN	IV.T1214	
...	...	...	...	

Figure 4.3: Schematic view of TEAM's early warning workflow for the October 2016 Norcia event ($M_w = 6.5$) 2.5 s after the first P wave pick (~3.5 s after origin time). **a.** An event is detected through triggering at multiple seismic stations. The waveform colours correspond to the stations highlighted with orange to magenta outlines. The circles indicate the approximate current position of P (dashed) and S (solid) wavefronts. **b.** TEAM's inputs are raw waveforms and station coordinates; it estimates probability densities for the PGA at a target set. A more detailed TEAM overview is given in Figure 4.4. **c.** The exceedance probabilities for a fixed set of PGA thresholds are calculated based on the estimated PGA probability densities. If the probability exceeds a threshold α, a warning is issued. The figure visualises a 10%g PGA level with $\alpha = 0.4$, resulting in warnings for the stations highlighted. The colours correspond to the stations with green outlines in a. **d.** The real-time shake map shows the highest PGA levels for which a warning is issued. Stations are coloured according to their current warning level. The table lists all stations for which warnings have already been issued.

be subdivided into three components: feature extraction, feature combination, and density estimation (Figure 4.4). Inputs to TEAM are three, respectively six (3 surface, 3 borehole), component waveforms at 100 Hz sampling rate from multiple stations and the corresponding station coordinates. Furthermore, the model is provided with a set of output locations, at which the PGA should be predicted. These can be anywhere within the spatial domain of the model and need not be identical with station locations in the training set.

TEAM extracts features from input waveforms using a convolutional neural network (CNN). The feature extraction is applied separately to each station but is identical for all stations. CNNs are well established for feature extraction from seismic waveforms, as they can recognise complex features independent of their position in the trace. On the other hand, CNN based feature extraction usually requires a fixed input length, inhibiting real-time processing. We allow real-time processing through the alignment of the waveforms and zero-padding: we align all input waveforms in time, i.e., all start at the same time

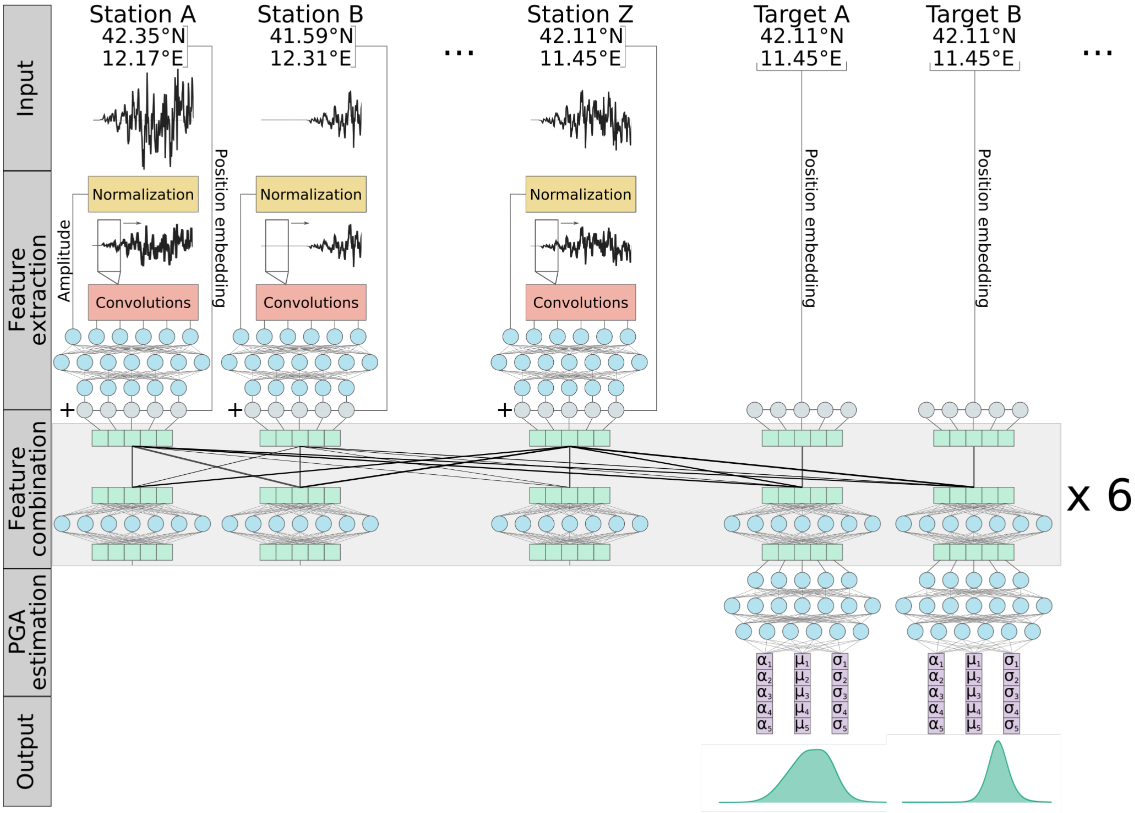


Figure 4.4: Architecture overview of the transformer earthquake alerting model, showing the input, the feature extraction, the feature combination, the PGA estimation and the output. For simplicity, not all layers are shown, but only their order and combination is visualised schematically. For the exact number of layers and the size of each layer please refer to Tables C.5 and C.6. Please note that the number of input stations and the number of targets are both variable, due to the self-attention mechanism in the feature combination. An ensemble of ten instances of this network is trained and the results are averaged in probability space. Training is conducted independently for each ensemble member.

t_0 and end at the same time t_1. We define t_0 to be 5 s before the first P wave arrival at any station, allowing the model to understand the noise characteristics. For t_1 we use the current time, i.e., the amount of available waveforms. We obtain constant length input, by padding all waveforms after t_1 with zeros up to a total length of 30 s. The feature extraction is described in more detail in Appendix C.2.1.

TEAM combines the feature vectors and maps them to representations at the targets using a transformer [Vaswani et al., 2017]. Transformers are attention-based neural networks for combining information from a flexible number of input vectors in a learnable way. To encode the location of the recording stations as well as of the prediction targets, we use sinusoidal vector representations. For input stations, we add these representations component-wise to the feature vectors, for target stations we directly use them as inputs to the transformer. This architecture, processing a varying number of inputs, together with the explicitly encoded locations, allows TEAM to handle dynamically varying sets of stations and targets. The transformer returns one vector for each target representing predictions at this target. Details on the feature combinations can be found in Appendix C.2.2.

From each of the vectors returned by the transformer, TEAM calculates the PGA predictions at one target. Similar to the feature extraction, the PGA prediction network is applied separately to each target but is identical for all targets. TEAM uses mixture density networks [Bishop, 1994] returning Gaussian mixtures, to compute PGA densities. Gaussian mixtures allow TEAM to predict more complex distributions and better capture realistic uncertainties than a point estimate or a single Gaussian. The full specifications for the final PGA estimation are provided in Appendix C.2.3.

TEAM is trained end-to-end using a negative log-likelihood loss. To increase the flexibility of TEAM and allow for real-time processing, we use training data augmentation. We randomly select the stations used as inputs and targets in each training iteration. In addition, again in each training iteration, we randomly replace all waveforms after a time t with zeros, matching the input representation of real-time data, to train TEAM for real-time application. These data augmentations and the complete training procedure are further described in Appendix C.2.4.

To mitigate the systematic underestimation of high PGA values observed in previous machine learning models, TEAM oversamples large events and PGA targets close to the epicenter during training, which reduces the inherent bias in data towards smaller PGAs. When learning from small catalogs or when applied to regions where events substantially larger than all training events can be expected, e.g., because of known locked fault patches or historic records, TEAM additionally can use domain adaptation. To this end, the training procedure is modified to include large events from other regions that are similar to the expected events in the target region. While records from those events will differ in certain aspects, e.g., site responses or the exact propagation patterns, other aspects, e.g., the average extent of strong shaking or the duration of events of a certain size, will mostly be independent of the region in question. The domain adaptation aims to enable the model to transfer the region immanent aspects of large events, at the cost of a certain blurring of the specific regional aspects of the target region. TEAM aims to mitigate the blurring of regional aspects by the choice of the training procedure.

Our Italy dataset is an example of this situation. Accordingly, TEAM applies domain adaptation to this case: It first trains a joint model using data from Japan and from Italy, which is then fine-tuned using the Italy data on its own, except for the addition of a few large, shallow, onshore events from Japan. We chose these events, as for Italy one also expects large, shallow, crustal events due to its tectonic setting and earthquake history. As we use events from Italy in both training steps and in particular in the second step the overwhelming number of events are from Italy, we expect that this scheme only results in a small degradation in the modelling of the regional specifics of the Italy region.

4.1.3 Baseline methods

We compare TEAM to two state-of-the-art early warning methods, one using source estimation and one propagation based. As a source estimation based method, we use an estimated point source approach (EPS), which estimates magnitudes from peak displacement during the P-wave onset [Kuyuk and Allen, 2013] and then applies a GMPE [Cua and Heaton, 2009] to predict the PGA. For simplicity, our implementation assumes knowledge of the final catalog epicentre, which is impossible in real-time, leading to overly optimistic results for EPS. As a propagation based method, we chose an adaptation of PLUM [Kodera et al., 2018], which issues warnings if a station within a radius r of the target exceeds the level of shaking. In contrast to the original PLUM, which operates on the Japanese seismic intensity scale, I_{JMA} [Shabestari and Yamazaki, 2001], our adaptation applies the concept of PLUM to PGA, thereby making it comparable to the other approaches. Whereas I_{JMA} is also a measure of the strongest acceleration and is thus strongly correlated with PGA, it considers a narrower frequency band and imposes a minimum duration of strong shaking. As such, although the performance might vary slightly for our PLUM-like approach compared to the original PLUM, it still exhibits its key features, in particular the effects of the localised warning strategy. Additionally, we apply the GMPE used in EPS to catalog location and magnitude as an approximate upper accuracy bound for point source algorithms (Catalog-GMPE or C-GMPE). C-CMPE is a theoretical bound that can not be realised in real-time. It can be considered as an estimate of the modelling error for point source approaches. A detailed description of the baseline methods can be found in Appendix C.3.

4.2 Results

4.2.1 Alert performance

We compare the alert performance of all methods for PGA thresholds from light (1%g) to very strong (20%g) shaking, regarding *precision*, the fraction of alerts actually exceeding the PGA threshold, and *recall*, the fraction of issued alerts among all cases where the PGA threshold was exceeded [Meier, 2017, Minson et al., 2019]. Precision and recall trade-off against each other depending on the alert threshold α. The PGA predictions of TEAM, EPS and the C-GMPE are probabilistic, with the probability distribution describing the uncertainty of the models, e.g., for the GMPE the apparent aleatoric uncertainty from aspects not accounted for. The thresholding transforms the predictions into alerts or non-alerts. The uncertainty in the prediction means that false and missed alerts are inevitable. The threshold value α controls the trade-off between both types of errors, and its appropriate value will depend on user needs, specifically, the costs associated with false and missed alerts. Therefore, to analyse the performance of the models across different user requirements, we look at the precision-recall curves for different thresholds α. In addition to precision and recall, we use two summary metrics: the *F1 score*, the harmonic mean of precision and recall, and the *AUC*, the area under the precision-recall curve. The evaluation metrics and full setup of the evaluation are defined in detail in Appendix C.4.

TEAM outperforms both EPS and the PLUM-like approach for both datasets and all PGA thresholds, indicated by the precision-recall curves of TEAM lying to the top-right of the baseline curves (Figure 4.5a). For any baseline method configuration, there is a TEAM configuration surpassing it both in precision and in recall. Improvements are larger for Japan, but still substantial for Italy. To compare the performance at fixed α, we selected α values yielding the highest F1 score separately for each PGA threshold and method. Again, TEAM outperforms both baselines on both datasets, irrespective of the PGA level (Figure 4.5b). Performance statistics in numerical form are available in Tables C.1 and C.2.

All methods degrade with increasing PGA levels, particularly for Japan. This degradation is intrinsic to early warning for high thresholds due to the very low prior probability of strong shaking [Meier, 2017, Minson et al., 2019, Meier et al., 2020]. Furthermore, the shortage of training data with high PGA values results in less well-constrained model parameters.

Using domain adaptation techniques, TEAM copes well with the Italy data, even though the largest test event ($M_w = 6.5$) is significantly larger than the largest training event ($M_w = 6.1$), and three further test events have $M_W \geq 5.8$. To assess the impact of this technique, we compared TEAM's results to a model trained without it

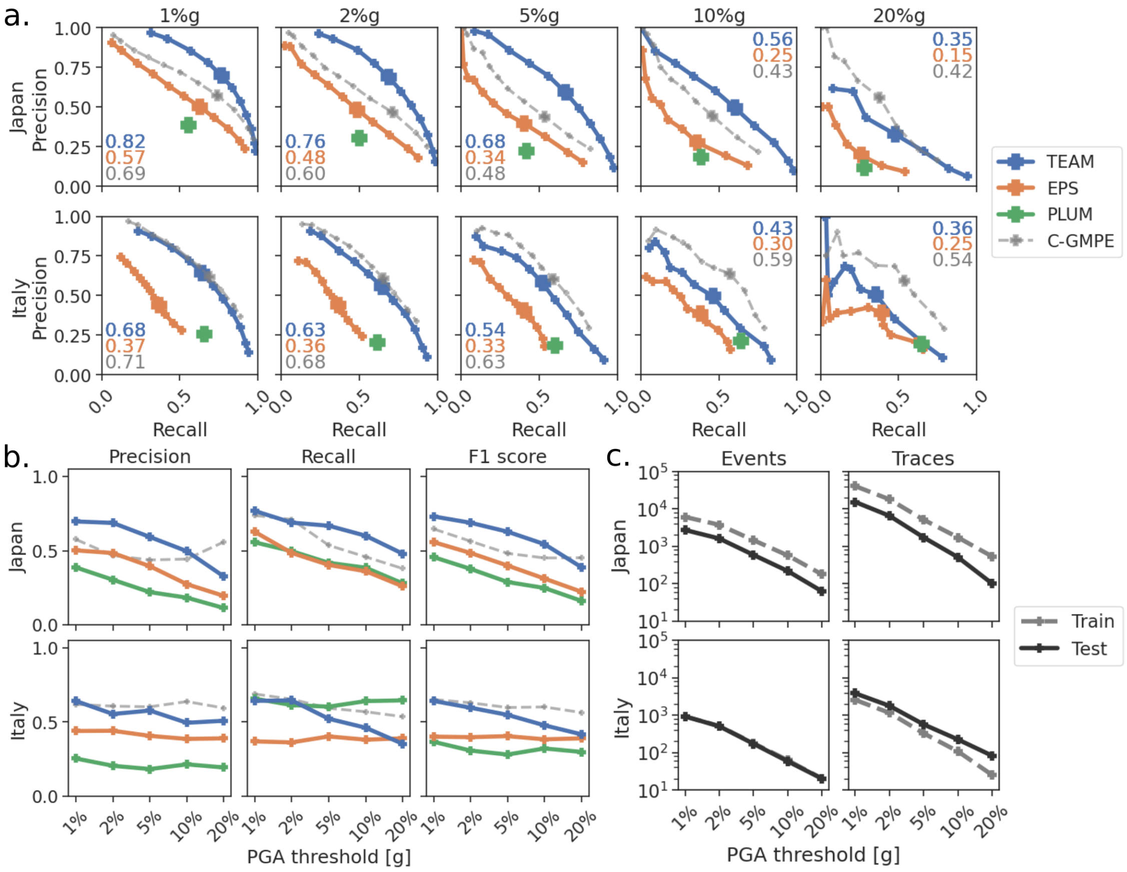


Figure 4.5: Warning statistics for the three early-warning models (TEAM, EPS, PLUM) for the Japan and Italy datasets. In addition, statistics are provided for C-GMPE, which can only be evaluated post-event due to its reliance on catalog magnitude and location. **a.** Precision and recall curves across different thresholds $\alpha = 0.05, 0.1, 0.2, \ldots, 0.8, 0.9, 0.95$. As the PLUM-like approach has no tuning parameter, its performance is shown as a point. Enlarged markers show the configurations yielding the highest F1 scores. Numbers in the corner give the area under the precision-recall curve (AUC), a standard measure quantifying the predictive performance across thresholds. **b.** Precision, recall and F1 score at different PGA thresholds using the F1 optimal value α. Threshold probabilities α were chosen independently for each method and PGA threshold. **c.** Number of events and traces exceeding each PGA threshold for training and test set. Training set numbers include development events and show the numbers before oversampling is applied. For Italy, training and test event curves are overlapping due to similar numbers of events.

(Figures 4.6, C.1). While for low PGA thresholds differences are small, at high PGA levels they grow to more than 20 points F1 score. Interestingly, for large events, TEAM strongly outperforms TEAM without domain adaptation even for low PGA thresholds. This shows that domain adaptation does not only allow the model to predict higher PGA values, but also to accurately assess the region of lighter shaking for large events. Domain adaptation, therefore, helps TEAM to remain accurate even for events far from the training distribution.

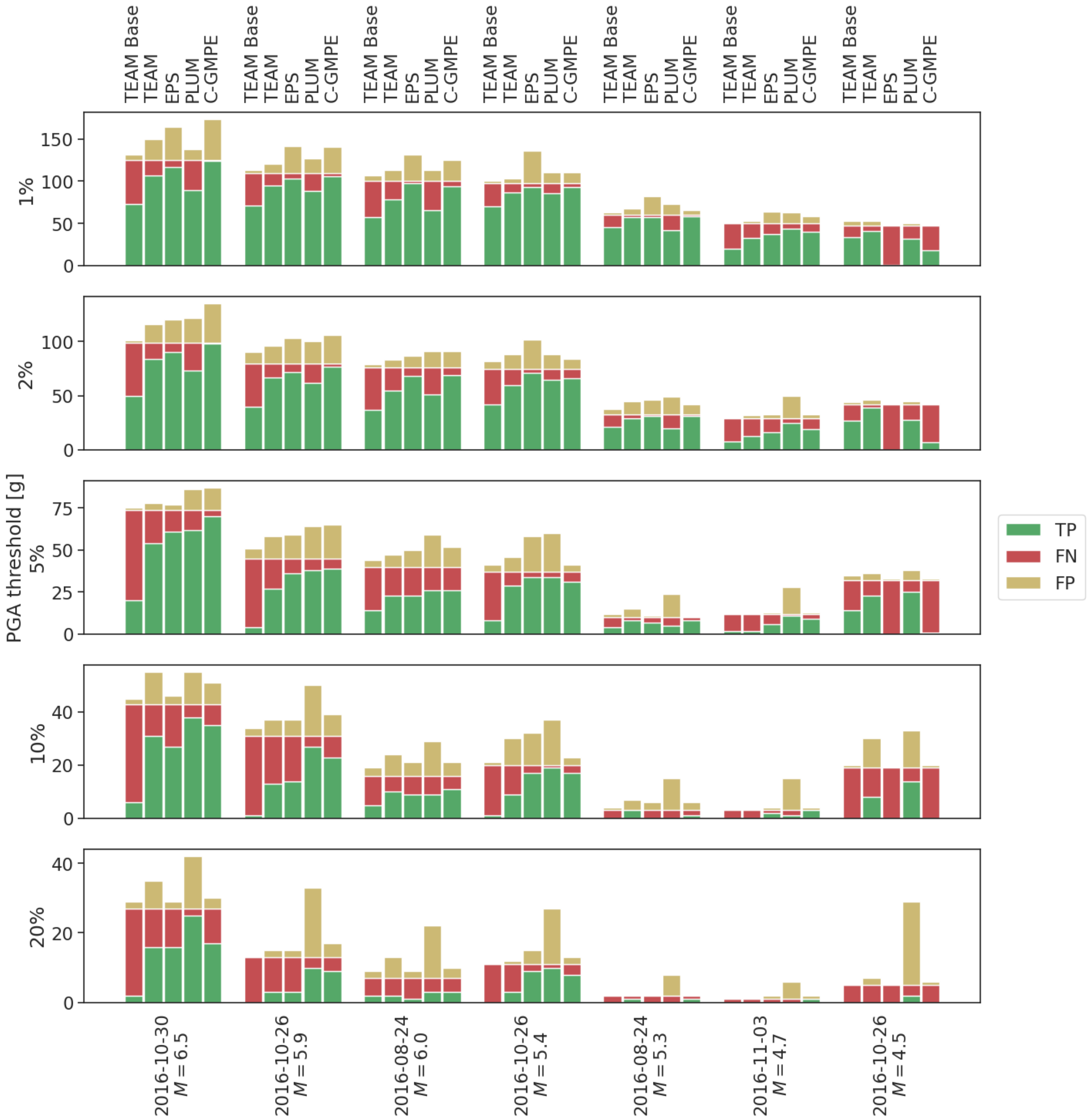


Figure 4.6: True positives (TP), false negatives (FN) and false positives (FP) for the events in the Italy test set causing the largest shaking. The methods are the transformer earthquake alerting model without domain adaptation (TEAM base), the transformer earthquake alerting model (TEAM), the estimated point source algorithm (EPS) and the PLUM-based approach. In addition, a GMPE with full catalog information is included for reference (C-GMPE). Values α were chosen separately for each threshold and method to yield the highest F1 score for the whole test set, but are kept constant across all events. TEAM with domain adaptation outperforms TEAM without domain adaptation consistently across all thresholds. This indicates that the domain adaptation not only allows TEAM to better predict higher levels of shaking but also to better assess large events in general.

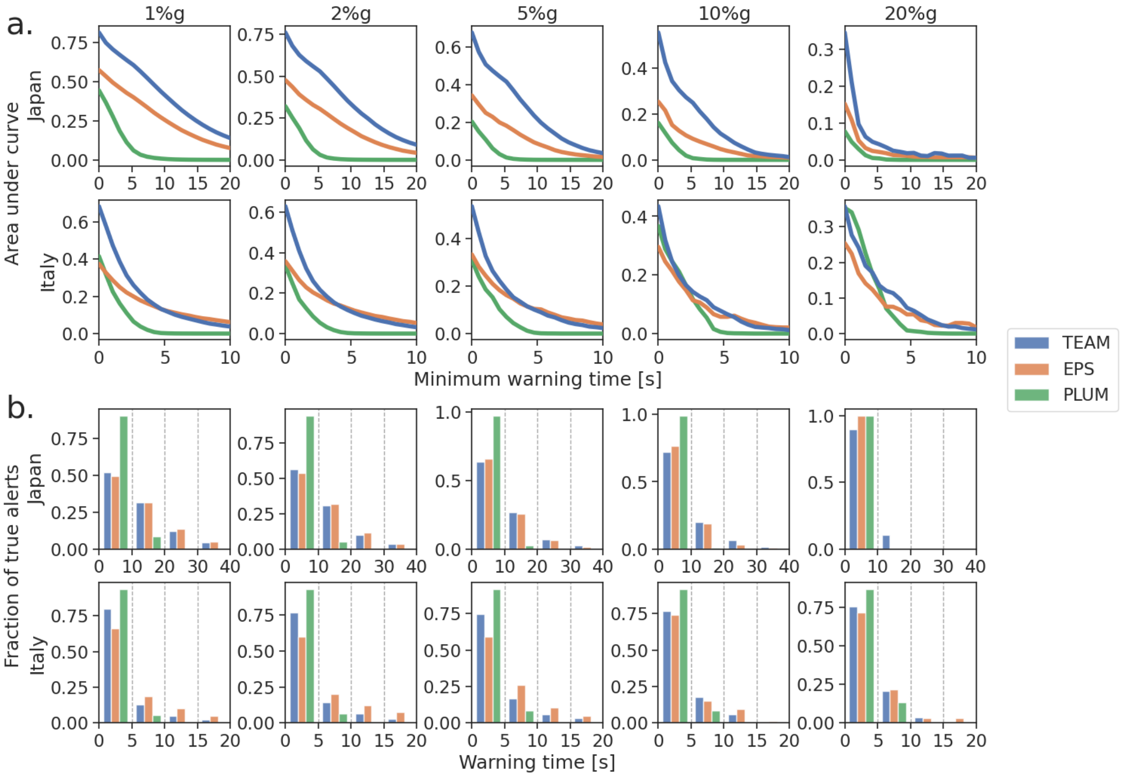


Figure 4.7: Warning time statistics. **a.** Area under the precision-recall curve for different minimum warning times. All alerts with shorter warning times are counted as false negatives. **b.** Warning time histograms showing the distribution of true alerts across distances for the different methods. Please note that the total number of true alerts differs by method and is not shown in this subplot. Therefore the values of different methods can not be directly compared, but only the differences in the distributions. TEAM and EPS are shown at F1-optimal α, chosen separately for each threshold and method. Warning time dependence on hypocentral distance is shown in Figure C.2.

4.2.2 Warning times

In application scenarios, a user will require a certain warning time, which is the time between issuing of the warning and the first exceedance of the level of shaking. This time is necessary for taking action. As the previous evaluation considered prediction accuracy irrespective of the warning time, we now compare the methods while imposing a certain minimum warning time. TEAM consistently outperforms both baselines across different required warning times and irrespective of the PGA threshold (Figure 4.7a). While the margin for TEAM compared to the baselines is smaller for Italy than for Japan, TEAM shows consistently strong performance across different warning times. In contrast, EPS performs worse at short warning times, the PLUM-based approach at longer warning times. The latter is inherent to the key idea of PLUM and makes the method only competitive at high PGA thresholds, where potential maximum warning times are naturally short due to the proximity between stations with strong shaking and the epicenter [Minson et al., 2018]. We further note that while the PLUM-like approach shows slightly higher AUC than TEAM for short warning times at 20 %g, this is only a hypothetical result. As PLUM does not have a tuning parameter between precision and recall, this performance can only be realised for a specific precision/recall threshold, where it performs slightly superior to TEAM (Figure 4.5a bottom right).

Warning times depend on α: a lower α value naturally leads to longer warning times but also to more false positive warnings. At F1-optimal thresholds α, EPS and TEAM have similar warning time distributions (Figure 4.7b, Table S3), but lowering α leads to stronger increases in warning times for TEAM. For instance, at 10%g, lowering α from 0.5 to 0.2 increases the average warning times of TEAM by 2.3 s/1.2 s (Japan/Italy), but only by 1.1 s/0.1 s for EPS. Short times as measured here are critical in real applications: First, they reduce the time available for countermeasures. Second, real warning times will be shorter than reported here due to telemetry and compute delays. However, compute delays for TEAM are very mild: analysing the Norcia event (25 input stations, 246 target sites) for one time step took only 0.15 s on a standard workstation using non-optimised code.

4.3 Discussion

4.3.1 Calibration of uncertainty estimates

Even though TEAM and EPS give probabilistic predictions, it is not clear whether these predictions are well-calibrated, i.e., if the predicted confidence values correspond to observed probabilities. Calibrated probabilities are essential for threshold selection, as they are required to balance expected costs of taking action versus expected costs of not taking action. We note that while good calibration is a necessary condition for a good model, it is not sufficient, as a model constantly predicting the marginal distribution of the labels would be always perfectly calibrated, yet not very useful.

To assess the calibration, we use calibration diagrams (Figures C.6, C.7) for Japan and Italy at different times after the first P arrival. These diagrams compare the predicted probabilities to the observed fraction of occurrences. In general, both models are well-calibrated, with a slightly better calibration for TEAM. Calibration is generally better for Japan, where only EPS is slightly underconfident at earlier times for the highest PGA thresholds. For Italy, EPS is generally slightly overconfident, while TEAM is well-calibrated, except for a certain overconfidence at 20%g. We suspect that the worse calibration for the largest events is caused by the domain adaptation strategy, but the better performance in terms of accuracy weighs out this downside of domain adaptation.

4.3.2 Insights into TEAM

We analyse differences between the methods using one example event from each dataset (Japan: Figure 4.8, Italy: Figure 4.9). All methods underestimate the shaking in the first seconds (left column Figures 4.8, 4.9). However, TEAM is the quickest to detect the correct extent of the shaking. Additionally, it estimates even fine-grained regional shaking details in real-time (middle and right columns). In contrast, shake maps for EPS remain overly simplified due to the assumptions inherent to GMPEs (right column and bottom left panel). For the Japan example, even late predictions of EPS underestimate the shaking, due to an underestimation of the magnitude. The PLUM-based approach produces very good PGA estimates but exhibits the shortest warning times.

Notably, TEAM predictions at later times correspond even better to the measured PGA than C-GMPE estimates, although these are based on the final magnitude (top right and bottom left panels). For the Japan data, this is not only the case for the example at hand but also visible in Figure 4.5, showing higher accuracy of TEAM's prediction compared to C-GMPE for all thresholds except 20%g on the full Japan dataset. We as-

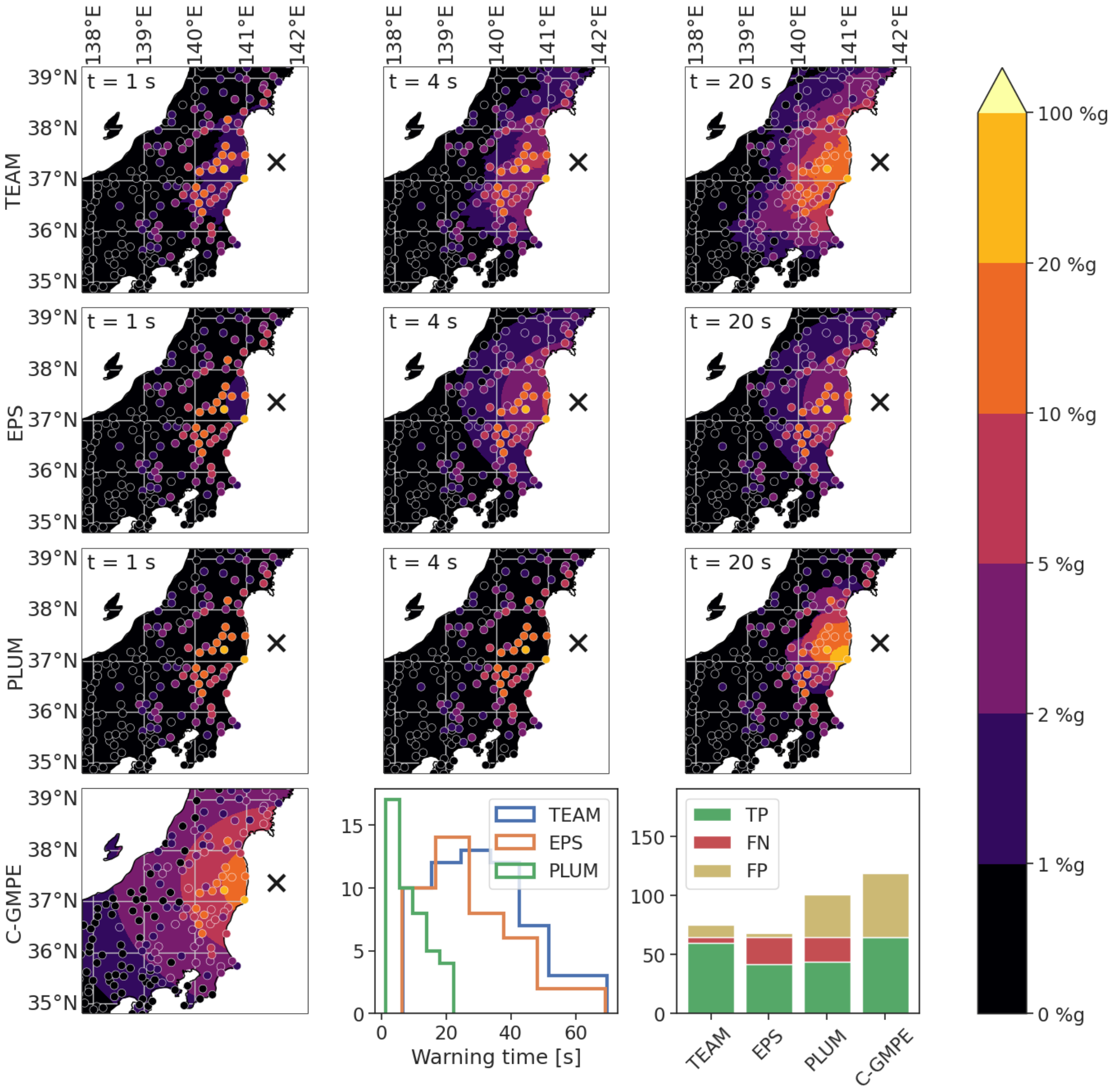


Figure 4.8: Scenario analysis of the 22nd November 2016 $M_J = 7.4$ Fukushima earthquake, the largest test event located close to shore. Maps show the warning levels for each method (top three rows) at different times (shown in the corners, $t = 0$ s corresponds to the P arrival at the closest station). Dots represent stations and are coloured according to the PGA recorded during the full event, i.e., the prediction target. The bottom row shows (left to right), the catalog based GMPE predictions, the warning time distributions, and the true positives (TP), false negatives (FN) and false positives (FP) for each method, both at a 2%g PGA threshold. EPS and GMPE shake map predictions do not include station terms, but they are included for the bottom row histograms.

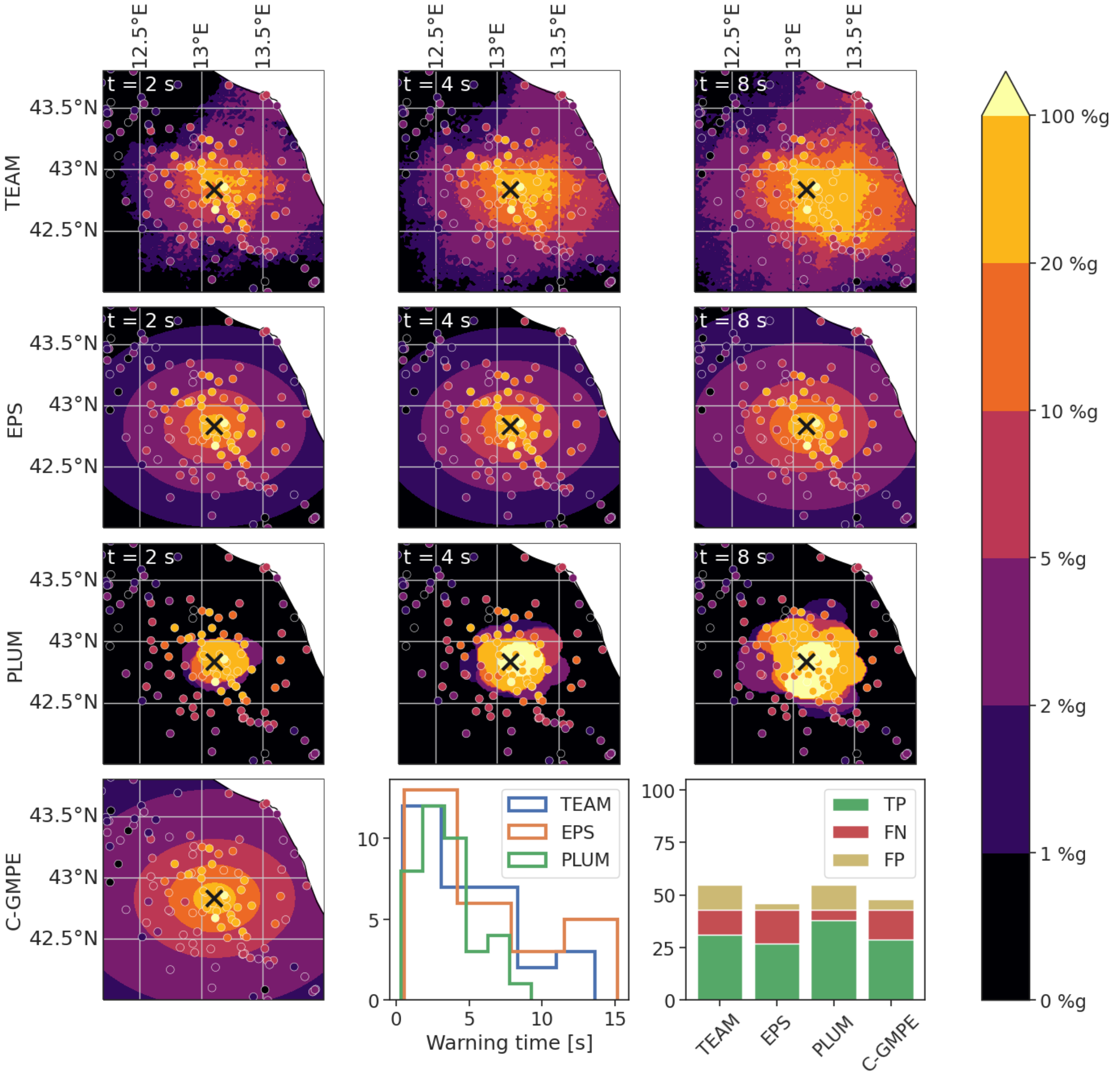


Figure 4.9: Scenario analysis of the 30th October 2016 $M_w = 6.5$ Norcia earthquake, the largest event in the Italy test set. See Figure 4.8 for further explanations. The bottom row diagrams for this scenario analysis use a 10%g PGA threshold.

sume TEAM's superior performance is rooted in both global and local aspects. Global aspects are the abilities to exploit variations in the waveforms, e.g., frequency content, to model complex event characteristics, such as stress drop, radiation pattern or directivity, and to compare to events in the training set. Local aspects include understanding regional effects, e.g., frequency-dependent site responses, and the ability to consider shaking at proximal stations. We note that for our Italy experiments, the modelling of local aspects resulting from regional characteristics might be slightly degraded by the domain adaptation. However, the first-order propagation effects such as, e.g., amplitude decay due to geometric spreading, are similar between regions and therefore not negatively affected by the domain adaptation. In conclusion, combining a global event view with propagation aspects, TEAM can be seen as a hybrid model between source estimation and propagation.

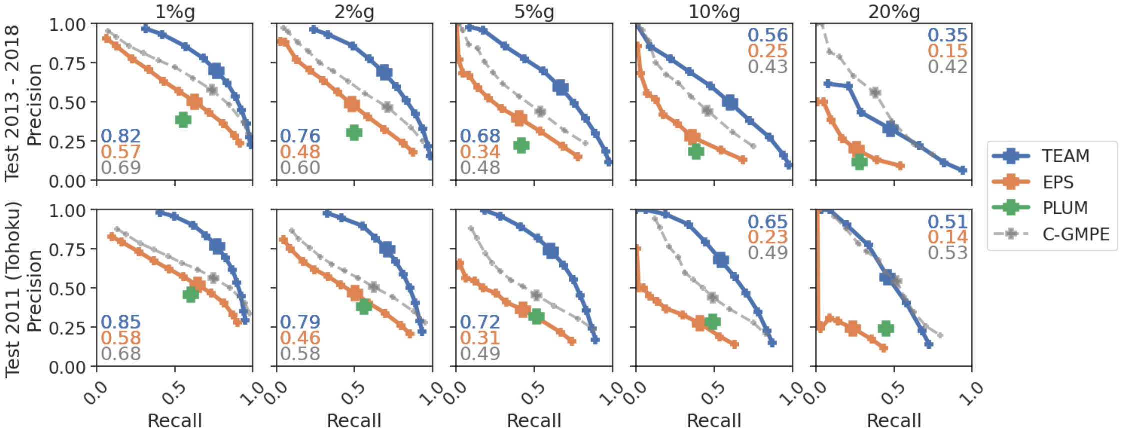


Figure 4.10: Precision recall curves for the Japanese dataset using the chronological split (top) and using the events in 2011 as test set (bottom). The year 2011 contains the $M_w = 9.1$ Tohoku event as well as its aftershocks.

4.3.3 TEAM performance on the Tohoku sequence

We evaluated TEAM for Japan on a chronological train/dev/test split, as this split ensures the evaluation closest to the actual application scenario. On the other hand, this split put the $M = 9.1$ Tohoku event in March 2011 into the training set. To evaluate the performance for this very large event and its aftershocks, we trained another TEAM instance using the year 2011 as test set and the remainder of the data for training and validation. Figure 4.10 shows the precision-recall curves for the chronological split and the year 2011 as test set. In general, the performance of all models stays similar when evaluated on the alternative split. A key difference between the curves is, that TEAM, in particular for high PGA thresholds, does not reach similar levels of recall for 2011 as for the chronological split, while achieving higher precision. As we will describe in the next paragraph, this trend probably results from a tendency to underestimate true PGA amplitudes, which will naturally reduce recall and boost precision. We suspect that this tendency for underestimation is either caused by the higher number of large events in the 2011 test set compared to the chronological split or by the lower number of high PGA events in the training set without 2011. Nevertheless, the performance of TEAM as quantified by the AUC improves, and significantly so for the highest thresholds.

Figure C.3 presents a scenario analysis for the Tohoku event. All models underestimate the event considerably, with the strongest underestimation for the EPS method. Even 20 s after the first P wave arrival, all methods underestimate both the severity and the extent of shaking. Due to its localised approach, the PLUM-based model achieves the highest number of true warnings, albeit at short warning times and a certain number of false positives, which due to the underestimation are absent from TEAM and EPS predictions. The performance of both EPS and TEAM is likely degraded by the slow onset of the Tohoku event [Koketsu et al., 2011]. According to Koketsu et al. [2011] the main subevent with a displacement of 36 m only initiated 20 s after the onset of the Tohoku event. Therefore only the first P waves for EPS or at most the first 25 s of waveforms for TEAM is most likely insufficient to correctly estimate the size of the Tohoku event.

For Italy, we showed that underestimation for large events can be mitigated using transfer learning. However, the Tohoku event clearly shows the limitations of this strategy, as practically no training data for events of comparable size are available, even when using events across the globe. Therefore, for the largest events, alternative strategies need to be developed, e.g., training using simulated data. Furthermore, the 25 s of waveforms used by TEAM in the current implementation may, for a very large event, not capture the largest subevent. While we decided to use only 25 s of event waveforms, as there is only insufficient training data of longer events, this window could be extended when developing training strategies and models for the largest events.

4.4 Conclusion

In this chapter, we presented the transformer earthquake alerting model (TEAM). We compared TEAM to two prototypical existing early warning methods, one source estimation based and one propagation based approach. TEAM outperforms both approaches in terms of alert performance and warning time. Using a flexible machine learning model, TEAM extracts information about an event from raw waveforms and leverages the information to model the complex dependencies of ground motion. Towards the goal of this thesis, the main contributions of this chapter are to prove that real-time assessment of earthquakes with deep learning is possible and to provide a model to conduct this assessment. Furthermore, our experiments with transfer learning and the scenario analyses gave us a first impression of the characteristics, limitations, and possible training strategies of this real-time assessment model. Building upon TEAM, we will develop TEAM-LM, a model for real-time magnitude and location estimation in the next chapter. We will use TEAM-LM to study the characteristics of real-time models in more detail.

Concurrently and subsequently to the publication of this chapter [Münchmeyer et al., 2021b], several closely related studies were published. Zhang et al. [2021] developed a CNN based model for real-time magnitude and location estimation and applied it to the Central Italy sequence of 2016 that was also studied in this chapter. While motivating their approach with early warning, they did not explicitly evaluate the alert performance. In contrast to TEAM, their approach is not flexible with regard to the set of input stations. As TEAM, the approach provides real-time capabilities but uses a sliding window approach instead of zero-blinding. [Jozinović et al., 2022] conducted a study on transfer learning for ground motion prediction from waveforms, comparing different approaches regarding their performance. They used a multi-station approach but employed a fixed set of stations. They did not study real-time application, and consequently also did not discuss alert performance. van den Ende and Ampuero [2020] presented a method for magnitude and location estimation from a flexible set of seismic stations using global pooling of features. We will compare their approach in detail to our work in the subsequent Chapter 5, where we present and study TEAM-LM, an adaptation of TEAM to magnitude and location estimation.

Resource availability

The code for TEAM is available at `https://doi.org/10.5880/10.1093/gji/ggaa609` and `https://github.com/yetinam/TEAM`. We made the Italy dataset publicly available at `https://doi.org/10.5880/GFZ.2.4.2020.004`. Due to licensing restrictions, we are not able to redistribute the Japan dataset, but instructions and code to convert it from the source files are available in the TEAM software repository.

5 Real-time earthquake magnitude and location estimation

In the previous chapter, we developed TEAM, an end-to-end approach for estimating ground motion parameters from waveforms in real-time. While we conducted some analyses of the predictions and experiments with transfer learning, many questions remain open. For example, under which circumstances does the model fail; how does the model performance depend on training data; which impact do different training strategies have on the performance? Similar questions arise from other studies of deep learning for earthquake assessment, where these questions are also not discussed [e.g., Lomax et al., 2019, Mousavi and Beroza, 2020b, van den Ende and Ampuero, 2020]. To address theses questions, in this chapter, we develop TEAM-LM, a method for real-time magnitude and location estimate, and conduct an in-depth analysis of the failure modes, the influence of the training data, and potential training strategies.[22] In contrast to the previous chapter, where we discussed ground motion, a key metric for early warning, we now investigate two source parameters: magnitude and location. This is beneficial for our analysis, as the source parameters are not affected by site conditions acting as confounding factors, which stands in contrast to ground motion parameters.

Recently, multiple studies investigated deep learning for the fast assessment of earthquake source parameters, such as magnitude [e.g., Lomax et al., 2019, Mousavi and Beroza, 2020b, van den Ende and Ampuero, 2020] and location [e.g., Kriegerowski et al., 2019, Mousavi and Beroza, 2020a, van den Ende and Ampuero, 2020]. Deep learning is well suited for these tasks, as it does not rely on manually selected features, but can learn to extract relevant information from the raw input data. This property allows the models to use the full information contained in the waveforms of an event. However, several desirable properties are missing from these models. First, the models can not be applied in real-time, instead requiring a fixed amount of waveforms after the event onset. Second, except for the model by van den Ende and Ampuero [2020], all models process either waveforms from only a single seismic station or rely on a fixed set of seismic stations defined at training time. However, this is desirable, as outlined in the introduction of Chapter 4. The model by van den Ende and Ampuero [2020] enables the use of a variable station set but combines measurements from multiple stations using a simple pooling mechanism. While it has not been studied so far in a seismological context, it has been shown in the general domain that set pooling architectures are in practice limited in the complexity of functions they can model [Lee et al., 2019].

In this chapter, we introduce a new model for magnitude and location estimation based on the architecture of TEAM (Chapter 4), a deep learning based earthquake early warning model. While TEAM estimated the PGA at target locations, our model estimates the magnitude and the hypocentral location of the event. We call our adaptation TEAM-LM, TEAM for location and magnitude estimation. We use TEAM as a basis due to its flexible multi-station approach and its ability to process incoming data effectively in real-time, issuing updated estimates as additional data become available. Similar to TEAM, TEAM-LM uses mixture density networks to provided probability distributions rather than merely point estimates as predictions.

To perform a comprehensive evaluation of TEAM-LM, we use three large and diverse datasets: the regional broadband dataset from Northern Chile with magnitudes calibrated in Chapter 3, and the two strong motion datasets from Japan and Italy that were intro-

[22] This chapter has been published as [Münchmeyer et al., 2021a]. Compared to the publication, the Introduction and Conclusion of this chapter have been modified to highlight the context of the chapter within this thesis. Minor modifications were introduced to the remaining text and figures.

duced in Chapter 4.[23] These datasets differ in their seismotectonic environment (Northern Chile and Japan: subduction zones; Italy: dominated by both convergent and divergent continental deformation), their spatial extent (Northern Chile: regional scale; Italy and Japan: national catalogs), and the instrument type (Northern Chile: broadband, Italy and Japan: strong motion). This selection of diverse datasets allows for a comprehensive analysis, giving insights for different use cases.

For magnitude estimation, our model outperforms two state-of-the-art baselines, one using deep learning [van den Ende and Ampuero, 2020] and one classical approach [Kuyuk and Allen, 2013]. For location estimation, our model outperforms a deep learning baseline [van den Ende and Ampuero, 2020] and shows promising performance in comparison to a classical localisation algorithm. However, our analysis also reveals limitations of the model. The performance degrades significantly when faced with training data sparsity: large magnitudes are systematically underestimated, events in previously seismically quiet regions systematically mislocated. Our experiments show that the characteristics of TEAM-LM are rooted in the principle structure, i.e., the black-box approach of learning a very flexible model from data, without imposing any physical constraints. As this black-box approach is common to all current fast assessment models using deep learning, we expect that our results can be generalised, i.e., that other deep learning models for earthquake assessment will exhibit similar characteristics. This finding is further backed by comparison to the results reported in previous studies.

5.1 Data and Methods

5.1.1 Datasets

For this study, we use three datasets (Table 5.1, Figure 5.1): one from Northern Chile, one from Italy and one from Japan. The Chile dataset is based on the catalog by Sippl et al. [2018] with the magnitude values obtained in Chapter 3. While there were minor changes in the seismic network configuration during the time covered by the catalog, the station set used in the construction of this catalog had been selected to provide a high degree of stability of location accuracy throughout the observational period [Sippl et al., 2018]. Similarly, we calibrated the magnitude scale carefully to achieve a high degree of consistency in spite of significant variations of attenuation (Chapter 3). This dataset, therefore, contains the highest quality labels among the datasets in this study. For the Chile dataset, we use broadband seismogramms from the fixed set of 24 stations used for the creation of the original catalog and magnitude scale. Although the Chile dataset has the smallest number of stations of the three datasets, it comprises three to four times as many waveforms as the other two due to a large number of events.

The datasets for Italy and Japan are identical to the ones used in the previous chapter. Here, we give a brief recap of their characteristics. The two datasets are more focused on early warning than the Chile dataset, containing fewer events and only strong motion waveforms. They are based on catalogs from the INGV [ISIDe Working Group, 2007] and the NIED KiKNet [National Research Institute For Earth Science And Disaster Resilience, 2019], respectively. The datasets each encompass a larger area than the Chile dataset and include waveforms from significantly more stations. In contrast to the Chile dataset, the station coverage differs strongly between different events, as only stations recording the event are considered. In particular, KiKNet stations do not record continuous waveforms,

[23] We did not use the Chile dataset for the evaluation of TEAM in Chapter 4 as it is lacking relevant characteristics to evaluation early warning: recordings are primarily from broadband instruments rather than from strong motion instruments and the spacing between the stations is too wide.

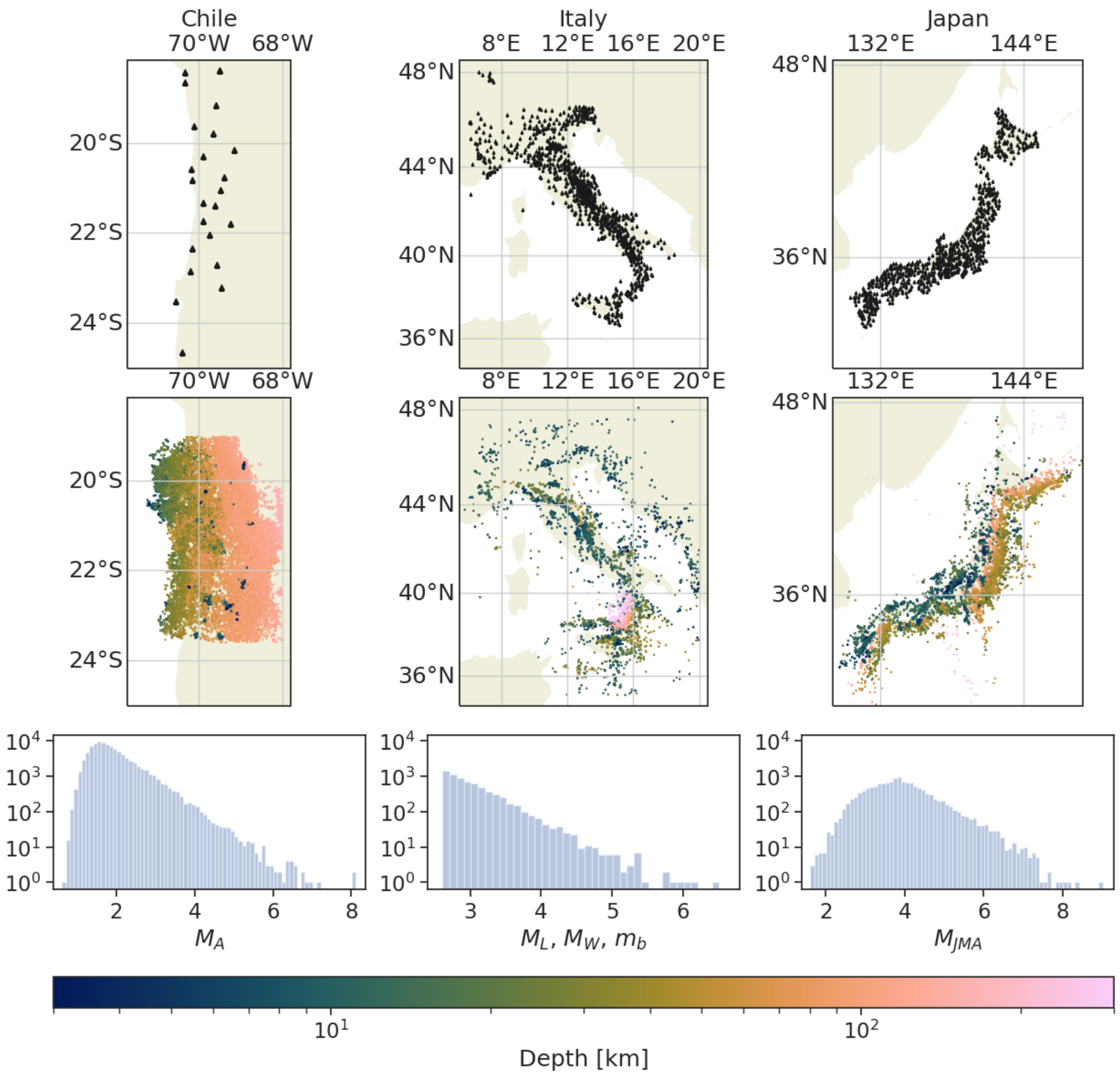


Figure 5.1: Overview of the datasets. The top row shows the spatial station distribution, the second row the spatial event distribution. The event depth is encoded using color. Higher resolution versions of the maps can be found in Figures D.1, D.2, D.3. The bottom row shows the distributions of the event magnitudes. The magnitude scales are the peak displacement based M_A, local magnitude M_L, moment magnitude M_W, body wave magnitude m_b and M_{JMA}, a magnitude primarily using peak displacement.

but operate in trigger mode, only saving waveforms if an event triggered at the station. For Japan each station comprises two sensors, one at the surface and one borehole sensor. Therefore for Japan we have 6 component recordings (3 surface, 3 borehole) available instead of the 3 component recordings for Italy and Chile. A full list of seismic networks used in this study can be found in Table D.1.

For each dataset we use the magnitude scale provided in the catalog. For the Chile catalog, this is M_A, a peak displacement based scale, but without the Wood-Anderson response and therefore saturation-free for large events [Deichmann, 2018b]. For Japan, M_{JMA} is used. M_{JMA} combines different magnitude scales but, similarly to M_A, primarily uses horizontal peak displacement [Doi, 2014]. For Italy, the catalog provides different magnitude types approximately dependent on the size of the event: M_L (>90 % of the

Table 5.1: Overview of the datasets. The lower boundary of the magnitude category is the 5th percentile of the magnitude; this limit is chosen as each dataset contains a small number of unrepresentative very small events. The upper boundary is the maximum magnitude. Magnitudes are given with two digit precision for Chile, as the precision of the underlying catalog is higher than for Italy and Japan. The Italy dataset uses different magnitude scales for different events, which are M_L (>90 % of the events), M_W (<10 %) and m_b (<1 %). For depth and distance minimum, median and maximum are stated. Distance refers to the epicentral distance between stations and events. Note that the count of traces refers to the number of waveform triplets (for Chile and Italy), or groups of six waveforms (for the Japanese stations). The sensor types are broadband (BB) and strong motion (SM).

	Chile	Italy	Japan
Years	2007 - 2014	2008 - 2019	1997 - 2018
Training	01/2007-08/2011	01/2008 - 12/2015 & 01/2017 - 12/2019	01/1997 - 03/2012
Test	08/2012 - 12/2014	01/2016 - 12/2016	08/2013 - 12/2018
Magnitudes	1.21 - 8.27	2.7 - 6.5	2.7 - 9.0
Magnitude scale	M_A	M_L, M_W, m_b	M_{JMA}
Depth [km]	0 - 102 - 183	0 - 10 - 617	0 - 19 - 682
Distance [km]	0.1 - 180 - 640	0.1 - 180 - 630	0.2 - 120 - 3190
Events	96,133	7,055	13,512
Unique stations	24	1,080	697
Traces	1,605,983	494,183	372,661
Traces per event	16.7	70.3	27.6
Sensor type	BB	SM	SM & SM-borehole
Catalog source	Münchmeyer et al. [2020]	INGV	NIED

events), M_W (<10 %) and m_b (<1 %). We note that while the primary magnitude scales for all datasets are peak-displacement based, the precision of the magnitudes vary, with the highest precision for Chile. This might lead to slightly worse magnitude estimation performance for Italy and Japan. We do not have quantitative data on the uncertainties of the magnitude values for the Italy and Japan datasets.

For all datasets, the data were not subselected based on the type of seismicity but only based on the location (for Chile and Italy) or depending on if they triggered (Japan). This guarantees that, even though we made use of a catalog to assemble our training data, the resulting datasets are suitable for training and assessing methods geared at real-time applications without any prior knowledge about the earthquakes. We focus on earthquake characterisation and do not discuss event detection or separation from noise; we refer the interested reader to, e.g., Perol et al. [2018] or Mousavi et al. [2019b].

We split each dataset into training, development and test set. For Chile and Japan, we apply a simple chronological split with approximate ratios of 60:10:30 between training, development and test set, with the most recent events in the test set. As the last 30% of the Italy dataset consist of less interesting events, in particular missing large events, we instead use all events from 2016 as test set and the remaining events as training and development sets. We reserve all of 2016 for testing, as it contains a long seismic sequence in central Italy with two mainshocks in August (M_W = 6.5) and October (M_W = 6.0). Notably, the largest event in the test set is significantly larger than the largest event in the training set (M_w = 6.1 L'Aquila event in 2007), representing a challenging test case.

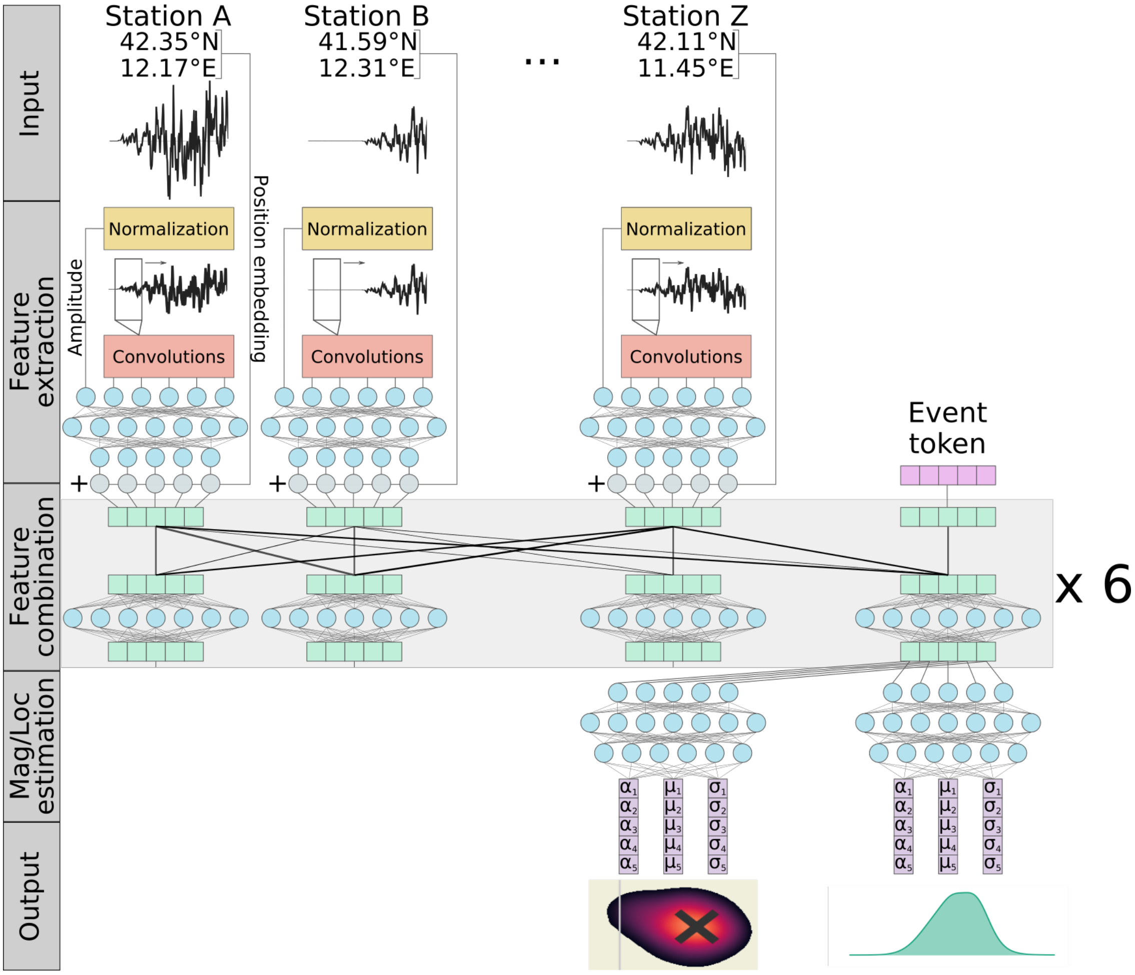


Figure 5.2: Overview of the adapted transformer earthquake alerting model (TEAM-LM), showing the input, the feature extraction, the feature combination, the magnitude/location estimation and the output. For simplicity, not all layers are shown, but only their order and combination is visualised schematically. For the exact number of layers and the size of each layer we refer to Tables D.2 to D.4. Please note that the number of input stations is variable, due to the self-attention mechanism in the feature combination.

For Italy, we assign the remaining events to training and development set randomly with a 6:1 ratio. The splits for Japan and Italy are identical to the ones used in Chapter 4.

5.1.2 The transformer earthquake alerting model for magnitude and location

In the last chapter we built TEAM, a method for real-time end-to-end estimation of ground shaking. Here, we adapt TEAM to calculate real-time probabilistic estimates of event magnitude and hypocentral location. As our model closely follows the architecture and key ideas of TEAM, we use the name TEAM-LM to refer to the location and magnitude estimation model.

Similar to TEAM, TEAM-LM consists of three major components (Figure 5.2): a feature extraction, which generates features from raw waveforms at single stations, a feature combination, which aggregates features across multiple stations, and an output estimation. Here, we briefly discuss the core ideas of the TEAM architecture and training and put a further focus on the necessary changes for magnitude and location estimation.

The input to TEAM consists of three component seismogramms from multiple stations and their locations. TEAM aligns all seismogramms to start and end at the same times t_0 and t_1. We choose t_0 to be 5 seconds before the first P arrival at any station. This allows the model to understand the noise conditions at all stations. We limit t_1 to be at latest $t_0 + 30$ s. In a real-time scenario t_1 is the current time, i.e., the available amount of waveforms, and we use the same approach to imitate real-time waveforms in training and evaluation. The waveforms are padded with zeros to a length of 30 s to achieve constant length input to the feature extraction.

TEAM uses a CNN architecture for feature extraction, which is applied separately at each station. The architecture consists of several convolution and pooling layers, followed by a multi-layer perceptron (Table D.2). To avoid scaling issues, each input waveform is normalised through division by its peak amplitude. As the amplitude is expected to be a key predictor for the event magnitude, we provide the logarithm of the peak amplitude as a further input to the multi-layer perceptron inside the feature extraction network. We ensure that this transformation does not introduce a knowledge leak by calculating the peak amplitude based only on the waveforms until t_1. The full feature extraction returns one vector for each station, representing the measurements at the station.

The feature vectors from multiple stations are combined using a transformer network [Vaswani et al., 2017]. Transformers are attention based neural networks, originally introduced for natural language processing. A transformer takes a set of n vectors as input, and outputs again n vectors which now incorporate the context of each other. The attention mechanism allows the transformer to put special emphasis on inputs that it considers particularly relevant and thereby model complex inter-station dependencies. Importantly, the parameters of the transformer are independent of the number of input vectors n, allowing to train and apply a transformer on variable station sets. To give the transformer a notion of the position of the stations, TEAM encodes the latitude, longitude and elevation of the stations using a sinusoidal embedding and adds this embedding to the feature vectors.

TEAM adds the position embeddings of the PGA targets as additional inputs to the transformer. In TEAM-LM, we aim to extract information about the event itself, where we do not know the position in advance. To achieve this, we add an event token, which is a vector with the same dimensionality as the positional embedding of a station location, and which can be thought of as a query vector. This approach is inspired by the so-called sentence tokens in NLP that are used to extract holistic information on a sentence [Devlin et al., 2018]. The elements of this event query vector are learned during the training procedure.

From the transformer output, we only use the output corresponding to the event token, which we term event embedding and which we pass through another multi-layer perceptron predicting the parameters of a Gaussian mixture [Bishop, 1994]. We use $N = 5$ Gaussians for magnitude and $N = 15$ Gaussians for location estimation. For computational and stability reasons, we constrain the covariance matrix of the individual Gaussians for location estimation to a diagonal matrix to reduce the output dimensionality. Even though uncertainties in latitude, longitude and depth are known to generally be correlated, this correlation can be modelled with diagonal covariance matrices by using the mixture.

The model is trained end-to-end using a log-likelihood loss with the Adam optimiser [Kingma and Ba, 2014]. We train separate models for magnitude and for location. As we observed difficulties in the onset of the optimisation when starting from a fully random initialisation, we pretrain the feature extraction network. To this end we add a mixture density network directly after the feature extraction and train the resulting network to predict magnitudes from single station waveforms. We then discard the mixture density network and use the weights of the feature extraction as initialisation for the end-to-end training. We use this pretraining method for both magnitude and localisation networks.

Similarly to the training procedure for TEAM, we make extensive use of data augmentation during training. First, we randomly select a subset of up to 25 stations from the available station set. We limit the maximum number to 25 for computational reasons. Second, we apply temporal blinding, by zeroing waveforms after a random time t_1. This type of augmentation allows TEAM-LM to be applied to real-time data. We note that this type of temporal blinding would most likely work for the previously published CNN approaches as well, making them applicable to real-time prediction. To avoid knowledge leaks for Italy and Japan, we only use stations as inputs that triggered before time t_1 for these datasets. This is not necessary for Chile, as there the maximum number of stations per event is below 25 and waveforms for all events are available for all stations active at that time, irrespective of whether the station actually recorded the event. Third, we oversample large magnitude events, as they are strongly underrepresented in the training dataset. We discuss the effect of this augmentation in further detail in the results section. In contrast to the station selection during training, in evaluation we always use the 25 stations picking first. Again, for Italy and Japan, we only use stations and their waveforms as input once they triggered, thereby ensuring that the station selection does not introduce a knowledge leak.

5.1.3 Baseline methods

Recently[24], van den Ende and Ampuero [2020] suggested a deep learning method capable of incorporating waveforms from a flexible set of stations. Their architecture uses a similar CNN based feature extraction as TEAM-LM. In contrast to TEAM-LM, for feature combination it uses maximum pooling to aggregate the feature vectors from all stations instead of a transformer. In addition they do not add predefined position embeddings, but concatenate the feature vector for each station with the location coordinates and apply a multi-layer perceptron to get the final feature vectors for each station. The model of van den Ende and Ampuero [2020] is both trained and evaluated on 100 s long waveforms. In its original form it is therefore not suitable for real-time processing, although the real-time processing could be added with the same zero-padding approach employed for TEAM and TEAM-LM. The detail differences in the CNN structure and the real-time processing capability make a comparison of the exact model of van den Ende and Ampuero [2020] to TEAM-LM difficult.

To still compare TEAM-LM to the techniques introduced in this approach, we implemented a model based on the key concepts of van den Ende and Ampuero [2020]. As we aim to evaluate the performance differences from the conceptual changes, rather than different hyperparameters, e.g., the exact size and number of the convolutional layers, we use the same architecture as TEAM-LM for the feature extraction and the mixture density output. Additionally we train the model for real-time processing using zero padding. In comparison to TEAM-LM we replace the transformer with a maximum pooling operation and remove the event token.

We evaluate two different representations for the position encoding. In the first, we concatenated the positions to the feature vectors as proposed by van den Ende and

[24] Compared to the original publication of the TEAM-LM method in [Münchmeyer et al., 2021a].

Ampuero [2020]. In the second, we add the position embeddings element-wise to the feature vectors as for TEAM-LM. In both cases, we run a three-layer perceptron over the combined feature and position vector for each station, before applying the pooling operation.

We use the fast magnitude estimation approach [Kuyuk and Allen, 2013] as a classical baseline for magnitude estimation, i.e., a baseline method not using deep-learning. The magnitude is estimated from the horizontal peak displacement in the first seconds of the P wave. As this approach estimates the attenuation using the hypocentral distance, it requires knowledge of the event location. We simply provide the method with the catalog hypocenter. While this would not be possible in real-time, and therefore gives the method an unfair advantage over the deep learning approaches, it allows us to focus on the magnitude estimation capabilities. Furthermore, in particular for Italy and Japan, the high station density usually allows for sufficiently well constrained location estimates at early times. For a full description of this baseline, see Appendix D.1.

As a classical location baseline we employ NonLinLoc [Lomax et al., 2000] with the 1D velocity models from Graeber and Asch [1999] for Chile, from Ueno et al. [2002] for Japan, and from Matrullo et al. [2013] for Italy. For the earliest times after the event detection usually only few picks picks are available. Therefore we apply two heuristics. Until at least 3/5/5 (Chile/Japan/Italy) picks are available, the epicenter is estimated as the arithmetic mean of the stations with picked arrivals so far, while the depth is set to the median depth in the training dataset. Until at least 4/7/7 picks are available, we apply NonLinLoc, but fix the depth to the median depth in the dataset. We require higher numbers of picks for Italy and Japan, as the pick quality is lower than in Chile but the station density is higher. For a constant number of stations, this leads to worse early NonLinLoc estimates in Italy and Japan compared to Chile, but improves the performance of the heuristics.

5.2 Results

5.2.1 Magnitude estimation performance

We first compare the estimation capabilities of TEAM-LM to the baselines in terms of magnitude (Figure 5.3). We evaluate the models at fixed times $t = 0.5$ s, 1 s, 2 s, 4 s, 8 s, 16 s, 25 s after the first P arrival at any station in the network. In addition to presenting selected results here, full tables with the results of further experiments are available in the supplementary material (Tables SM 5–SM 15) of [Münchmeyer et al., 2021a].[25]

TEAM-LM outperforms the classical magnitude baseline consistently. On two datasets, Chile and Italy, the performance of TEAM-LM with only 0.5 s of data is superior to the baseline with 25 s of data. Even on the third dataset, Japan, TEAM-LM requires only approximately a quarter of the time to reach the same precision as the classical baseline and achieves significantly higher precision after 25 s. The RMSE for TEAM-LM stabilises after 16 s for all datasets with final values of 0.08 m.u. for Chile, 0.20 m.u. for Italy and 0.22 m.u. for Japan. The performance differences between TEAM-LM and the classical baseline result from the simplified modelling assumptions for the baseline. While the relationship between early peak displacement and magnitude only holds approximately, TEAM-LM can extract more nuanced features from the waveform. In addition, the relationship for the baseline was originally calibrated for a moment magnitude scale. While

[25] Contrary to the remaining supplementary materials of the publications we decided not to include the tables in the appendix due to their enormous space requirements.

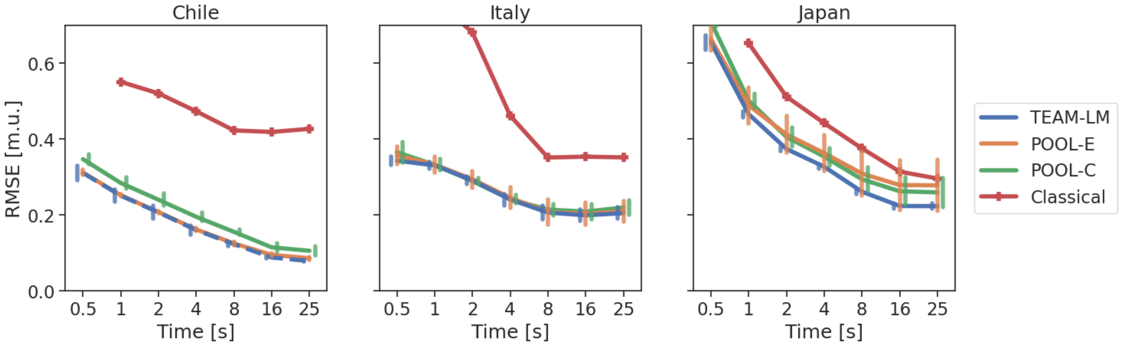


Figure 5.3: RMSE of the mean magnitude predictions from TEAM-LM, the pooling model with sinusoidal location embeddings (POOL-E), the pooling model with concatenated positions (POOL-C) and the classical baseline method. The time indicates the time since the first P arrival at any station, the RMSE is provided in magnitude units [m.u.]. Error bars indicate ±1 standard deviation when training the model with different random initialisations. For better visibility error bars are provided with a small x-offset. Standard deviations were obtained from six realisations. Note that the uncertainty of the provided means is by a factor $\sqrt{6}$ smaller than the given standard deviation, due to the number of samples. We provide no standard deviation for the baseline, as it does not depend on a model initialisation.

all magnitude scales have an approximate 1:1 relationship with moment magnitude, this might introduce further errors.

We further note that the performance of the classical baseline for Italy is consistent with the results reported by Festa et al. [2018]. They analysed early warning performance in a slightly different setting, looking only at the 9 largest events in the 2016 Central Italy sequence. However, they report a RMSE of 0.28 m.u. for the PRESTO system 4 s after the first alert, which matches approximately the 8 s value in our analysis. Similarly, Leyton et al. [2018] analyse how fast magnitudes can be estimated in subductions zones and obtain residuals of 0.01 ± 0.28 (mean and standard deviation) across all events and -0.70 ± 0.30 for the largest events ($M_w > 7.5$) at 30 s after origin time. This matches the observed performance of the classical baseline for Japan. For Chile, our classical baseline performs considerably worse, likely caused by the many small events with bad SNR compared to the event set considered by Leyton et al. [2018]. However, TEAM-LM still outperforms the performance numbers reported by Leyton et al. [2018] by a factor of more than 2.

Improvements for TEAM-LM in comparison to the deep learning baseline variants are much smaller than to the classical approach. Still, for the Japan dataset at late times, TEAM-LM offers improvements of up to 27 % for magnitude. For the Italy dataset, the baseline variants are on par with TEAM-LM. For Chile, only the baseline with position embeddings is on par with TEAM-LM. Notably, for the Italy and Japan datasets, the standard deviation between multiple runs with different random model initialisation is considerably higher for the baselines than for TEAM-LM (Figure 5.3, error bars). This indicates that the training of TEAM-LM is more stable regarding model initialisation.

The gains of TEAM-LM can be attributed to two differences: the transformer for station aggregation and the position embeddings. In our experiments, we ruled out further differences, e.g. size and structure of the feature extraction CNN, by using identical network architectures for all parts except the feature combination across stations. Regarding

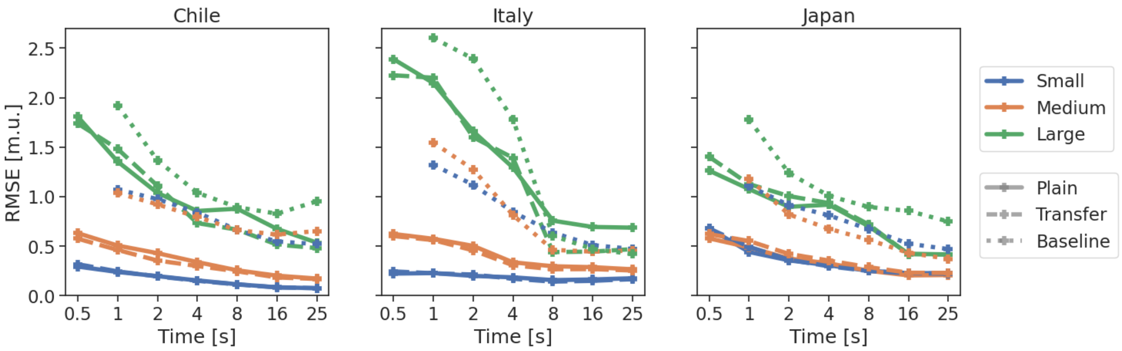


Figure 5.4: RMSE comparison of the TEAM-LM mean magnitude predictions for different magnitude buckets. Line styles indicate the model type: trained only on the target data (solid line), using transfer learning (dashed), classical baseline (dotted). For Chile/Italy/Japan we count events as small if their magnitude is below 3.5/3.5/4 and as large if their magnitude is at least 5.5/5/6. The time indicates the time since the first P arrival at any station, the RMSE is provided in magnitude units [m.u.].

the impact of position embeddings, the results do not show a consistent pattern. Gains for Chile seem to be solely caused by the position embeddings; gains for Italy are generally lowest, but again the model with position embeddings performs better; for Japan, the concatenation model performs slightly better, although the variance in the predictions makes the differences non-significant. We suspect these different patterns to be caused by the different catalog and network sizes as well as the station spacing.

We think that gains from using a transformer can be explained with its attention mechanism. The attention allows the transformer to focus on specific stations, for example, the stations which have recorded the longest waveforms so far. In contrast, the maximum pooling operation is less flexible. We suspect that the high gains for Japan result from the wide spatial distribution of seismicity and therefore very variable station distribution. While in Italy most events are in Central Italy and in Chile the number of stations is limited, the seismicity in Japan occurs along the whole subduction zone with additional onshore events. This complexity can likely be handled better with the flexibility of the transformer than using a pooling operation. This indicates that the gains from using a transformer compared to pooling with position embeddings are likely modest for small sets of stations, and highest for large heterogeneous networks.

5.2.2 Magnitude estimation performance for large events

In many use cases, the performance of magnitude estimation algorithms for large magnitude events is of particular importance. In Figure 5.4 we compare the RMSE of TEAM-LM and the classical baselines binned by catalog magnitude into small, medium and large events. For Chile/Italy/Japan we count events as small if their magnitude is below 3.5/3.5/4 and as large if their magnitude is at least 5.5/5/6. We observe a clear dependence on the event magnitude. For all datasets, the RMSE for large events is higher than for intermediate-sized events, which is again higher than for small events. On the other hand, the decrease in RMSE over time is strongest for larger events. This general pattern can also be observed for the classical baseline, even though the difference in RMSE between magnitude buckets is smaller. As both variants of the deep learning baseline show very similar trends to TEAM-LM, we omit them from this discussion.

We discuss two possible causes for these effects: (i) the magnitude distribution in the

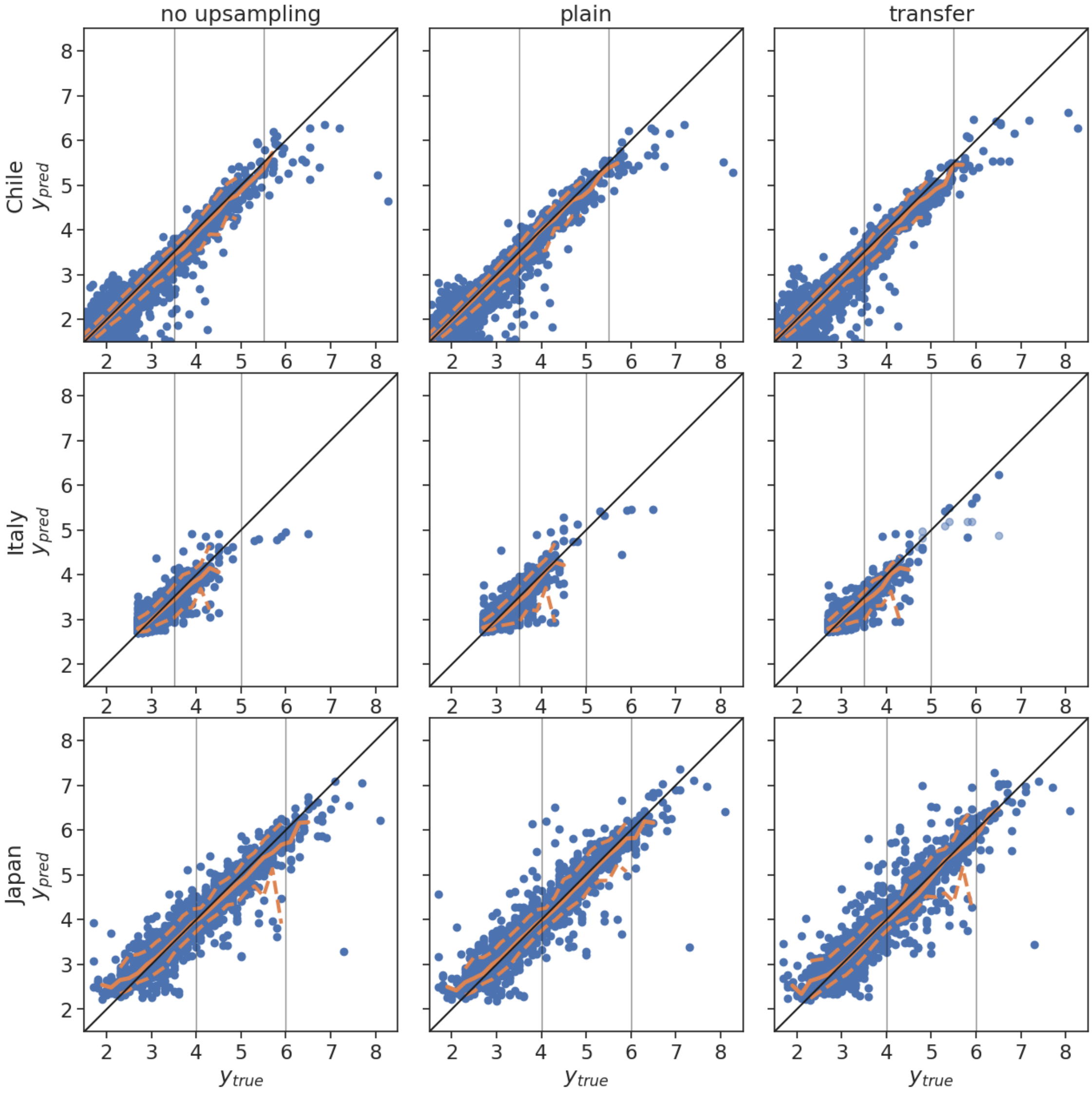


Figure 5.5: True and predicted magnitudes without upsampling or transfer learning (left column), with upsampling but without transfer learning (middle column) and with upsampling and transfer learning (right column). All plots show predictions after 8 seconds. In the transfer column, for Chile and Japan, we show results after fine-tuning on the target dataset; for Italy, we show results from the model without fine-tuning as this model performed better. For the largest events in Italy ($M > 4.5$), we additionally show the results after fine-tuning with pale blue dots. We suspect the degraded performance in the fine tuned model results from the fact that the largest training event ($M_W = 6.1$) is considerably smaller than the largest test event ($M_W = 6.5$). Vertical lines indicate the borders between small, medium and large events as defined in Figure 5.4. The orange lines show the running 5th, 50th and 95th percentile in 0.2 m.u. buckets. Percentile lines are only shown if sufficiently many data points are available. The very strong outlier for Japan (true ~7.3, predicted ~3.3) is an event far offshore (>2000 km).

training set restricts the quality of the model optimisation, (ii) inherent characteristics of large events. Cause (i) arise from the Gutenberg-Richter distribution of magnitudes. As large magnitudes are rare, the model has significantly fewer examples to learn from for large magnitudes than for small ones. This should impact the deep learning models the strongest, due to their high number of parameters. Cause (ii) has a geophysical origin. As large events have longer rupture durations, the information gain from longer waveform recordings is larger for large events. At which point during the rupture the final rupture size can be accurately predicted is a point of open discussion [e.g., Meier et al., 2017, Colombelli et al., 2020]. We probe the likely individual contributions of these causes in the following.

Estimations for large events not only show lower precision but are also biased (Figure 5.5, middle column). For Chile and Italy, a clear saturation sets in for large events. Interestingly the saturation starts at different magnitudes, which are around 5.5 for Italy and 6.0 for Chile. For Japan, events up to magnitude 7 are predicted without obvious bias. This saturation behaviour is not only visible for TEAM-LM but has also been observed in prior studies [e.g., Mousavi and Beroza, 2020b, Fig. 3, 4]. In their work, with a network trained on significantly smaller events, the saturation already sets in around magnitude 3. The different saturation thresholds indicate that the primary cause for saturation is not the longer rupture duration of large events or other inherent event properties, as in cause (ii), but is instead likely related to the low number of training examples for large events, rendering it nearly impossible to learn their general characteristics, as in cause (i). This explanation is consistent with the much higher saturation threshold for the Japanese dataset, where the training dataset contains a comparably large number of high magnitude events, encompassing the year 2011 with the Tohoku event and its aftershocks.

As a further check of cause (i), we trained models without upsampling large magnitude events during training, thereby reducing the occurrence of large magnitude events to the natural distribution observed in the catalog (Figure 5.5, left column). While the overall performance stays similar, the performance for large events is degraded on each of the datasets. Large events are on average underestimated even more strongly. We tried different upsampling rates but were not able to achieve significantly better performance for large events than the configuration of the preferred model presented in the paper. This shows that upsampling yields improvements but can not solve the issue completely, as it does not introduce actual additional data. On the other hand, the performance gains for large events from upsampling seem to cause no observable performance drop for smaller events. As the magnitude distribution in most regions approximately follows a Gutenberg-Richter law with $b \approx 1$, upsampling rates similar to the ones used in this paper will likely work for other regions as well.

The expected effects of cause (ii), inherent limitations to the predictability of rupture evolutions, can be approximated with physical models. To this end, we look at the model from Trugman et al. [2019], which suggests weak rupture predictability, i.e., predictability after 50 % of the rupture duration. Trugman et al. [2019] discuss the saturation of early peak displacement and the effects for magnitude predictions based on peak displacements. Following their model, we would expect magnitude saturation at approximately magnitude 5.7 after 1 s; 6.4 after 2 s; 7.0 after 4 s; 7.4 after 8 s. Comparing these results to Figure 5.5, the saturation for Chile and Italy occurs below these thresholds, and even for Japan the saturation is slightly below the modelled threshold. As we assumed a model with only weak rupture predictability, this makes it unlikely that the observed saturation is caused by limitations of rupture predictability. This implies that our result does not allow us to draw any conclusions on rupture predictability, as the possible effects of rupture predictability are masked by the data sparsity effects.

5.2.3 Location estimation performance

We evaluate the epicentral error distributions in terms of the 50^{th}, 90^{th}, 95^{th} and 99^{th} error percentiles (Figure 5.6). In terms of the median epicentral error, TEAM-LM outperforms all baselines in all cases, except for the classical baseline at late times in Italy. For all datasets, TEAM-LM shows a clear decrease in median epicentral error over time. The decrease is strongest for Chile, going from 19 km at 0.5 s to 2 km at 25 s. For Italy, the decrease is from 7 km to 2 km, for Japan from 22 km to 14 km. For all datasets, the error distributions are heavy-tailed. While for Chile even the errors at high quantiles decrease considerably over time, these quantiles stay nearly constant for Italy and Japan.

Similar to the difficulties for large magnitudes, the characteristics of the location estimation point to insufficient training data as the source of errors. The Chile dataset covers the smallest region and has by far the lowest magnitude of completeness, leading to the highest event density. Consequently, the location estimation performance is best and outliers are very rare. For the Italy and Japan datasets, significantly more events occurred in regions with only a few training events, causing strong outliers. The errors for the Japanese dataset are highest, presumably related to a large number of offshore events with consequently poor azimuthal coverage.

We expect a further difference from the number of unique stations. While for a small number of unique stations, as in the Chile dataset, the network can mostly learn to identify the stations using their position embeddings, it might be unable to do so for a larger number of stations with fewer training examples per station. Therefore the task is significantly more complicated for Italy and Japan, where the concept of station locations has to be learned simultaneously to the localisation task. This holds even though we encode the station locations using continuously varying position embeddings. Furthermore, whereas for moderate and large events waveforms from all stations of the Chilean network will contain the earthquake and can contribute information, the limitation to 25 stations of the current TEAM-LM implementation does not allow full exploitation of the information contained in the hundreds of recordings of larger events in the Japanese and Italian datasets. This will matter in particular for out-of-network events, where the wavefront curvature and thus event distance can only be estimated properly by considering stations with later arrivals.

Looking at the classical baseline, we see that it performs considerably worse than TEAM-LM in the Chile dataset in all location quantiles; better than TEAM-LM in all but the highest quantiles at late times in the Italy dataset; and worse than TEAM-LM at late times in the Japan dataset. This strongly different behaviour can largely be explained with the pick quality and the station density in the different datasets. While the Chile dataset contains high-quality automatic picks, obtained using the MPX picker [Aldersons, 2004], the Italy dataset uses a simple STA/LTA and the Japan dataset uses triggers from KiKNet. This reduces location quality for Italy and Japan, in particular in the case of a low number of picks available for location estimation. On the other hand, the very good median performance of the classical approach for Italy can be explained from the very high station density, giving a strong prior on the location. An epicentral error of around 2 km after 8 s is furthermore consistent with the results from Festa et al. [2018]. Considering the reduction in error due to the high station density in Italy, we note that the wide station spacing in Chile likely causes higher location errors than would be achievable with a denser seismic network designed for early warning.

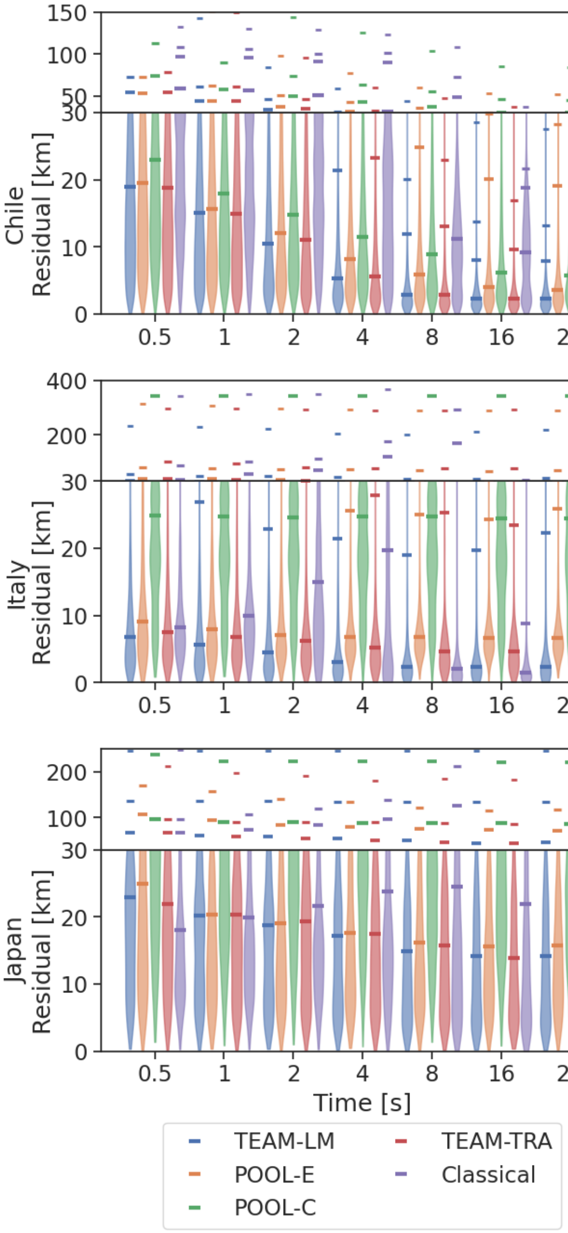


Figure 5.6: Violin plots and error quantiles of the distributions of the epicentral errors for TEAM-LM, the pooling baseline with position embeddings (POOL-E), the pooling baseline with concatenated position (POOL-C), TEAM-LM with transfer learning (TEAM-TRA), and a classical baseline. Vertical lines mark the 50^{th}, 90^{th}, 95^{th} and 99^{th} error percentiles, with smaller markers indicating higher quantiles. The time indicates the time since the first P arrival at any station. We compute errors based on the mean location predictions. A similar plot for hypocentral errors is available in Figure D.4.

In addition to the pick quality, the assumption of a 1D velocity model for NonLinLoc introduces a systematic error into the localisation, in particular for the subduction regions in Japan and Chile where the 3D structure deviates considerably from the 1D model. Because of these limitations, the classical baseline could be improved by employing more proficient pickers and fine-tuned velocity models. Nonetheless, in particular the results from Chile, where the classical baseline has access to high-quality P picks, suggest that TEAM-LM can, given sufficient training data, outperform classical real-time localisation algorithms.

For magnitude estimation no consistent performance differences between the baseline approach with position embeddings and the approach with concatenated coordinates, as originally proposed by van den Ende and Ampuero [2020], are visible. In contrast, for location estimation, the approach with embeddings consistently outperforms the approach with concatenated coordinates. The absolute performance gains between the baseline with concatenation and the baseline with embeddings is even higher than the gains from adding the transformer to the embedding model. We speculate that the positional embeddings might show better performance because they explicitly encode information on how to interpolate between locations at different scales, enabling improved exploitation of the information from stations with few or no training examples. This is more important for location estimation, where an explicit notion of relative position is required. In contrast, magnitude estimation can use further information, like frequency content, which is less position-dependent.

5.2.4 Transfer learning

A common strategy for mitigating data sparsity is the injection of additional information from related datasets through transfer learning [Pan and Yang, 2009], in our use case, waveforms from other source regions. This way the model is supposed to be taught the properties of earthquakes that are consistent across regions, e.g., attenuation due to geometric spreading or the magnitude dependence of source spectra. Note that a similar knowledge transfer implicitly is part of the classical baseline, as it was calibrated using records from multiple regions.

Here, we conduct a transfer learning experiment inspired by the transfer learning used for TEAM. We first train a model jointly on all datasets and then fine-tune it to each of the target datasets. This way, the model has more training examples, which is of special relevance for the rare large events but still is adapted specifically to the target dataset. As the Japan and Italy datasets contain acceleration traces, while the Chile dataset contains velocity traces, we first integrate the Japan and Italy waveforms to obtain velocity traces. This does not have a significant impact on the model performance, as visible in the full results tables.

Transfer learning reduces the saturation for large magnitudes (Figure 5.5, right column). For Italy, the saturation is eliminated. For Chile, while the largest magnitudes are still underestimated, we see a lower level of underestimation than without transfer learning. Results for Japan for the largest events show nearly no difference, which is expected as the Japan dataset contains the majority of large events and therefore does not gain significant additional high-magnitude training examples using transfer learning. The positive impact of transfer learning is also reflected in the lower RMSE for large and intermediate events for Italy and Chile (Figure 5.4). These results do not only offer a way of mitigating saturation for large events but also represent further evidence for data sparsity as the reason for the underestimation.

We tried the same transfer learning scheme for mitigating mislocations (Figure 5.6). For this experiment, we shifted the coordinates of stations and events such that the datasets spatially overlap. We note that this shifting is not expected to have any influence on the single dataset performance, as the relative locations of events and stations within a dataset stay unchanged and nowhere the model uses absolute locations. The transfer learning approach is reasonable, as mislocations might result from data sparsity, similarly to the underestimation of large magnitudes. However, none of the models shows significantly better performance than the original models, and in some instances, performance even degrades. We conducted additional experiments where shifts were applied separately for each event, but observed even worse performance.

We hypothesise that this behaviour indicates that the TEAM-LM localisation does not primarily rely on travel time analysis, but rather employs some form of fingerprinting of earthquakes. These fingerprints could be specific scattering patterns for certain source regions and receivers. Note that similar fingerprints are exploited in the traditional template matching approaches [e.g., Shelly et al., 2007]. While the travel time analysis should be mostly invariant to shifts and therefore be transferable between datasets, the fingerprinting is not invariant to shifts. This would also explain why the transfer learning, where all training samples were already in the pretraining dataset and therefore their fingerprints could be extracted, outperforms the shifting of single events, where fingerprints do not relate to earthquake locations. Similar fingerprinting is presumably also used by other deep learning methods for location estimation, e.g., by Kriegerowski et al. [2019] or by Perol et al. [2018]. However, further experiments are required to prove this hypothesis.

5.3 Discussion

5.3.1 Multi-task learning

While transfer learning is one option to improve model performance in face of data sparsity by incorporating further information, other approaches exist. One common method is multi-task learning [Ruder, 2017], i.e., having a network with multiple outputs for different objectives and training it simultaneously on all objectives. This approach has previously been employed for seismic source characterisation [Lomax et al., 2019], but without an empirical analysis on the specific effects of multi-task learning.

We perform an experiment, in which we train TEAM-LM to predict magnitude and location concurrently. The feature extraction and the transformer parts are shared and only the final MLPs and the mixture density networks are specific to the task. This method is known as hard parameter sharing. The intuition is that the individual tasks share some similarity, e.g., in our case the correct estimation of the magnitude likely requires an assessment of the attenuation and geometric spreading of the waves and therefore some understanding of the source location. This similarity is then expected to drive the model towards learning a solution for the problem that is more general, rather than specific to the training data. The reduced number of free parameters implied by hard parameter sharing is also expected to improve the generality of the derived model if the remaining degrees of freedom are still sufficient to extract the relevant information from the training data for each sub-task.

Unfortunately, we actually experience a moderate degradation of performance for either location or magnitude in any dataset when following a multi-task learning strategy (see full results tables in the supplement of Münchmeyer et al. [2021a]). The RMSE of the mean epicenter estimate increases by at least one third for all times and datasets, and the RMSE for magnitude stays nearly unchanged for the Chile and Japan datasets, but increases by ∼20% for the Italy dataset. Our results, therefore, exhibit a case of negative transfer.

While it is generally not known, under which circumstances multi-task learning shows positive or negative influence [Ruder, 2017], a negative transfer usually seems to be caused by insufficiently related tasks. In our case, we suspect that while the tasks are related in a sense of the underlying physics, the training dataset is large enough that similarities relevant for both tasks can be learned already from a single objective. At the same time, the particularities of the two objectives can be learned less well. Furthermore, we earlier discussed that both magnitude and location might not actually use travel time or attenuation based approaches, but rather frequency characteristics for magnitude and a fingerprinting scheme for location. These approaches would be less transferable between the two tasks. We conclude that hard parameter sharing does not improve magnitude and location estimation. Future work is required to see if other multi-task learning schemes can be applied beneficially.

5.3.2 Location outlier analysis

As all location error distributions are heavy-tailed, we visually inspect the largest deviations between predicted and catalog locations to understand the behaviour of the localisation mechanism of TEAM-LM. We base this analysis on the Chile dataset (Figure 5.7), as it has generally the best location estimation performance, but observations are similar for the other datasets (Figures D.5 and D.6).

Nearly all mislocated events are outside the seismic network and location predictions are generally biased towards the network. This matches the expected errors for traditional localisation algorithms. In contrast to traditional algorithms, events are not only predicted to be closer to the network but they are also predicted as lying in regions with a higher event density in the training set (Figure 5.7, inset). This suggests that not enough similar events were included in the training dataset. Similarly, Kriegerowski et al. [2019] observed a clustering tendency when predicting the location of swarm earthquakes with deep learning.

We investigated two subgroups of mislocated events: the Iquique sequence, consisting of the Iquique mainshock, foreshocks and aftershocks, and mine blasts. The Iquique sequence is visible in the North-Western part of the study area. All events are predicted approximately 0.5° too far east. The area is both outside the seismic network and has no events in the training set. This systematic mislocation may pose a serious threat in applications, such as early warning, when confronted with a major change in the seismicity pattern, as is common in the wake of major earthquakes or during sudden swarm activity, typical periods of heightened seismic hazard.

For mine blasts, we see one mine in the North-East and one in the South-West (marked by red circles in Figure 5.7). While all events are located close by, the locations are both systematically mispredicted in the direction of the network and exhibit scatter. Mine-blasts show a generally lower location quality in the test set. While they make up only ∼1.8% of the test set, they make up 8% of the top 500 mislocated events. This is surprising as they occur not only in the test set but also in similar quantities in the training set. We, therefore, suspect that the difficulties are caused by the strongly different waveforms of mine blasts compared to earthquakes. One waveform of each, a mine blast and an earthquake, recorded at a similar distances are shown as an inset in Figure 5.7. While for the earthquake both a P and an S wave are visible, the S wave can not be identified for the mine blast. In addition, the mine blast exhibits a strong surface wave, which is not

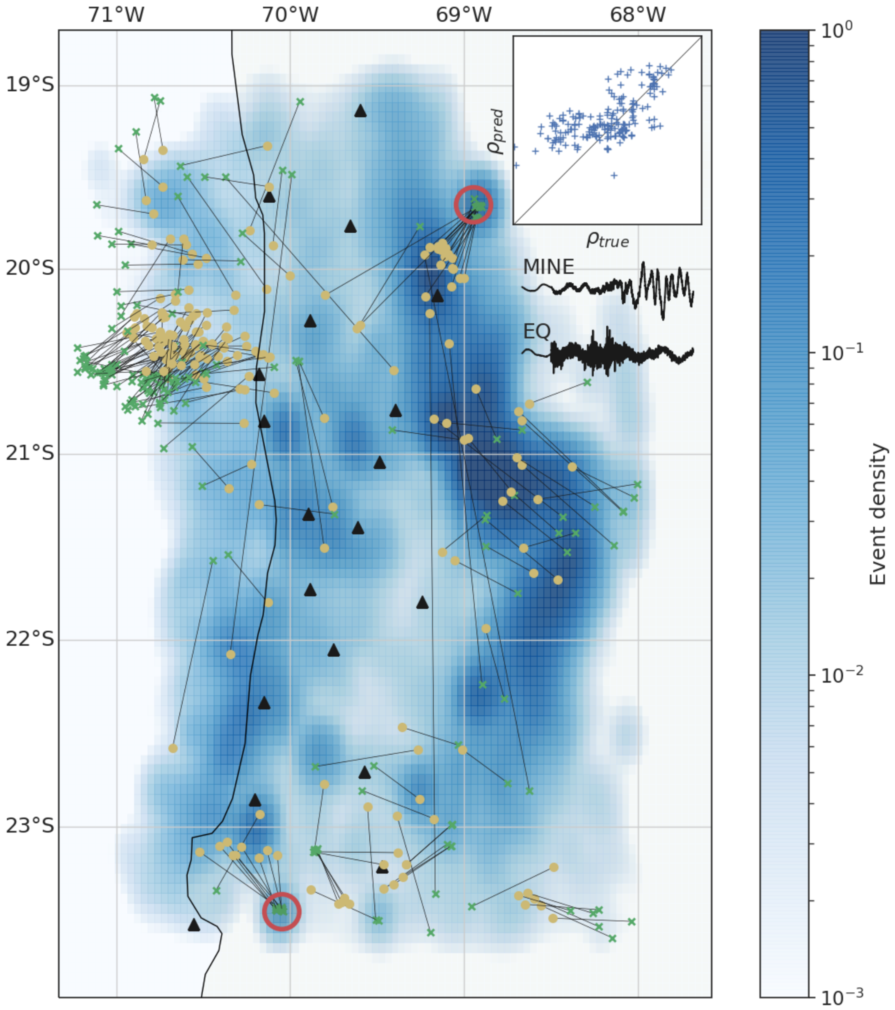


Figure 5.7: The 200 events with the highest location errors in the Chile dataset overlayed on top of the spatial event density in the training dataset. The location estimations use 16 s of data. Each event is denoted by a yellow dot for the estimated location, a green cross for the true location and a line connecting both. Stations are shown by black triangles. The event density is calculated using a Gaussian kernel density estimation and does not take into account the event depth. The inset shows the event density at the true event location in comparison to the event density at the predicted event location for the 200 events. Red circles mark locations of mine blast events. The inset waveforms show one example of a waveform from a mine blast (top) and an example waveform of an earthquake (bottom, 26 km depth) of similar magnitude ($M_A = 2.5$) at a similar distance (60 km) on the transverse component. Similar plots for Italy and Japan can be found in Figures D.5 and D.6.

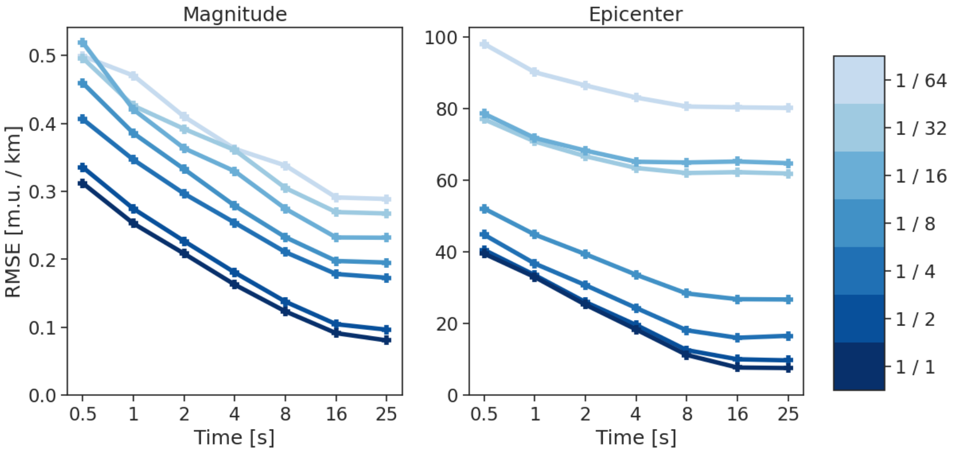


Figure 5.8: RMSE for magnitude and epicentral location at different times for models trained on differently sized subsets of the training set in Chile. The line colour encodes the fraction of the training and validation set used in training. All models were evaluated on the full Chilean test set. We note that the variance of the curves with fewer data is higher, due to the increased stochasticity from model training and initialisation.

visible for the earthquake. The algorithm therefore can not use the same features as for earthquakes to constrain the distance to a mine blast event.

5.3.3 The impact of dataset size and composition

Our analysis so far showed the importance of the amount of training data qualitatively. To quantify the impact of data availability on magnitude and location estimation, we trained models using only fractions of the training and validation data (Figure 5.8). We use the Chile dataset for this analysis, as it contains by far the most events. We subsample the events by only using each k^{th} event in chronological order, with $k = 2, 4, 8, 16, 32, 64$. This strategy approximately maintains the magnitude and location distribution of the full set. We point out, that TEAM-LM only uses information of the event under consideration and does not take the events before or afterwards into account. Therefore, the 'gaps' between events introduced by the subsampling do not negatively influence TEAM-LM.

For all times after the first P arrival, we see a clear increase in the RMSE for magnitude when reducing the number of training samples. While the impact of reducing the dataset by half is relatively small, using only a quarter of the data already leads to a twofold increase in RMSE at late times. Even more relevant in an early warning context, a fourfold smaller dataset results in an approximately fourfold increase in the time needed to reach the same precision as with the full data. This relationship seems to hold approximately across all subsampled datasets: reducing the dataset k fold increases the time to reach a certain precision by a factor of k.

We make three further observations by comparing the predictions to the true values (Figure D.7). First, for nearly all models the RMSE changes only marginally between 16 s and 25 s, but the RMSE of this plateau increases significantly with a decreasing number of training events. Second, the lower the amount of training data, the lower is the saturation threshold above which all events are strongly underestimated. In addition, for 1/32 and 1/64 of the full dataset, an 'inverse saturation' effect is noticeable for the smallest magnitudes. Third, while for the full dataset and the largest subsets all large events are estimated at approximately the saturation threshold if at most one quarter of the training data is used, the largest events even fall significantly below the saturation threshold. For the models trained on the smallest subsets (1/8 to 1/64), the higher the true magnitude the lower the predicted magnitude becomes. We assume that the larger the event is, the further away from the training distribution it is and therefore it is estimated approximately at the densest region of the training label distribution. These observations support the hypothesis that underestimations of large magnitudes for the full dataset are caused primarily by insufficient training data.

While the RMSE for epicenter estimation shows a similar behaviour as the RMSE for magnitude, there are subtle differences. If the amount of training data is halved, the performance only degrades mildly and only at later times. However, the performance degradation is much more severe than for magnitude if only a quarter or less of the training data are available. This demonstrates that location estimation with high accuracy requires catalogs with a high event density.

The strong degradation further suggests insights into the inner working of TEAM-LM. Classically, localisation should be a task where interpolation leads to good results, i.e., the travel times for an event in the middle of two others should be approximately the average between the travel times for the other events. Following this argument, if the network would be able to use interpolation, it should not suffer such significant degradation when faced with fewer data. This provides further evidence that the network does not actually learn some form of triangulation, but only an elaborate fingerprinting scheme, backing the finding from the qualitative analysis of location errors.

5.3.4 Training TEAM-LM on large events only

Often, large events are of the greatest concern, and as discussed, generally showed poorer performance because they are not well represented in the training data. It, therefore, appears plausible that a model optimised for large events might perform better than a model trained on both large and small events. To test this hypothesis, we employed an extreme version of the upscaling strategy by training a set of models only on large events, which might avoid tuning the model to seemingly irrelevant small events. In fact, these models perform significantly worse than the models trained on the full dataset, even for the large events [Münchmeyer et al., 2021a, Supplementary Tables SM5 to SM11]. Therefore, even if the events of interest are only the large ones, training on more complete catalogs is still beneficial, presumably by giving the network more comprehensive information on the regional propagation characteristics and possibly site effects.

5.3.5 Interpretation of the predicted uncertainties

So far, we only analysed the mean predictions of TEAM-LM. As for many application scenarios, for example, early warning, quantified uncertainties are required, TEAM-LM outputs not only these mean predictions but a probability density. Figure 5.9 shows the development of magnitude uncertainties for events from different magnitude classes in the Chile dataset. The left panel shows the absolute predictions, while the right panel shows the difference between prediction and true magnitude and focuses on the first 2 s. As we average over multiple events, each set of lines can be seen as a prototype event of a certain magnitude.

For all magnitude classes, the estimation shows a sharp jump at $t = 0$, followed by a slow convergence to the final magnitude estimate. We suspect that the magnitude estimation always converges from below, as due to the Gutenberg-Richter distribution, lower magnitudes are more likely *a priori*. The uncertainties are largest directly after

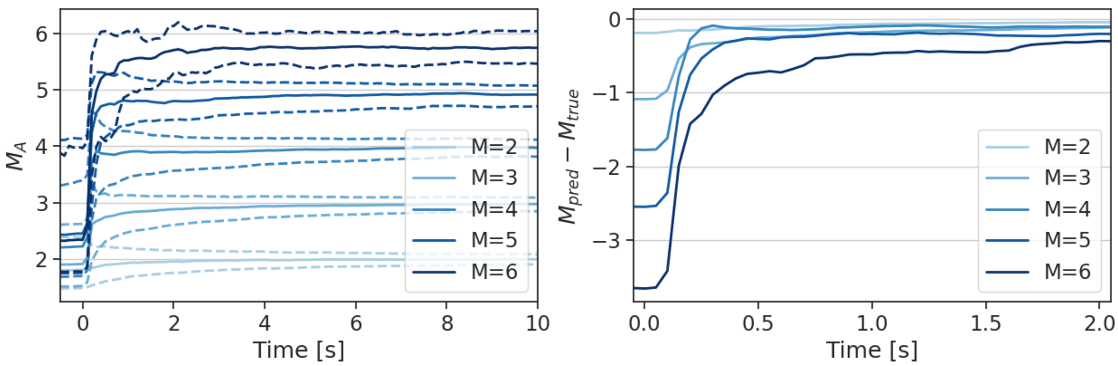


Figure 5.9: Magnitude predictions and uncertainties in the Chile dataset as a function of time since the first P arrival. Solid lines indicate median predictions, while dashed lines (left panel only) show the 20th and 80th quantiles of the prediction. The left panel shows the predictions, while the right panel shows the differences between the predicted and true magnitude. The right panel is focused on a shorter time frame to show the early prediction development in more detail. In both plots, each colour represents a different magnitude bucket. For each magnitude bucket, we sampled 1,000 events around this magnitude and combined their predictions. If less than 1,000 events were available within ±0.5 m.u. of the bucket centre, we use all events within this range. We only use events from the test set. To ensure that the actual uncertainty distribution is visualised, rather than the distribution of magnitudes around the bucket centre, each prediction is shifted by the magnitude difference between bucket centre and catalog magnitude.

$t = 0$ and subsequently decrease, with the highest uncertainties for the largest events. As we do not use transfer learning in this approach, there is a consistent underestimation of the largest magnitude events, visible from the incorrect median predictions for magnitudes 5 and 6. We note that the predictions for magnitude 4 converge slightly faster than the ones for magnitude 3, while in all other cases the magnitude convergence is faster the smaller the events are. We suspect that this is caused by the accuracy of the magnitude estimation being driven by both the number of available events and by the signal to noise ratio. While magnitude 4 events have significantly less training data than magnitude 3 events, they have a better signal to noise ratio, which could explain their more accurate early predictions.

While the Gaussian mixture model is designed to output uncertainties, it cannot be assumed that the predicted uncertainties are indeed well-calibrated, i.e., that they match the real error distribution. Having well-calibrated uncertainties is crucial for downstream tasks that rely on the uncertainties. Neural networks trained with a log-likelihood loss generally tend to be overconfident [Snoek et al., 2019, Guo et al., 2017], i.e., underestimate the uncertainties. This overconfidence is caused by the strong overparametrisation of neural network models. To assess the quality of our uncertainty estimations for magnitude, we assess the empirical quantiles of the true values relative to the predictions. For a prediction with cumulative distribution function F^i_{pred}, the empirical quantile can be calculated as $u_i = F^i_{pred}(y^i_{true})$. For a perfectly calibrated model, u_i should be distributed uniformly in $[0, 1]$, as discussed in Chapter 2.4.3.

Figure 5.10 shows the P-P plots of u in comparison to a uniform distribution. For all datasets and all times, the model is significantly miscalibrated, as estimated using Kolmogorov-Smirnov test statistics (see also Appendix D.2). Miscalibration is considerably stronger for Italy and Japan than for Chile. More precisely, the model is always

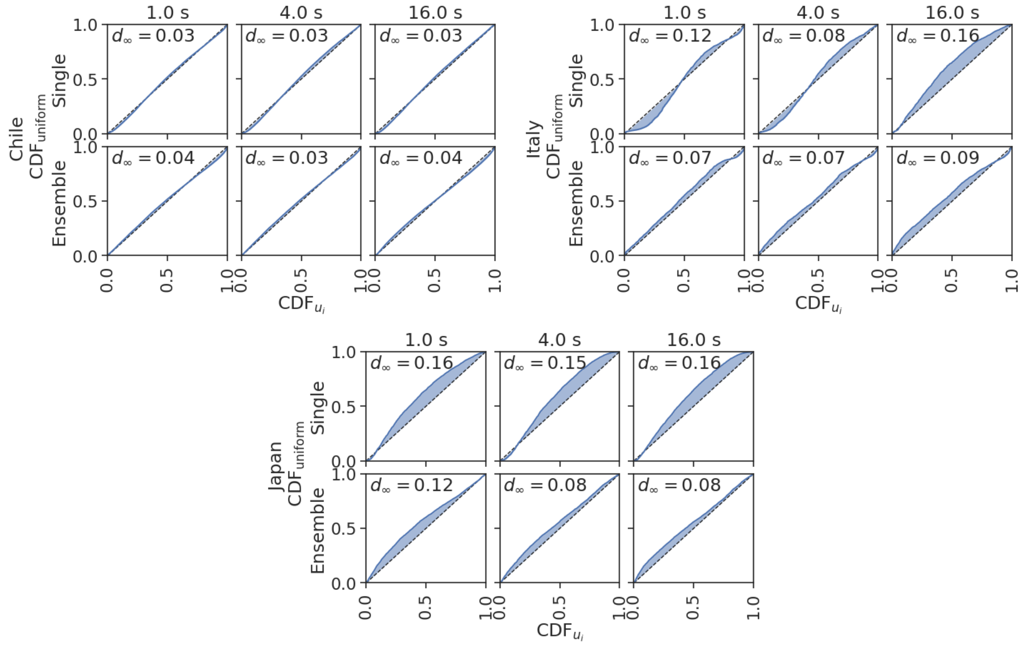


Figure 5.10: P-P plots of the CDFs of the empirical quantile of the magnitude predictions compared to the expected uniform distribution. The P-P plot shows $(\mathrm{CDF}_{u_i}(z), \mathrm{CDF}_{\mathrm{uniform}}(z))$ for $z \in [0, 1]$. The expected uniform distribution is shown as the diagonal line, the misfit is indicated by the shaded area. The value in the upper corner provides d_∞, the maximum distance between the diagonal and the observed CDF. d_∞ can be interpreted as the test statistic for a Kolmogorov-Smirnov test. Curves consistently above the diagonal indicate a bias to underestimation, and below the diagonal to overestimation. Sigmoidal curves indicate over-confidence, mirrored sigmoids indicate under-confidence. See Appendix D.2 for a further discussion of the plotting methodology and its connection to the Kolmogorov-Smirnov test.

overconfident, i.e., estimates narrower confidence bands than the observed errors. Further, in particular at later times, the model is biased towards underestimating the magnitudes. This is least visible for Chile. We speculate that this is a result of the large training dataset for Chile, which ensures that for most events the density of training events in their magnitude range is high.

To mitigate the miscalibration, we trained ensembles [Hansen and Salamon, 1990], a classical method to improve calibration. Instead of training a single neural network, a set of n neural networks, in our case $n = 10$, are trained, which all have the same structure, but different initialisation and batching in training. The networks, therefore, represent a sample of size n from the posterior distribution of the model parameters given the training data. For Italy and Japan, this improves calibration considerably (Figure 5.10). For Chile, the ensemble model, in contrast to the single model, exhibits underconfidence, i.e., estimates too broad uncertainty bands. While the ensembles improve the calibration, the distribution of u_i still deviates highly significantly from a uniform distribution for all datasets (Kolmogorov-Smirnov test with $p \ll 10^{-5}$).

To evaluate the location uncertainties qualitatively, we plot confidence ellipses for a set of events in Chile (Figure 5.11). Again we compare the predictions from a single model to the predictions of an ensemble. At early times, the uncertainty regions mirror the seismicity around the station with the first arrival, showing that the model correctly

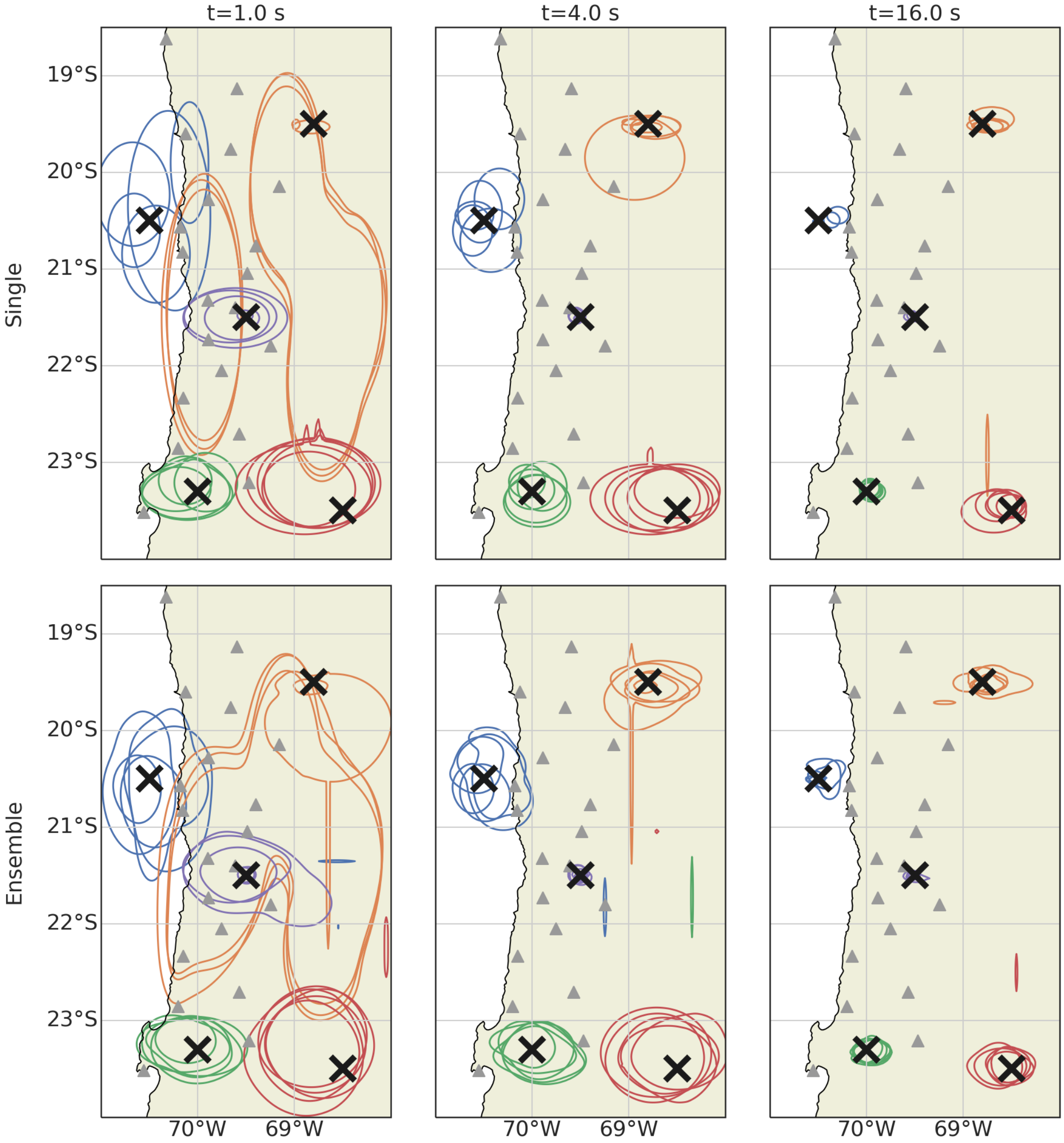


Figure 5.11: The figure shows 90% confidence areas for example events around 5 locations. For each location, the 5 closest events are shown. Confidence areas belonging to the same location are visualised using the same colour. Confidence areas were chosen as curves of constant likelihood, such that the probability mass above the likelihood equals 0.9. To visualise the result in 2D we marginalise out the depth. Triangles denote station locations for orientation. The top row plots show results from a single model, while the bottom row plots show results from an ensemble of 10 models.

learned the prior distribution. Uncertainty ellipses at late times approximately match the expected uncertainty ellipses for classical methods, i.e., they are small and fairly round for events inside the seismic network, where there is good azimuthal coverage, and larger and elliptical for events outside the network. Location uncertainties are not symmetric around the mean prediction but show a higher likelihood towards the network than further outwards. Location errors for the ensemble model are more smooth than from the single model but show the same features. The uncertainty ellipses are slightly larger, suggesting that the single model is again overconfident.

In addition to improving calibration, ensembles also lead to slight improvements regarding the accuracy of the mean predictions [Münchmeyer et al., 2021a, Supplementary tables SM 5 to SM 11]. Improvements in terms of magnitude RMSE range up to $\sim 10\%$, for epicentral location error up to $\sim 20\%$. Due to the high computational demand of training ensembles, all other results reported in this chapter are calculated without ensembling. We note that in addition to ensembles a variety of methods have been developed to improve calibration or obtain calibrated uncertainties [Snoek et al., 2019]. One of these methods, Monte-Carlo Dropout, has already been employed in the context of fast assessment by van den Ende and Ampuero [2020].

5.4 Conclusion

In this chapter, we adapted TEAM to build TEAM-LM, a real-time earthquake source characterisation model and used it to study the pitfalls and particularities of deep learning for this task. We showed that TEAM-LM achieves state-of-the-art performance in magnitude estimation, outperforming both a classical baseline and a deep learning baseline. Given sufficiently large catalogs, magnitudes can be assessed with a standard deviation of 0.2 m.u. within 2 s of the first P arrival and a standard deviation of 0.07 m.u. within the first 25 s. For location estimation, TEAM-LM outperforms a state-of-the-art deep learning baseline and compares favourably with a classical baseline.

Our analysis showed that the quality of model predictions depends crucially on the training data. While performance with abundant data is excellent, in face of data sparsity, prediction quality degrades significantly. For magnitude estimation, this effect results in the underestimation of large magnitude events; for location estimation, events in regions with few or no training events tend to be mislocated severely. This results in a heavy-tailed error distribution for location estimation. Large deviations in both magnitude and location estimation can have a significant impact in application scenarios, e.g., for early warning where large magnitudes are of the biggest interest.

As in the previous chapter, relevant contributions to the topic have been published since the original publication of this chapter as [Münchmeyer et al., 2021a]. Again, we point out the study of Zhang et al. [2021] for real-time magnitude and location estimation, mentioned in the previous chapter. Zhang et al. [2021] address the issue of systematic underestimation of large magnitude through a splitting approach. Instead of directly predicting magnitude, they decompose the magnitude prediction into the peak displacement, which can be extracted directly from the waveforms, and an attenuation term that is output by the network. This way, they observe no saturation, even for the largest events. However, we tried a similar approach for Northern Chile, where magnitudes are up to 2 m.u. larger than in Italy, with mixed results [Hauffe, 2021]. This suggests that the approach might not be applicable to very large events. In this chapter, we used a fairly simple scheme for transfer learning. [Jozinović et al., 2022] conducted a more extensive study, comparing different transfer learning schemes for ground motion estimation. Their applicability to magnitude and location estimation still needs to be evaluated.

A key conclusion of this chapter is that regional datasets, even if very complete, are insufficient to assess rupture predictability, at least with the current methods (see Figure 5.5 and Chapter 5.2.2). There are several possibilities to address this limitation. First, the employed models can be improved, in particular by incorporating physical knowledge into the model [Raissi et al., 2019]. Within this thesis, we do not pursue this approach, but we will discuss its potential and its challenges in our conclusion (Chapter 7.3). Second, the training data selection can be modified to include more events. As we already included two of the most seismically active regions worldwide in this chapter, Northern Chile and Japan, this is only possible when studying data from diverse regions together. Third, the models can be applied to a simpler task, thereby requiring less training data. For example, the models can be provided with preprocessed information about the events obtained using physical knowledge, such as source time functions, instead of raw waveforms. We will explore the second and the third approach in detail in the next chapter.

Resource availability

The code for TEAM-LM is available at `https://doi.org/10.5880/10.1093/gji/ggaa609` and `https://github.com/yetinam/TEAM`. We made the Italy dataset publicly available at `https://doi.org/10.5880/GFZ.2.4.2020.004`. We made the Chile dataset publicly available at `https://doi.org/10.5880/GFZ.2.4.2021.002`. Due to licensing restrictions, we are not able to redistribute the Japan dataset, but instructions and code to convert it from the source files are available in the TEAM software repository.

6 A probabilistic view on rupture predictability

In the previous chapters, we focused primarily on developing models for the fast and accurate assessment of earthquakes. In this chapter, we put the spotlight on rupture predictability.[26] First, we study previous results regarding rupture predictability and show that these results are inconclusive. The majority of these results uses a deterministic view of rupture predictability. However, in this chapter we show that such a deterministic view is insufficient to describe possible modes of rupture predictability. Therefore, in a second step, we develop a principled, probabilistic formulation of rupture predictability as a more expressive alternative. We then show how the conditional distributions in this formulation can be estimated from data using neural networks and variational inference.

Subsequently, we apply the framework to two types of observables. First, we study rupture predictability from teleseismic waveforms, building upon the TEAM-LM method from Chapter 5. Second, we complement this analysis with a study of rupture predictability based on moment rate functions, for which we develop a similar real-time magnitude estimation model. For both types of observables, teleseismic waveforms and moment rate functions, we find no indication of early rupture predictability. The final magnitude of an earthquake can only be assessed after the peak of the moment rate function, usually around half of the event duration. Even then, it is impossible to foresee further rupturing asperities.

6.1 The deterministic view on rupture predictability

To contextualise and motivate our approach, we start with an overview of different models for rupture predictability, both models implying predictability and models not implying predictability. A common theory implying predictability is the preslip model [Ellsworth and Beroza, 1995], in which failure starts aseismically until the process reaches a critical size and becomes unstable. Here, the final moment of the earthquake might be derivable at the event onset time from properties of the nucleation zone, i.e., its size or the amount of slip. Other models also suggest early predictability, but only after several seconds. For example, Melgar and Hayes [2017] argue that ruptures of large events propagate as self-healing pulses and that the pulse properties allow identification of very large events after ∼15 s. Support for such theories has been provided by the analysis of, e.g., waveform onsets [Ellsworth and Beroza, 1995], moment rate functions [Danré et al., 2019], and early ground motion parameters [Colombelli et al., 2020].

The opposing hypothesis, often termed cascade model [Ellsworth and Beroza, 1995], suggests a universal initiation behaviour: small and large earthquakes start identically and are differentiated only after the peak moment release, which occurs approximately at half of the rupture duration. Rupture evolution is controlled by heterogeneous local conditions, such as pre-event stress distribution or the presence of mechanical barriers. Studies supporting this theory also analysed properties like moment rate functions [Meier et al., 2017], waveform onsets [Ide, 2019], or peak displacement [Trugman et al., 2019].

While reaching contradicting conclusions, predictability studies often follow the same principle: analysing correspondences between earthquake size and real-time observables [Ellsworth and Beroza, 1995, Meier et al., 2017, Danré et al., 2019, Ide, 2019, Trugman

[26] This chapter is based on a manuscript currently under review. Compared to the manuscript, the Introduction and Conclusion of this chapter have been modified to highlight the context of the chapter within this thesis. Furthermore, we added a discussion on the estimation error and moved several figures and the section on comparison to related work from the supplementary material into the main text. Minor modifications were introduced to the remaining text and figures.

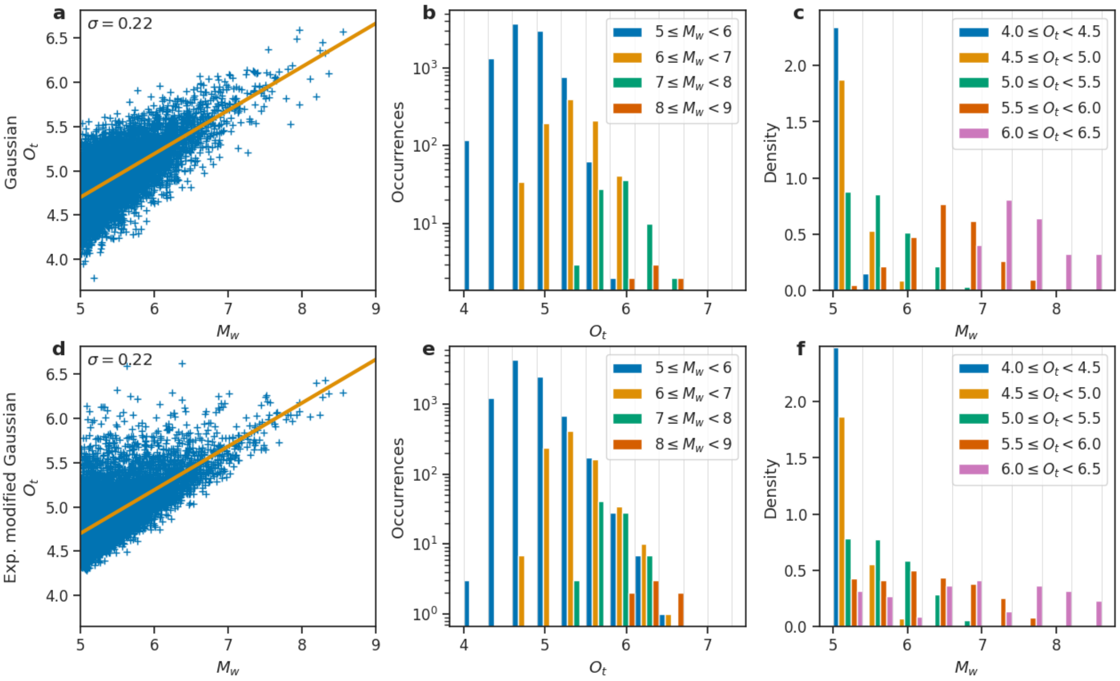


Figure 6.1: Synthetic samples of an arbitrary scalar observable O_t and magnitude M_w assuming a linear connection with Gaussian error (**a-c**) or with exponentially modified Gaussian error (**d-f**), i.e., the sum of a Gaussian and an exponential random variable. Both observables have the same linear connection and standard deviation. **a** and **d** show scatter plots of O_t and M_w. **b** and **e** show histograms of O_t for M_w bins with log-scaled y axis. **c** and **f** show histograms of M_w for O_t bins which are normed to represent densities. For both cases, events with large magnitudes cause large observables. However, only in the Gaussian case does a similar connection hold for small events causing small observables, i.e., in the second case, small events can cause large observables as well. The observable distributions for different magnitudes are mostly distinct in the first case, but overlap strongly for the second case. The magnitude distributions for different observables are mostly distinct for the first case while for the second case, small observables only give an upper bound on the magnitude, i.e., small observables rule out large magnitudes, but large observables do not imply large magnitudes. M_w samples were generated according to a Gutenberg-Richter distribution with $b = 1$. O_t samples were generated using the linear connection and random samples from the error distribution.

et al., 2019, Colombelli et al., 2020]. Earthquake size is commonly quantified by seismic moment/moment magnitude, as large, high-quality catalogs thereof are openly available [Ekström et al., 2012]. A common practice is calculating parametric fits between magnitude and observables, and assessing at which time they become significant using standard deviations [Olson and Allen, 2005, Zollo et al., 2006, Noda and Ellsworth, 2016, Meier et al., 2017, Melgar and Hayes, 2017, Danré et al., 2019, Colombelli et al., 2020]. However, this point-estimator approach hides the residual distribution and thereby potentially obscures distinct modes of rupture predictability, especially when distributions are non-Gaussian.

To illustrate the importance of this restriction, we created a synthetic toy example (Figure 6.1). The example shows two sets of observables with an identical linear fit and standard deviation. However, the examples differ in their residual distributions. The first

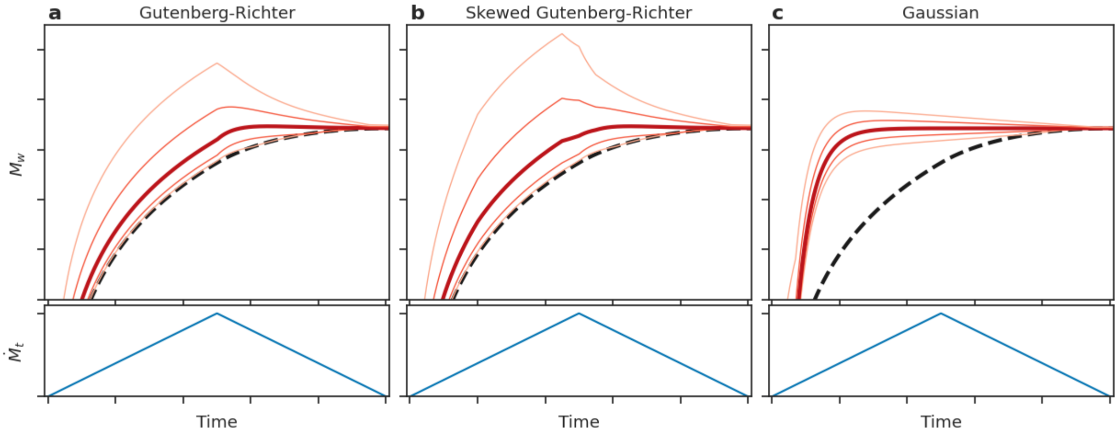


Figure 6.2: Conditional magnitude distribution development for three different predictability models: Gutenberg-Richter (GR) (not predictable during the growth phase), skewed GR (total magnitude not point-predictable, but information gain compared to the prior already during the growth phase of the rupture) and Gaussian (predictable). The panels in the top row show the predictive distributions by their 0.05, 0.2, 0.5, 0.8, 0.95 quantiles over time. The cumulative moment release is indicated by the dashed black line. The bottom plots show the moment rate $\dot{M}_t$ over time. For each model, we use the same hypothetical event with a prototypical triangular moment rate function. The prototypical moment rate function is meant to represent the first-order moment release history; for predictable models to be viable, other features are required, such as further observables or second-order features of the moment rate function. **a** In the GR case, the prediction follows a GR distribution above the moment released so far. Only after the peak moment release, the prediction quickly transforms into a Gaussian, although with a decreasing GR portion that relates to the possibility of future asperities. **b** The skewed GR case behaves similarly to the GR case, but the distribution is skewed towards higher magnitudes, i.e., from early on, it is more likely for the event to become large. **c** For the Gaussian case, the magnitude can be determined early on with a small error that decreases further over time. No quantitative x and y labels are provided to highlight the prototypical character of the figure. A cross-section view of the three different options at a fixed time is shown in Figure E.1a.

example exhibits a Gaussian residual, i.e., the observables allow to predict the magnitude up to an unbiased uncertainty term. In contrast, the second example shows an exponentially modified Gaussian residual. While small observables uniquely identify small events, larger observables can result from any size of event. Pinpointing the final magnitude is impossible in this case. However, in a probabilistic sense, large observables still considerably increase the likelihood of a large event compared to the marginal Gutenberg-Richter (GR) distribution. Notably, exactly such residual distributions occur for real observables. Olson and Allen [2005] analyse the dominant period of the initial 4 s of the P wave and find such residuals (their Figure 3). Noda and Ellsworth [2016] derive an observable from the early P displacement waveform and obtain a similar residual (their Figure 5).

6.2 A probabilistic framework for rupture predictability

We argue that a rigorous, probabilistic approach can overcome the limitations of the deterministic approach discussed in the previous section. To formalise this probabilistic approach, we interpret the magnitude M of an event as a random variable and introduce a stochastic process $(O_t)_{t\in\mathbb{R}}$, the observables at time t.[27] $t = 0$ identifies the event onset. The observables $(O_t)_{t\in\mathbb{R}}$ can be any information, as long as O_t only describes the event until t, e.g., waveforms up to the P travel time plus t.

Two events with magnitudes $M_1 \neq M_2$ differ at the time t if the conditional distributions $\mathbb{P}(O_t|M_1)$ and $\mathbb{P}(O_t|M_2)$ differ. However, while describing $\mathbb{P}(O_t|M)$ for scalar O_t is feasible, it becomes intractable for higher-dimensional O_t. Furthermore, for early warning, the objective is estimating M from O_t and not vice versa. Therefore, we analyse $\mathbb{P}(M|O_t)$, directly investigating to what degree the observables constrain the magnitude. While this type of analysis has been conducted for peak ground displacement, where Meier et al. [2017] considered $\mathbb{P}(O_t|M)$ and Trugman et al. [2019] analysed both $\mathbb{P}(O_t|M)$ and $\mathbb{P}(M|O_t)$, an analysis for higher dimensional observables is still missing. This leaves many observables unexplored that might potentially contain information on future rupture development, e.g., seismic waveforms.

There are two distinct aspects of rupture predictability: (i) the future development of the current asperity and (ii) the probability of further asperities to rupture. Figure 6.2a shows an example of $\mathbb{P}(M|O_t)$ with no predictability in the growing rupture, as suggested, e.g., by Meier et al. [2017]. Before the peak moment release, the distribution equals a GR distribution with a lower bound at the currently released moment, accounting for both aspects of rupture predictability. After the peak, the distribution becomes Gaussian (i), with a decreasing GR component accounting for potential future asperities (ii). Figure 6.2b shows a skewed GR case: magnitude cannot be pinpointed, but from early on the event is more likely to become large than the marginal GR distribution. Skewed GR distributions might occur, e.g., in slip pulse models [Melgar and Hayes, 2017], where pulse properties define the likelihood of the rupture to arrest soon. Figure 6.2c shows the predictable case: magnitude can be pinpointed early and uncertainties decrease steadily, implying correct assessment of both aspects.

The different evolution of $\mathbb{P}(M|O_t)$ has consequences for early warning: a shifted tail for $\mathbb{P}(M|O_t)$ shifts the estimated distribution of ground shaking and possibly the warning decision. However, several results from previous research do not allow a clear distinction of the presented cases. For example, multiple studies [Abercrombie and Mori, 1994, Mori and Kanamori, 1996, Kilb and Gomberg, 1999, Ide, 2019] reported that for most large events, small events with similar onsets exist. While this rules out the predictable case, events might still differ strongly in their likelihood of becoming large.

For practical analysis, $\mathbb{P}(M|O_t)$ needs to be derived from observed samples

$$\{(M^i, O_t^i)\}_{i=1,\dots,n} \sim_{iid} \mathbb{P}(M, O_t) \ . \tag{6.1}$$

As a direct description is infeasible for high dimensional O_t, we propose to instead use a variational approximation $\mathbb{P}_\theta(M|O_t) \approx \mathbb{P}(M|O_t)$, i.e., approximate the true distribution with a parametrised distribution. The parameters θ are learned to fit $\mathbb{P}(M|O_t)$ using the samples $\{(M^i, O_t^i)\}_{i=1,\dots,n}$ and a proper loss function/scoring rule [Gneiting and Raftery, 2007]. Specifically, we suggest parameterising $\mathbb{P}_\theta(M|O_t)$ using neural networks with Gaussian mixture outputs [Bishop, 1994]. Both neural networks and Gaussian mixtures (Figure E.1b) have universal approximator properties, making them particularly well suited for our case [Cybenko, 1989, Bengio et al., 2017]. This enables us to obtain probabilistic magnitude estimates, while not being restricted to single dimensional observables. Notably,

[27] Throughout this analysis, we will be using moment magnitude values M_w. As both super- and subscripts will be used as indexes into the stochastic process and for different samples, we drop the w from the notation in most places.

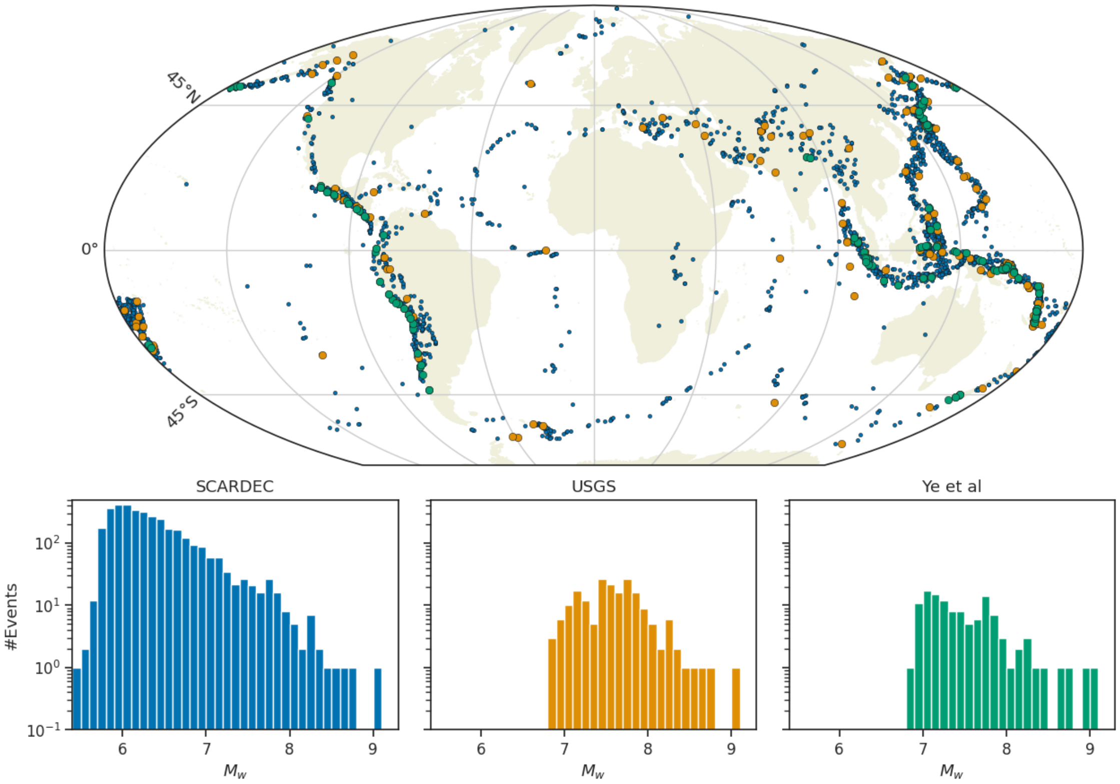


Figure 6.3: Distribution of events and histograms for magnitude distribution for the three STF datasets. The events are colour coded by their dataset. Ye et al is plotted on top of USGS, on top of SCARDEC. This might lead to a few events not being visible due to overlaps.

this approach can be applied directly to any type of observables, simply by designing an appropriate neural network.

6.3 Predictions from moment rate functions

We first apply this framework to source time functions (STFs), also known as moment rate functions, a commonly used observable in predictability studies [Meier et al., 2017, Danré et al., 2019]. For our analysis we use three STF databases: SCARDEC (3514 events, $5.4 \leq M_w \leq 9.1$) [Vallée and Douet, 2016] and those from USGS [Hayes, 2017] (190 events, $6.8 \leq M_w \leq 9.1$) and from Ye et al. [2016] (119 events, $6.8 \leq M_w \leq 9.1$). The spatial and magnitude distributions for all three datasets are shown in Figure 6.3. The three STF databases were generated using two different methodologies. SCARDEC uses a point source approximation and conducts a constrained deconvolution of body waves. In contrast to SCARDEC, USGS [Hayes, 2017] and Ye et al. [2016] calculate finite fault solutions from both body and surface waves assuming constant rupture velocity within each event. As the spatial extent of the source is modelled, the STFs from finite fault solutions generally represent more high-frequency details than the SCARDEC ones. On the other hand, the SCARDEC method applies to smaller events that can not be processed with the finite-fault inversion schemes. Further details on the methodologies of the STF datasets are provided in E.1.

As a neural network model for the prediction of total moment magnitude based on

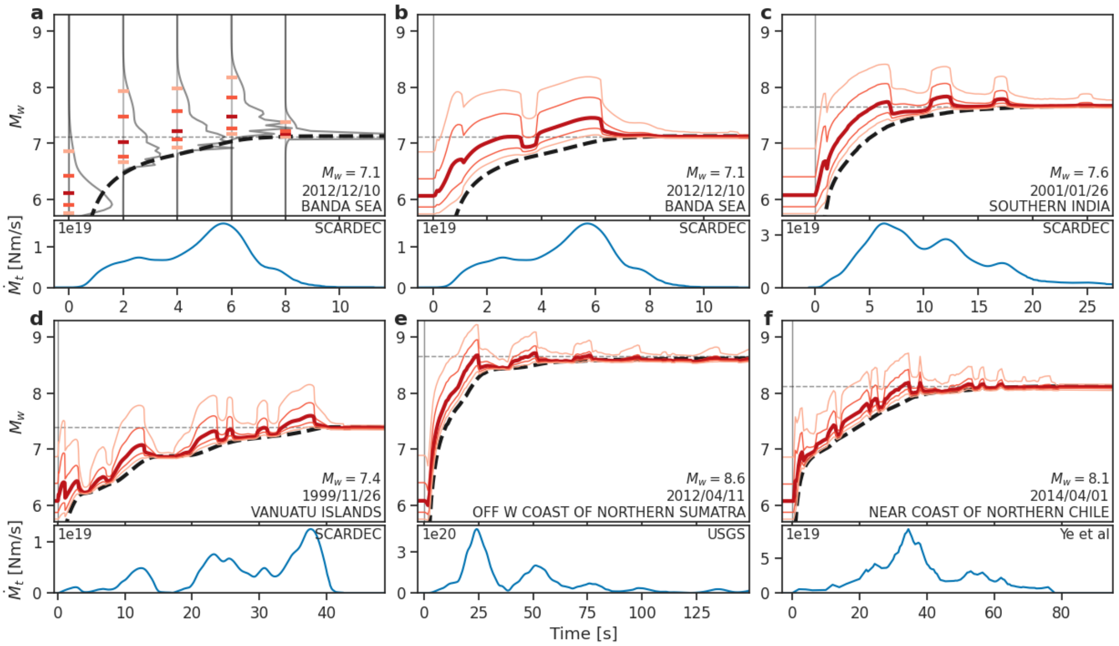


Figure 6.4: **a** Probability density functions (PDFs) calculated from the STF model just before onset, and at 2, 4, 6 and 8 s after onset. Coloured ticks on the PDFs indicate 0.05, 0.2, 0.5, 0.8, 0.95 quantiles. **b-f** Example predictions from the STF model visualised by the 0.05, 0.2, 0.5, 0.8, 0.95 quantiles over time. **b** shows the same event as **a**. The lower right gives information on the event. The black dashed line shows the magnitude equivalent to the moment released so far, i.e., the trivial lower bound. The bottom plots show the STFs used for prediction. The annotations in the upper right of these subplots indicate the STF database used.

(partial) source time functions, we use a simple multi-layer perceptron. The model has five hidden layers with 200 neurons each and ReLU activation. As input, we use five observables derived from the source time function at time t: (1) cumulative moment M_t; (2) current moment rate $\dot{M}_t$; (3) average moment rate $\frac{1}{t}M_t$; (4) peak moment rate $\max_{\tau\leq t}\dot{M}_\tau$; (5) current moment acceleration $\ddot{M}_t$. We use features instead of full STFs to avoid the danger of overfitting due to the high dimensionality of time series in contrast to the low number of training examples. Still, these features describe the STFs in sufficient detail to represent the observables considered in most previous STF-based predictability studies [Meier et al., 2017, Melgar and Hayes, 2019]. For improved learning behaviour, we log-transformed features (1) to (4) and multiplied them by 0.1. As feature (5) can take negative values as well, we transformed the feature with the function $f(x) = 0.01\,\mathrm{sign}(x)\max(0, \log(|x|/(10^{15}\,\mathrm{Nm/s^2})))$, i.e., we apply a signed and scaled log-transform. To mitigate slight differences in the onset times, we rebase the STF times such that the last sample with a moment rate below 10^{15} Nm is at $t = 0$.

As the output, we use a Gaussian mixture density network [Bishop, 1994]. The network outputs mixture weights α_i, mean values μ_i, and standard deviations σ_i. The probability density function (PDF) of the mixture is $f(x) = \sum_i \alpha_i \sigma_i^{-1}\varphi(\frac{x-\mu_i}{\sigma_i})$, where φ denotes the PDF of a standard normal random variable. For the mixture weights, we use softmax activation, for the mean values no activation function and for the standard deviation softplus activation. As we observed a mode collapse of the Gaussian mixture, i.e., all mixture components except one or two having mixture weights very close to zero, we introduce a Dirichlet prior on the mixture weights [Ormoneit and Tresp, 1995], forcing mixture weights away from zero.

We train the model on SCARDEC, as it is the largest of the datasets, with further results from models trained on the USGS datasets available in Figure E.2. For training, we use a ten-fold cross-validation scheme. We use the continuous ranked probability score as loss, as its optimisation behaviour is more favourable in face of highly skewed underlying distributions than the behaviour of log-likelihood. Further details on the training procedure are provided in Appendix E.2.

For qualitative insights into the predictions and as a basis for interpreting the average results, we visualise a few representative examples (Figure 6.4). We show their PDFs at different times (Figure 6.4a) and their quantiles (Figure 6.4b-f). In all cases, the sign of the moment acceleration largely defines the anticipated potential for growth: positive acceleration, i.e., the growth phase, indicates high growth potential, negative acceleration low potential. Furthermore, the higher the current moment release is, the higher the growth potential. This results from the STF's smoothness: at high moment rates, it will likely take longer to arrest than at low rates. Notably, the model does not predict future asperities within a multiple asperities rupture (Figure 6.4d, e); for times after the peak of the moment rate function has been passed, the model expects a steady decay. Once the moment rate approaches zero, the estimated further growth is low (e.g., Figure 6.4d at 15 s, 6.4e at 40 s). If moment release accelerates again, the model immediately expects another asperity to break and higher growth potential is inferred yet again. These effects lead to sudden changes of the PDF at local maxima and minima of the STF (e.g., Figure 6.4d at 20 s, 6.4e at 25 s).

For a systematic analysis, we average $\mathbb{P}(M|O_t)$ by magnitude buckets (Figure 6.5a-c). For the datasets using finite fault solutions (Figure 6.5b, c), during the first 2 s of the STF, the predicted distributions are mostly identical across buckets. Afterwards, the buckets split up over time: $M_w = 6.5$ to 7.0 at ~2 s, $M_w = 7.0$ to 7.5 at ~8 s, $M_w = 7.5$ to 8.0 at ~16 s, $M_w = 8.0$ to 8.5 at 25–40 s. These times match typical half-durations of events in these magnitude ranges [Gomberg et al., 2016].

SCARDEC (Figure 6.5a) shows similar splitting over time, but exhibits an apparent skew in the early predictions: higher magnitude buckets exhibit a higher likelihood for becoming large. Furthermore, lower bounds for the highest magnitude buckets (brown, purple), are higher than for the remaining buckets already after 1 s. Similarly, the SCARDEC examples in Figure 6.4b-d show high predictions within the first 2 s and abruptly fall afterwards. We attribute this apparent predictability to artefacts of the SCARDEC processing, in particular uncertainties in onset timing and the point source approximation. We further discuss this apparent predictability and its causes in Appendix E.1. This matches previous studies [Meier et al., 2021] reporting a strong correlation between early samples of SCARDEC STFs and the final magnitude. Additionally, we trained the model with the much smaller USGS dataset and evaluated the model again on all three datasets. The results lack the apparent early predictability, which is consistent with this explanation (Figure E.2).

Predictions at a fixed time t after onset describe both moment release until t and future development, with only the latter being relevant for the predictability. To isolate this aspect, we define $\mathbb{P}(M|O_{\bar{M}}) = \mathbb{P}(M|O_{t_{\bar{M}}})$, where $t_{\bar{M}} = \sup_t\{M(t) \leq \bar{M}\}$ is the time when the cumulative moment release equals $\bar{M}$. When analysing $\mathbb{P}(M|O_{\bar{M}})$, all three datasets exhibit the same trends (Figure 6.5d-f). All magnitude buckets with lower bounds at least $\bar{M} + 0.5$ show nearly identical predictions: a sharp increase in likelihood from $\bar{M}$ to $\sim \bar{M} + 0.2$ and an exponential tail. $\bar{M} + 0.2$ represents roughly twice the

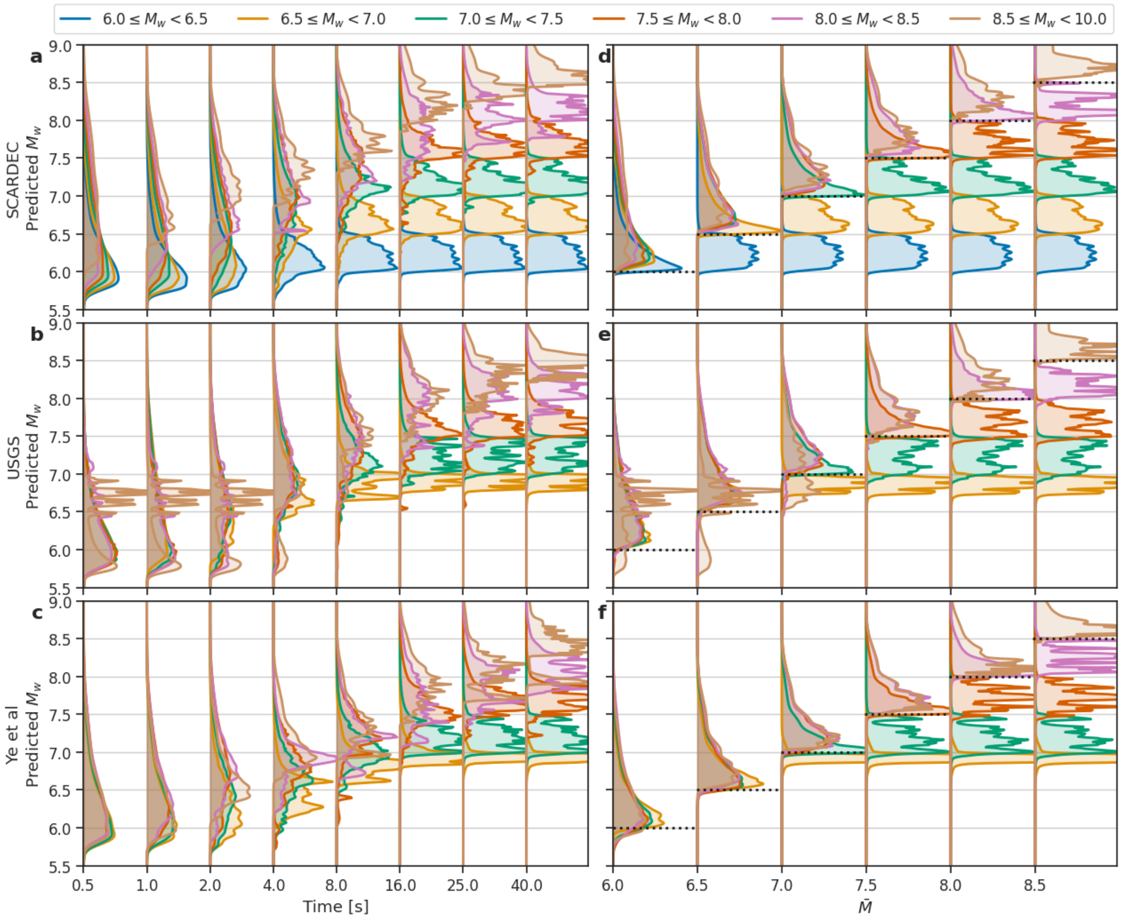


Figure 6.5: Average predicted PDFs based on STFs grouped by magnitude bin. The left column shows results at time t after onset ($\mathbb{P}(M|O_t)$), the right column after cumulative moment equals magnitude $\bar{M}$ ($\mathbb{P}(M|O_{\bar{M}})$). The model has been trained on the SCARDEC dataset and evaluated on each STF dataset. See Figure E.2 for STF results from a neural network trained with the USGS dataset. PDFs were truncated in the visualisation to avoid overlap between different times/base magnitudes. Black dotted lines in **d-f** indicate the current base magnitude.

seismic moment of $\bar{M}$ and, due to the symmetry of STFs [Meier et al., 2017], half the event duration. For buckets with lower bound equal to $\bar{M}$, peak likelihood occurs around $\bar{M}$, again with exponential tails. The decay is steeper for these buckets, as most events are already past the peak and substantial future growth can therefore only result from future asperities, but not from further growth of the current one. The results are independent of the faulting mechanism (Figures E.3, E.4, E.5). The systematic analysis, therefore, confirms the hypothesis that the final magnitude can only be assessed after the peak of the STF has been passed and that the rupture of further asperities cannot be anticipated.

6.4 Predictions from teleseismic P arrivals

STFs have limited temporal resolution, giving only a low-pass filtered view of the source process. Consequently, potential higher frequency details indicative of future rupture development might be hidden. To resolve this issue, we apply our approach to teleseismic P arrival waveforms, which in contrast to STFs contain full spectral information up to

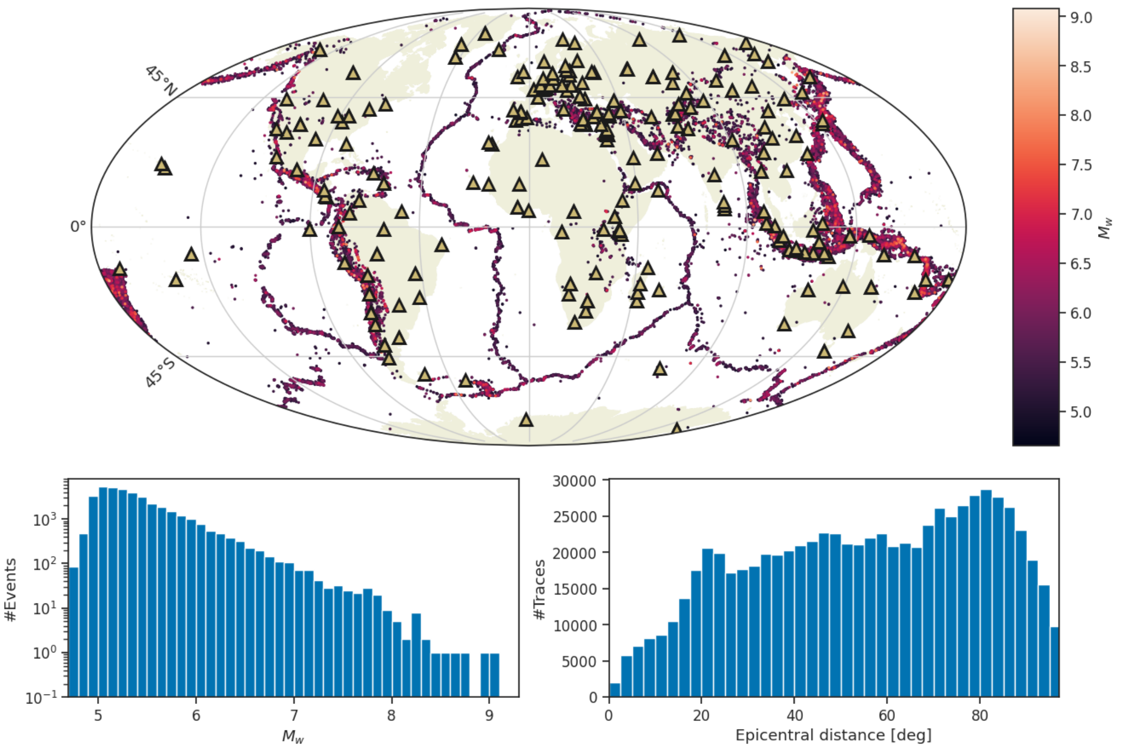


Figure 6.6: Distribution of stations and events, and histograms for magnitude and epicentral distance distributions for the teleseismic P arrival dataset. In the map, triangles denote stations and dots denote events. Events are colour-coded by magnitude.

~ 1 Hz, above which they will be hidden by attenuation. We collated a dataset of $\sim 35,000$ events with $\sim 750,000$ manually labelled first P arrivals. The picks were obtained from the ISC [International Seismological Centre, 2021] and the USGS [U.S. Geological Survey, 2017], the magnitudes from the Global CMT project [Ekström et al., 2012]. The station and event distribution is visualised in Figure 6.6, alongside the magnitude and epicentral distance distributions. The dataset is primarily comprised of events with magnitudes $M_w > 5$. We include picks from high quality global seismic networks. We limit the maximum epicentral distance to 97° to avoid core phases. Further details on the dataset and the preprocessing are provided in Appendix E.3.

As a neural network, we adapted TEAM-LM, introduced in Chapter 5. Compared to the earlier chapter, we introduced several modifications to TEAM-LM to fit our application. First, as the traces are teleseismic, it is not possible to align the traces between stations by wall time. Instead, we align the traces by their P picks, such that the P pick is at the same sample for each station. Second, we now model real-time application through a sliding window instead of zero padding [Zhang et al., 2021]. To model the data available at time t, where t is relative to the P pick, we provide the model with the waveforms from $t - 30$ s to t. This allows to (i) apply the model to times more than 30 s after the P arrival; (ii) give the model more information on the noise at early times; (iii) make the model less sensitive to pick inaccuracies. Third, we do not encode station positions. We experimented with encoding the positions relative to the event, but it became apparent that the station distribution in our dataset in many cases is indicative of the magnitude. However, at teleseismic distances, the locations generally tend to have a lower impact on the waveform than at regional distances, which is also visible in our results. In addition,

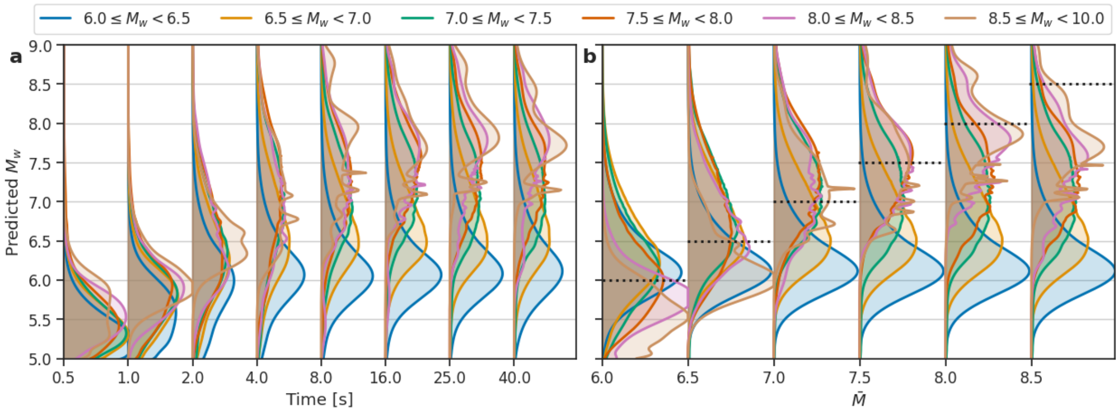


Figure 6.7: Average predicted PDFs based on teleseismic waveforms grouped by magnitude bin. **a** shows results at time t after onset ($\mathbb{P}(M|O_t)$). **b** shows results after the cumulative moment equals magnitude $\bar{M}$ ($\mathbb{P}(M|O_{\bar{M}})$). PDFs were truncated in visualisation to avoid overlap between different times/base magnitudes. Black dotted lines in **b** indicate the current base magnitude. For determining $t_{\bar{M}}$ in **b** we used the SCARDEC dataset. See Figure E.6 for plots with the other STF datasets. The apparent skew between buckets in panel **b** for $\bar{M} = 6.0$ likely results from SCARDEC processing artefacts (see Appendix E.1). Events differ between the panels: **b** only includes those events present in both the teleseismic dataset and SCARDEC (∼3,500 events) and **a** all of the former (∼38,000 events).

we modified the mixture density output to be consistent with the one for the STF model, i.e., we added the Dirichlet regularisation and switched to softplus for the sigma values.

Compared to the STF model, predictions $\mathbb{P}(M|O_t)$ and $\mathbb{P}(M|O_{\bar{M}})$ show higher uncertainties and systematic underestimation of the largest magnitudes at all times (Figure 6.7). Higher uncertainties result from the fact that assessing magnitude from waveforms is harder than from STFs, whereas underestimation can be attributed to data sparsity as discussed in Chapter 5. Due to the higher model uncertainties, all tails look rather like exponentially modified Gaussians than exponential distributions, as observed in the STF case. We note that the apparent lower uncertainty for the highest magnitude buckets compared to the lower magnitude buckets results from the number of samples in each bucket: with fewer samples available, the result gets less smooth but also less wide. Nonetheless, the general trends are highly similar to the results from the STF analysis before. Early predictions ($t \leq 2$ s) are indistinguishable, except for the bin $M_w = 6.0$ to 6.5, where event durations are often < 4 s. Bins split over time, similar to the STF model, although with a higher overlap in predictions between bins for the teleseismic results than for the STF case. Splits occur around 4 s for $M_w = 6.5$ to $M_w = 7.0$, 8 s for $M_w = 7.0$ to $M_w = 7.5$, and 16 to 25 s for both $M_w = 7.5$ to $M_w = 8.0$ and $M_w = 8.0$ to $M_w = 8.5$, again representing typical event half-durations.

As for $\mathbb{P}(M|O_t)$, predictions for $\mathbb{P}(M|O_{\bar{M}})$ exhibit similar behaviour to the ones from the STF model (Figure 6.7b). While $\bar{M}$ is considerably lower than the final magnitude, the predictions are indistinguishable between the buckets. Splitting of buckets occurs slightly later than for the STF model, i.e., clear differences only become apparent once $\bar{M}$ exceeds the upper bound of the bucket. This likely results from the higher uncertainties. We would therefore argue that assessment up to potential further asperities is likely still possible after the moment rate peak.

6.5 Comparison with previous results

Our results find no predictability from both STFs and teleseismic waveforms. These observations seem contradictory to several previously published results. We discuss potential reasons for the differences to some studies below. Melgar and Hayes [2019] found differences in moment acceleration for earthquakes of different sizes. As our STF model has access to the acceleration parameter investigated in this study, we would expect to be able to reproduce this effect. However, Meier et al. [2021] demonstrated that these results were caused by a sampling bias, i.e., an artificial selection of events based on a combination of magnitude and moment acceleration. Consequently, our results support the findings by Meier et al. [2021] regarding this sampling bias.

Danré et al. [2019] analysed STFs as well, decomposing them into subevents, and also found predictability. Large events exhibited higher moment in early subevents and in addition, showed higher complexity, i.e., featured more subevents. We suspect that their different conclusion might also result from the SCARDEC processing, which hides small subevents within large earthquakes and thereby artificially inflates the first identifiable subevent within a large event comparably to within a smaller event. For a further description of artefacts in the SCARDEC dataset, we refer to Appendix E.1.

Melgar and Hayes [2017] analysed slip pulse behaviour and found a correlation between rise time and moment magnitude, making magnitude assessment possible after ~ 15 s. While this conclusion would contradict our results, the significance of the findings for events with $M_w > 7.5$ is unclear, given the low number of very large events and several intermediate events with high rise time. On the other hand, ~ 15 s does not imply any further predictability than found in our study for events with $M_w \leq 7.5$, due to their comparatively short duration. Furthermore, the study by Melgar and Hayes [2017] uses geodetic observations in contrast to the STFs and teleseismic waveforms used in our analysis. While teleseismic P arrivals should allow for good rupture tracking, similar to geodetic recordings, specific patterns of slip pulses might not be identifiable.

Colombelli et al. [2020] found differences in the slope of early peak ground motion parameters at local distances between earthquakes of magnitude 4 to 9. In our analysis of teleseismic waves, this effect could be hidden by the attenuation of high-frequency waveforms. Therefore, our results do neither confirm nor contradict Colombelli et al. [2020]. Similarly, while our study practically rules out predictability given STFs and teleseismic waveforms, it still leaves the option of approaches, where the tell-tale signals might only be observable in local waveforms or require geodetic observations.

6.6 Implications of the estimation error

For our analysis, we used a variational approximation $\mathbb{P}_\theta(M|O_t)$ to $\mathbb{P}(M|O_t)$. However, we so far we did not take into account the estimation error, i.e., the difference between the true distribution and the variational approximation. While the use of a proper loss function guarantees us that this error is negligible in the limiting scenario of infinite data, in our finite-data scenario we observed estimation errors, e.g., higher uncertainties for the waveform-based model compared to the STF model. Therefore, in this section, we discuss the potential effects of the estimation error. In particular, we argue why we still interpret the predictions in terms of rupture predictability, even given the estimation error.

First, we analysed the model for teleseismic waveforms extensively on regional waveforms in Chapter 5. While we identified shortcomings for very large events, model errors are small as long as sufficient samples are available: the saturation threshold, from which underestimation occurs, varies between datasets and is determined by the number of large events. Consequently, this indicates a similar shift in saturation threshold for the teleseismic dataset, consistent with the observed saturation behaviour at late times t. Second, our results are not only consistent between two different observables, but they are also consistent with a well established physical model, i.e., symmetric moment rate functions with predictability only on the downward slope [e.g., Meier et al., 2017, Trugman et al., 2019]. Therefore, it would be an unlikely coincidence to fit exactly this model for both observables. Nonetheless, this observation allows for another, valid conclusion: estimating the final magnitude on the upward slope might not be impossible, but considerably more difficult than on the downward slope. Consequently, while the model is able to assess the magnitude on the downward slope, the model architecture or the amount of training data might be insufficient to correctly derive the distribution on the upward branch.

The missing constraints on the estimation error are closely related to an inherent issue in the discussion of rupture predictability: the question is asymmetric, i.e., while certain observations could conclusively prove rupture predictability, disproving rupture predictability is by far harder. Proving a certain level of predictability is straightforward, i.e., one can use statistical tests to show that a derived parameter differs between events with different final magnitudes at time t. However, the contrary is virtually impossible as one needs to show that *no* significant difference exists for *any* derived parameter.[28] Our approach puts the burden of analysing all derived parameters onto the deep learning model, i.e., instead of validating all derived parameters, we look for the best one, as defined by the proper loss function and the scoring rule. The missing constraints on the estimation error can therefore be interpreted as the deep learning analogue to the impossibility of enumerating all possible derived parameters for statistical analysis.

While we think it will not be possible to fully disprove rupture predictability due to the theoretical concerns discussed above, this chapter provides evidence against rupture predictability. Further evidence could be obtained through general bounds on the estimation error or by gaining further insights into the models to improve interpretability. We will discuss this aspect in Chapter 7.2.

6.7 Conclusion

In this chapter, we built a probabilistic framework for a principled discussion of rupture predictability, the primary goal of this thesis. Using variational approximation, we developed a method to estimate the relevant conditional probabilities from collected samples. We conclude that there are no signs of early rupture predictability in either STFs or broadband teleseismic P waveforms. Instead, our analysis indicates that the total moment of an event based on such data can only be estimated after the peak moment release. However, even then it is not possible to anticipate future asperities.

While our analysis finds no early predictability, it again highlights the feasibility of real-time rupture tracking, at least using STFs and teleseismic waveforms. In particular, we showed that magnitudes for larger events ($M_w > 7$) can be correctly estimated from real-time waveforms using TEAM-LM if sufficient training data is available. This expands upon the results from Chapter 5, where systematic underestimation already set in at smaller magnitudes. Nonetheless, systematic underestimation still occurs for the largest events.

[28] One might argue that it is only necessary to show that no difference exists for any *physically reasonable* parameter. However, enumerating all possible physically reasonable parameters is still virtually impossible.

Resource availability

The source time function datasets are available from the US Geological Survey (`https://earthquake.usgs.gov/data/finitefault/`), Linling Ye (`https://sites.google.com/site/linglingye001/earthquakes/slip-models`), and the SCARDEC project (`http://scardec.projects.sismo.ipgp.fr/`). We downloaded manual phase picks from the ISC [International Seismological Centre, 2021] and USGS [U.S. Geological Survey, 2017]. Seismic waveforms were downloaded from the IRIS and GEOFON data centers. We use waveforms from the GE [GEOFON Data Centre, 1993], G [Institut De Physique Du Globe De Paris (IPGP) and Ecole Et Observatoire Des Sciences De La Terre De Strasbourg (EOST), 1982], GT [Albuquerque Seismological Laboratory (ASL)/USGS, 1993], IC [Albuquerque Seismological Laboratory (ASL)/USGS, 1992], II [Scripps Institution Of Oceanography, 1986], and IU [Albuquerque Seismological Laboratory (ASL)/USGS, 1988] seismic networks. We have not published a precompiled teleseismic waveform dataset or code for the experiments yet.

7 Conclusion and Outlook

In this thesis, we studied earthquake rupture predictability through the real-time assessment of earthquake source parameters. We developed machine learning based methods for source parameter estimation in both post hoc and real-time scenarios. Furthermore, we developed a method for end-to-end assessment of ground motion parameters and showed how this method improves on traditional earthquake early warning methods. In this conclusion, we will first summarise the main findings and contributions of each chapter, and then discuss open questions and potential further directions.

In Chapter 2 we introduced relevant terms and concepts for the thesis. To account for the interdisciplinary nature of this thesis, we discuss both basics of seismology and of machine learning. The chapter closes with an overview of recent machine learning approaches in seismology.

Each of the four main chapters discusses an estimation task of the form $\mathbb{P}(X|O_t)$, with X either magnitude, location, or ground motion. In Chapter 3, we investigated the task $\mathbb{P}(M|O_t)$ for $t \to \infty$, i.e., magnitude scale calibration in a post hoc scenario. This analysis provides both a lower bound for the quality of real-time estimation methods and a gold-standard catalog for the later studies. For the magnitude scale calibration, we introduced a hybrid method based on mathematical optimisation and gradient boosted trees. Compared to standard, single-dimensional attenuation terms used for local magnitude calculation, we reduced residuals and thereby uncertainties by up to 23 % through the addition of correction terms for depth and spatial attenuation patterns. Furthermore, our method for combining several waveform features through gradient boosted trees led to a further reduction in uncertainties by nearly a factor of two. We applied our method to a catalog from Northern Chile with $\sim 100,000$ events, obtaining high-confidence magnitude values for the catalog.

After analysing the post hoc scenario in Chapter 3, we turned towards the real-time assessment of earthquakes. As a first step, in Chapter 4, we introduced TEAM, a deep learning based earthquake early warning method. TEAM conducts end-to-end ground motion estimation, i.e., estimates $\mathbb{P}(GM|O_t)$. TEAM outperforms traditional early warning schemes on two datasets from Italy and Japan in terms of both alert performance and warning times. We explained the performance gains with the end-to-end modelling approach. In contrast to source based approaches, this reduces the modelling errors, as it does not use two steps (source estimation and ground motion prediction) but only a single step. At the same time, TEAM still maintains a global view of the event, which stands in contrast to previous end-to-end approaches, so-called propagation based methods. This allows TEAM to achieve considerably longer warning times than these approaches. However, while outperforming traditional approaches, we also showed that TEAM is more susceptible to data sparsity issues than traditional methods. While this can partially be mitigated through transfer learning, correctly estimating strong ground shaking remains challenging.

For a comprehensive understandings of the advantages and limitations of deep learning for real-time earthquake assessment, we conducted an in-depth study on three datasets from Italy, from Japan and from Northern Chile. To this end, in Chapter 5, we adapted TEAM to magnitude and location estimation, $\mathbb{P}(M, Loc|O_t)$, yielding TEAM-LM. We use magnitude and location for this study, as, in contrast to ground shaking, these parameters are not affected by local site conditions. TEAM-LM strongly outperforms classical approaches, in particular for magnitude estimation, and also improves upon the performance of previous deep learning approaches. Nonetheless, we identified severe shortcomings in low data scenarios that are rooted in the black-box modelling approach. For location estimation, events in regions with low training event density are systematically mislocated towards regions with higher densities. For magnitude estimation, these shortcomings manifest in a systematic underestimation of large magnitude events. We showed that the systematic underestimation of large magnitudes can be reduced, and in some cases even completely resolved, with transfer learning, as long as an appropriate source dataset for transfer learning is available. Nonetheless, we showed that our method does not allow to draw conclusions on rupture predictability from regional datasets, as the performance limitations due to data sparsity mask potential effects of rupture predictability.

Following the inconclusive results from regional data, in Chapter 6, we moved to different observables: moment rate functions and teleseismic waveforms. In both cases, we used observations from earthquakes worldwide, thereby obtaining a significantly higher number of large events compared to the regional case. We discussed prior studies on rupture predictability and identified drawbacks in the commonly used deterministic viewpoint. To alleviate these deficiencies, we introduced a probabilistic formulation of rupture predictability in terms of the conditional distribution $\mathbb{P}(M|O_t)$ and an estimation scheme through variational approximation. Applying this scheme to both types of observables, we identify no signs of rupture predictability. Estimating the final size of an earthquake is only possible after the moment rate peak, and even then only up to the rupture of further asperities. Notably, this holds even in a probabilistic sense: no events are more or less likely to become large early on than the marginal.

Several key results of this thesis are, at least to some extent, unfortunate. First, the lack of rupture predictability poses an inherent limitation to early warning, as it is merely limited to tracking the moment release in real-time rather than anticipating the future growth of an event. Second, while we showed the excellent performance of deep learning methods for the real-time assessment of earthquakes, our detailed analysis also repeatedly revealed limitations in data-sparse scenarios. Therefore, in the next sections, we discuss potential future research directions to address these issues. We will look at both methodological contributions and potential opportunities in seismology to still identify signs of rupture predictability.

7.1 Standardisation for machine learning in seismology

In this thesis, we presented several novel machine learning methods for seismology. In total, we used seven datasets for their evaluation: waveforms and catalogs from Northern Chile, from Japan, from Italy, and from teleseismic arrivals, and three moment rate function datasets. We obtained or compiled all these datasets from publicly available sources and converted them to a format suitable for machine learning. This approach, model development being inherently coupled with dataset collection, is detrimental to the progress of machine learning research in seismology for several reasons. First, compiling datasets requires time, resources and expertise. Thereby it increases the entry hurdle for new researchers and reduces the resources spent on model development and evaluation. Second, the performance of different machine learning models can not be compared unless they are evaluated on the same dataset. Therefore, if datasets are specifically created in the same process as the model development, a comparison among published models is impossible. Third, as the datasets are created for a specific task, the employed data format is often tailored towards the task and does not follow any standard. Even if the compiled dataset is made publicly available, this limits the reusability or at least increases the effort for reuse. Consequently, subsequent developments can usually not be evaluated

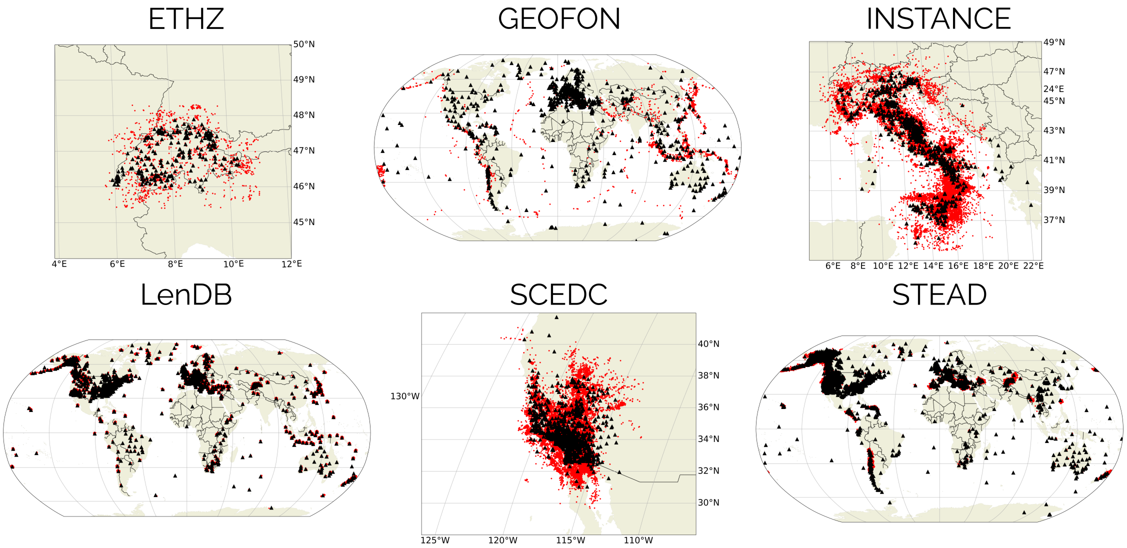


Figure 7.1: Map views of six datasets included in the *SeisBench* library. Each dataset is provided in a common format using the same API. The ETHZ, GEOFON, and SCEDC datasets were compiled for *SeisBench*; the INSTANCE, LenDB and STEAD datasets were published independently and converted to the *SeisBench* data format. The remaining datasets are not visualised as the location metadata for events and stations is not available. Figure from Woollam et al. [2022].

on a range of different datasets without significant effort in dataset conversion, making their evaluation less robust.

Similarly to dataset standardisation, model standardisation needs to be discussed. Several times within this thesis, we compared our approaches to previous approaches. However, in each case, we reimplemented the competing approaches ourselves, either because no implementation was publicly available, or because the existing implementation was incompatible with our processing pipelines. While we made our best efforts to ensure a truthful comparison with these methods, our reimplemented versions will, without question, deviate from the original approaches. This makes the conclusions achieved from the performance comparison less reliable.

There is another perspective on model standardisation, coming from observatory practice. Even though several authors published implementations of their methods [e.g. Mousavi et al., 2020, van den Ende and Ampuero, 2020, or our implementations of TEAM and TEAM-LM], we identify two major limitations in these approaches. First, these implementations evolved from research code and were often not primarily designed with the need of practitioners seeking to apply these methods in mind. Consequently, applying these methods requires a certain knowledge of deep learning, and even more of seismic waveform processing in Python. This is problematic, as many monitoring services use different tools, such as SeisComp [Helmholtz-Centre Potsdam-GFZ German Research Centre For Geosciences and GEMPA GmbH, 2008] or EarthWorm [Johnson et al., 1995], leading to a gap between model developers and seismological practitioners. Second, each of these implementations provides a different interface. It is therefore laborious to apply different models, both for application purposes and for benchmarking.

Several contributions towards standardisation have been put forward recently. In particular, multiple benchmark datasets have been published. Mousavi et al. [2019a] presented the STanford EArthquake Dataset (STEAD), a collection of 1.2 million waveforms with a rich collection of metadata. Among other applications, STEAD has been successfully used for training and evaluation models for phase picking [Mousavi et al., 2020], for magnitude estimation [Mousavi and Beroza, 2020a], and for earthquake localisation [Mousavi and Beroza, 2020b]. Similar to STEAD, Michelini et al. [2021] published INSTANCE, a collection of 1.3 million waveforms containing earthquakes in Italy and noise examples recorded on the same stations, again accompanied by rich metadata. Another published dataset is LenDB [Magrini et al., 2020], which in contrast to providing manually labelled phase picks as in STEAD and INSTANCE, contains estimated picks from travel time calculations. While all three of these datasets use similar data formats, the subtle differences do not allow for direct interchangeability.

With regards to models, we are not aware of frameworks by other authors unifying model APIs for different models solving the same task. However, several implementations of deep learning models aimed at practitioners have recently been published. QuakeFlow [Zhu et al., 2021] is a Kubernetes based system to build data processing pipelines, including machine learning based picking, denoising, and phase association models. PhaseWorm [Retailleau et al., 2022] integrates the deep learning based PhaseNet model [Zhu and Beroza, 2019] with the EarthWorm [Johnson et al., 1995] seismic processing software.

We developed a further solution to the standardisation issue: *SeisBench - A toolbox for machine learning in seismology* [Woollam et al., 2022].[29] SeisBench is an open-source python framework for machine learning in seismology. It jointly addresses the standardisation of datasets and models and works towards bridging the gap between model developers and practitioners. SeisBench consists of three main modules: *data*, *models* and *generate*. The *data* module provides a data format specification and a unified interface for accessing datasets in this format. Furthermore, it contains a collection of currently ten benchmark datasets, encompassing both previously published datasets like STEAD or INSTANCE, and newly assembled datasets (Figure 7.1). Each benchmark dataset can be easily accessed, downloaded, and, if necessary, converted through SeisBench. The *models* module contains implementations of currently six deep learning based detection and phase picking models, and one model for waveform denoising. Each model is implemented in pytorch [Paszke et al., 2019] and offers two interfaces. First, a regular pytorch interface to train the model using standard deep learning techniques. Second, an interface to apply models directly to obspy streams [Beyreuther et al., 2010] and thereby easily incorporating them into seismological analysis workflows. Furthermore, SeisBench offers a rich collection of pretrained weights for the models that can easily be accessed. The *generate* module offers functionality for building data generation pipelines, containing standard building blocks required in preprocessing, such as window selection strategies, frequency filters, or data augmentations. This allows developers to easily build training pipelines based on SeisBench datasets.

Using SeisBench, we conducted a large scale benchmark of detection and phase picking approaches, comparing six deep learning models and one traditional picking algorithm on eight datasets covering local to teleseismic distances [Münchmeyer et al., 2022].[30] Overall we observed the best performance for EQTransformer [Mousavi et al., 2020], GPD [Ross et al., 2018b], and PhaseNet [Zhu and Beroza, 2019], with advantages for EQTransformer

[29] SeisBench was developed within the Helmholtz AI project REPORT-DL. SeisBench was implemented by Jack Woollam and Jannes Münchmeyer. Andreas Rietbrock, Frederik Tilmann, Dietrich Lange, Thomas Bornstein, Tobias Diehl, Carlo Giunchi, Florian Haslinger, Dario Jozinović, Alberto Michelini, Joachim Saul, and Hugo Soto contributed to the design and concept of SeisBench.

[30] The benchmark was implemented by Jannes Münchmeyer within the REPORT-DL project. Jannes Münchmeyer wrote the manuscript about the benchmark. All contributors of SeisBench listed above contributed to the benchmark study and the manuscript preparation as well.

on teleseismic data. We analysed model performance both in-domain, evaluating models on the datasets they were trained on, and cross-domain, evaluating on other datasets than the training datasets. Our results showed good model transferability between different world regions, as long as the distance ranges of the datasets matched. A transfer between regional and teleseismic examples however yielded significantly worse results than in-domain application.

We think that further standardisation and development of tools for machine learning in seismology is necessary to build a robust foundation. This can happen through the extension of existing tools, such as SeisBench, or through the introduction of novel tools and standards.

7.2 The limitations of black-box machine learning

In large parts of this thesis, with TEAM, TEAM-LM, and the application of TEAM-LM to rupture predictability, we employed a machine learning approach called *black-box* machine learning. Such algorithms, once trained, can be applied to test data and thereby evaluated, at the same time, it is not possible to explain how the algorithms obtain their predictions. The algorithms are not *interpretable*, their inner workings remain a black box. To understand the implications of this black box on our work, we reiterate the two central goals of this thesis. First, investigating rupture predictability, i.e., gaining scientific understanding of physical processes. Second, improving earthquake early warning, i.e., a practical application of real-time assessment. The consequences of the lacking interpretability manifest differently for each of these goals.

For highlighting the implications regarding scientific insights, let us assume that our model found the contrary result for rupture predictability, i.e., it can predict the final magnitude early on. What does this tell us about rupture predictability? First of all, and importantly, this provides evidence for the existence of rupture predictability. However, because of the black-box approach, it does not tell us, how events of different sizes differ early on. Consequently, we can not draw conclusions on the underlying physical mechanisms. This also reduces trust in the findings. Data processing pipelines and modern neural networks are highly complex, often creating so-called knowledge leaks, i.e., high-quality telltale signs introduced only through the processing [Lapuschkin et al., 2019]. Without a physical understanding of the underlying mechanism, it is practically impossible to rule out such artefacts.

For early warning, the aspect of physical explainability itself is less relevant than for knowledge discovery. However, the resulting question of trust in the model is nonetheless pressing. For traditional early warning systems, guarantees can be given. For example, magnitude estimations from peak displacement [Kuyuk and Allen, 2013] will fulfil monotonicity constraints: if the peak displacement increases, the magnitude estimate increases. No similar guarantees can be given for black-box models, and our analysis of TEAM-LM (Chapter 5) showed that monotonicity is strongly violated in practice once event magnitudes fall sufficiently far outside the training data range. More generally, deep learning networks are known to exhibit strong performance degradation and wildly incorrect uncertainty estimates for these out-of-distribution examples [Snoek et al., 2019]. In practical applications, every very large earthquake will be an out-of-distribution example, due to their recurrence cycles of tens to hundreds of years.

To solve or mitigate these shortcomings of black-box machine learning, two main strategies can be pursued: either analysing the black-box model post hoc or employing models that are interpretable by design. Several post hoc interpretation techniques for neural networks have been proposed, for example, synthesising preferred inputs to identify learned features [Nguyen et al., 2016] or methods for generating saliency maps of feature importance [Bach et al., 2015, Montavon et al., 2017]. These methods have been applied to analyse models for seismological tasks. For example, Rouet-Leduc et al. [2020] trained a model to classify waveforms into noise and tremor waveforms and used interpretation techniques to show that the decision is indeed dependent on the parts of the waveforms containing the tremors. However, post hoc interpretation methods only give partial insights and, in addition, different methods frequently contradict each other in their results [Linardatos et al., 2021].

The alternative to post hoc interpretation is building interpretable models. Simple examples would be linear or logistic regression, where the coefficients can directly be interpreted. Some more complex models allow for direct interpretation as well, such as gradient boosted trees where the information gain from each split can be interpreted as feature importance [Chen and Guestrin, 2016]. Using this approach, Rouet-Leduc et al. [2017] identified acoustic signals indicative of the timing until the next rupture in laboratory shear experiments. We applied the same approach to identify the most relevant waveform features in Chapter 3. While such models offer direct interpretations, in most cases their performance is considerably inferior to deep learning models, in particular for complex, high dimensional input data [Linardatos et al., 2021].

For the application of black-box models in early warning, there is a further option: assessing the credibility of each prediction through an external algorithm. For example, a black box algorithm could be coupled with a traditional algorithm. When the traditional algorithm detects a magnitude above a certain threshold, the prediction of the black box algorithm is discarded, following the observation that predictions for very large earthquakes are unreliable. While this would limit the usefulness of the black-box approach for very large events, it could still improve warnings for intermediate-sized events. The black box model would then be one among an ensemble of models. Such model ensembles, even though without pure machine learning algorithms, are already deployed in early warning systems, such as ShakeAlert [Böse et al., 2015]. Nonetheless, future work is required regarding the interpretability and trustworthiness of black-box models in seismology for both the discovery of scientific knowledge and the application in high-stakes scenarios.

7.3 Mitigating data sparsity

Throughout this thesis, we repeatedly identified data sparsity as a key issue limiting the performance and applicability of machine learning models. In general, there is an abundance of samples, but for relevant corner cases, such as large events, only a few samples are available. Unfortunately, this issue is going to persist. While novel developments will continue increasing the completeness of catalogs [Tan et al., 2021, Jiang et al., 2022], thereby making more samples of small events available, all large events are already cataloged. Their number will only increase linearly with time as new events occur, which is not going to solve the data sparsity problem for machine learning models. It is therefore worthwhile discussing potential directions to circumvent this problem.

One solution studied extensively within this thesis is transfer learning, a form of domain adaptation. In Chapter 4 we showed how transfer learning from Japan to Italy can improve ground motion estimates. In Chapter 5 we showed similar results for magnitude estimation, but also pointed out that transfer learning does not yield benefits for location estimation. In general, two points are limiting the performance of transfer learning: the similarity between source and target domain, and the source dataset itself [Pan and

Yang, 2009]. The first point explains why transfer learning worked for ground motion and magnitude estimation, but not for location estimation: the location task simply is more region-specific than the other tasks. The second point becomes particularly apparent looking back at the magnitude estimation in Japan where transfer learning did not yield any improvements for large events. This is natural, as the source datasets from Chile and Italy only added a negligible number of large events on top of the ones already available in the Japan dataset. The requirement of a suitable source dataset is an inherent limitation for transfer learning. For the largest events globally, transfer learning will stay insufficient for training machine learning models, because there will simply never be a suitable source dataset available.

Therefore, other means are required for training models towards these cases. One option is to incorporate physical knowledge about earthquakes into the models, i.e., replacing the missing records with the knowledge of the underlying physical principles. The most straightforward way for this is through the data, i.e., training on synthetic waveforms. The practical applicability of this approach has been validated, for example through the application to prompto elasto-gravity signals [Licciardi et al., 2021]. However, there are several drawbacks to this approach. First, training deep learning models on synthetic waveforms requires high-quality synthetics, even though the exact requirements on the synthetics are yet to be determined. This is problematic, as creating high-quality synthetics is difficult and computationally expensive [Breuer et al., 2014]. Second, while it might improve the performance of methods such as early warning systems, this approach is unlikely to generate new scientific knowledge. Taking the example of rupture predictability, one would need to incorporate assumptions about rupture physics into the generating process of the synthetic waveforms. Therefore, the model is likely to recover these assumptions, leading to circular reasoning and consequently no new insights.

Another option for incorporating physical knowledge into the models is to incorporate it directly into the neural networks, leading to so-called physics informed neural networks [PINNs, Raissi et al., 2019]. PINNs incorporate physical laws, expressed as partial differential equations, into the structure or training of a neural network. This is usually achieved by incorporating the PDE into the loss function [Raissi et al., 2019], but also by other means, such as learning Hamiltonians [Greydanus et al., 2019] or Lagrangians [Cranmer et al., 2020]. In seismology, PINNs have been proposed for travel time estimation [Smith et al., 2020] and solutions of the seismic wave equation [Song et al., 2022]. Successive work employed these methods for both event localisation [Smith et al., 2022] and travel time tomography [Gao et al., 2021] based on arrival times. However, the application of PINNs to location or magnitude estimation directly from waveforms is difficult, as no suitable PDE connecting these properties is known.

As a more promising route, we suggest task composition and task decomposition in a way suiting the machine learning models. Task composition means combining several consecutive tasks into one end-to-end task that is more favourable to learn the the separate tasks. An example of task composition is TEAM, presented in Chapter 4. TEAM estimates ground motion directly from waveforms, instead of first estimating magnitude and location and then applying a ground motion model. When comparing TEAM to the results for TEAM-LM, TEAM still shows good performance for events where TEAM-LM significantly underestimates the magnitude. Furthermore, systematic underestimation of the largest PGA values is less severe than the magnitude underestimation. The better performance of TEAM-LM can be explained with the training data. While for TEAM-LM each large event provides one label, for TEAM each event has one label per station, even though these examples are highly correlated. Furthermore, high PGA values occur more frequently than large magnitude events, as they can also be caused by smaller events at short distances. This way, the composition of two tasks, one of which is at high magnitudes unsuitable for machine learning, creates a task that is more suitable for machine learning.

The opposite approach to task composition is task decomposition, i.e., splitting a task into a part that can be solved with a classical approach and a part that is favourable for machine learning. The classical part is a way of incorporating physical knowledge into the task. The machine learning part models a quantity not affected by the data sparsity, i.e., a quantity invariant to magnitude or location. An example for this approach would be to predict magnitude based on scaling laws. Instead of directly predicting the magnitude, one can predict a rough estimate from peak displacement and use machine learning only to estimate the residual between this traditional estimate and the true value. This approach has been employed for magnitude estimation in Central Italy, using a decomposition into the logarithm of peak ground velocity and a residual term [Zhang et al., 2021]. However we evaluated a similar approach, decomposing into the logarithm of peak displacement and a residual term, and found mixed results [Hauffe, 2021].[31] We propose that this particular type of decomposition becomes unreliable at high magnitudes due to saturation effects. A further advantage of such decomposed models is their improved interpretability, as at least parts of the prediction can be explained explicitly. However, these models also tend to suffer from similar drawbacks as other interpretable models, i.e., they often show worse average performance than the non-decomposed models [Hauffe, 2021].

Data sparsity is a major challenge for the application of machine learning to earthquake early warning and the study of very large events. The approaches presented in this section, synthetic training data, PINNs, and task (de)composition, are potential candidates to address this issue. We expect that a breakthrough regarding data sparsity will greatly improve the usefulness of machine learning for large seismic events.

7.4 Rupture nucleation and preparatory phases

After discussing technical aspects in the previous parts of this conclusion, as the last point, we now turn towards a seismological question: how could other observables influence the outcomes of our study? Our conclusion of Chapter 6 showed that there is no predictability from teleseismic P arrival waveforms or source time functions. However, this conclusion can not be generalised to other observables without further studies. We are going to focus on two aspects. First, we will discuss near field observations. These observations are promising as they contain high-frequency features that are attenuated at larger distances. Second, we will discuss observables from the time leading up to the event, the so-called preparatory phase. Such a preparatory phase might indicate certain properties of a large earthquake even before the event onset. We visualise potential observables and precursors in Figure 7.2.

Near field observations have been used to study rupture nucleation for several decades [e.g., Ellsworth and Beroza, 1995, Nakatani et al., 2000]. In principle, near field observations contain more information about the earthquake source, as they are less impacted by path effects, such as scattering or attenuation, than recordings at higher distances. The further away from the source a recording is obtained, the fewer detail on the source can usually be resolved. Signs of predictability in near-field observables have been identified, for example, by observing nucleation phases [Ellsworth and Beroza, 1995], early

[31] This work was conducted by Viola Hauffe within her master's thesis. Jannes Münchmeyer proposed the study design and supervised the thesis.

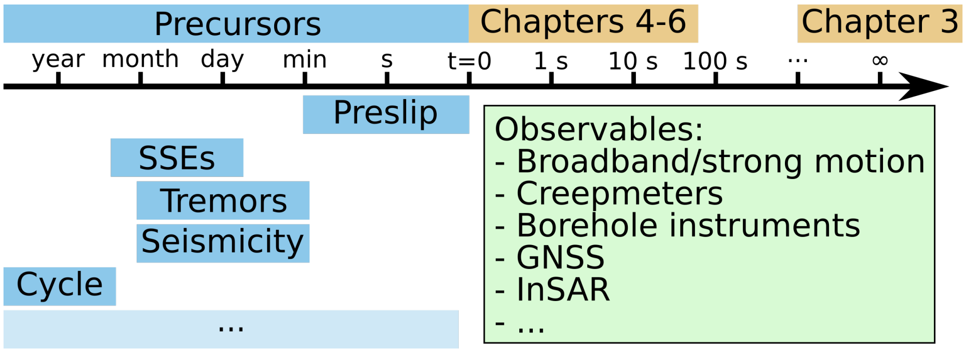


Figure 7.2: Schematic timeline indicating potential precursors with their time scales and the time scales analysed in each Chapter of this thesis. As potential precursors, we list preslip, slow slip events (SSEs), tremors, changes in the seismicity pattern, and the seismic cycle. As all of these are highly debated, the indicated time scales are only intended as a rough orientation. In addition, the green box lists potential observables. These might help identify precursors and also signs of rupture predictability. GNSS denotes geodetic observations, InSAR remote sensing data.

amplitude differences [Nakatani et al., 2000], dominant periods [Olson and Allen, 2005], or ground motion parameters [Colombelli et al., 2020]. Notably, not all studies of near field observables find predictability. Examples include studies of early peak displacement [Meier et al., 2016, Trugman et al., 2019] or onset waveforms [Abercrombie and Mori, 1994, Kilb and Gomberg, 1999, Mori and Kanamori, 1996, Okuda and Ide, 2018].

While near field observations provide a high resolution of the earthquake source process, they are unfortunately rare, as they require dense instrumentation around potential faults. For purely geometric reasons, the number of available records gets smaller the closer to the source one goes. A specific case of near field observations with even better source resolution are borehole instruments, due to their favourable noise conditions and (potential) closer proximity to the fault [Ellsworth et al., 2005, Kılıç et al., 2020]. However, these instruments are more costly and therefore rarer, in addition to the mentioned geometric reasons. Consequently, near field studies to this day remain rather anecdotal. In particular, the data can likely not be analysed with deep learning methods as introduced in this thesis, as long as not either considerably more data become available, or more data-efficient models are developed.

As a second type of alternative observables, we look at a potential precursory phase. Figure 7.2 provides a timeline centered around an earthquake at $t = 0$, indicating the time ranges covered in the chapters of this thesis. So far, when discussing the conditional probability $\mathbb{P}(M|O_t)$ we only looked at observables O_t with $t > 0$ and slightly before, i.e., only observables describing the rupture itself. However, this leaves out a potential preparatory phase, which might be lasting from seconds to months or even years. Studying preparatory phases poses a slightly different question than rupture predictability: can the size of an earthquake be constrained already before its onset? Note that pinpointing events before they occur falls into earthquake forecasting, which, as discussed before, is widely regarded as impossible. Therefore, studies of preparatory phases rather aim to identify typical and required preconditions of large events, as well as their relation to the event size.

Several changes have been observed before large earthquakes around their future fault zones. One line of research focuses on seismicity patterns, finding, for example, alterations in b-value [Nanjo and Yoshida, 2021, Derode et al., 2021], localisation of seismicity along a fault plane [Kato and Ben-Zion, 2020], or changes in frequency content of seismic events [Socquet et al., 2017]. Along another line of research, numerous studies found aseismic deformation in the days and months leading to major events, either in geodetic records or using strainmeters [Kaneko et al., 2017, Ruiz et al., 2017, Socquet et al., 2017, Bedford et al., 2020]. We visualise the approximate time scales of potential precursors in Figure 7.2. At the moment, most of these observations are anecdotal and not systematic, i.e., it is yet unknown for which fraction of large events such alterations occur in the lead up to the event, and whether similar alterations can also happen without a subsequent major failure. However, various real-world observations of precursory phases, such as a preceding aseismic slip or an increase in seismicity rate, have also been observed in laboratory experiments [Johnson et al., 2013, McLaskey, 2019].

When the effects of precursors are formulated probabilistic, they can be readily incorporated into the real-time assessment of ruptures. For this, we decompose the conditional probability

$$\mathbb{P}(M|O_t) \approx \mathbb{P}(M) * \tilde{\mathbb{P}}(M|O_{<0}) * \tilde{\mathbb{P}}(M|O_{t\geq 0}) \tag{7.1}$$

where $\tilde{\mathbb{P}}(M|O_t) = \mathbb{P}(M|O_t)\mathbb{P}(M)^{-1}$, i.e., the likelihood ratio compared to the marginal distribution. The decomposition consists of three terms: the long-term marginal $\mathbb{P}(M)$, the preparatory phase $\tilde{\mathbb{P}}(M|O_{<0})$, and the rupture itself $\tilde{\mathbb{P}}(M|O_{t\geq 0})$.[32] This decomposition is only an approximation as it assumes independence between preparatory phase and rupture evolution, which in practice will not be given. However, in contrast to a model incorporating the interactions between the preparatory phase and the rupture directly, which will most likely be infeasible to estimate, the decomposition can be estimated as each components can be estimated. The long term marginal is generally known [Shearer, 2009, Chapter 9.7.1]. This thesis showed, that estimation of the rupture related term in real-time is feasible up to rupture predictability, even though improvements for large magnitudes are still required. Some approximations for the term related to the preparatory phase exist, in particular from statistical seismology, such as observed changes in b-value [Nanjo and Yoshida, 2021, Derode et al., 2021]. Still, we are not aware of any models, in particular probabilistic models, based on more complex precursory processes, such as precursory slow slip events. Nonetheless, we think that with the growing number of observations of these phenomena, a probabilistic formulation will become possible in the near future. This could improve real-time earthquake assessment, for example, in the context of early warning.

With this outlook on combining real-time estimates of rupture development with information from a precursory phase, we conclude this thesis. Within the thesis, we developed a probabilistic formulation of rupture predictability and methods for the real-time analysis of earthquakes, finding no signs of early rupture predictability. In addition, we showed the applicability of these methods to improve earthquake early warning. We hope that future work, for example, along the directions outlined in the sections above, will help improving real-time assessment methods further and potentially even identify rupture predictability or preparatory phases in novel, high-quality observables.

[32] This decomposition should be seen as rather prototypical. In particular, it is debatable which information should be incorporated in the long-term marginal and which in the preparatory phase. For example, the seismic cycle itself is a long-term preparatory phase, even though it would be uncommon to refer to it this way. As we present this as a general concept, we refrain from defining an exact separation between the terms.

List of acronyms

$\bar{\mathbb{R}}$	$\mathbb{R} \cup \{-\infty, \infty\}$
$\mathbb{R}^+$	$\{x \in \mathbb{R} \mid x \geq 0\}$
δ_x	Dirac delta, $\delta_x(y)$ is 1 for $x = y$ and 0 otherwise
API	application programming interface
CDF	cumulative distribution function
CNN	convolutional neural network
CPU	central processing unit
CRPS	continuous ranked probability score
DL	deep learning
EEW	earthquake early warning
GMPE	ground motion prediction equation
GNSS	global navigation satellite system
GPU	graphics processing unit
GR	Gutenberg-Richter
iid	independently identically distributed
InSAR	interferometric synthetic aperture radar
MAE	mean absolute error
ML	machine learning
MLP	multilayer perceptron
MSE	mean squared error
m.u.	magnitude units
NN	neural network
PINN	physics informed neural network
PDE	partial differential equation
PDF	probability density function
PGA	peak ground acceleration
PGD	peak ground displacement
PGV	peak ground velocity
RMSE	root mean squared error
RNN	recurrent neural network
SGD	stochastic gradient descent
SSE	slow slip event
STF	source time function/moment rate function
TEAM	transformer earthquake alerting model
TEAM-LM	TEAM for location and magnitude estimation
TPU	tensor processing unit

List of Figures

List of Tables

References

M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283, 2016.

R. E. Abercrombie and J. Mori. Local observations of the onset of a large earthquake: 28 June 1992 Landers, California. *Bulletin of the Seismological Society of America*, 84 (3):725–734, June 1994. ISSN 0037-1106.

N. Abrahamson, N. Gregor, and K. Addo. BC Hydro Ground Motion Prediction Equations for Subduction Earthquakes. *Earthq. Spectra*, 32(1):23–44, Feb. 2016. ISSN 8755-2930. doi: 10.1193/051712EQS188MR.

Albuquerque Seismological Laboratory (ASL)/USGS. Global Seismograph Network (GSN - IRIS/USGS), 1988. doi: 10.7914/SN/IU.

Albuquerque Seismological Laboratory (ASL)/USGS. New China Digital Seismograph Network, 1992. doi: 10.7914/SN/IC.

Albuquerque Seismological Laboratory (ASL)/USGS. Global Telemetered Seismograph Network (USAF/USGS), 1993. doi: 10.7914/SN/GT.

F. Aldersons. Toward three-dimensional crustal structure of the Dead Sea region from local earthquake tomography. *PhD thesis*, 2004.

R. M. Allen. The ElarmS Earthquake Early Warning Methodology and Application across California. In P. Gasparini, G. Manfredi, and J. Zschau, editors, *Earthquake Early Warning Systems*, pages 21–43. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. ISBN 978-3-540-72240-3. doi: 10.1007/978-3-540-72241-0_3.

R. M. Allen and D. Melgar. Earthquake Early Warning: Advances, Scientific Challenges, and Societal Needs. *Annual Review of Earth and Planetary Sciences*, 47(1):361–388, 2019. doi: 10.1146/annurev-earth-053018-060457.

R. M. Allen, P. Gasparini, O. Kamigaichi, and M. Bose. The Status of Earthquake Early Warning around the World: An Introductory Overview. *Seismological Research Letters*, 80(5):682–693, Sept. 2009. ISSN 0895-0695. doi: 10.1785/gssrl.80.5.682.

G. Asch, F. Tilmann, B. Schurr, and T. Ryberg. Seismic network 5E: MINAS Project (2011/2013), 2011. doi: 10.14470/ab466166.

J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

M. Baer and U. Kradolfer. An automatic phase picker for local and teleseismic events. *Bulletin of the Seismological Society of America*, 77(4):1437–1445, 1987.

J. W. Baker. An introduction to probabilistic seismic hazard analysis. *White paper version*, 2(1):79, 2013.

W. B. Banerdt, S. E. Smrekar, D. Banfield, D. Giardini, M. Golombek, C. L. Johnson, P. Lognonné, A. Spiga, T. Spohn, C. Perrin, et al. Initial results from the InSight mission on Mars. *Nature Geoscience*, 13(3):183–189, 2020.

P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

J. R. Bedford, M. Moreno, Z. Deng, O. Oncken, B. Schurr, T. John, J. C. Báez, and M. Bevis. Months-long thousand-kilometre-scale wobbling before great subduction earthquakes. *Nature*, 580(7805):628–635, 2020.

M. Belkin, D. Hsu, S. Ma, and S. Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

Y. Bengio, I. Goodfellow, and A. Courville. *Deep learning*, volume 1. MIT press Massachusetts, USA:, 2017.

K. J. Bergen, P. A. Johnson, V. Maarten, and G. C. Beroza. Machine learning for data-driven discovery in solid Earth geoscience. *Science*, 363(6433), 2019.

M. Beyreuther, R. Barsch, L. Krischer, T. Megies, Y. Behr, and J. Wassermann. Obspy: A python toolbox for seismology. *Seismological Research Letters*, 81(3):530–533, 2010.

D. Bindi, B. Schurr, R. Puglia, E. Russo, A. Strollo, F. Cotton, and S. Parolai. A Magnitude Attenuation Function Derived for the 2014 Pisagua (Chile) Sequence Using Strong-Motion DataShort Note. *BSSA*, 104(6):3145–3152, Dec. 2014. ISSN 0037-1106. doi: 10.1785/0120140152.

C. M. Bishop. Mixture density networks. Technical report, Aston University, 1994.

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

P. Bormann. *New Manual of Seismological Observatory Practice (NMSOP-2)*. IASPEI, GeoForschungsZentrum, 2012. doi: 10.2312/GFZ.NMSOP-2.

P. Bormann, R. Liu, Z. Xu, K. Ren, L. Zhang, and S. Wendt. First application of the new IASPEI teleseismic magnitude standards to data of the China National Seismographic Network. *Bulletin of the Seismological Society of America*, 99(3):1868–1891, 2009. doi: 10.1785/0120080010.

P. Bormann, K. Klinge, and S. Wendt. Data analysis and seismogram interpretation. *New Manual of Seismological Observatory Practice 2 (NMSOP2)*, 2013a. doi: 10.2312/GFZ.NMSOP-2_CH11.

P. Bormann, S. Wendt, and D. DiGiacomo. Seismic sources and source parameters. *New Manual of Seismological Observatory Practice 2 (NMSOP2)*, 2013b. doi: 10.2312/GFZ.NMSOP-2_CH3.

M. Böse, F. Wenzel, and M. Erdik. PreSEIS: A Neural Network-Based Approach to Earthquake Early Warning for Finite Faults. *Bulletin of the Seismological Society of America*, 98(1):366–382, Feb. 2008. ISSN 0037-1106. doi: 10.1785/0120070002.

M. Böse, T. H. Heaton, and E. Hauksson. Real-time Finite Fault Rupture Detector (FinDer) for large earthquakes. *Geophysical Journal International*, 191(2):803–812, Nov. 2012. ISSN 0956-540X. doi: 10.1111/j.1365-246X.2012.05657.x.

M. Böse, C. Felizardo, and T. H. Heaton. Finite-fault rupture detector (FinDer): Going real-time in Californian ShakeAlert warning system. *Seismological Research Letters*, 86 (6):1692–1704, 2015.

M. Böse, D. E. Smith, C. Felizardo, M.-A. Meier, T. H. Heaton, and J. F. Clinton. FinDer v.2: Improved real-time ground-motion predictions for M2–M9 with seismic finite-source characterization. *Geophysical Journal International*, 212(1):725–742, Jan. 2018. ISSN 0956-540X. doi: 10.1093/gji/ggx430.

A. Breuer, A. Heinecke, S. Rettenberger, M. Bader, A.-A. Gabriel, and C. Pelties. Sustained petascale performance of seismic simulations with SeisSol on SuperMUC. In *International Supercomputing Conference*, pages 1–18. Springer, 2014.

D. R. Brillinger and H. K. Preisler. An exploratory analysis of the Joyner-Boore attenuation data. *BSSA*, 74(4):1441–1450, Aug. 1984. ISSN 0037-1106.

E. E. Brodsky and N. J. van der Elst. The uses of dynamic earthquake triggering. *Annual Review of Earth and Planetary Sciences*, 42:317–339, 2014.

M. M. Bronstein, J. Bruna, T. Cohen, and P. Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

S. Carrol and D. Goodstein. Defining the scientific method. *Nat Methods*, 6:237, 2009.

C. Cauzzi, R. Sleeman, J. Clinton, J. D. Ballesta, O. Galanis, and P. Kaestli. Introducing the European Rapid Raw Strong-Motion Database. *Seismological Research Letters*, 84 (4):977–986, 2016.

S. Cesca, M. Sobiesiak, A. Tassara, M. Olcay, E. Günther, S. Mikulla, and T. Dahm. The Iquique Local Network and PicArray, 2009. doi: 10.14470/vd070092.

J. C. Chang, D. A. Lockner, and Z. Reches. Rapid Acceleration Leads to Rapid Weakening in Earthquake-Like Laboratory Experiments. *Science*, 338(6103):101–105, Oct. 2012. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1221195.

T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4232-2. doi: 10.1145/2939672.2939785.

A. I. Chung, I. Henson, and R. M. Allen. Optimizing Earthquake Early Warning Performance: ElarmS-3. *Seismological Research Letters*, 90(2A):727–743, Mar. 2019. ISSN 0895-0695. doi: 10.1785/0220180192.

E. S. Cochran, J. Bunn, S. E. Minson, A. S. Baltay, D. L. Kilb, Y. Kodera, and M. Hoshiba. Event Detection Performance of the PLUM Earthquake Early Warning Algorithm in Southern California. *Bulletin of the Seismological Society of America*, 109(4):1524–1541, Aug. 2019. ISSN 0037-1106. doi: 10.1785/0120180326.

S. Colombelli, G. Festa, and A. Zollo. Early rupture signals predict the final earthquake size. *Geophysical Journal International*, 223(1):692–706, Sept. 2020. ISSN 0956-540X. doi: 10.1093/gji/ggaa343.

F. Corbi, L. Sandri, J. Bedford, F. Funiciello, S. Brizzi, M. Rosenau, and S. Lallemand. Machine Learning Can Predict the Timing and Size of Analog Earthquakes. *Geophysical Research Letters*, 46(3):1303–1311, 2019. ISSN 1944-8007. doi: 10.1029/2018GL081251.

F. Crameri. Geodynamic diagnostics, scientific visualisation and StagLab 3.0. *Geoscientific Model Development*, 11(6):2541–2562, 2018.

M. Cranmer, S. Greydanus, S. Hoyer, P. Battaglia, D. Spergel, and S. Ho. Lagrangian neural networks. *arXiv preprint arXiv:2003.04630*, 2020.

G. Cua and T. H. Heaton. Characterizing Average Properties of Southern California Ground Motion Amplitudes and Envelopes. EERL Report, Earthquake Engineering Research Laboratory, Pasadena, CA, 2009.

G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

H. Dai and C. MacBeth. Automatic picking of seismic arrivals in local earthquake data using an artificial neural network. *Geophysical journal international*, 120(3):758–774, 1995.

P. Danré, J. Yin, B. P. Lipovsky, and M. A. Denolle. Earthquakes Within Earthquakes: Patterns in Rupture Complexity. *Geophysical Research Letters*, 46(13):7352–7360, July 2019. ISSN 0094-8276. doi: 10.1029/2019GL083093.

H. M. Dawood and A. Rodriguez-Marek. A Method for Including Path Effects in Ground-Motion Prediction Equations: An Example Using the Mw 9.0 Tohoku Earthquake Aftershocks. *Bulletin of the Seismological Society of America*, 103(2B):1360–1372, May 2013. ISSN 0037-1106. doi: 10.1785/0120120125.

N. Deichmann. The relation between ME, ML and Mw in theory and numerical simulations for small to moderate earthquakes. *J Seismol*, 22(6):1645–1668, Nov. 2018a. ISSN 1573-157X. doi: 10.1007/s10950-018-9786-1.

N. Deichmann. Why Does ML Scale 1:1 with 0.5logES? *Seismological Research Letters*, 89(6):2249–2255, Nov. 2018b. ISSN 0895-0695. doi: 10.1785/0220180121.

B. Derode, R. Madariaga, and J. Campos. Seismic rate variations prior to the 2010 Maule, Chile MW 8.8 giant megathrust earthquake. *Scientific reports*, 11(1):1–9, 2021.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Dipartimento di Fisica, Università degli studi di Napoli Federico II. Irpinia Seismic Network (ISNet), 2005.

K. Doi. Seismic Network and Routine Data Processing - Japan Meteorological Agency. *Summary of the Bulletin of the International Seismological Centre*, 47(7-12):25–42, 2014.

M. Dolce and D. Di Bucci. The 2016–2017 Central Apennines Seismic Sequence: Analogies and Differences with Recent Italian Earthquakes. In K. Pitilakis, editor, *Recent Advances in Earthquake Engineering in Europe: 16th European Conference on Earthquake Engineering-Thessaloniki 2018*, Geotechnical, Geological and Earthquake Engineering, pages 603–638. Springer International Publishing, Cham, 2018. ISBN 978-3-319-75741-4. doi: 10.1007/978-3-319-75741-4_26.

F. U. Dowla, S. R. Taylor, and R. W. Anderson. Seismic discrimination with artificial neural networks: preliminary results with regional spectral data. *Bulletin of the Seismological Society of America*, 80(5):1346–1373, 1990.

B. Duignan. Occam's razor. In *Encyclopedia Britannica*, 2021. `https://www.britannica.com/topic/Occams-razor`, Accessed 10 January 2022.

P. S. Dysart and J. J. Pulli. Regional seismic event classification at the NORESS array: seismological measurements and the use of trained neural networks. *Bulletin of the Seismological Society of America*, 80(6B):1910–1933, 1990.

A. M. Dziewonski and D. L. Anderson. Preliminary reference Earth model. *Physics of the earth and planetary interiors*, 25(4):297–356, 1981.

A. M. Dziewonski, T.-A. Chou, and J. H. Woodhouse. Determination of Earthquake Source Parameters from Waveform Data for Studies of Global and Regional Seismicity. *J. Geophys. Res. Solid Earth*, 86(B4):2825–2852, Apr. 1981. ISSN 2156-2202. doi: 10.1029/JB086iB04p02825.

S. Earp and A. Curtis. Probabilistic neural network-based 2d travel-time tomography. *Neural Computing and Applications*, 32(22):17077–17095, 2020.

M. L. Eaton. A Group Action on Covariances with Applications to the Comparison of Linear Normal Experiments. *Lect. Notes-Monogr. Ser.*, 22:76–90, 1992. ISSN 0749-2170.

G. Ekström, M. Nettles, and A. Dziewoński. The global CMT project 2004–2010: Centroid-moment tensors for 13,017 earthquakes. *Phys. Earth Planet. Inter.*, 200-201: 1–9, June 2012. ISSN 00319201. doi: 10.1016/j.pepi.2012.04.002.

W. Ellsworth, S. Hickman, and M. Zoback. Seismology in the Source: The San Andreas Fault Observatory at Depth. 2005.

W. L. Ellsworth and G. C. Beroza. Seismic Evidence for an Earthquake Nucleation Phase. *Science*, 268(5212):851–855, May 1995. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.268.5212.851.

EMERSITO Working Group. Seismic Network for Site Effect Studies in Amatrice Area (Central Italy) (SESAA), 2018. doi: 10.13127/SD/7TXeGdo5X8.

G. Festa, A. Zollo, and M. Lancieri. Earthquake magnitude estimation from early radiated energy. *Geophys. Res. Lett.*, 35(22), Nov. 2008. ISSN 0094-8276. doi: 10.1029/2008GL035576.

G. Festa, M. Picozzi, A. Caruso, S. Colombelli, M. Cattaneo, L. Chiaraluce, L. Elia, C. Martino, S. Marzorati, M. Supino, and A. Zollo. Performance of Earthquake Early Warning Systems during the 2016–2017 Mw 5–6.5 Central Italy Sequence. *Seismological Research Letters*, 89(1):1–12, Jan. 2018. ISSN 0895-0695. doi: 10.1785/0220170150.

J. Frankle and M. Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

J. H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378, 2002.

A. Gao, J. Castellanos, Y. Yue, Z. Ross, and K. Bouman. Deepgem: Generalized expectation-maximization for blind inversion. *Advances in Neural Information Processing Systems*, 34, 2021.

GEOFON Data Centre. GEOFON Seismic Network, 1993. URL `http://geofon.gfz-potsdam.de/doi/network/GE`. doi: 10.14470/TR560404.

Geological Survey-Provincia Autonoma di Trento. Trentino Seismic Network, 1981. doi: 10.7914/SN/ST.

GFZ German Research Centre For Geosciences and Institut Des Sciences De L'Univers-Centre National De La Recherche CNRS-INSU. IPOC Seismic Network, 2006. doi: 10.14470/pk615318.

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*, pages 249–256, 2010.

T. Gneiting and A. E. Raftery. Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, 102(477):359–378, Mar. 2007. ISSN 0162-1459. doi: 10.1198/016214506000001437.

J. Gomberg, A. Wech, K. Creager, K. Obara, and D. Agnew. Reconsidering earthquake scaling. *Geophysical Research Letters*, 43(12):6243–6251, 2016. ISSN 1944-8007. doi: 10.1002/2016GL069967.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

F. M. Graeber and G. Asch. Three-dimensional models of P wave velocity and P-to-S velocity ratio in the southern central Andes by simultaneous inversion of local earthquake data. *Journal of Geophysical Research: Solid Earth*, 104(B9):20237–20256, Sept. 1999. ISSN 0148-0227. doi: 10.1029/1999JB900037.

S. Greydanus, M. Dzamba, and J. Yosinski. Hamiltonian neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017.

Gurobi Optimization LLC. Gurobi optimizer reference manual, 2018. URL `http://www.gurobi.com`.

J. F. Hall, T. H. Heaton, M. W. Halling, and D. J. Wald. Near-source ground motion and its effects on flexible buildings. *Earthquake spectra*, 11(4):569–605, 1995.

T. C. Hanks and H. Kanamori. A moment magnitude scale. *Journal of Geophysical Research: Solid Earth*, 84(B5):2348–2350, 1979.

T. C. Hanks and R. K. McGuire. The character of high-frequency strong ground motion. *Bulletin of the Seismological Society of America*, 71(6):2071–2095, Dec. 1981. ISSN 0037-1106.

L. K. Hansen and P. Salamon. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, 12(10):993–1001, 1990.

C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, Sept. 2020. doi: 10.1038/s41586-020-2649-2. URL `https://doi.org/10.1038/s41586-020-2649-2`.

V. Hauffe. A hybrid deep-learning approach for reliable real-time assessment of high magnitude earthquakes. *Master thesis, Computer Science, Otto-von-Guericke-Universität Magdeburg*, 2021.

G. P. Hayes. The finite, kinematic rupture properties of great-sized earthquakes since 1990. *Earth and Planetary Science Letters*, 468:94–100, June 2017. ISSN 0012-821X. doi: 10.1016/j.epsl.2017.04.003.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Helmholtz-Centre Potsdam-GFZ German Research Centre For Geosciences and GEMPA GmbH. The SeisComP seismological software package, 2008. URL `https://www.seiscomp.de/`. doi: 10.5880/GFZ.2.4.2020.003.

C. Hulbert, B. Rouet-Leduc, P. A. Johnson, C. X. Ren, J. Rivière, D. C. Bolton, and C. Marone. Similarity of fast and slow earthquakes illuminated by machine learning. *Nature Geoscience*, 12(1):69, Jan. 2019. ISSN 1752-0908. doi: 10.1038/s41561-018-0272-8.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi: 10.1109/MCSE.2007.55.

S. Ide. Frequent observations of identical onsets of large and small earthquakes. *Nature*, 573(7772):112–116, Sept. 2019. ISSN 1476-4687. doi: 10.1038/s41586-019-1508-5.

S. Ide, G. C. Beroza, D. R. Shelly, and T. Uchide. A scaling law for slow earthquakes. *Nature*, 447(7140):76–79, 2007.

Institut De Physique Du Globe De Paris (IPGP) and Ecole Et Observatoire Des Sciences De La Terre De Strasbourg (EOST). GEOSCOPE, French Global Network of broad band seismic stations, 1982. URL `http://geoscope.ipgp.fr/networks/detail/G/`. doi: 10.18715/GEOSCOPE.G.

International Seismological Centre. ISC bulletin, 2021. doi: 10.31905/d808b830.

S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

ISIDe Working Group. Italian Seismological Instrumental and Parametric Database (ISIDe), 2007. URL `http://iside.rm.ingv.it/`. doi: 10.13127/ISIDE.

Istituto Nazionale di Geofisica e Vulcanologia (INGV). INGV experiments network, 2008.

Istituto Nazionale di Geofisica e Vulcanologia (INGV), Istituto di Geologia Ambientale e Geoingegneria (CNR-IGAG), Istituto per la Dinamica dei Processi Ambientali (CNR-IDPA), Istituto di Metodologie per l'Analisi Ambientale (CNR-IMAA), and Agenzia Nazionale per le nuove tecnologie, l'energia e lo sviluppo economico sostenibile (ENEA). Centro di microzonazione sismica Network, 2016 Central Italy seismic sequence (CentroMZ), 2018. doi: 10.13127/SD/ku7Xm12Yy9.

Istituto Nazionale di Geofisica e Vulcanologia (INGV), Italy. Rete Sismica Nazionale (RSN), 2006. doi: 10.13127/SD/X0FXnH7QfY.

C. Jiang, P. Zhang, M. C. White, R. Pickle, and M. S. Miller. A Detailed Earthquake Catalog for Banda Arc–Australian Plate Collision Zone Using Machine-Learning Phase Picker and an Automated Workflow. *The Seismic Record*, 2(1):1–10, 2022.

C. E. Johnson, A. Bittenbinder, B. Bogaert, L. Dietz, and W. Kohler. Earthworm: A flexible approach to seismic network processing. *Iris newsletter*, 14(2):1–4, 1995.

P. A. Johnson, B. Ferdowsi, B. M. Kaproth, M. Scuderi, M. Griffa, J. Carmeliet, R. A. Guyer, P.-Y. Le Bas, D. T. Trugman, and C. Marone. Acoustic emission and microslip precursors to stick-slip failure in sheared granular material. *Geophysical Research Letters*, 40(21):5627–5631, 2013.

P. A. Johnson, B. Rouet-Leduc, L. J. Pyrak-Nolte, G. C. Beroza, C. J. Marone, C. Hulbert, A. Howard, P. Singer, D. Gordeev, D. Karaflos, et al. Laboratory earthquake forecasting: A machine learning competition. *Proceedings of the National Academy of Sciences*, 118(5), 2021.

T. H. Jordan, Y.-T. Chen, P. Gasparini, R. Madariaga, I. Main, W. Marzocchi, G. Papadopoulos, G. Sobolev, K. Yamaoka, and J. Zschau. Operational earthquake forecasting. State of knowledge and guidelines for utilization. *Annals of Geophysics*, 54(4), 2011.

D. Jozinović, A. Lomax, I. Štajduhar, and A. Michelini. Rapid prediction of earthquake ground shaking intensity using raw waveform data and a convolutional neural network. *Geophysical Journal International*, 222(2):1379–1389, 2020. doi: 10.1093/gji/ggaa233.

D. Jozinović, A. Lomax, I. Štajduhar, and A. Michelini. Transfer learning: Improving neural network based prediction of earthquake ground shaking for an area with insufficient training data. *Geophysical Journal International*, 229(1):704–718, 2022. doi: 10.1093/gji/ggab488.

J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, et al. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, 2021.

Y. Kaneko, B. M. Carpenter, and S. B. Nielsen. Nucleation process of magnitude 2 repeating earthquakes on the San Andreas Fault predicted by rate-and-state fault models with SAFOD drill core data. *Geophysical research letters*, 44(1):162–173, 2017.

K. R. Karim and F. Yamazaki. Correlation of JMA instrumental seismic intensity with strong motion parameters. *Earthquake Engineering & Structural Dynamics*, 31(5): 1191–1212, 2002. ISSN 1096-9845. doi: 10.1002/eqe.158.

A. Kato and Y. Ben-Zion. The generation of large earthquakes. *Nature Reviews Earth & Environment*, pages 1–14, Nov. 2020. ISSN 2662-138X. doi: 10.1038/s43017-020-00108-w.

A. Katsumata. Relationship between displacement and velocity amplitudes of seismic waves from local earthquakes. *Earth and Planetary Science Letters*, 53:347–355, May 2001. doi: 10.1186/BF03352391.

B. L. Kennett, E. Engdahl, and R. Buland. Constraints on seismic velocities in the Earth from traveltimes. *Geophysical Journal International*, 122(1):108–124, 1995.

D. Kilb and J. Gomberg. The initial subevent of the 1994 Northridge, California, earthquake: Is earthquake size predictable? *Journal of Seismology*, 3(4):409–420, Oct. 1999. ISSN 1573-157X. doi: 10.1023/A:1009890329925.

T. Kılıç, R. F. Kartal, F. T. Kadirioğlu, M. Bohnhoff, M. Nurlu, D. Acarel, P. M. Garzon, G. Dresen, V. Özsarac, and P. E. Malin. Geophysical borehole observatory at the North Anatolian Fault in the Eastern Sea of Marmara (GONAF): Initial results. *Journal of Seismology*, 24(2):375–395, 2020.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Y. Kodera, Y. Yamada, K. Hirano, K. Tamaribuchi, S. Adachi, N. Hayashimoto, M. Morimoto, M. Nakamura, and M. Hoshiba. The Propagation of Local Undamped Motion (PLUM) Method: A Simple and Robust Seismic Wavefield Estimation Approach for Earthquake Early Warning. *Bulletin of the Seismological Society of America*, 108(2): 983–1003, Apr. 2018. ISSN 0037-1106. doi: 10.1785/0120170085.

K. Koketsu, Y. Yokota, N. Nishimura, Y. Yagi, S. Miyazaki, K. Satake, Y. Fujii, H. Miyake, S. Sakai, Y. Yamanaka, et al. A unified source model for the 2011 Tohoku earthquake. *Earth and Planetary Science Letters*, 310(3-4):480–487, 2011.

A. Kolmogorov. Sulla determinazione empirica di una lgge di distribuzione. *Inst. Ital. Attuari, Giorn.*, 4:83–91, 1933.

Q. Kong. USGS Earthquake Hazards Program, Final Report: Deep Learning Based Approach to Integrate MyShake's Trigger Data with ShakeAlert for Faster and Robust EEW Alerts (Award No. G20AP00058, May 2020 through April 2021). 2021. URL `https://earthquake.usgs.gov/cfusion/external_grants/reports/G20AP00058.pdf`.

Q. Kong, D. T. Trugman, Z. E. Ross, M. J. Bianco, B. J. Meade, and P. Gerstoft. Machine learning in seismology: Turning data into insights. *Seismological Research Letters*, 90 (1):3–14, 2019.

S. R. Kotha, D. Bindi, and F. Cotton. Partially non-ergodic region specific GMPE for Europe and Middle-East. *Bulletin of Earthquake Engineering*, 14(4):1245–1263, 2016.

M. Kriegerowski, G. M. Petersen, H. Vasyura-Bathke, and M. Ohrnberger. A Deep Convolutional Neural Network for Localization of Clustered Earthquakes Based on Multistation Full Waveforms. *Seismological Research Letters*, 90(2A):510–516, 2019. doi: 10.1785/0220180320.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25: 1097–1105, 2012.

H. S. Kuyuk and R. M. Allen. A global approach to provide magnitude estimates for earthquake early warning alerts. *Geophysical Research Letters*, 40(24):6329–6333, 2013. ISSN 1944-8007. doi: 10.1002/2013GL058580.

M. Lancieri and A. Zollo. A Bayesian approach to the real-time estimation of magnitude from the early P and S wave displacement peaks. *J. Geophys. Res.*, 113(B12), Dec. 2008. ISSN 0148-0227. doi: 10.1029/2007JB005386.

S. Lapuschkin, S. Wäldchen, A. Binder, G. Montavon, W. Samek, and K.-R. Müller. Unmasking Clever Hans predictors and assessing what machines really learn. *Nature communications*, 10(1):1–8, 2019.

S. Latour, A. Schubnel, S. Nielsen, R. Madariaga, and S. Vinciguerra. Characterization of nucleation during laboratory earthquakes. *Geophysical Research Letters*, 40(19): 5064–5069, 2013. ISSN 1944-8007. doi: 10.1002/grl.50974.

T. Lay, H. Kanamori, C. J. Ammon, M. Nettles, S. N. Ward, R. C. Aster, S. L. Beck, S. L. Bilek, M. R. Brudzinski, R. Butler, et al. The great Sumatra-Andaman earthquake of 26 December 2004. *science*, 308(5725):1127–1133, 2005.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436, 2015.

J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, and Y. W. Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning*, pages 3744–3753. PMLR, 2019.

F. Leyton, S. Ruiz, J. C. Baez, G. Meneses, and R. Madariaga. How Fast Can We Reliably Estimate the Magnitude of Subduction Earthquakes? *Geophysical Research Letters*, 45 (18):9633–9641, Sept. 2018. ISSN 0094-8276. doi: 10.1029/2018GL078991.

A. Licciardi, Q. Bletery, B. Rouet-Leduc, J.-P. Ampuero, and K. Juhel. Timeliness of earthquake magnitude estimation from the prompt elasto-gravity signal using deep learning. In *EGU General Assembly Conference Abstracts*, pages EGU21–14790, 2021.

P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis. Explainable ai: A review of machine learning interpretability methods. *Entropy*, 23(1):18, 2021.

A. Lomax, J. Virieux, P. Volant, and C. Berge-Thierry. Probabilistic earthquake location in 3d and layered models. In *Advances in seismic event location*, pages 101–134. Springer, 2000.

A. Lomax, A. Michelini, and D. Jozinović. An Investigation of Rapid Earthquake Characterization Using Single-Station Waveforms and a Convolutional Neural Network. *Seismological Research Letters*, 2019. doi: 10.1785/0220180311.

F. Magrini, D. Jozinović, F. Cammarano, A. Michelini, and L. Boschi. Local earthquakes detection: A benchmark dataset of 3-component seismograms built on a global scale. *Artificial Intelligence in Geosciences*, 1:1–10, 2020.

A. Marcellinus. *Res gestae.* around 390.

J. E. Matheson and R. L. Winkler. Scoring rules for continuous probability distributions. *Management science*, 22(10):1087–1096, 1976.

E. Matrullo, R. De Matteis, C. Satriano, O. Amoroso, and A. Zollo. An improved 1-D seismic velocity model for seismological studies in the Campania–Lucania region (Southern Italy). *Geophysical Journal International*, 195(1):460–473, 2013.

W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

G. C. McLaskey. Earthquake Initiation From Laboratory Observations and Implications for Foreshocks. *Journal of Geophysical Research: Solid Earth*, 124(12):12882–12904, 2019. ISSN 2169-9356. doi: 10.1029/2019JB018363.

G. C. McLaskey and D. A. Lockner. Preslip and cascade processes initiating laboratory stick slip. *Journal of Geophysical Research: Solid Earth*, 119(8):6323–6336, 2014. ISSN 2169-9356. doi: 10.1002/2014JB011220.

MedNet Project Partner Institutions. Mediterranean Very Broadband Seismographic Network (MedNet), 1990. doi: 10.13127/SD/fBBBtDtd6q.

M.-A. Meier. How “good” are real-time ground motion predictions from Earthquake Early Warning systems? *Journal of Geophysical Research: Solid Earth*, 122(7):5561–5577, 2017. ISSN 2169-9356. doi: 10.1002/2017JB014025.

M.-A. Meier, T. Heaton, and J. Clinton. Evidence for universal earthquake rupture initiation behavior. *Geophysical Research Letters*, 43(15):7991–7996, Aug. 2016. ISSN 00948276. doi: 10.1002/2016GL070081.

M.-A. Meier, J. P. Ampuero, and T. H. Heaton. The hidden simplicity of subduction megathrust earthquakes. *Science*, 357(6357):1277–1281, Sept. 2017. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aan5643.

M.-A. Meier, Y. Kodera, M. Böse, A. Chung, M. Hoshiba, E. Cochran, S. Minson, E. Hauksson, and T. Heaton. How Often Can Earthquake Early Warning Systems Alert Sites With High-Intensity Ground Motion? *Journal of Geophysical Research: Solid Earth*, 125(2):e2019JB017718, 2020. ISSN 2169-9356. doi: 10.1029/2019JB017718.

M.-A. Meier, J.-P. Ampuero, E. Cochran, and M. Page. Apparent earthquake rupture predictability. *Geophysical Journal International*, 225(1):657–663, 2021. doi: 10.1093/gji/ggaa610.

D. Melgar and G. P. Hayes. Systematic Observations of the Slip Pulse Properties of Large Earthquake Ruptures. *Geophysical Research Letters*, 44(19):9691–9698, 2017. ISSN 1944-8007. doi: 10.1002/2017GL074916.

D. Melgar and G. P. Hayes. Characterizing large earthquakes before rupture is complete. *Science Advances*, 5(5):eaav2032, May 2019. ISSN 2375-2548. doi: 10.1126/sciadv.aav2032.

Met Office. *Cartopy: a cartographic python library with a matplotlib interface*. Exeter, Devon, 2010 - 2015. URL `http://scitools.org.uk/cartopy`.

A. Michelini, S. Cianetti, S. Gaviano, C. Giunchi, D. Jozinović, and V. Lauciani. INSTANCE – the Italian seismic dataset for machine learning. *Earth System Science Data*, 13(12):5509–5544, 2021.

S. E. Minson, M.-A. Meier, A. S. Baltay, T. C. Hanks, and E. S. Cochran. The limits of earthquake early warning: Timeliness of ground motion estimates. *Science Advances*, 4(3):eaaq0504, Mar. 2018. ISSN 2375-2548. doi: 10.1126/sciadv.aaq0504.

S. E. Minson, A. S. Baltay, E. S. Cochran, T. C. Hanks, M. T. Page, S. K. McBride, K. R. Milner, and M.-A. Meier. The Limits of Earthquake Early Warning Accuracy and Best Alerting Strategy. *Scientific Reports*, 9(1):2478, Feb. 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-39384-y.

G. Montavon, S. Lapuschkin, A. Binder, W. Samek, and K.-R. Müller. Explaining nonlinear classification decisions with deep Taylor decomposition. *Pattern Recognition*, 65: 211–222, 2017.

J. Mori and H. Kanamori. Initial rupture of earthquakes in the 1995 Ridgecrest, California Sequence. *Geophysical Research Letters*, 23(18):2437–2440, 1996. ISSN 1944-8007. doi: 10.1029/96GL02491.

N. Mori, T. Takahashi, T. Yasuda, and H. Yanagisawa. Survey of 2011 Tohoku earthquake tsunami inundation and run-up. *Geophysical research letters*, 38(7), 2011.

S. Mousavi, A. Bagchi, and V. K. Kodur. Review of post-earthquake fire hazard to building structures. *Canadian Journal of Civil Engineering*, 35(7):689–698, 2008.

S. M. Mousavi and G. C. Beroza. Bayesian-deep-learning estimation of earthquake location from single-station observations. *IEEE Transactions on Geoscience and Remote Sensing*, 58(11):8211–8224, 2020a.

S. M. Mousavi and G. C. Beroza. A Machine-Learning Approach for Earthquake Magnitude Estimation. *Geophysical Research Letters*, 47(1):e2019GL085976, 2020b. ISSN 1944-8007. doi: 10.1029/2019GL085976.

S. M. Mousavi, Y. Sheng, W. Zhu, and G. C. Beroza. STanford EArthquake Dataset (STEAD): A Global Data Set of Seismic Signals for AI. *IEEE Access*, pages 1–1, 2019a. ISSN 2169-3536. doi: 10.1109/ACCESS.2019.2947848.

S. M. Mousavi, W. Zhu, Y. Sheng, and G. C. Beroza. CRED: A deep residual network of convolutional and recurrent units for earthquake signal detection. *Scientific reports*, 9 (1):1–14, 2019b.

S. M. Mousavi, W. L. Ellsworth, W. Zhu, L. Y. Chuang, and G. C. Beroza. Earthquake transformer - an attentive deep-learning model for simultaneous earthquake detection and phase picking. *Nature Communications*, 11(1):3952, Aug. 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-17591-w.

J. Münchmeyer, D. Bindi, C. Sippl, U. Leser, and F. Tilmann. Low uncertainty multifeature magnitude estimation with 3-D corrections and boosting tree regression: Application to North Chile. *Geophysical Journal International*, 220(1):142–159, Jan. 2020. ISSN 0956-540X. doi: 10.1093/gji/ggz416.

J. Münchmeyer, D. Bindi, U. Leser, and F. Tilmann. Earthquake magnitude and location estimation from real time seismic waveforms with a transformer network. *Geophysical Journal International*, 226(2):1086–1104, 2021a. ISSN 0956-540X. doi: 10.1093/gji/ggab139.

J. Münchmeyer, D. Bindi, U. Leser, and F. Tilmann. The transformer earthquake alerting model: A new versatile approach to earthquake early warning. *Geophysical Journal International*, 225(1):646–656, 2021b. ISSN 0956-540X. doi: 10.1093/gji/ggaa609.

J. Münchmeyer, J. Woollam, A. Rietbrock, F. Tilmann, D. Lange, T. Bornstein, T. Diehl, C. Giunchi, F. Haslinger, D. Jozinović, et al. Which picker fits my data? A quantitative evaluation of deep learning based seismic pickers. *Journal of Geophysical Research: Solid Earth*, page e2021JB023499, 2022. doi: 10.1029/2021JB023499.

V. Muthukumar, K. Vodrahalli, V. Subramanian, and A. Sahai. Harmless interpolation of noisy data in regression. *IEEE Journal on Selected Areas in Information Theory*, 1 (1):67–83, 2020.

J. Nábělek. *Determination of earthquake source parameters from inversion of body waves*. PhD thesis, M. I. T., Dept. of Earth, Atmospheric and Planetary Sciences, 1984.

J. Nábělek and G. Xia. Moment-tensor analysis using regional data: Application to the 25 March, 1993, Scotts Mills, Oregon, Earthquake. *Geophysical Research Letters*, 22 (1):13–16, 1995.

V. Nair and G. E. Hinton. Rectified linear units improve restricted Boltzmann machines. In *International Conference on Machine Learning*, 2010.

M. Nakatani, S. Kaneshima, and Y. Fukao. Size-dependent microearthquake initiation inferred from high-gain and low-noise observations at Nikko district, Japan. *Journal of Geophysical Research: Solid Earth*, 105(B12):28095–28109, 2000. ISSN 2156-2202. doi: 10.1029/2000JB900255.

K. Nanjo and A. Yoshida. Changes in the b value in and around the focal areas of the M6. 9 and M6. 8 earthquakes off the coast of Miyagi prefecture, Japan, in 2021. *Earth, Planets and Space*, 73(1):1–10, 2021.

National Research Institute For Earth Science And Disaster Resilience. NIED K-NET, KiK-net, 2019. doi: 10.17598/NIED.0004.

A. Nguyen, A. Dosovitskiy, J. Yosinski, T. Brox, and J. Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. *Advances in neural information processing systems*, 29:3387–3395, 2016.

S. Noda and W. L. Ellsworth. Scaling relation between earthquake magnitude and the departure time from P wave similar growth. *Geophysical Research Letters*, 43(17): 9053–9060, 2016. ISSN 1944-8007. doi: 10.1002/2016GL070069.

OGS (Istituto Nazionale di Oceanografia e di Geofisica Sperimentale). North-East Italy Seismic Network (NEI), 2016. doi: 10.7914/SN/OX.

OGS (Istituto Nazionale di Oceanografia e di Geofisica Sperimentale) and University of Trieste. North-East Italy Broadband Network (NI), 2002. doi: 10.7914/SN/NI.

T. Okuda and S. Ide. Hierarchical rupture growth evidenced by the initial seismic waveforms. *Nature Communications*, 9(1):3714, Sept. 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-06168-3.

E. L. Olson and R. M. Allen. The deterministic nature of earthquake rupture. *Nature*, 438(7065):212–215, Nov. 2005. ISSN 1476-4687. doi: 10.1038/nature04214.

D. Ormoneit and V. Tresp. Improved Gaussian Mixture Density Estimates Using Bayesian Penalty Terms and Network Averaging. *Neural information processing systems*, page 7, 1995.

R. Otake, J. Kurima, H. Goto, and S. Sawada. Deep Learning Model for Spatial Interpolation of Real-Time Seismic Intensity. *Seismological Research Letters*, 91(6):3433–3443, Nov. 2020. ISSN 0895-0695. doi: 10.1785/0220200006.

S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

Y. Park, G. C. Beroza, and W. L. Ellsworth. A Deep Earthquake Catalog for Oklahoma and Southern Kansas Reveals Extensive Basement Fault Networks. *ESSOAr preprint*, 2021. doi: 10.1002/essoar.10508504.1.

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

T. Perol, M. Gharbi, and M. Denolle. Convolutional neural network for earthquake detection and location. *Science Advances*, 4(2):e1700578, Feb. 2018. ISSN 2375-2548. doi: 10.1126/sciadv.1700578.

M. Picozzi, D. Bindi, D. Spallarossa, D. Di Giacomo, and A. Zollo. A rapid response magnitude scale for timely assessment of the high frequency seismic radiation. *Sci. Rep.*, 8(1), Dec. 2018. ISSN 2045-2322. doi: 10.1038/s41598-018-26938-9.

Presidency of Counsil of Ministers - Civil Protection Department. Italian Strong Motion Network (RAN), 1972. doi: 10.7914/SN/IT.

N. Qian. On the momentum term in gradient descent learning algorithms. *Neural networks*, 12(1):145–151, 1999.

J. Quinteros, A. Strollo, P. L. Evans, W. Hanka, A. Heinloo, S. Hemmleb, L. Hillmann, K.-H. Jaeckel, R. Kind, J. Saul, et al. The GEOFON program in 2020. *Seismological Society of America*, 92(3):1610–1622, 2021.

M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

RESIF - Réseau Sismologique et géodésique Français. RESIF-RLBP French Broad-band network, RESIF-RAP strong motion network and other seismic stations in metropolitan France, 1995a. doi: 10.15778/RESIF.FR.

RESIF - Réseau Sismologique et géodésique Français. Réseau Accélérométrique Permanent (French Accelerometrique Network) (RAP), 1995b. doi: 10.15778/RESIF.RA.

L. Retailleau, J.-M. Saurel, W. Zhu, C. Satriano, G. C. Beroza, S. Issartel, P. Boissier, OVPF Team, and OVSM Team. A wrapper to use a machine-learning-based algorithm for earthquake monitoring. *Seismological Research Letters*, 2022.

D. A. Rhoades, A. Christophersen, and S. Hainzl. Statistical seismology. *Encyclopedia of Solid Earth Geophysics*, pages 1–5, 2019.

C. F. Richter. An instrumental earthquake magnitude scale. *Bulletin of the Seismological Society of America*, 25(1):1–32, Jan. 1935. ISSN 0037-1106.

E. Roland and J. J. McGuire. Earthquake swarms on transform faults. *Geophysical Journal International*, 178(3):1677–1690, 2009.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

Z. E. Ross, M.-A. Meier, and E. Hauksson. P Wave Arrival Picking and First-Motion Polarity Determination With Deep Learning. *J. Geophys. Res. Solid Earth*, 123(6): 5120–5129, June 2018a. ISSN 2169-9356. doi: 10.1029/2017JB015251.

Z. E. Ross, M.-A. Meier, E. Hauksson, and T. H. Heaton. Generalized seismic phase detection with deep learning. *Bulletin of the Seismological Society of America*, 108 (5A):2894–2901, 2018b.

Z. E. Ross, Y. Yue, M.-A. Meier, E. Hauksson, and T. H. Heaton. Phaselink: A deep learning approach to seismic phase association. *Journal of Geophysical Research: Solid Earth*, 124(1):856–869, 2019.

B. Rouet-Leduc, C. Hulbert, N. Lubbers, K. Barros, C. J. Humphreys, and P. A. Johnson. Machine learning predicts laboratory earthquakes. *Geophysical Research Letters*, 44 (18):9276–9282, 2017.

B. Rouet-Leduc, C. Hulbert, and P. A. Johnson. Continuous chatter of the Cascadia subduction zone revealed by machine learning. *Nature Geoscience*, 12(1):75, Jan. 2019. ISSN 1752-0908. doi: 10.1038/s41561-018-0274-6.

B. Rouet-Leduc, C. Hulbert, I. W. McBrearty, and P. A. Johnson. Probing slow earthquakes with deep learning. *Geophysical Research Letters*, 47(4):e2019GL085870, 2020.

S. Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

S. Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

S. Ruiz, F. Aden-Antoniow, J. Baez, C. Otarola, B. Potin, F. Del Campo, P. Poli, C. Flores, C. Satriano, F. Leyton, et al. Nucleation phase and dynamic inversion of the Mw 6.9 Valparaíso 2017 earthquake in Central Chile. *Geophysical Research Letters*, 44(20): 10–290, 2017.

S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry. How does batch normalization help optimization? In *Proceedings of the 32nd international conference on neural information processing systems*, pages 2488–2498, 2018.

H. Sato, M. C. Fehler, and T. Maeda. *Seismic wave propagation and scattering in the heterogeneous earth.* Springer Science & Business Media, 2012.

C. Satriano, L. Elia, C. Martino, M. Lancieri, A. Zollo, and G. Iannaccone. PRESTo, the earthquake early warning system for Southern Italy: Concepts, capabilities and future perspectives. *Soil Dynamics and Earthquake Engineering*, 31(2):137–153, Feb. 2011. ISSN 0267-7261. doi: 10.1016/j.soildyn.2010.06.008.

M. K. Savage and J. G. Anderson. A local-magnitude scale for the western Great Basin-eastern Sierra Nevada from synthetic Wood-Anderson seismograms. *BSSA*, 85(4):1236–1243, Aug. 1995. ISSN 0037-1106.

F. Scherbaum and M.-P. Bouin. FIR filter effects and nucleation phases. *Geophysical Journal International*, 130(3):661–668, Sept. 1997. ISSN 0956-540X. doi: 10.1111/j.1365-246X.1997.tb01860.x.

C. H. Scholz. *The mechanics of earthquakes and faulting.* Cambridge university press, 2nd edition, 2012.

Scripps Institution Of Oceanography. IRIS/IDA Seismic Network, 1986. doi: 10.7914/SN/II.

L. Seydoux, R. Balestriero, P. Poli, M. De Hoop, M. Campillo, and R. Baraniuk. Clustering earthquake signals and background noises in continuous seismic data with unsupervised deep learning. *Nature communications*, 11(1):1–12, 2020.

K. T. Shabestari and F. Yamazaki. A proposal of instrumental seismic intensity scale compatible with mmi evaluated from three-component acceleration records. *Earthquake Spectra*, 17(4):711–723, 2001.

P. M. Shearer. *Introduction to seismology.* Cambridge university press, 2009.

D. R. Shelly, G. C. Beroza, and S. Ide. Non-volcanic tremor and low-frequency earthquake swarms. *Nature*, 446(7133):305–307, 2007.

W. Shih and S. Chai. Data-driven vs. hypothesis-driven research: making sense of big data. In *Academy of Management Proceedings*, number 1 in 2016, page 14843. Academy of Management Briarcliff Manor, NY 10510, 2016.

C. Sippl, B. Schurr, G. Asch, and J. Kummerow. Seismicity Structure of the Northern Chile Forearc From >100,000 Double-Difference Relocated Hypocenters. *J. Geophys. Res. Solid Earth*, 123(5):4063–4087, May 2018. ISSN 2169-9356. doi: 10.1002/2017JB015384.

N. Smirnov. Table for estimating the goodness of fit of empirical distributions. *The annals of mathematical statistics*, 19(2):279–281, 1948.

J. D. Smith, K. Azizzadenesheli, and Z. E. Ross. Eikonet: Solving the eikonal equation with deep neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 2020.

J. D. Smith, Z. E. Ross, K. Azizzadenesheli, and J. B. Muir. HypoSVI: Hypocentre inversion with Stein variational inference and physics informed neural networks. *Geophysical Journal International*, 228(1):698–710, 2022.

J. Snoek, Y. Ovadia, E. Fertig, B. Lakshminarayanan, S. Nowozin, D. Sculley, J. Dillon, J. Ren, and Z. Nado. Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*, pages 13969–13980, 2019.

A. Socquet, J. P. Valdes, J. Jara, F. Cotton, A. Walpersdorf, N. Cotte, S. Specht, F. Ortega-Culaciati, D. Carrizo, and E. Norabuena. An 8 month slow slip event triggers progressive nucleation of the 2014 Chile megathrust. *Geophysical Research Letters*, 44 (9):4046–4053, 2017.

C. Song, T. Alkhalifah, and U. B. Waheed. A versatile framework to solve the Helmholtz equation using physics-informed neural networks. *Geophysical Journal International*, 228(3):1750–1762, 2022.

D. Spallarossa, S. R. Kotha, M. Picozzi, S. Barani, and D. Bindi. On-site earthquake early warning: A partially non-ergodic perspective from the site effects point of view. *Geophys. J. Int.*, 216(2):919–934, Feb. 2019. ISSN 0956-540X, 1365-246X. doi: 10.1093/gji/ggy470.

S. Stein and M. Wysession. *An introduction to seismology, earthquakes, and earth structure*. Blackwell Publishing, 2003.

J. M. Stokes, K. Yang, K. Swanson, W. Jin, A. Cubillos-Ruiz, N. M. Donghia, C. R. MacNair, S. French, L. A. Carfrae, Z. Bloom-Ackermann, et al. A deep learning approach to antibiotic discovery. *Cell*, 180(4):688–702, 2020.

D. A. Storchak, J. Schweitzer, and P. Bormann. The IASPEI standard seismic phase list. *Seismological Research Letters*, 74(6):761–772, 2003.

T. Swift and further authors. Shake it off: A collection of inspirational tunes, 2022. URL `https://open.spotify.com/playlist/1uqU9ZQBikHvl4COCYZV5B?si=2f50f380f43b4cd1`.

Y. J. Tan, F. Waldhauser, W. L. Ellsworth, M. Zhang, W. Zhu, M. Michele, L. Chiaraluce, G. C. Beroza, and M. Segou. Machine-Learning-Based High-Resolution Earthquake Catalog Reveals How Complex Fault Structures Were Activated during the 2016–2017 Central Italy Sequence. *The Seismic Record*, 1(1):11–19, 2021.

The pandas development team. pandas-dev/pandas: Pandas, Feb. 2020. doi: 10.5281/zenodo.3509134.

The Pyrocko Developers. Pyrocko: A versatile seismology toolkit for Python., 2018. URL `http://pyrocko.org`. doi: 10.5880/GFZ.2.1.2017.001.

A. Trnkoczy. Understanding and parameter setting of STA/LTA trigger algorithm. In *New Manual of Seismological Observatory Practice (NMSOP)*, pages 1–20. Deutsches GeoForschungsZentrum GFZ, 2009.

D. T. Trugman, M. T. Page, S. E. Minson, and E. S. Cochran. Peak Ground Displacement Saturates Exactly When Expected: Implications for Earthquake Early Warning. *Journal of Geophysical Research: Solid Earth*, 124(5):4642–4653, 2019. ISSN 2169-9356. doi: 10.1029/2018JB017093.

G.-A. Tselentis and L. Danciu. Empirical relationships between modified Mercalli intensity and engineering ground-motion parameters in Greece. *Bulletin of the Seismological Society of America*, 98(4):1863–1875, 2008.

H. Ueno, S. Hatakeyama, J. Aketagawa, J. Funasaki, and N. Hamada. Improvement of hypocenter determination procedures in the Japan Meteorological Agency. *Quart. J. Seism.*, 65:123–134, 2002.

Universidad de Chile. Red Sismologica Nacional, 2013. doi: 10.7914/SN/C1.

Universita della Basilicata. UniBAS, 2005. doi: 10.17598/NIED.0004.

University of Genova. Regional Seismic Network of North Western Italy. International Federation of Digital Seismograph Networks, 1967. doi: 10.7914/SN/GU.

U.S. Geological Survey. Advanced National Seismic System (ANSS) Comprehensive Catalog of Earthquake Events and Products, 2017. doi: 10.5066/F7MS3QZH.

M. Vallée and V. Douet. A new database of source time functions (STFs) extracted from the SCARDEC method. *Physics of the Earth and Planetary Interiors*, 257:149–157, Aug. 2016. ISSN 00319201. doi: 10.1016/j.pepi.2016.05.012.

M. Vallée, J. Charléty, A. M. G. Ferreira, B. Delouis, and J. Vergoz. SCARDEC: A new technique for the rapid determination of seismic moment magnitude, focal mechanism and source time functions for large earthquakes using body-wave deconvolution. *Geophysical Journal International*, 184(1):338–358, Jan. 2011. ISSN 0956-540X. doi: 10.1111/j.1365-246X.2010.04836.x.

M. P. A. van den Ende and J.-P. Ampuero. Automated Seismic Source Characterization Using Deep Graph Neural Networks. *Geophysical Research Letters*, 47(17): e2020GL088690, 2020. ISSN 1944-8007. doi: 10.1029/2020GL088690.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

E. von Rebeur-Paschwitz. The earthquake of Tokio, April 18, 1889. *Nature*, 40(1030): 294–295, 1889.

D. J. Wald, V. Quitoriano, T. H. Heaton, and H. Kanamori. Relationships between Peak Ground Acceleration, Peak Ground Velocity, and Modified Mercalli Intensity in California. *Earthquake Spectra*, 15(3):557–564, Aug. 1999. ISSN 8755-2930. doi: 10.1193/1.1586058.

Z. Wang. Seismic hazard vs. seismic risk. *Seismological Research Letters*, 80(5):673–674, 2009.

M. L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021. doi: 10.21105/joss.03021. URL `https://doi.org/10.21105/joss.03021`.

P. Wigger, P. Salazar, J. Kummerow, W. Bloch, G. Asch, and S. Shapiro. West-Fissure-and Atacama-Fault Seismic Network (2005/2012), 2016. doi: 10.14470/3s7550699980.

D. R. Wilson and T. R. Martinez. The general inefficiency of batch training for gradient descent learning. *Neural networks*, 16(10):1429–1451, 2003.

J. Woollam, J. Münchmeyer, F. Tilmann, A. Rietbrock, D. Lange, T. Bornstein, T. Diehl, C. Giunchi, F. Haslinger, D. Jozinović, A. Michelini, J. Saul, and H. Soto. SeisBench - A toolbox for machine learning in seismology. *Seismological Research Letters (accepted)*, 2022.

Y. Yang, A. F. Gao, J. C. Castellanos, Z. E. Ross, K. Azizzadenesheli, and R. W. Clayton. Seismic wave propagation and inversion with neural operators. *The Seismic Record*, 1 (3):126–134, 2021.

L. Ye, T. Lay, H. Kanamori, and L. Rivera. Rupture characteristics of major and great (Mw $\geq$ 7.0) megathrust earthquakes from 1990 to 2015: 1. Source parameter scaling relationships. *Journal of Geophysical Research: Solid Earth*, 121(2):826–844, 2016. ISSN 2169-9356. doi: 10.1002/2015JB012426.

W. L. Yeck, J. M. Patton, Z. E. Ross, G. P. Hayes, M. R. Guy, N. B. Ambruz, D. R. Shelly, H. M. Benz, and P. S. Earle. Leveraging Deep Learning in Global 24/7 Real-Time Earthquake Monitoring at the National Earthquake Information Center. *Seismological Society of America*, 92(1):469–480, 2021.

C. E. Yoon, O. O'Reilly, K. J. Bergen, and G. C. Beroza. Earthquake detection through computationally efficient similarity search. *Science advances*, 1(11):e1501057, 2015.

R. Zhang. Making convolutional networks shift-invariant again. In *International conference on machine learning*, pages 7324–7334. PMLR, 2019.

X. Zhang, M. Zhang, and X. Tian. Real-Time Earthquake Early Warning With Deep Learning: Application to the 2016 M 6.0 Central Apennines, Italy Earthquake. *Geophysical Research Letters*, 48(5):2020GL089394, 2021.

X. Zhao, A. Curtis, and X. Zhang. Bayesian seismic tomography using normalizing flows. *Geophysical Journal International*, 228(1):213–239, 2022.

W. Zhu and G. C. Beroza. PhaseNet: a deep-neural-network-based seismic arrival-time picking method. *Geophysical Journal International*, 216(1):261–273, 2019.

W. Zhu, S. M. Mousavi, and G. C. Beroza. Seismic signal denoising and decomposition using deep neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 57(11):9476–9488, 2019.

W. Zhu, A. Hou, R. Yang, A. Datta, M. Mousavi, M. Zhang, Y. Park, I. McBrearty, W. Ellsworth, and G. Beroza. Quakeflow: A scalable deep-learning-based earthquake monitoring workflow with cloud computing. In *AGU Fall Meeting Abstracts*, volume 2021, 2021.

A. Zollo, M. Lancieri, and S. Nielsen. Earthquake magnitude estimation from peak amplitudes of very early seismic signals on strong motion records. *Geophys. Res. Lett.*, 33 (23), Dec. 2006. ISSN 0094-8276. doi: 10.1029/2006GL027795.

Appendix

A Software acknowledgements

This work relies heavily on open source software. While citations are provided in the main text where appropriate, we want to use this place for a condensed account of open source software central to this thesis. Among others, we used the following open source software (in alphabetic order):

- cartopy [Met Office, 2010 - 2015]
- matplotlib [Hunter, 2007]
- numpy [Harris et al., 2020]
- obspy [Beyreuther et al., 2010]
- pandas [The pandas development team, 2020]
- pyrocko [The Pyrocko Developers, 2018]
- pytorch [Paszke et al., 2019]
- scipy [Virtanen et al., 2020]
- seaborn [Waskom, 2021]
- tensorflow [Abadi et al., 2016]

We use scientific colour scales from Crameri [2018]. We thank all authors for making these tools openly available.

B Supplement to Chapter 3

B.1 High-pass frequency selection

Table B.1 shows the candidate intervals for high-pass filtering. The last line indicates the fall-back filter, which is used for all events for which the minimum SNR of 4 is not attained with any of the other filters. For velocity (acceleration) the SNR is larger than 2 in 96% (98%) of the waveforms, whereas for displacement this is only true for 70%. Therefore in some cases, particularly for features based on displacement, some of our data might be strongly affected by ambient noise. We nonetheless do not remove these measurements, as the information that the feature is close to noise is still valuable.

The distribution of chosen high-pass frequencies by event magnitude is shown in Figure B.1. As expected, for larger events lower frequencies are chosen. Especially for the largest events, only the lowest frequencies are chosen.

Table B.1: Intervals for high-pass filtering

f_{low} [Hz]	f_{high} [Hz]
0.001	0.3
0.1	0.5
0.3	1.0
0.5	1.5
0.75	-

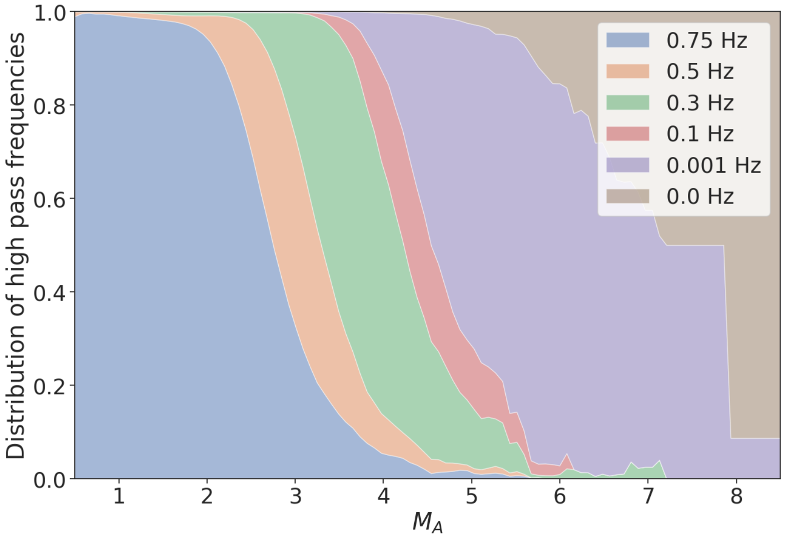


Figure B.1: Distribution of applied high-pass frequencies by event magnitude. Strong motion records were not high-pass filtered and are therefore denoted with a high-pass frequency of 0 Hz.

B.2 Choice of hyperparameters and envelope times

In this section, we give some advice on the selection of hyperparameters and envelope delays. As the experiments, both feature extraction and calibration of the correction functions, are computationally expensive, a grid search for hyperparameter selection is intractable. Hyperparameters, therefore, need to be tuned by hand. Therefore, we explain the significance of and interaction between the different hyperparameters. For practical applications, we suggest starting with the hyperparameters used in this study.

λ_r and λ_d determine the smoothness of the distance-depth correction function. We settled for a higher value of λ_r as we expect a generally lower lateral than vertical variability in ground structure. Both values might need to be increased in the presence of fewer data points and the other way around. λ_d should be increased, if fewer M_w values are available for the calibration of attenuation with depth. The choice of suitable values can be assisted by plots, as in Figure 3.9.

λ_L controls the level of deviation from the distance-depth correction that is caused by the source-path correction. It interacts with the number of neighbours k chosen for averaging and the subsampling rate $|\bar{E}_s|/|E_s|$. In general, a low number of neighbours k or a high subsampling requires a higher λ_L, as the number of free parameters is increased and the parameters are less constrained by the data.

k determines the smoothing of the source-path correction. A higher value will generally cause a smoother function, while a lower value will cause a rougher function. In contrast, a higher subsampling rate (at constant k) will cause a rougher function, a lower subsampling rate a smoother function. The choice of subsampling rate will most likely be governed by the available computational capacities. We experienced a superquadratic increase in runtime and memory consumption with the subsampling rate. If the computational capacities are limiting factors, we recommend slowly increasing the subsampling rate and observing the effect on RMSE.

λ_{M_w} determines the trade-off between the deviation from the prescribed M_w values and the smoothness of the correction functions. A higher value λ_w will lead to a smaller

Table B.2: Hyperparameters used for the correction functions

Hyperparameter	Value
G	$\{20\text{ km} + 9.8\text{ km} * i \mid i \in \{0, 49\}\}$ $\times$ $\{10\text{ km} + 10\text{ km} * i \mid i \in \{0, 19\}\}$
λ_r	10^3 km^4
λ_d	10^2 km^4
λ_L	10
λ_{M_w}	10^{-1}
k	10
$\lvert\bar{E}_s\rvert/\lvert E_s\rvert$	10^{-1}

deviation from M_w but increases the roughness of the correction functions. As the calibration with M_w is mostly required for the calibration of the depth-dependent attenuation, we generally recommend small values for λ_{M_w}.

A good measure for the suitability of hyperparameters is the difference between the RMSE on the training and development sets. In general, we recommend a slightly higher RMSE on the training set, indicating some level of overfitting. No overfitting at all suggests that the model is regularised too strongly, while strong deviations between the training and development performance suggest that overfitting negatively impacts performance on the development and test set.

For the envelope delays, we chose 5 s and 20 s. The 5 s value is intended to capture the early high energy portion of the event and provides a more stable measurement than the peak. We tried putting the second value as late as possible to approach the diffusive regime and thereby minimise the effects of the radiation pattern and distance uncertainties. As most of our events are small, we can not resort to the classical rule of assuming a diffusive regime after twice the S wave travel time, as this value is below noise level for most measurements. Therefore, we needed to find a sensible trade-off between diffusiveness and SNR. Whereas we did not carry out systematic testing, we confirmed 20 s as a good choice by comparing the value of the envelope at this time to the noise level 5 s before the P pick, as measured by the envelope value. We found that the noise exceeds the signal in only $\sim 3\%$ of cases. In addition, we expect the boosting tree to appropriately handle low-SNR 20 s envelope values.

The proper choice of envelope delays will usually depend on the dataset. In our case, we had a favourable dataset for long envelope delays, as most IPOC stations are low noise hard rock stations. To choose appropriate values we recommend first visually inspecting the signal envelopes for a subsample of the measurements and second looking at the SNRs for multiple candidate delay times. It is possible to include more than two envelope times. We did not conduct experiments with more than two envelope times, due to computational constraints.

B.3 Determination of moment magnitudes for moderate-size events

The global CMT catalog only covers earthquakes above moment magnitude 5–5.5 reliably. To extend our database of events with M_W, additional moment magnitudes were determined with regional moment tensor inversion with the approach of Nábělek [1984] and Nábělek and Xia [1995]. We constrained moment tensors to be deviatoric (i.e. no isotropic component), used the period band between 10 s and 35 s and assumed quality

Table B.3: Hyperparameters used in the boosting experiments. We use the naming conventions from XGBoost. We only denote parameters that were changed from the defaults for XGBoost version 0.80.

Hyperparameter	Value
Depth	11
Epochs	250
Eta	0.1

factors (inverse attenuation) of 225 for P and 100 for S waves for the calculation of Green's functions. Scalar moments were converted to moment magnitudes using the relation of Hanks and Kanamori [1979]. At the utilised long periods, physical attenuation effects only play a minor role.

B.4 Effect of SNR thresholding

No explicit SNR threshold is imposed but an implicit threshold exists because the dataset is assembled based on pre-existing picks, which require reasonable visibility of at least the P wave. We analysed the impact of imposing an additional SNR threshold on the RMSE and the resulting uncertainties (Figure B.2), initially using the vertical displacement magnitude as an example. We obtain the noise level for this analysis as the peak value in the 30 s before the P pick, with an additional safety margin of 1 s. For each SNR threshold, we calculate the RMSE using only measurements with a higher SNR and estimate the uncertainty on the mean. We estimate the uncertainty as the RMSE divided by the square root of the number of stations for each event minus one. As a higher SNR threshold causes a lower number of measurements, the average uncertainty can increase, even if the RMSE falls.

As we see in B.2, the RMSE falls for SNRs of up to ~ 2 and grows afterwards. The growth can be explained by the fact that measurements with a higher SNR are more often from events with higher magnitudes, which exhibit an increased RMSE in general (Figure 3.6). In contrast to the RMSE, the uncertainty does not show any decreasing behaviour, but a steady growth due to the decreasing number of measurements. We observe similar behaviour for velocity and acceleration. This means that the general quality of our estimates is highest if we do not impose a further SNR threshold. In addition, we expect the boosting tree regression to act as denoising, as it combines multiple features representing different frequency spectra.

B.5 Determination of magnitude uncertainties

To obtain magnitude values and uncertainties for each event, we combine the measurements from multiple stations. As the results from multiple stations might not be independent, the stated uncertainty of the magnitude estimate could be erroneous if it is calculated by ignoring possible correlations. Figures B.3 and B.4 show the correlations between the residuals at pairs of stations and their dependency on inter-station distance. Interestingly, correlation shows a strong dependence on distance and especially turns negative for distances above $\sim$ 100km. The negative values are partially caused by analysing the residuals with respect to the mean rather than the (unknown) true value. This effect alone causes some apparent negative correlation, but for truly independent errors this would be much smaller than observed.

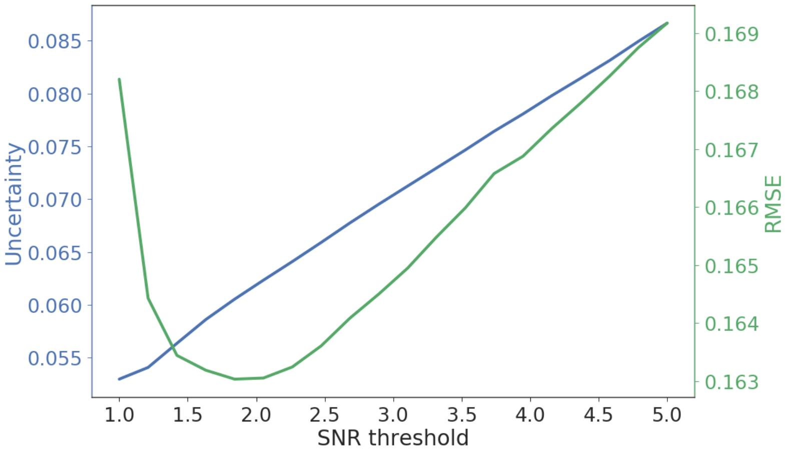


Figure B.2: RMSE and resulting uncertainties for the single feature magnitude scale from displacement on the vertical component at different signal to noise thresholds.

Determining the optimal estimator and the effective sample size has been discussed by Eaton [1992]. Unfortunately, the suggested method uses the inverse of the correlation matrix, which is unstable regarding minor variations of the covariance matrix. This is especially problematic, as we do not have access to the actual correlation matrix, but only to an empirical covariance matrix. In addition, we are missing some elements of the matrix, for stations with too few events in common. Therefore, the proposed method is not applicable.

Nonetheless, we want to present two main results from Eaton [1992]. First, a growing correlation does not always reduce effective sample size but can increase it as well. Second, negative correlations in general increase the effective sample size.

Following these observations, we adapt a simple ad hoc procedure. The mean observed correlation between pairs of stations is close to zero (-0.1). Therefore, we use the mean of all stations as the event magnitude and the standard deviation between the single station estimates divided by the square root of the number of contributing stations minus one as the event magnitude standard deviation. Even though this is not the theoretically optimal way, following the discussion above, we believe this achieves reasonable uncertainty estimates.

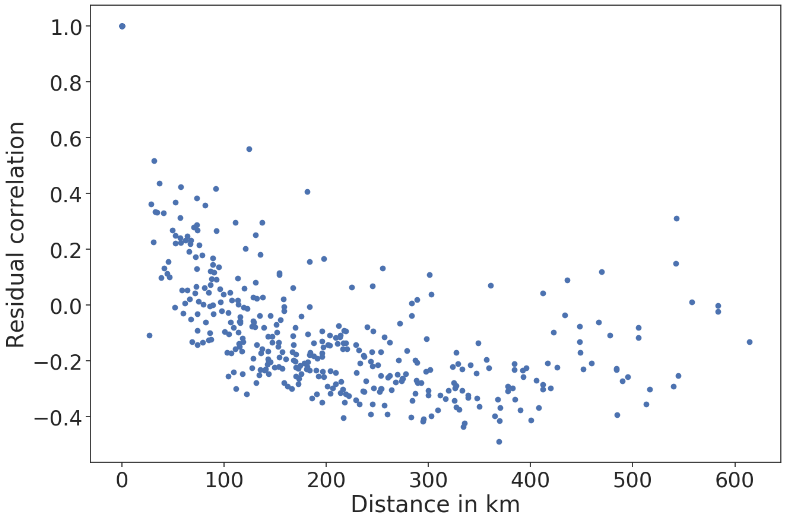


Figure B.3: Empirical correlation of the residuals for peak horizontal displacement as a function of inter-station distance. Each dot represents a pair of stations. Station pairs with less than 500 events in common are discarded.

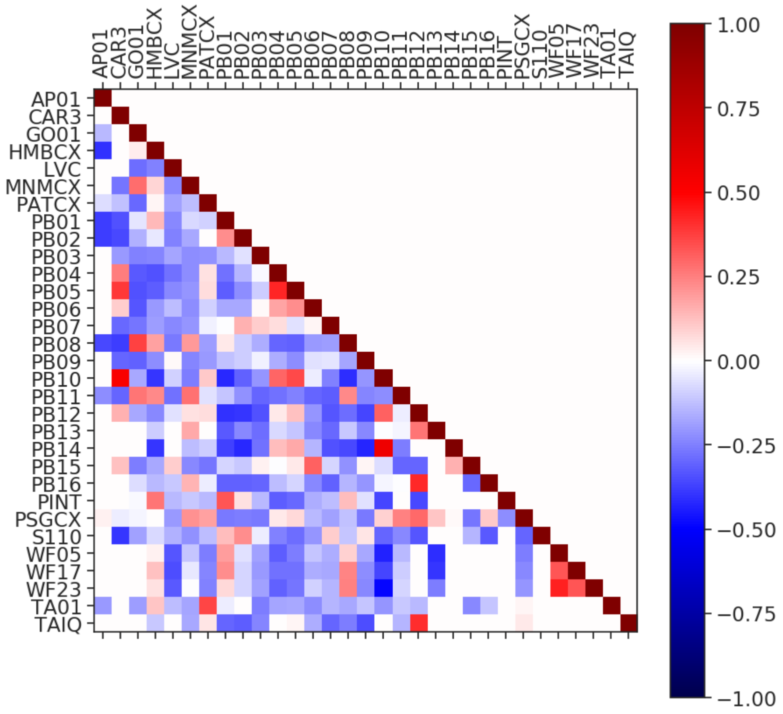


Figure B.4: Empirical correlation of the residuals for peak horizontal displacement for station pairs. Station pairs with less than 500 events in common are discarded.

Table B.4: Seismic networks and stations used. Stations including strong motion records are printed in bold. The stations are identical to those used by Sippl et al. [2018] except that station PB17 from the CX network was removed because it showed non-documented gain changes over time and for the different components.

Network		Stations
MINAS	(5E)	S110
CSN	(C)	AP01 GO01 TA01
IPOC	(CX)	CAR3 **HMBCX** **MNMCX** **PATCX** **PB01** **PB02** **PB03** **PB04** **PB05** **PB06** **PB07** **PB08** **PB09** **PB10** **PB11** **PB12** **PB13** PB14 **PB15** **PB16** **PSGCX** TAIQ
GEOFON	(GE)	LVC
Iquique	(IQ)	PINT
WestFissure	(8F)	WF05 WF17 WF23

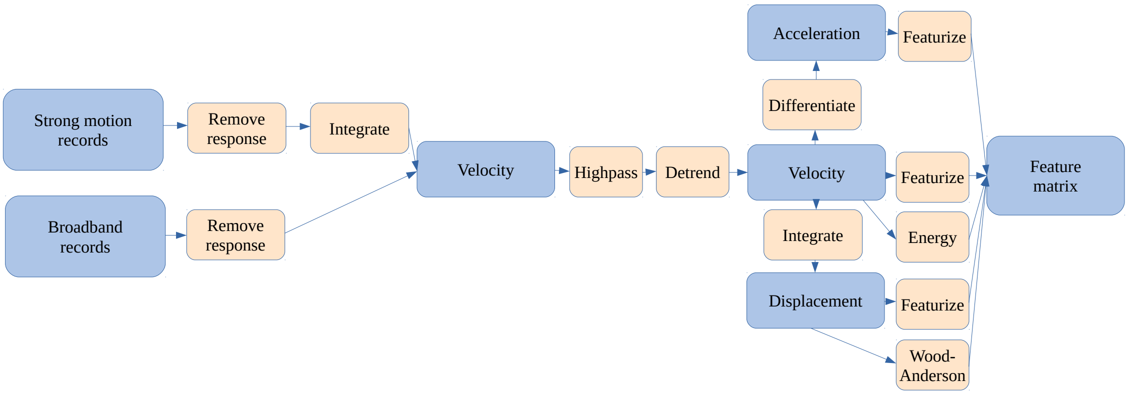


Figure B.5: Schematic overview of the preprocessing and feature extraction workflow. The split into different components is not visualised to keep the figure simple. Featurize refers to the process of extracting the peak and envelope values from the traces.

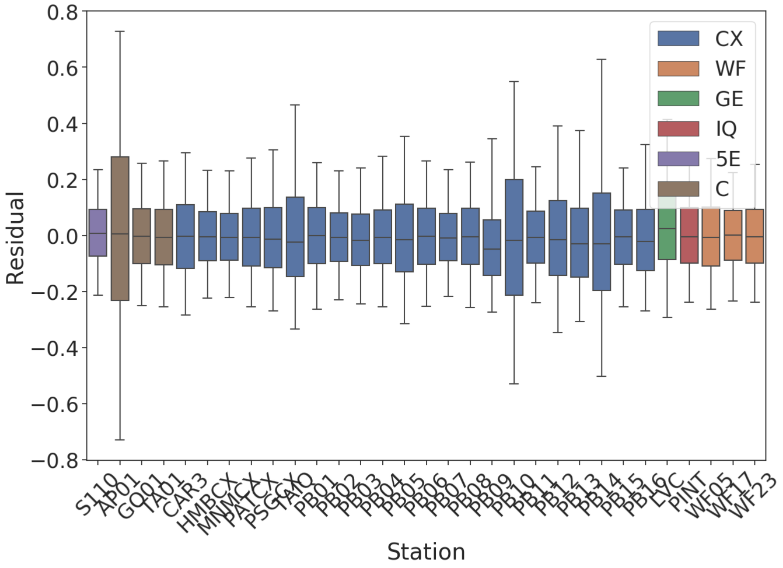


Figure B.6: Residual distribution by stations for displacement on the horizontal component. The middle bar denotes the median, the boxes show the quartile ranges, the whiskers show the 5^{th} and 95^{th} percentiles. Most stations have residuals of similar magnitudes, while a few show significantly higher residuals, e.g. AP01, TAIQ, PB10 and PB15.

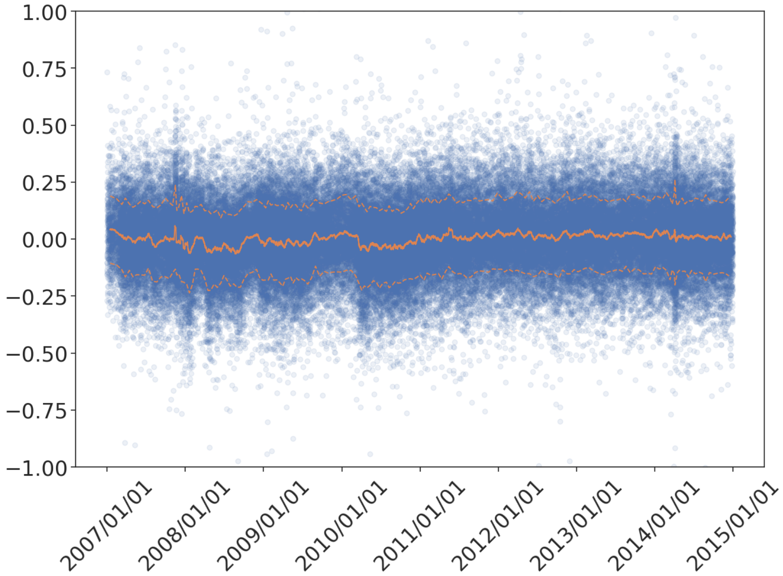


Figure B.7: Development of residuals for displacement on the horizontal component for station PB01 over time. The lines show running mean and standard deviation over 500 consecutive events. While we observed slight changes in the station bias over time, we were not able to ensure that these changes are not caused by measurement artifacts.

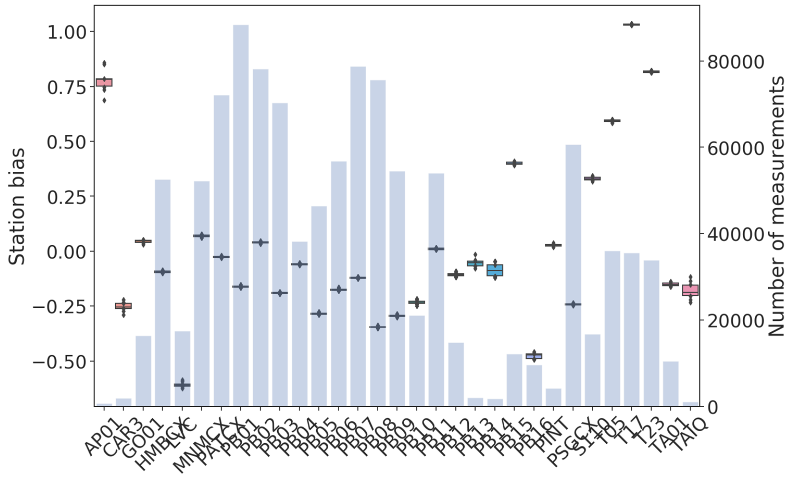


Figure B.8: Station bias for peak displacement on the horizontal component. The bias is shown for ten suboptimisations, each containing 10% of the events. Boxes indicate quartiles. The blue bars show the total number of measurements per station.

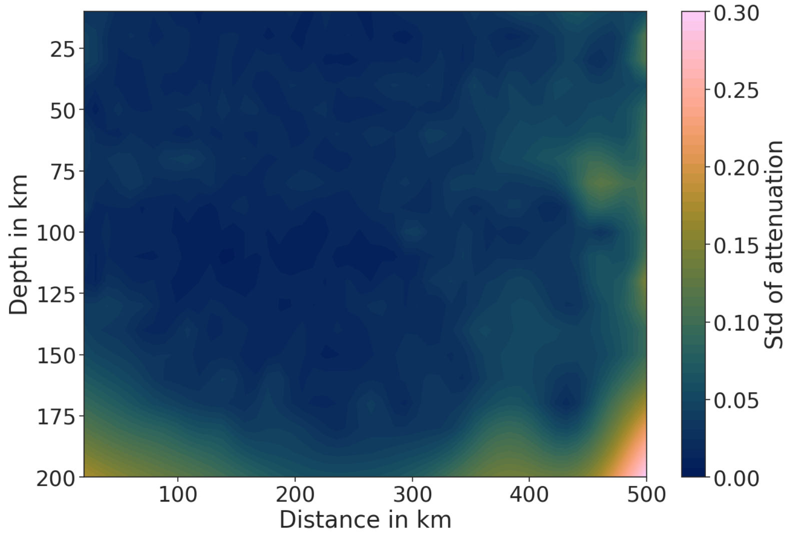


Figure B.9: Standard deviation of the distance and depth correction function for peak displacement on the horizontal component. Standard deviation is calculated across the subsets of a 10-fold split of the full dataset.

C Supplement to Chapter 4

C.1 Data and Preprocessing

For our study, we use two datasets, one from Japan, one from Italy. The Japan dataset consists of 13,512 events between 1997 and 2018 from the NIED KiK-net catalog [National Research Institute For Earth Science And Disaster Resilience, 2019]. The data was obtained from NIED and consists of triggered strong motion records. Each trace contains 15 s of data before the trigger and has a total length of 120 s. Each station consists of two three-component strong motion sensors, one at the surface and one borehole sensor. We split the dataset chronologically with ratios of 60:10:30 between training, development and test set. The training set ends in March 2012, the test set begins in August 2013. Events in between are used as development set. We decided to use a chronological split to ensure a scenario most similar to the actual application in early warning.

The Italy dataset consists of 7,055 events between 2008 and 2019 from the INGV catalog. We use data from the 3A [Istituto Nazionale di Geofisica e Vulcanologia (INGV) et al., 2018], BA [Universita della Basilicata, 2005], FR [RESIF - Réseau Sismologique et géodésique Français, 1995a], GU [University of Genova, 1967], IT [Presidency of Counsil of Ministers - Civil Protection Department, 1972], IV [Istituto Nazionale di Geofisica e Vulcanologia (INGV), Italy, 2006], IX [Dipartimento di Fisica, Università degli studi di Napoli Federico II, 2005], MN [MedNet Project Partner Institutions, 1990], NI [OGS (Istituto Nazionale di Oceanografia e di Geofisica Sperimentale) and University of Trieste, 2002], OX [OGS (Istituto Nazionale di Oceanografia e di Geofisica Sperimentale), 2016], RA [RESIF - Réseau Sismologique et géodésique Français, 1995b], ST [Geological Survey-Provincia Autonoma di Trento, 1981], TV [Istituto Nazionale di Geofisica e Vulcanologia (INGV), 2008] and XO [EMERSITO Working Group, 2018] networks. We use all events from 2016 as test set and the remaining events as training and development sets. The test set consists of 31% of the events, a similar fraction as in the Japan dataset. We shuffle events between training and development set. While a chronological split would have been the default choice, we decided to use 2016 for testing, as it contains a long seismic sequence in central Italy containing several very large events in August and October. Further details on the statistics of both datasets can be found in Table C.4.

Before training, we extract, align and preprocess the waveforms and store them in hdf5 format. As alignment requires the first P pick, we need approximate picks for the datasets. For Japan, we use the trigger times provided by NIED. Our preprocessing accounts for misassociated triggers. For Italy, we use an STA/LTA trigger around the predicted P arrival. While triggering needs to be handled differently in an application scenario, we use this simplified approach as our evaluation metrics depend only very weakly on the precision of the picks.

C.2 TEAM - The transformer earthquake alerting model

C.2.1 Feature extraction network

The feature extraction of TEAM is conducted separately for each station. Nonetheless, the same convolutional neural network (CNN) for feature extraction is applied at all stations, i.e., the same model with the same model weights.

As amplitudes of seismic waveforms can span several orders of magnitude, the first layer of the network normalizes the traces by dividing through their peak value observed so far. All components of one station are normalised jointly, such that the amplitude ratio between the components stays unaltered. Notably, we only use the peak value observed so far, i.e., the waveforms after t_1, which have been blinded with zeros, are not considered, as this would introduce a knowledge leak. As the peak amplitude of the trace is likely a key predictor, we logarithmise the value and concatenate it to the feature vector after passing through all the convolutional layers, prior to the fully connected layers.

We apply a set of convolutional and max-pooling layers to the waveforms. We use convolutional layers as this allows the model to extract translation-invariant features and as convolutional kernels can be interpreted as modelling frequency features. We concatenate the output of the convolutions and the logarithm of the peak amplitude. This vector is fed into a multi-layer perceptron to generate the final feature vector for the station. All layers use ReLu activations. A detailed overview of the number and specifications of the layers in the feature extraction model can be found in Table C.5.

C.2.2 Feature combination network

The feature extraction provides one feature vector per input station representing the waveforms. As an additional input, the model is provided with the location of the stations, represented by latitude, longitude and elevation. The targets for the PGA estimation are specified by the latitude, longitude, and elevation.

We use a transformer network [Vaswani et al., 2017] for the feature combination. Given a set of n input vectors, a transformer produces n output vectors capturing combined information from all the vectors in a learnable way. We use transformers for two main reasons. First, they are permutation equivariant, i.e., changing the order of input or output stations does not have any impact on the output. This is essential, as there exists no natural ordering on the input stations or target locations. Second, they can handle variable input sizes, as the number of parameters of a transformer is independent of the number of input vectors. This property allows applying the model to different sets of stations and a flexible number of target locations.

To incorporate the locations of the stations we use predefined position embeddings. As proposed by Vaswani et al. [2017], we use pairs of sinusoidal functions, $\sin(\frac{2\pi}{\lambda_i}x)$ and $\cos(\frac{2\pi}{\lambda_i}x)$, with different wavelengths λ_i. We use 200 dimensions for latitude and longitude, respectively, and the remaining 100 dimensions for elevation. We anticipate two advantages of sinusoidal embeddings for representing the station position. First, keeping the position embeddings fixed instead of learnable reduces the parameters and therefore likely provides better representations for stations with only a few input measurements or sites not contained in the training set. Second, sinusoidal embeddings guarantee that shifts can be represented by linear transformations, independent of the location it applies to. As the attention mechanism in transformers is built on linear projections and dot products, this should allow for more efficient attention scores at least in the first transformer layers. As proposed in the original transformer paper [Vaswani et al., 2017], the position embeddings are added element-wise to the feature vectors to form the input of the transformer. We calculate position embeddings of the target locations in the same way.

As in our model input and output size of the transformer are identical, we only use the transformer encoder stack [Vaswani et al., 2017] with six encoder layers. Inputs are the feature vectors with position embeddings from all input stations and the position embeddings of the output locations. We apply masking to the attention to ensure that no attention weight is put on the vectors corresponding to the output locations. This guarantees that each target only affects its own PGA value and not any other PGA values. As the self-attention mechanism of the transformer has quadratic computational complexity in the number of inputs, we restrict the maximum number of input stations to 25 (see training details for the selection procedure). Further details on the hyperparameters can be found in Table C.6. The transformer returns one output vector for each input vector. We discard the vectors corresponding to the input stations and only keep the vectors corresponding to the targets.

C.2.3 Mixture density output

Similar to the feature extraction, the output calculation is conducted separately for each target, while sharing the same model and weights between all targets. We use a mixture density network to predict probability densities for the PGA [Bishop, 1994]. We model the probability as a mixture of $m = 5$ Gaussian random variables. Using a mixture of Gaussians instead of a single Gaussian allows the model to predict more complex distributions, like non-Gaussian distributions, e.g., asymmetric distributions. The functional form of the Gaussian mixture is $\sum_{i=1}^{m} \alpha_i \varphi_{\mu_i,\sigma_i}(x)$. We write φ_{μ_i,σ_i} for the density of a standard normal with mean μ_i and standard deviation σ_i. The values α_i are non-negative weights for the different Gaussians with the property $\sum_{i=1}^{m} \alpha_i = 1$. The mixture density network uses a multi-layer perceptron to predict the parameters α_i, μ_i and σ_i. The hidden dimensions are 150, 100, 50, 30, 10. The activation function is ReLu for the hidden layers, linear for the μ outputs, ReLu for the σ outputs, and softmax for the α output.

C.2.4 Training details

We train the model end-to-end using negative log-likelihood as the loss function. All components are trained jointly end-to-end. The model has about 13.3 million parameters in total. To increase the amount of training data and to train the model on shorter segments of data we apply various forms of data augmentation. Each data augmentation is calculated separately each time a particular waveform sample is shown, such that the effective training samples vary.

First, if our dataset contains more stations for an event than the maximum number of 25 allowed by the model, we subsample. We introduce a bias to the subsampling to favour stations closer to the event. We use up to twenty targets for PGA prediction. Similarly to the input station, we subsample if more targets are available and bias the subsampling to stations close to the event. This bias ensures that targets with higher PGA values are shown more often during training.

Second, we apply station blinding, meaning we zero out a set of stations in terms of both waveforms and coordinates. The number of stations to blind is uniformly distributed between zero and the total number of stations available minus one. In combination with the first point, this guarantees that the model also learns to predict PGA values at sites where no waveform inputs are available.

Third, we apply temporal blinding. We uniformly select a time t that is between 1 s before the first P pick and 25 s after. All waveforms are set to zero after time t. The model therefore only uses data available at time t. Even though we never apply TEAM to times before the first P pick, we include these in the training process to ensure TEAM learns a sensible prior distribution. We observed that this leads to better early predictions. As information about the triggering station distribution would introduce a knowledge leak, if available from the beginning, we zero out all waveforms and coordinates from stations that did not trigger until time t.

Fourth, we oversample large magnitude events. As large magnitude events are rare, we artificially increase their number in the training set. An event with magnitude $M \geq M_0$

is used λ^{M-M_0} times in each training epoch with $\lambda = 1.5$ and $M_0 = 5$ for Japan and $M_0 = 4$ for Italy. This event-based oversampling implicitly increases the number of high PGA values in the training set too.

We apply all data augmentation on the training and the development set, to ensure that the development set properly represents the task we are interested in. As this introduces stochasticity into the development set metrics, we evaluate the development set three times after each epoch and average the result. In contrast, at test time we do not apply any data augmentation, except temporal blinding for modelling real-time application. If more than 25 stations are available for a test set event, we select the 25 stations with the earliest arrivals for evaluation.

We train our model using the Adam optimiser [Kingma and Ba, 2014]. We emphasise that the model is only trained on predicting the PGA probability density and does not use any information on the PGA thresholds used for evaluation. We start with a learning rate of 10^{-4} and decrease the learning rate by a factor of 3 after 5 epochs without a decrease in validation loss. For the final evaluation, we use the model from the epoch with the lowest loss on the development set. We apply gradient clipping with a value of 1.0. We use a batch size of 64. We train the model for 100 epochs.

To improve the calibration of the predicted probability densities we use ensembles [Snoek et al., 2019]. We use an ensemble size of 10 models and average the predicted probability densities. We weigh each ensemble member identically. To increase the entropy between the ensembles, we also modify the position encodings between the ensemble members by rotating the latitude and longitude values of stations and targets. The rotations for the 10 ensemble members are $0°, 5°, \ldots, 40°, 45°$.

For the Italy model, we use domain adaptation by modifying the training procedure. We first train a model jointly on the Italy and Japan datasets, according to the configuration described above. We use the resulting model weights as initialisation for the Italy model. For this training we reduce the number of PGA targets to 4, leading to a higher fraction of high PGA values in the training data, and the learning rate to 10^{-5}. In addition, we train jointly on an auxiliary dataset, comprised of 77 events from Japan. The events were chosen to be shallow, crustal and onshore, having a magnitude between 5.0 and 7.2. We shift the coordinates of the stations to lie in Italy. We use 85% of the auxiliary events in the training set and 15% in the development set.

We implemented the model using Tensorflow. We trained each model on one GeForce RTX 2080 Ti or Tesla V100. Training of a single model takes approximately 5 h for the Japan dataset, 10 h for the joint model and 1 h for the Italy dataset. We benchmarked the inference performance of TEAM on a common workstation with GPU acceleration (Intel i7-7700, Nvidia Quadro P2000). Running TEAM with ensembling at a single timestep took 0.15 s for all 246 PGA targets of the Norcia event. As our implementation is not optimised for run time, we expect an optimised implementation to yield multifold lower run times, enabling a real-time application of TEAM with a high update rate and low compute latency.

Figure C.4 shows the training and validation loss curves for the Japan TEAM model and the fine-tuning step of the Italy TEAM model. While there is some variation between the ensemble members, all show similar characteristics. We note, that the early appearance of the optima for the Italy fine-tuning is expected because of the transfer learning applied. We validated through a comparison of the fine-tuned and the non-fine-tuned models, that the fine-tuning step still leads to considerable improvement in the model performance.

As visible from the by far lower training than validation loss, all models exhibit overfitting. This is expected, as the number of model parameters (13.3M) is very high in comparison to the number of training examples ($< 10,000$). However, multiple publications [e.g., Belkin et al., 2019, Muthukumar et al., 2020] have provided theoretical and empirical evidence, that overfitting for deep learning is not necessarily problematic and can even lead to considerably better performance than a not overfitted model if proper model selection on a validation set is employed. We conduct this model selection by using the model with the lowest validation score.

C.3 Baseline methods

We compare TEAM to two baseline methods, EPS and a PLUM-based approach. We do not compare to any deep learning baseline, because we are not aware of any published deep learning method for early warning that can be applied in real-time. For the EPS method, we use a GMPE based on the functional form by Cua and Heaton [2009] and add a quadratic magnitude term as proposed by Meier [2017]. We make further minor adjustments to accommodate the wider range of magnitudes in our datasets. The functional form of the GMPE is:

$$\log(pga) = a_1 M + a_2 \max(M - M_0, 0)^2 + b(R_d + C(M)) + d \log(R_d + C(M)) + e + \delta_S + \mathscr{N}(0, \sigma^2) \tag{C.1}$$

$$C(M) := c_1 \exp(c_2 \max(0, M - 5))(\arctan(M - 5) + \pi/2) \tag{C.2}$$

$$R_d := \sqrt{R^2 + H_d^2} \tag{C.3}$$

We write M for magnitude, R for epicentral distance, δ_S for the station bias, and e for a station-independent bias term. We use m/s^2 as unit for PGA and km as unit for all length measurements. We use a pseudo-depth H_d, depending on the event depth and the dataset. This allows modelling the stronger attenuation with distance for shallow events. For Italy, we set $H_d = 5$ km for events shallower than 20 km and $H_d = 50$ km for all other events. For Japan, we set $H_d = 5$ km for events shallower than 20 km, $H_d = 40$ km for events between 20 km and 200 km and set H_d to the actual depth for all deeper events, to account for a few very deep events. We set $M_0 = 4$ for Italy and $M_0 = 6$ for Japan.

We fix $c_1 = 1.48$ and $c_2 = 1.11$, as proposed by Cua and Heaton [2009], and optimise the other parameters using linear regression. We perform the optimisation iteratively to obtain station bias terms, using the union of training and development set. To avoid noise samples in calibration we only use stations for which $R_d < (M - 3.5) * 200$ km for Japan and $R_d < (M - 3) * 50$ km for Italy. The calibrated GMPEs have residual values σ of 0.29 for Italy and 0.33 for Japan, matching the value of $\sim$0.3 proposed as the approximate current optimum for GMPEs [Minson et al., 2019]. Residual plots can be found in Figure C.5.

We note that our GMPE model is using a point source assumption, which is incorrect for larger events. We chose this simplification, as it is common in source based early warning and makes the GMPE performance an upper bound for any method relying on magnitude and location estimate. While there are early warning methods based on extended fault models [Böse et al., 2018], they perform equally well as point source approaches for all but the largest events [Meier et al., 2020]. As lower thresholds are dominated by smaller events, for which the point source approximation is valid, the inferior performance of the GMPE compared to TEAM is not an artefact of the point source assumption, but probably related to its inability to account for systematic propagation effects caused by regional structure, and variability of the earthquake source (focal mechanism, stress drop) not captured by the magnitude and location.

For magnitude estimation, we use the peak displacement based method proposed by Kuyuk and Allen [2013]. We bandpass filter the signal between 0.5 Hz and 3 Hz and discard traces with insufficient signal to noise ratio. We extract peak displacement from the horizontal components in the first 6 s of the P wave. We stop the time window at the latest at the S onset. We use the relationship

$$M = c_1 \log(PD) + c_2 \log(R) + c_3 + \mathscr{N}(0, \sigma^2) \tag{C.4}$$

to estimate magnitudes from peak displacement. We use $c_1 = 1.23$, $c_2 = 1.38$, $c_3 = 5.69$ (Italy) / $c_3 = 5.89$ (Japan) and $\sigma = 0.31$. These are the values from Kuyuk and Allen [2013], except for c_3 which we needed to adjust as we do not use moment magnitude. We combine the predictions in probability space assuming independence between the predictions from different stations. We weight stations based on the length of the P wave window recorded so far. We use the mean value of the single-station magnitude estimates for PGA estimation. For both the application of the GMPE and the magnitude estimation we use the catalog hypocenters. As the quality of real-time location estimates will be worse, this leads to inflated performance measures for EPS.

As a second baseline, we adapted the PLUM algorithm [Kodera et al., 2018]. While the original paper applies PLUM to seismic intensities, we apply it to PGA values. This adaptation is possible, as approximate linear and especially monotonic relations exist between intensity and PGA [Karim and Yamazaki, 2002]. However, as seismic intensity incorporates a narrower frequency band and also considers the duration of strong shaking [Shabestari and Yamazaki, 2001], the PLUM adaptation to PGA might exhibit a slightly different performance. The PGA prediction $\hat{pga}_t^s$ at a station s at time t is the maximum of all observed PGA values $pga_t^{s'}$ at stations s' within a radius r of s. Therefore a warning for a certain threshold for a station is issued once the threshold has been exceeded at any station within the radius r. Due to different station densities in Italy and Japan, we used different values for r. For Italy, we used $r = 15$ km; for Japan, we used $r = 30$ km. Following the findings of Cochran et al. [2019], we do not use site correction terms in our implementation of PLUM as they only have a minor impact on the performance.

C.4 Evaluation metrics

We analyse the performance of the early warning algorithms using PGA thresholds of 1%g, 2%g, 5%g, 10%g and 20%g, approximately matching Modified Mercalli Intensity (MMI) III (light) to VII (very strong) [Wald et al., 1999]. We calculate PGA from the absolute value of the two horizontal components. To determine the PGA values for the Japanese data, we use the surface stations and not the borehole stations.

A warning at a site should be issued if anytime during the event the PGA threshold is exceeded at the site. We consider a warning correct (true positive, TP) if a warning for a certain threshold was issued and the threshold was actually exceeded later during the event. Missed warnings (false negative, FN) are all cases, where the PGA threshold was exceeded, but no warning was issued or the warning was issued after the PGA threshold was first exceeded. We consider a warning false (false positive, FP) if a warning was issued, but the threshold was not exceeded. All remaining cases are true negatives (TN).

As the number of true negatives depends strongly on the inclusion criteria of the catalog, we use metrics independent of the true negatives. As summary statistics we use *precision*, TP/(TP+FP), measuring the fraction of correct warnings among all warnings, and *recall*, TP/(TP+FN), measuring the fraction of possible correct warnings that were issued. We use the *F1 score* $= 2 * \text{precision} * \text{recall}/(\text{precision} + \text{recall})$ as a combined statistic. Any analysis using a fixed α uses the value maximising the F1 score, which is specific to each method and PGA threshold. For an analysis independent of the threshold α we use the area under the precision-recall curve (AUC). We use values $\alpha = 0.05, 0.1, 0.2, \dots, 0.8, 0.9, 0.95$ and add additional points at $(0, 1)$ and $(1, 0)$ to the precision-recall curve to approximate the AUC. For comparison of the PLUM-based model using AUC in Figure 3, we introduce an artificial precision-recall line for PLUM with a slope of -1 going through the observed precision and recall values.

We define the warning time as the time between the issuance of a warning and the first exceedance of the threshold. We consider a zero-latency system and do not impose a minimum warning time. For comparing warning times between methods or different parameter combinations, we only use the subset of station event pairs, where both methods/parameter combinations issued correct warnings.

We evaluate our PLUM-based implementation continuously, i.e., warnings are issued immediately at the exceedance of a threshold. TEAM and EPS are evaluated every 0.1 s, starting 1 s after the first P arrival for EPS and 0.5 s after the first P arrival for TEAM. We use a longer time before the first prediction for EPS as the early results of EPS are unstable. Warnings are not retracted, i.e., even if the model later estimates a shake level below the warning threshold, the warning stays active.

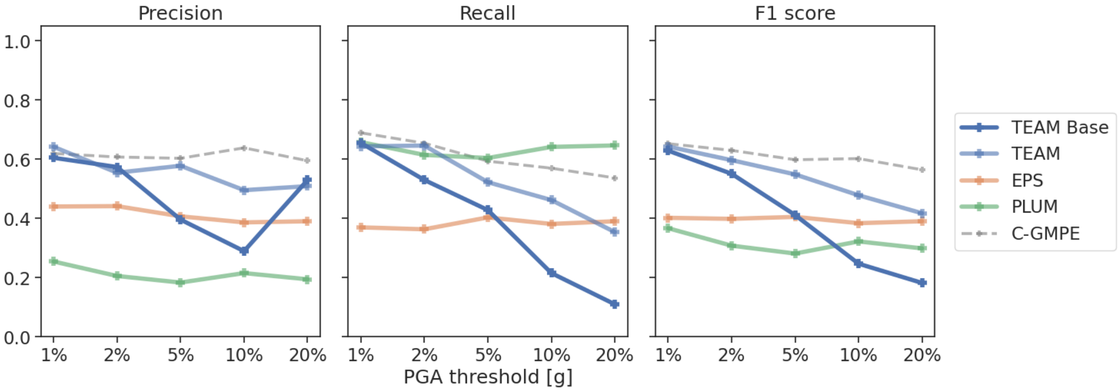


Figure C.1: Precision, recall and F1 score at different PGA thresholds for Italy including TEAM without domain adaptation. Threshold values α were chosen independently for each method and PGA threshold to yield the highest F1 score. The methods are the transformer earthquake alerting model without domain adaptation (TEAM Base), the transformer earthquake alerting model (TEAM), the estimated point source (EPS) model and the PLUM-based model. In addition the graph shows the performance of C-GMPE, a GMPE with full catalog information for reference.

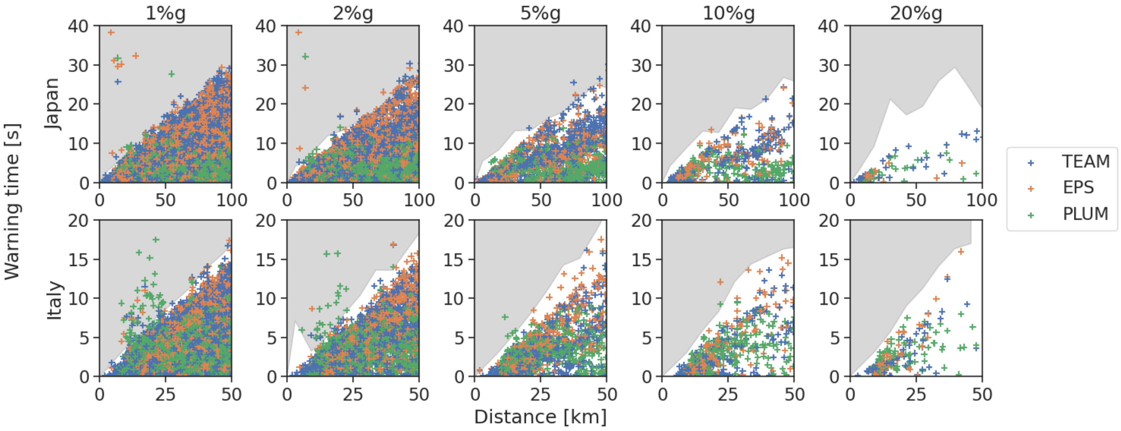


Figure C.2: Warning time and hypocentral distance between station and event for each true alert at F1-optimal α. The white area corresponds roughly to the range of possible warning times and is bounded by the 90^{th} percentile of the times between first detection of an event (i.e., arrival of P wave at the closest station) and first exceedance of the PGA threshold in recordings at that approximate distance.

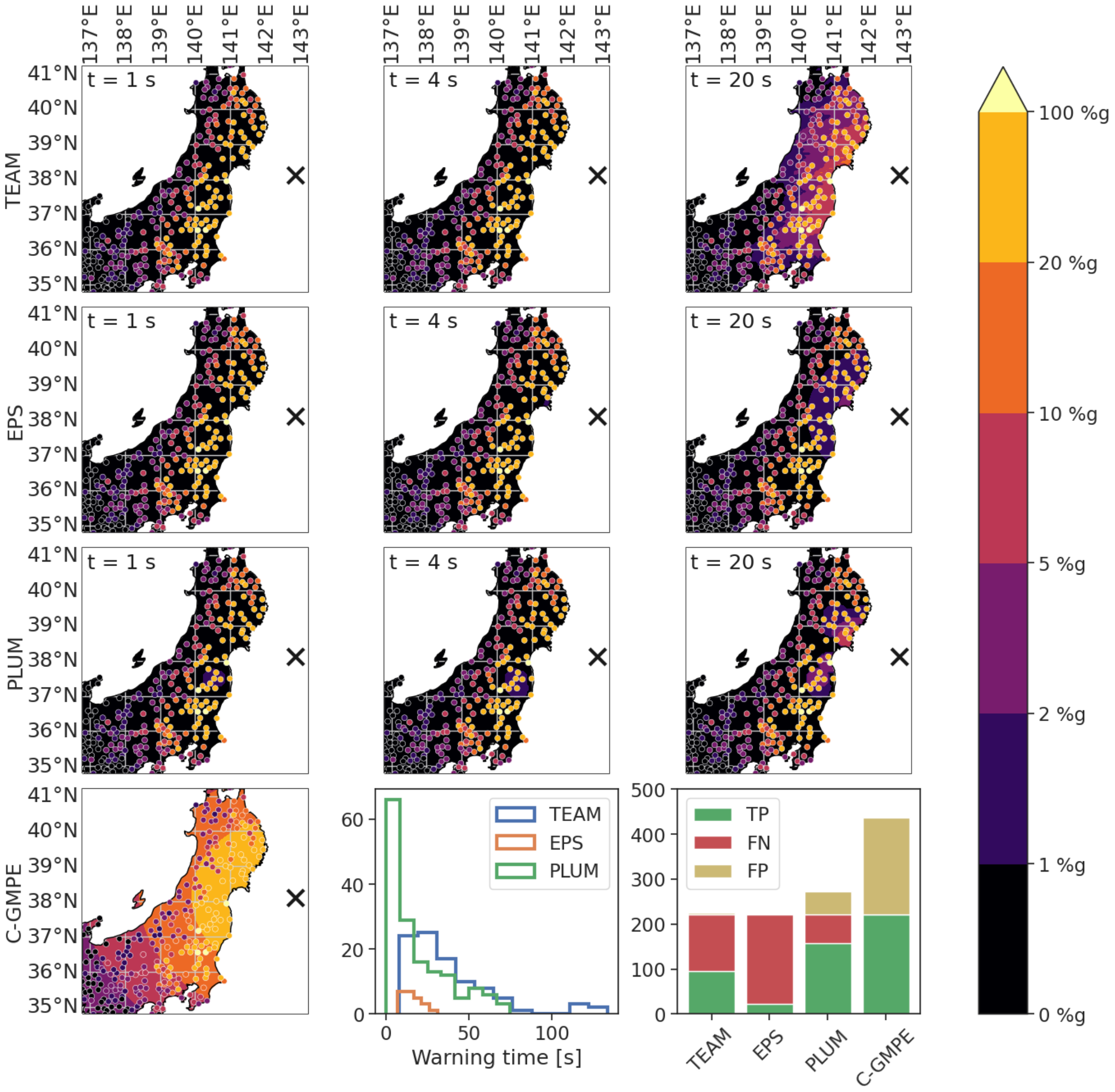


Figure C.3: Scenario analysis of the 11th March 2011 $M_w = 9.1$ Tohoku earthquake, the largest event in the Japan dataset. See Figure 4.8 for further explanations. The bottom row diagrams for this scenario analysis use a 2%g PGA threshold.

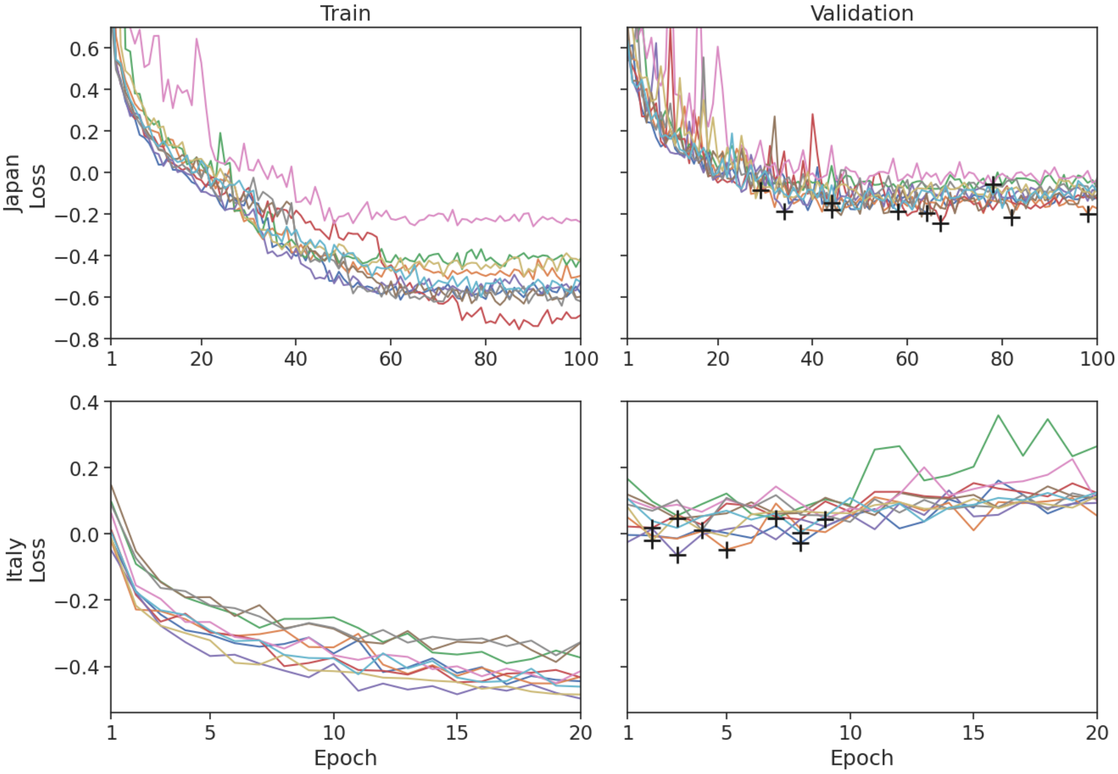


Figure C.4: Training and validation loss curves for the Japan TEAM and the fine-tuning step of the Italy TEAM. Each line shows the loss curve for one ensemble member with colors matching between training and validation curves. The models used are determined by the minimum validation loss and are denoted by black crosses. The models were evaluated after the training epoch indicated on the x-axis, i.e., the leftmost point of each curve already includes one epoch of training.

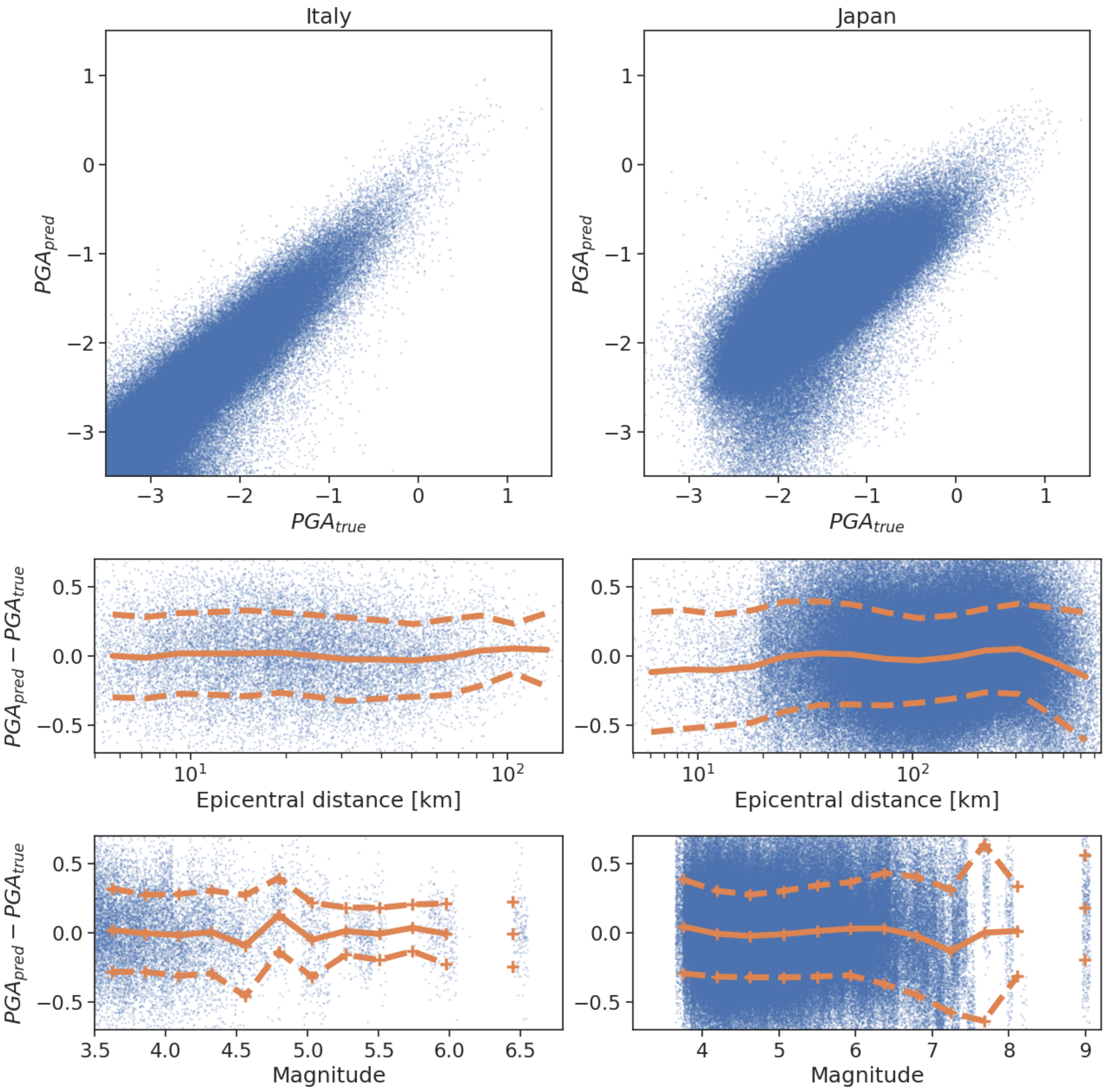


Figure C.5: Predictions and residuals of the GMPEs derived in this study. All PGA values are given as log units using m/s^2. Every point refers to one recording. Solid lines indicate running means, dashed lines denote the running standard deviation around the running mean. Orange crosses denote mean and standard deviations for magnitude ranges with insufficient data to infer a continuous line. Window sizes are 0.24 m.u./10 km (Italy) and 0.44 m.u./53 km (Japan). Overall σ is 0.29 for Italy and 0.33 for Japan. The plotted magnitude values have been offset by random values between -0.05 and 0.05 m.u. for increased visibility.

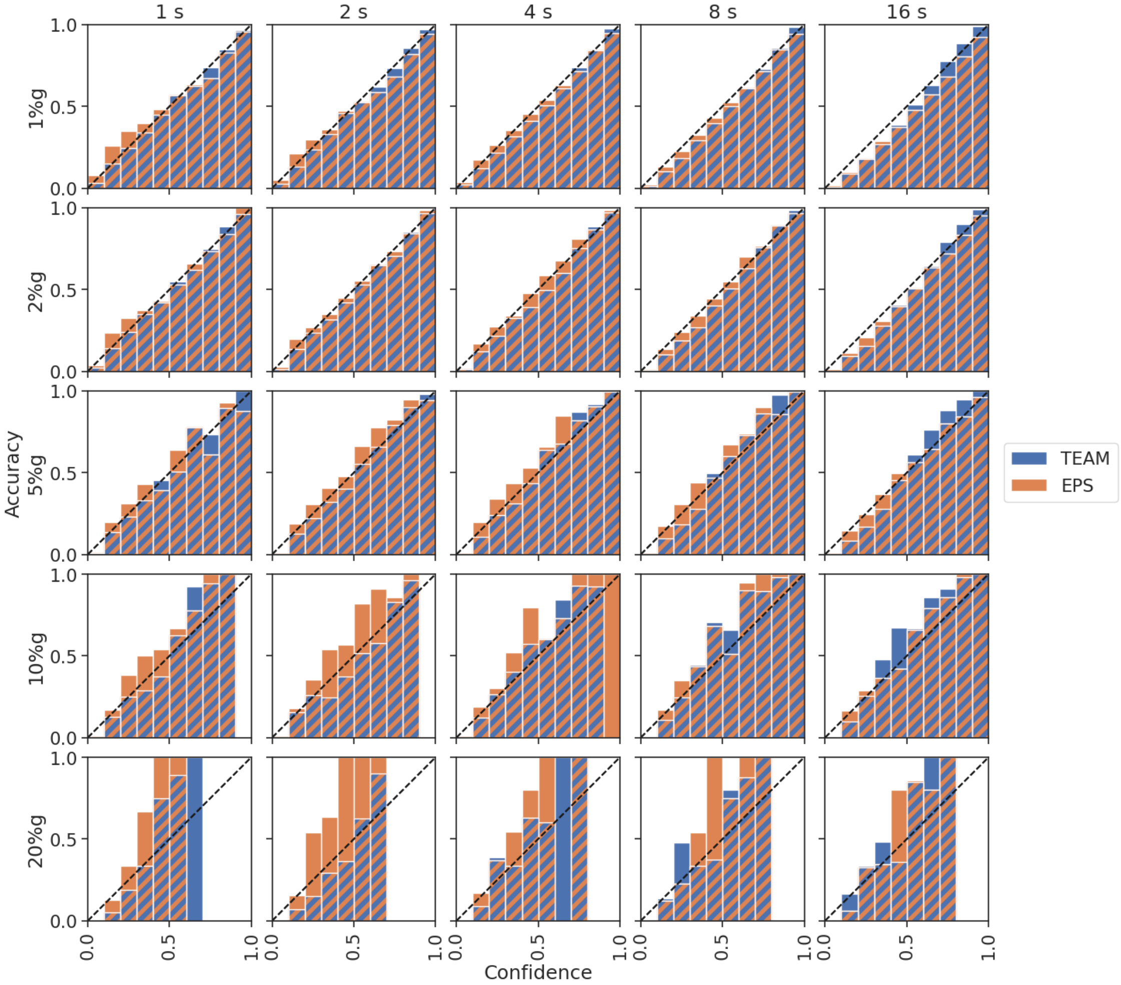


Figure C.6: Calibration diagrams for Japan at different times after the first P detection and different PGA thresholds. The confidence is defined as the probability of exceeding the PGA threshold as predicted by the model. Each bar represents the traces with a confidence value inside the x axis limits of the bar. Its height is given by the accuracy, the fraction of traces actually exceeding the threshold among all traces in the bar. For a perfectly calibrated model, the confidence equals the accuracy. This is indicated by the dashed line. We note that accuracy estimations for the high PGA thresholds are strongly impacted by stochasticity due to the small number of samples.

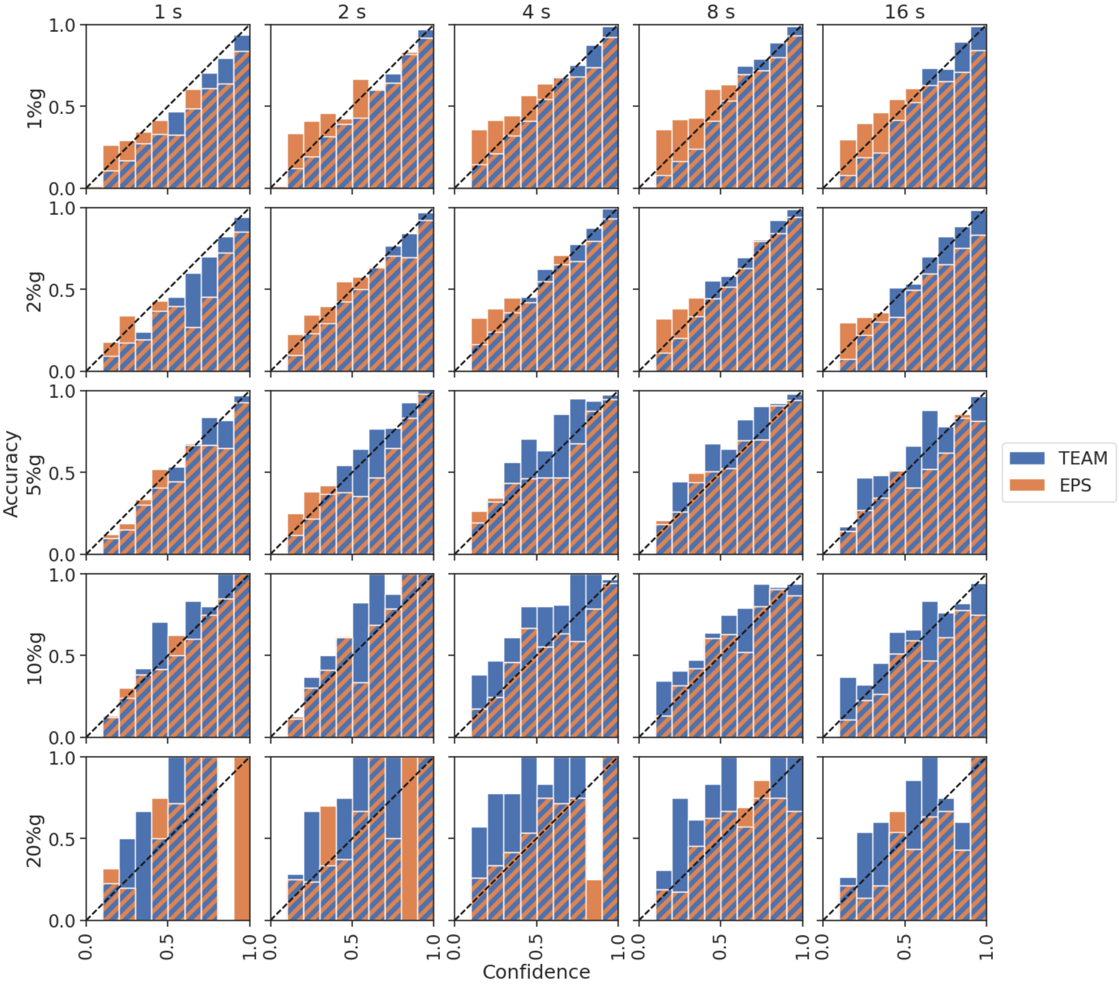


Figure C.7: Calibration diagrams for Italy at different times after the first P detection and different PGA thresholds. For a further description see the caption of Figure C.6.

Table C.1: Performance statistics for Japan. Probability thresholds α were chosen to maximise F1 scores and are shown in the last column. The AUC value does not depend on the threshold α. PGA indicates the used PGA threshold.

	PGA [g]	Precision	Recall	F1	AUC	α
TEAM	1%	0.70	0.77	0.73	0.82	0.60
	2%	0.69	0.69	0.69	0.76	0.60
	5%	0.59	0.67	0.63	0.68	0.50
	10%	0.50	0.60	0.54	0.56	0.40
	20%	0.33	0.48	0.39	0.35	0.30
EPS	1%	0.50	0.63	0.56	0.57	0.40
	2%	0.48	0.48	0.48	0.48	0.40
	5%	0.40	0.40	0.40	0.34	0.30
	10%	0.27	0.36	0.31	0.25	0.20
	20%	0.20	0.26	0.22	0.15	0.20
PLUM	1%	0.39	0.56	0.46	-	-
	2%	0.30	0.50	0.38	-	-
	5%	0.22	0.42	0.29	-	-
	10%	0.18	0.39	0.25	-	-
	20%	0.11	0.28	0.16	-	-
C-GMPE	1%	0.58	0.74	0.65	0.69	0.30
	2%	0.47	0.71	0.56	0.60	0.20
	5%	0.44	0.54	0.48	0.48	0.20
	10%	0.44	0.46	0.45	0.43	0.20
	20%	0.56	0.38	0.45	0.42	0.30

Table C.2: Performance statistics for Italy. Probability thresholds α were chosen to maximise F1 scores and are shown in the last column. The AUC value does not depend on the threshold α. PGA indicates the used PGA threshold.

	PGA [g]	Precision	Recall	F1	AUC	α
TEAM	1%	0.64	0.64	0.64	0.68	0.60
	2%	0.55	0.65	0.60	0.63	0.50
	5%	0.58	0.52	0.55	0.54	0.50
	10%	0.50	0.46	0.48	0.43	0.40
	20%	0.51	0.35	0.42	0.36	0.30
EPS	1%	0.44	0.37	0.40	0.37	0.30
	2%	0.44	0.36	0.40	0.36	0.40
	5%	0.41	0.40	0.40	0.33	0.40
	10%	0.39	0.38	0.38	0.30	0.40
	20%	0.39	0.39	0.39	0.25	0.40
PLUM	1%	0.25	0.66	0.37	-	-
	2%	0.21	0.61	0.31	-	-
	5%	0.18	0.60	0.28	-	-
	10%	0.22	0.64	0.32	-	-
	20%	0.19	0.65	0.30	-	-
C-GMPE	1%	0.62	0.69	0.65	0.71	0.30
	2%	0.61	0.65	0.63	0.68	0.30
	5%	0.60	0.59	0.60	0.63	0.30
	10%	0.64	0.57	0.60	0.59	0.30
	20%	0.59	0.54	0.56	0.54	0.30

Table C.3: Relative warning times of the algorithms in seconds. Positive values indicate longer average warning times for the second method, negative values shorter warning times. The difference in average warning times is calculated from all event station pairs, where both methods issued correct warnings. No value is reported if this set is empty. We set α for TEAM and EPS to the optimal value in terms of F1 score.

		Japan					Italy				
PGA [g]		1%	2%	5%	10%	20%	1%	2%	5%	10%	20%
EPS	TEAM	0.39	0.43	0.70	0.31	0.61	0.18	0.26	-0.49	-0.65	-1.19
PLUM	TEAM	8.98	8.24	6.35	5.01	0.55	1.49	1.60	1.03	-0.03	0.03
PLUM	EPS	8.53	7.74	5.29	3.08	-0.04	2.95	3.11	2.35	0.81	1.08

Table C.4: Dataset statistics for the full datasets and the test sets. The lower boundary of the magnitude category is the 5th percentile of the magnitude; this limit is chosen as each dataset contains a small number of unrepresentative very small events. The upper boundary is the maximum magnitude. The lower part of the table shows how often each PGA threshold was exceeded. An event is counted as exceeding a threshold if at least one station exceeded this threshold during the event. The number of exceedances in the test set for Italy is disproportionally high compared to the number of events in the test set. This is caused by the high seismic activity and the higher station density in 2016. Traces for Japan always refer to 6 component traces, while for Italy it refers to 3 component traces.

	Japan				Italy			
	Full		Test		Full		Test	
Years	1997 - 2018		08/2013 - 12/2018		2008 - 2019		01/2016 - 12/2016	
Magnitudes	2.7 - 9.0		2.7 - 8.1		2.7 - 6.5		2.7 - 6.5	
Events	13,512		4,054		7,055		2,123	
Unique stations	697		632		1,080		621	
Traces	372,661		104,573		494,183		253,454	
Avg. traces per event	27.6		25.9		70.3		119.4	
PGA [g]	Events	Traces	Events	Traces	Events	Traces	Events	Traces
1%	8,761	55,618	2,710	15,215	1,841	6,379	923	3,826
2%	5,324	24,396	1,601	6,489	1,013	2,921	503	1,771
5%	2,026	6,802	583	1,712	348	888	171	563
10%	782	2,223	216	506	120	330	58	223
20%	238	631	62	100	40	107	20	82

Table C.5: Architecture of the feature extraction network. The input dimensions of the waveform data are (time, channels). FC denotes fully connected layers. As FC layers can be regarded as 0D convolutions, we write the output dimensionality in the filters column. The "Concatenate scale" layer concatenates the log of the peak amplitude to the output of the convolutions. Depending on the existence of borehole data the number of input filters for the first Conv1D layer is 64 instead of 32 in the non-borehole case.

Layer	Filters	Kernel size	Stride
Conv2D	8	5, 1	5, 1
Conv2D	32	16, 3	1, 3
Flatten to 1D			
Conv1D	64	16	1
MaxPool1D		2	2
Conv1D	128	16	1
MaxPool1D		2	2
Conv1D	32	8	1
MaxPool1D		2	2
Conv1D	32	8	1
Conv1D	16	4	1
Flatten to 0D			
Concatenate scale			
FC	500		
FC	500		
FC	500		

Table C.6: Architecture of the transformer network. Please note that even though the transformer in TEAM does not apply dropout, we explicitly state this in the table, as transformers commonly use dropout.

Feature	Value
# Layers	6
Dimension	500
Feed forward dimension	1000
# Heads	10
Maximum number of stations	25
Dropout	0
Activation	GeLu

D Supplement to Chapter 5

D.1 Classical magnitude estimation baseline

For magnitude estimation, we compare TEAM-LM to a classical baseline. To this end, we use the peak displacement based approach proposed by Kuyuk and Allen [2013]. At each station, we bandpass filter the signal between 0.5 Hz and 3 Hz and discard traces with insufficient signal to noise ratio. We extract peak displacement PD from the horizontal components in the first 6 s of the P wave, while only including samples before the S onset. We use the relationship

$$M = c_1 \log(PD) + c_2 \log(R) + c_3 + \mathscr{N}(0, \sigma^2) \tag{D.1}$$

from Kuyuk and Allen [2013] to estimate magnitudes from peak displacement. We use $c_1 = 1.23$, $c_2 = 1.38$ and $\sigma = 0.31$ from Kuyuk and Allen [2013]. These parameters were calibrated using data from California and Japan, but the authors state that the relationship can be applied to earthquake source zones around the world. To account for a constant offset between different magnitude scales, we optimized c_3 separately for each dataset such that the predictions do not have a systematic bias compared to the ground truth.

We average the predictions from multiple stations, effectively assuming independence between the predictions. To obtain earlier predictions, we already calculate magnitude estimates at a station once at least 1 s of P wave data has been recorded. We assign higher weights to stations with longer P wave records, with weights linearly increasing from 0.11 for 1 s of waveforms, to 1.0 for 6 s of data. Thereby, while getting early estimates from the first stations, new data from later stations does not perturb the prediction strongly until enough data has been recorded.

As the estimation relies on the hypocentral distance R between station and event, the method requires an estimate of the hypocentral location. We provide the method with the catalogued hypocentral location. While this is an unrealistically optimistic assumption for an actual real-time determination, it allows us to put our focus on the magnitude estimation capabilities. We note that this gives the baseline an advantage compared to TEAM-LM, which has no information on the earthquake location.

For some events in the Chile catalog, the SNR criterion is not fulfilled at any station due to the inclusion of smaller magnitude events and the higher distances between stations and events. For these events, the baseline does not issue a magnitude estimation. We exclude these events from the evaluation of the baseline, leading to an optimistic assessment of the performance of the baseline.

D.2 Calibration estimation

Calibration of a model describes whether the predicted uncertainties match the observed values, i.e., if the observation y_{true} was drawn from a distribution with cumulative distribution function (CDF) F_{pred}. Unfortunately, for each event i, only one prediction observation of the magnitude y^i_{true} and one prediction F^i_{pred} is available. To this end, we define the random variable $u_i = F^i_{pred}(y^i_{true})$. If y^i_{true} is distributed according to F^i_{pred} than u_i must be uniformly distributed on $[0, 1]$. This follows from the definition of the CDF. If F is a CDF and U a uniform random variable on $[0, 1]$, then $F^{-1}(U)$ is distributed according to F.

We take the u_i of all events as samples of a random variable U and compare U to a uniform random variable on $[0, 1]$. The maximum difference between the empirical CDF of U and a uniform variable is the test statistic of a Kolmogorov-Smirnov test, d_∞. As the number of events n is large, critical values d_α to a confidence threshold α can be estimated as:

$$d_\alpha = \frac{\sqrt{-\frac{1}{2}\log\frac{\alpha}{2}}}{\sqrt{n}} \tag{D.2}$$

For $\alpha = 10^{-5}$, this gives values d_α of 0.015 (Chile), 0.054 (Japan) and 0.039 (Italy). This is considerably below the observed values d_∞, even using ensembles, indicating that U differs highly significantly from a uniform distribution.

Table D.1: Seismic networks

Region	Network	Reference
Chile	GE	GEOFON Data Centre [1993]
	C, C1	Universidad de Chile [2013]
	8F	Wigger et al. [2016]
	IQ	Cesca et al. [2009]
	5E	Asch et al. [2011]
Italy	3A	Istituto Nazionale di Geofisica e Vulcanologia (INGV) et al. [2018]
	BA	Universita della Basilicata [2005]
	FR	RESIF - Réseau Sismologique et géodésique Français [1995a]
	GU	University of Genova [1967]
	IT	Presidency of Counsil of Ministers - Civil Protection Department [1972]
	IV	Istituto Nazionale di Geofisica e Vulcanologia (INGV), Italy [2006]
	IX	Dipartimento di Fisica, Università degli studi di Napoli Federico II [2005]
	MN	MedNet Project Partner Institutions [1990]
	NI	OGS (Istituto Nazionale di Oceanografia e di Geofisica Sperimentale) and University of Trieste [2002]
	OX	OGS (Istituto Nazionale di Oceanografia e di Geofisica Sperimentale) [2016]
	RA	RESIF - Réseau Sismologique et géodésique Français [1995b]
	ST	Geological Survey-Provincia Autonoma di Trento [1981]
	TV	Istituto Nazionale di Geofisica e Vulcanologia (INGV) [2008]
	XO	EMERSITO Working Group [2018]
Japan	KiK-Net	National Research Institute For Earth Science And Disaster Resilience [2019]

Table D.2: Architecture of the feature extraction network. The input shape of the waveform data is (time, channels). FC denotes fully connected layers. As FC layers can be regarded as 0D convolutions, we write the output dimensionality in the filters column. The "Concatenate scale" layer concatenates the log of the peak amplitude to the output of the convolutions. Depending on the existence of borehole data the number of input filters for the first Conv1D varies.

Layer	Filters	Kernel size	Stride
Conv2D	8	5, 1	5, 1
Conv2D	32	16, 3	1, 3
Flatten to 1D			
Conv1D	64	16	1
MaxPool1D		2	2
Conv1D	128	16	1
MaxPool1D		2	2
Conv1D	32	8	1
MaxPool1D		2	2
Conv1D	32	8	1
Conv1D	16	4	1
Flatten to 0D			
Concatenate scale			
FC	500		
FC	500		
FC	500		

Table D.3: Architecture of the transformer network.

Feature	Value
# Layers	6
Dimension	500
Feed forward dimension	1000
# Heads	10
Maximum number of stations	25
Dropout	0
Activation	GeLu

Table D.4: Architecture of the mixture density networks.

Feature	Value
Dimensions fully connected layers (magnitude)	150, 100, 50, 30, 10
Dimensions fully connected layers (location)	150, 100, 50, 50, 50
Mixture size (magnitude)	5
Mixture size (location)	15
Base distribution	Gaussian

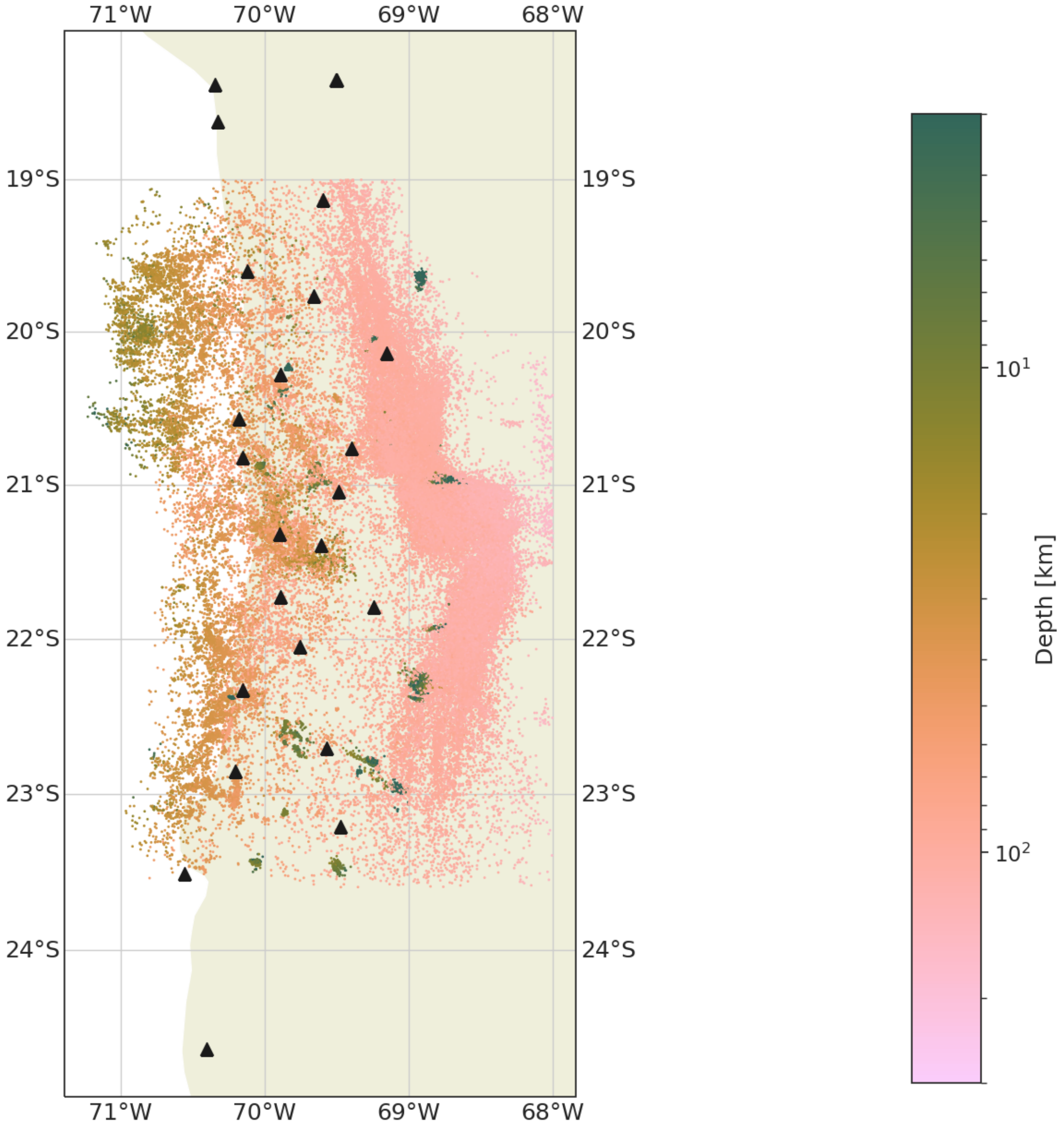


Figure D.1: Event and station distribution for Chile. In the map, events are indicated by dots, stations by triangles. The event depth is encoded using colour.

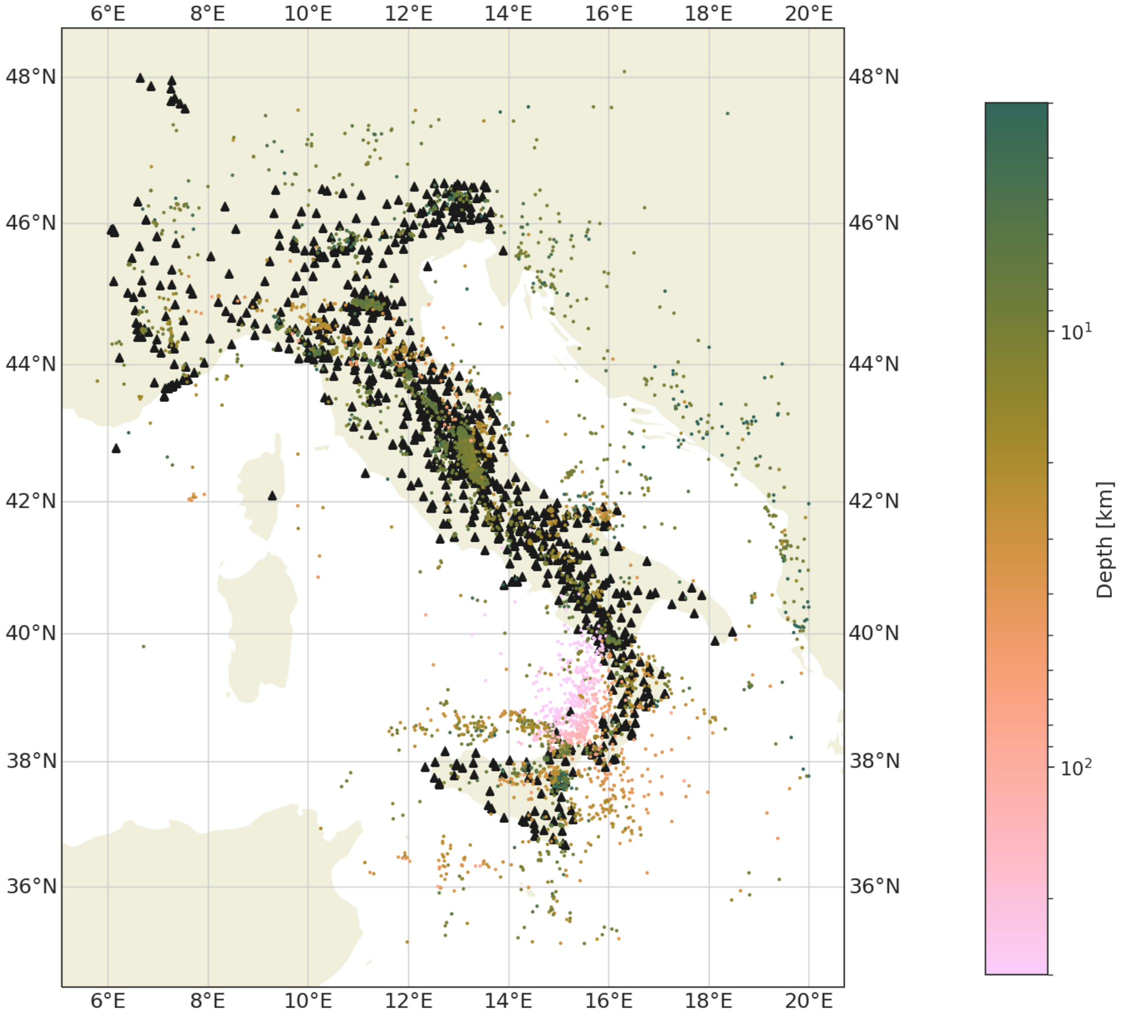


Figure D.2: Event and station distribution for Italy. In the map, events are indicated by dots, stations by triangles. The event depth is encoded using colour.

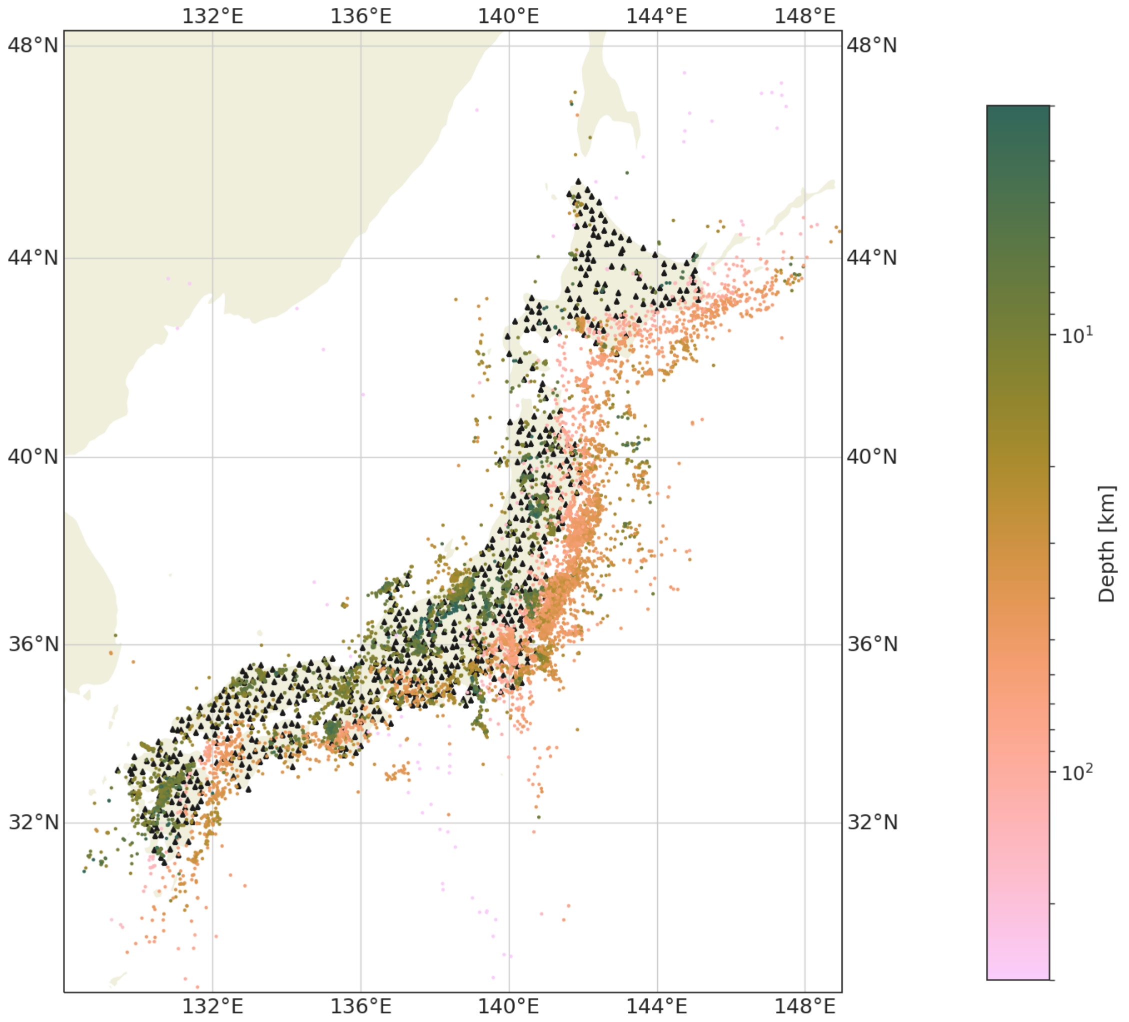


Figure D.3: Event and station distribution for Japan. In the map, events are indicated by dots, stations by triangles. The event depth is encoded using colour. There are ∼20 additional events far offshore in the catalog, which are outside the displayed map region.

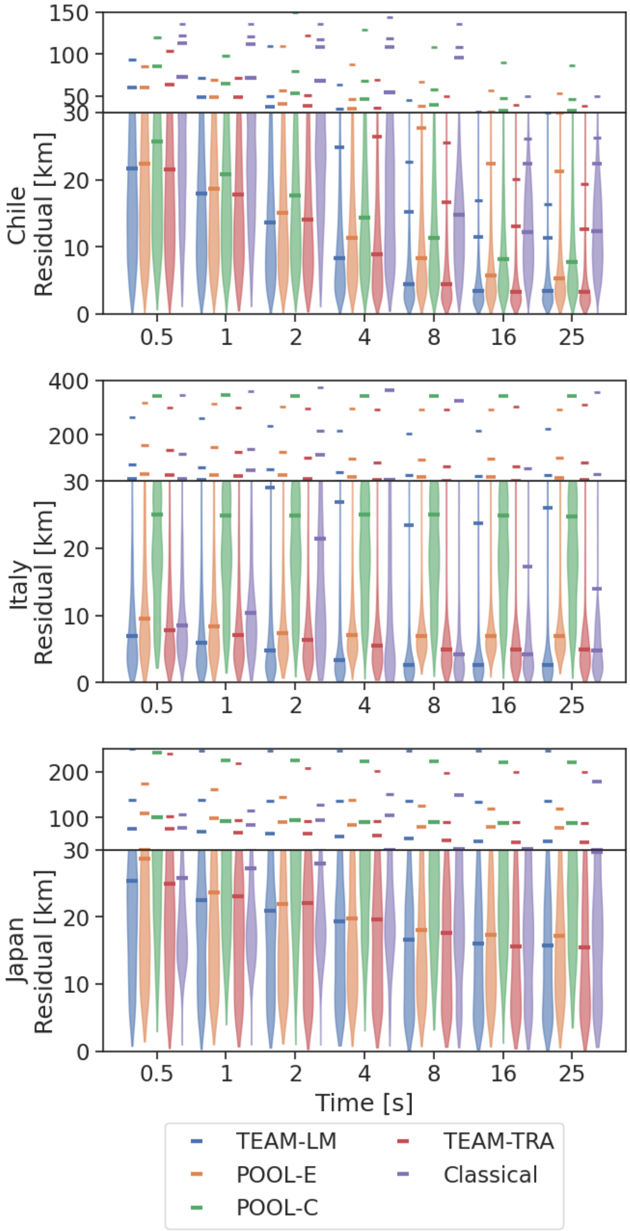


Figure D.4: Distribution of the hypocentral errors for TEAM-LM, the pooling baseline with position embeddings (POOL-E), the pooling baseline with concatenated position (POOL-C), TEAM-LM with transfer learning (TEAM-TRA) and a classical baseline. Vertical lines mark the 50^{th}, 90^{th}, 95^{th} and 99^{th} error percentiles. The time indicates the time since the first P arrival at any station. We use the mean predictions.

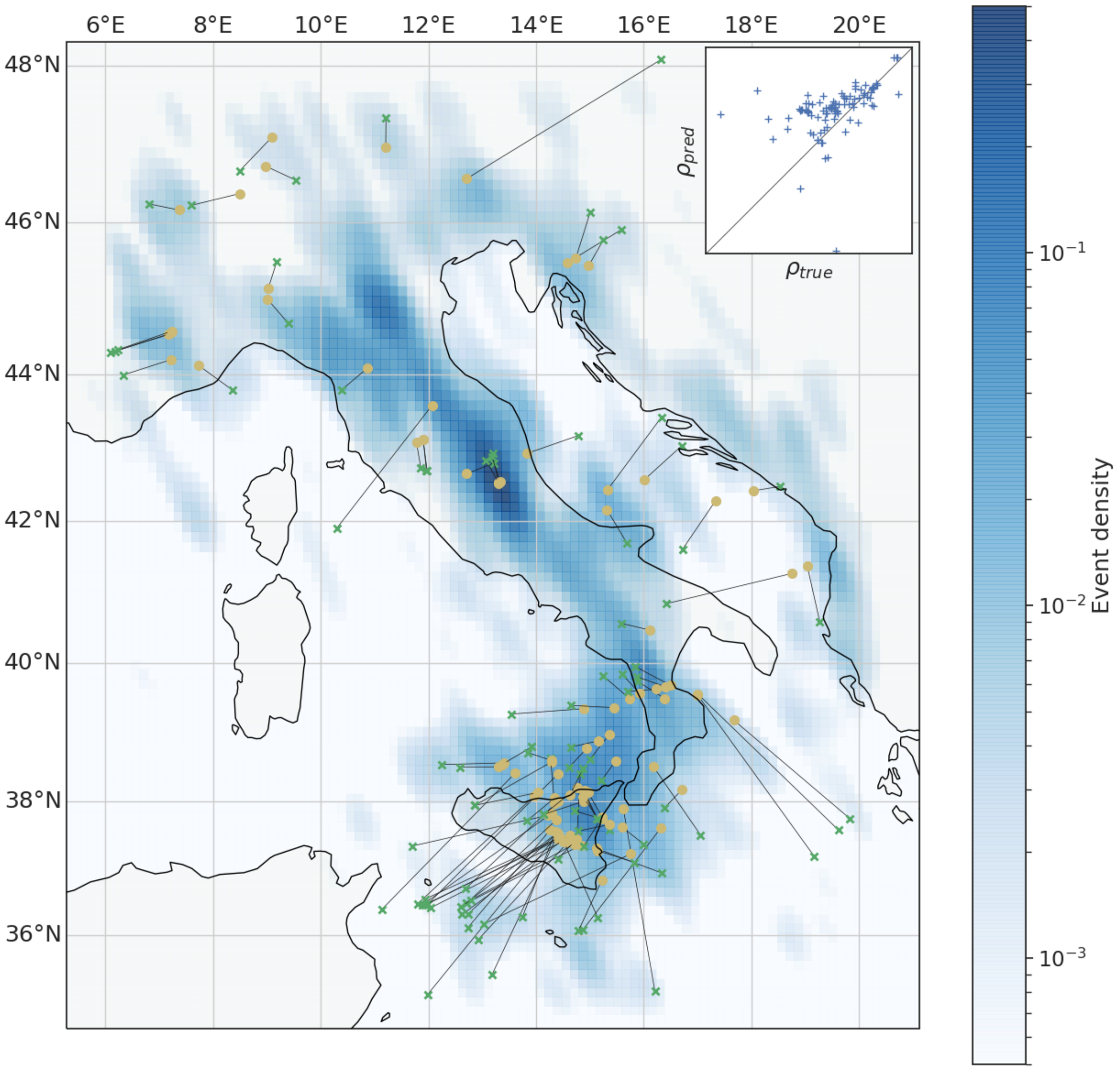


Figure D.5: The 100 events with the highest location error in the Italy dataset overlayed on top of the spatial event density in the training dataset. The estimations use 16 s of data. Each event is denoted by a dot for the estimated location, a cross for the true location and a line connecting both. Stations are not shown as station coverage is dense. The event density is calculated using a Gaussian kernel density estimation and does not take into account the event depth. The inset shows the event density at the true event location in comparison to the event density at the predicted event location.

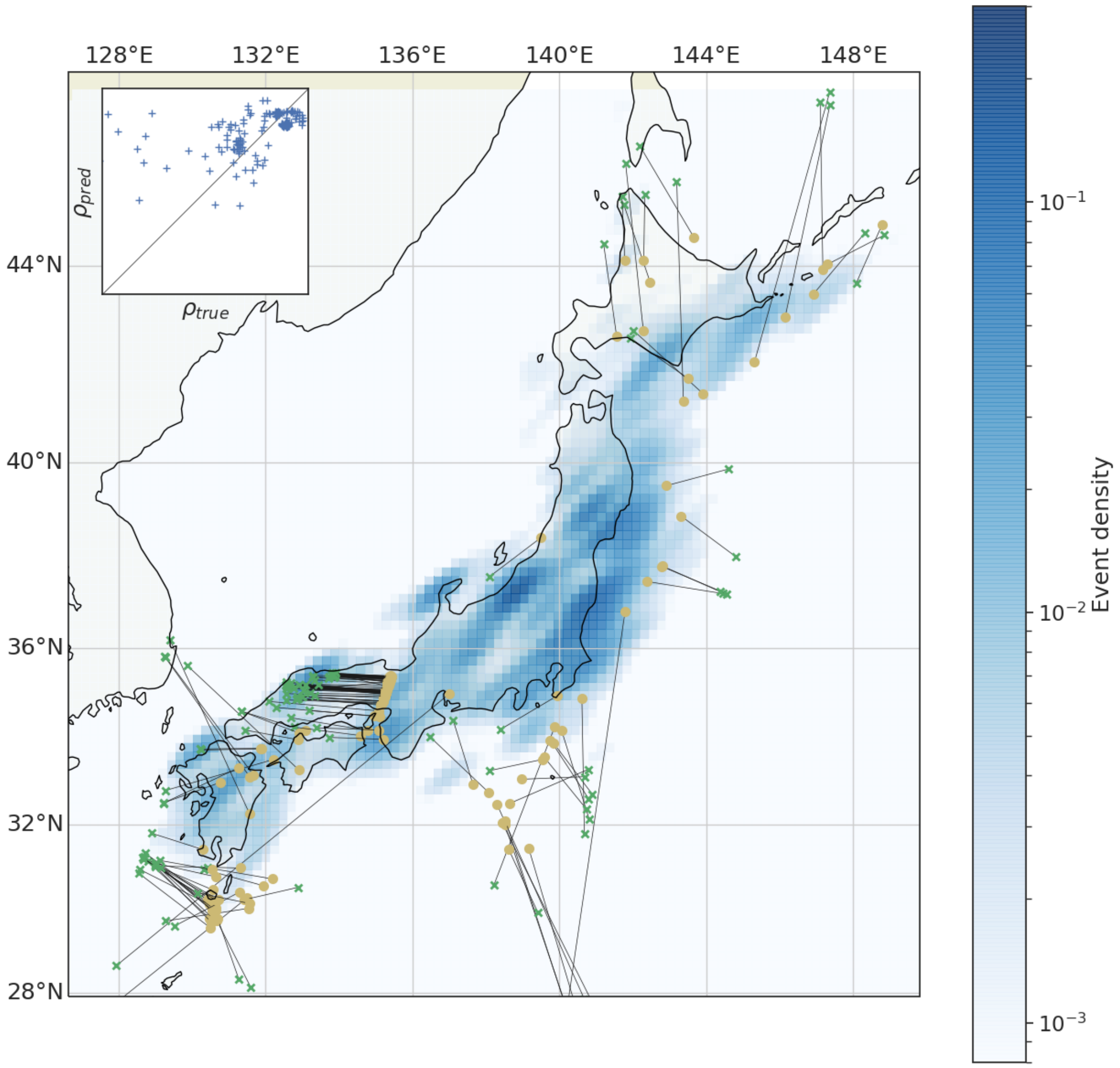


Figure D.6: The 200 events with the highest location error in the Japan dataset overlayed on top of the spatial event density in the training dataset. The estimations use 16 s of data. Each event is denoted by a dot for the estimated location, a cross for the true location and a line connecting both. Stations are not shown as station coverage is dense. The event density is calculated using a Gaussian kernel density estimation and does not take into account the event depth. The inset shows the event density at the true event location in comparison to the event density at the predicted event location.

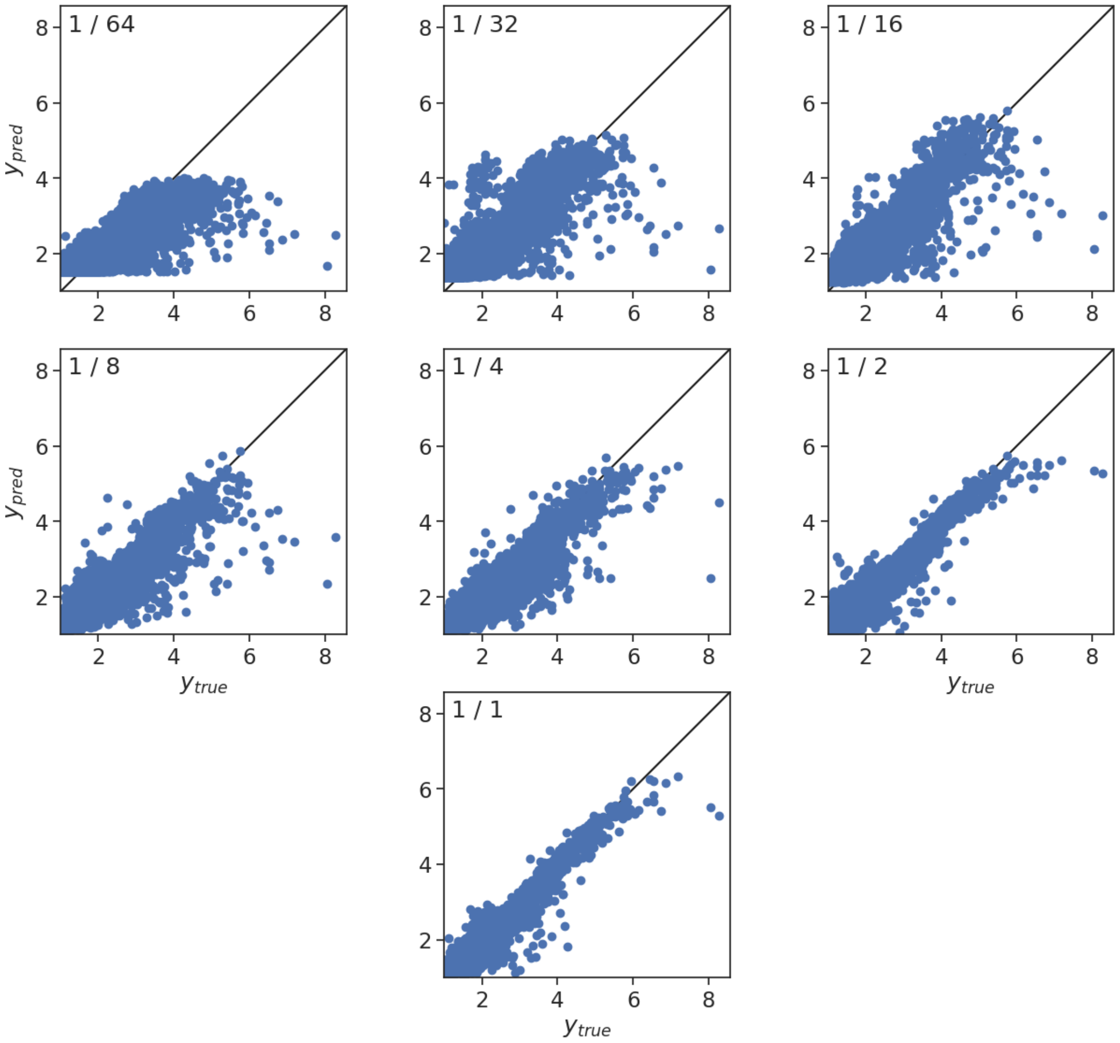


Figure D.7: True and predicted magnitudes after 8 seconds using only parts of the datasets for training. All plots show the Chile dataset. The fraction in the corner indicates the amount of training and validation data used for model training. All models were evaluated on the full test dataset.

E Supplement to Chapter 6

E.1 Apparent early predictability in SCARDEC

To investigate the apparent early predictability in the SCARDEC result, we first review the methodologies applied for creating the datasets. The three STF databases were generated using two different methodologies. Notably, none of the methods has originally been developed for real-time assessment. SCARDEC uses a point source approximation and conducts a constrained deconvolution of body waves with five constraints on the STF: positivity, causality, boundedness in time, fixed cumulative moment and low inter-station variation [Vallée et al., 2011]. To extend the SCARDEC methodology to events with $M < 7$, SCARDEC uses empirical relationships to determine event duration and filter frequencies. This methodology returns one apparent STF for each station. From these, SCARDEC publishes the average STF and the so-called optimal STF, i.e., a single-station STF with a high agreement with the average STF. Due to possible timing offsets between stations and stacking out of uncorrelated fluctuations, average STFs have lower high-frequency content than single-station STFs. For our study, we used the average STFs. To check the influence of average and optimal STFs, we trained models on the optimal STFs and obtained similar results as for the average ones (Figures E.7 and E.8). The only visible difference is a slightly higher uncertainty, which is to be expected as the optimal STFs are less smooth than the average STFs.

In contrast to SCARDEC, USGS [Hayes, 2017] and Ye et al. [2016] calculate finite fault solutions from both body and surface waves assuming constant rupture velocity within each event. This method returns only one STF per event instead of one per station. As the spatial extent of the source is modelled, the STFs generally represent more high-frequency details than the SCARDEC ones. On the downside, these methods do not apply to intermediate size events ($M_w < 7$).

To identify the source of the different behaviour of the model between the STF datasets, we analyse the cumulative and current moment release at fixed early times and after a fixed moment release (Figure E.9). For SCARDEC, cumulative and current moment release at early times differs between different magnitude bins, with higher magnitude events already exhibiting higher moment release. The same, although with a higher overlap between bins, is true for the current moment release at the time when magnitude 6 is reached. No differences between bins are visible at the times when magnitudes 6.5 and 7 are reached. These observations match the predictive results, where predicted magnitudes differed for early times and the time of reaching magnitude $\bar{M} = 6$, but not for higher base magnitudes $\bar{M}$.

In contrast to SCARDEC, for the other two STF datasets, we observe no systematic difference in any of the observables between the magnitude bins. This matches the predictive results, where predictions did not show systematic differences between buckets. Given these observations and the processing of SCARDEC, in particular the point source approximation, we attribute the difference in the early observables to a processing artefact rather than interpreting them to be physically based. In particular, the difference can be explained with uncertainties in the onset times for the SCARDEC STFs. SCARDEC onset times are defined by the first time the STF exceeds a few percent of the peak moment rate. This is necessary, as for an event with peak moment rate $> 10^{20}$ Nm/s it will be impossible to identify the first exceedance of a low threshold such as 10^{17} Nm/s due to model approximations, in particular, due to the point source approximation. On the other hand, for an event with peak moment rate $\sim 10^{18}$ Nm/s, this first exceedance is easy to determine. This introduces a systematic bias in the first seconds of the event [Vallée and Douet, 2016]. This bias has also been analysed quantitatively in prior publications [Meier et al., 2021].

As a further validation, we trained our neural network model on the USGS dataset, which is the larger of the two datasets using finite fault solutions. The results confirm that no signs of rupture determinism are visible (Figure E.2). We observe no systematic difference between different magnitude buckets until at least half of the time has passed or half of the moment has been released on either, the USGS dataset we trained the model on, or on the other two datasets. Note that, due to the different marginal distribution of magnitudes in the USGS dataset compared to SCARDEC, early estimates are considerably higher than for the SCARDEC model. In addition, the smallest SCARDEC events are systematically overestimated. This behaviour is expected, as neural networks are usually unable to extrapolate.

E.2 Training details for the STF model

In this section, we provide details on the training of the STF model. We train the model using a continuous ranked probability score (CRPS). The CRPS is defined as

$$CRPS(F, x) = -\int_{-\infty}^{\infty} (F(y) - \mathbb{1}_{\{y \geq x\}})^2 dy \tag{E.1}$$

with F the cumulative distribution function of the predicted probability, x the observed value and $\mathbb{1}_{\{y \geq x\}}$ the indicator function, being 1 for $y \geq x$ and 0 otherwise. The CRPS measures the distance in probability mass between true and predicted cumulative distribution functions [Matheson and Winkler, 1976], and thereby not only takes into account the prediction at the observed value as the more common log-likelihood. This is particularly useful for gradient-based optimisation in face of the highly skewed Gutenberg-Richter (GR) prior distribution. The CRPS of a Gaussian mixture has a closed-form representation and is differentiable with respect to the mixture parameters, making it amenable to gradient-based optimisation (Appendix E.4).

We train the model using ten-fold cross-validation with random splits. In each split, we use eight folds for training, one fold for validation and the last fold as test set. For each split, we train five models and average the predictions in probability space. We use the Adam optimiser with a learning rate of 10^{-4} and a batch size of 128. We reduce the learning rate by a factor of 0.3 after 15 epochs without a reduction in validation loss. We train for 100 epochs and use from each ensemble member the model with the lowest validation loss for evaluation. To avoid degenerated mixture weights, we introduce a Dirichlet prior as regulariser. This regulariser takes the form $-\gamma \sum_i \log \alpha_i$. For positive γ this enforces that no mixture weight is close to zero. We use $\gamma = 10^{-4}$.

In Chapter 5 we showed that neural network models suffer from data sparsity for large events, causing systematic underestimation of magnitudes. As a simple mitigation, we suggested upsampling these events in training, i.e., artificially increasing their occurrence. We follow this approach by upsampling events above magnitude 6 with the factor $\rho(M) = \lambda^{M-6}$, where we use $\lambda = 2$. As we analyse probabilistic predictions, we need to take the introduced skew on the distribution into account. For this, we analyse Bayes' rule $\mathbb{P}(M|O_t) \sim \mathbb{P}(O_t|M)\mathbb{P}(M)$. The upsampling replaces $\mathbb{P}(M)$ by $\tilde{\mathbb{P}}(M) = c_1\rho(M)\mathbb{P}(M)$. The model therefore estimates $\tilde{\mathbb{P}}(M|O_t) = c_2\rho(M)\mathbb{P}(M|O_t)$, where c_1 and c_2 are normalisation constants. Note that usually $c_1 \neq c_2$, as the normalisation constant $\mathbb{P}(O_t)$ will change with the upsampling as well. To get true estimates of $\mathbb{P}(M|O_t)$, one would need to rescale the predictions with $1/\rho(M)$. Notably, this scale factor is independent of O_t.

For the results presented in Chapter 6, we refrained from rescaling the predictions for several reasons. First, our upsampling rate of 2 per magnitude step is considerably weaker than the GR law with a tenfold decrease in event occurrence with each magnitude step. Therefore upsampling will not obscure GR tails. In fact, the lower decay rate with magnitude obtained by upsampling allows for better visual representation. Second, our key results compare predictions in different buckets. As each bucket is equally affected by the upsampling, their relative behaviour stays unchanged. However, we note that any quantitative evaluation should take the effect of upsampling into account.

E.3 Teleseismic arrival dataset and model

We downloaded all available manual phase picks for events with magnitudes above 5 from the ISC [International Seismological Centre, 2021] and USGS. We matched the event references to the Global CMT catalog [Ekström et al., 2012] and discarded all events that could not be matched. For all analyses, we used the moment magnitude from Global CMT as the target value. We only use picks with phase label P and discarded all but the first pick for each station and event. We only use picks within an epicentral distance below 97° to avoid core phases. For consistency, we calculated expected first P arrival times using the GCMT event onset times and the ak135 velocity model [Kennett et al., 1995]. If a pick was not within 4 s of the first predicted arrival, we discarded the pick. We use broadband waveforms from the following seismic networks: GE [GEOFON Data Centre, 1993], G [Institut De Physique Du Globe De Paris (IPGP) and Ecole Et Observatoire Des Sciences De La Terre De Strasbourg (EOST), 1982], GT [Albuquerque Seismological Laboratory (ASL)/USGS, 1993], IC [Albuquerque Seismological Laboratory (ASL)/USGS, 1992], II [Scripps Institution Of Oceanography, 1986] and IU [Albuquerque Seismological Laboratory (ASL)/USGS, 1988]. We downloaded the waveforms from the GEOFON and IRIS FDSN webservices. We excluded all stations within 10 km of the coastline as they showed high levels of short-period noise. We do not enforce any further constraint on the signal to noise ratio but note that the usage of manual picks provides an implicit constraint. All waveforms are resampled to 20 Hz sampling rate, filtered between 0.025 Hz and 8 Hz and cut from 35 s before the phase pick to 90 s after the phase pick. We removed the instrument sensitivity but did not restitute the instrument response as we observed acausal artefacts from restitution. We manually inspected the resulting dataset and removed stations with timing errors. As a sanity check, we applied our model to $t = -0.5$ s, i.e., 0.5 s before the annotated P arrivals. The results showed no significant difference from the marginal distribution of magnitudes, indicating no or at least very few cases with severe timing errors or other knowledge leaks. The resulting catalog consists of 37,646 events with 747,824 manually labelled P arrivals from 307 unique seismic stations.

We train the model using ten-fold cross-validation with random splits. In each split, we use 8 folds for training, 1 fold for validation and the last fold as test set. Due to the massively higher computing requirements for the TEAM-LM model compared to the STF model, we did not train an ensemble but only a single model for each split. We use the Adam optimiser with learning rate 10^{-4} and a batch size of 1024. We reduce the learning rate by a factor of 0.3 after 5 epochs without a reduction in validation loss. We train for 100 epochs and use the model with the lowest validation loss for evaluation. We clip gradients to a maximum norm of 1. We use at most 50 input stations. As for the STF model we use upsampling of large magnitude events and did not rescale the outputs. As for TEAM-LM in Chapter 5, we pretrain the feature extraction and the mixture density layers on single station magnitude estimation. As the extensive data augmentation incorporates stochasticity in the validation score, the validation set is evaluated five times with different augmentations after each epoch.

E.4 CRPS of a Gaussian mixture

As we train our network with a CRPS, here, we derive the closed-form solution of the CRPS for a Gaussian mixture. The probability density function (PDF) of a Gaussian mixture is defined as $f(x) = \sum_i \alpha_i \sigma_i^{-1} \varphi(\frac{x-\mu_i}{\sigma_i})$, where φ denotes the PDF of a standard normal random variable. Similarly, we use $\varPhi$ for the cumulative distribution function (CDF) of a standard normal random variable. For deriving the closed-form solution, we use three identities. First, Gneiting and Raftery [2007] note that the CPRS can be written as

$$CRPS(F, x) = \frac{1}{2}\mathbb{E}_F|X - X'| - \mathbb{E}_F|X - x| \tag{E.2}$$

with X and X' independent copies of random variables with CDF F and $\mathbb{E}_F|\cdot|$ the expectation of the absolute value. Note that this identity requires a finite first moment of X, which for a finite Gaussian mixture is always true. Second, for two independent Gaussian random variables $Z \sim \mathscr{N}(\mu, \sigma^2)$ and $Z' \sim \mathscr{N}(\mu', \sigma'^2)$, the sum $Z + Z'$ is a Gaussian random variable with $Z + Z' \sim \mathscr{N}(\mu + \mu', \sigma^2 + \sigma'^2)$. Third, for a Gaussian random variable $Z \sim \mathscr{N}(\mu, \sigma^2)$ the expected absolute value has the following closed-form solution:

$$\mathbb{E}|Z| = 2\sigma^2 \varphi\left(\frac{\mu}{\sigma}\right) + \mu\left(1 - 2\varPhi\left(-\frac{\mu}{\sigma}\right)\right) \tag{E.3}$$

Let X and X' be the Gaussian mixtures and $Z_i, Z_i' \sim \mathscr{N}(\mu_i, \sigma_i^2)$ be the mixture components. We can now calculate the terms of (E.2). For the first term we get:

$$\mathbb{E}|X - X'| = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j \mathbb{E}|Z_i - Z_j'| \tag{E.4}$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j \mathbb{E}|N_{ij}| \tag{E.5}$$

Here, $N_{ij} \sim \mathscr{N}(\mu_i - \mu_j, \sigma_i^2 + \sigma_j^2)$, using the summation of independent Gaussian random variables. The expected value can be computed using (E.3).

Similarly, for the second term of (E.2) we get

$$\mathbb{E}|X - x| = \sum_{i=1}^{n} \alpha_i \mathbb{E}|Z_i - x| \tag{E.6}$$

$$= \sum_{i=1}^{n} \alpha_i \mathbb{E}|Y_i| \tag{E.7}$$

with $Y_i \sim \mathscr{N}(\mu_i - x, \sigma_i)$. This again allows us to calculate the term using (E.3). Therefore, the CRPS of the Gaussian mixture can be computed in closed form. Furthermore, the solution is differentiable in α_i, μ_i and σ_i, which is required for neural network training. For α_i differentiability is clear, as the CRPS only depends linearly on the mixture weights. For μ_i and σ_i, differentiability results from the differentiability of φ and $\varPhi$. While this does not hold for $\sigma_i = 0$, our network architectures ensure $\sigma_i > 0$. Calculating the closed-form has compute complexity in $\mathcal{O}(n^2)$. As the number of mixture components is low ($n < 25$) and the calculation can trivially be vectorised, this does not pose a computational issue and computation times are negligible compared to the neural network computations.

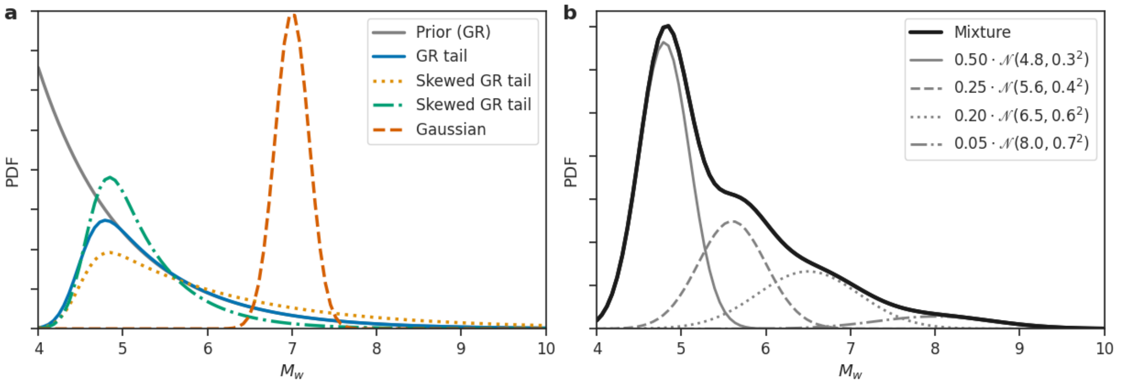


Figure E.1: **a** Possible shapes of $\mathbb{P}(M|O_t)$ for an ongoing event. The Gutenberg-Richter prior is rescaled to fit the tail behaviour of the other distributions. **b** Exemplary Gaussian mixture with mixture size 4, showing both the individual components and the resulting mixture PDF.

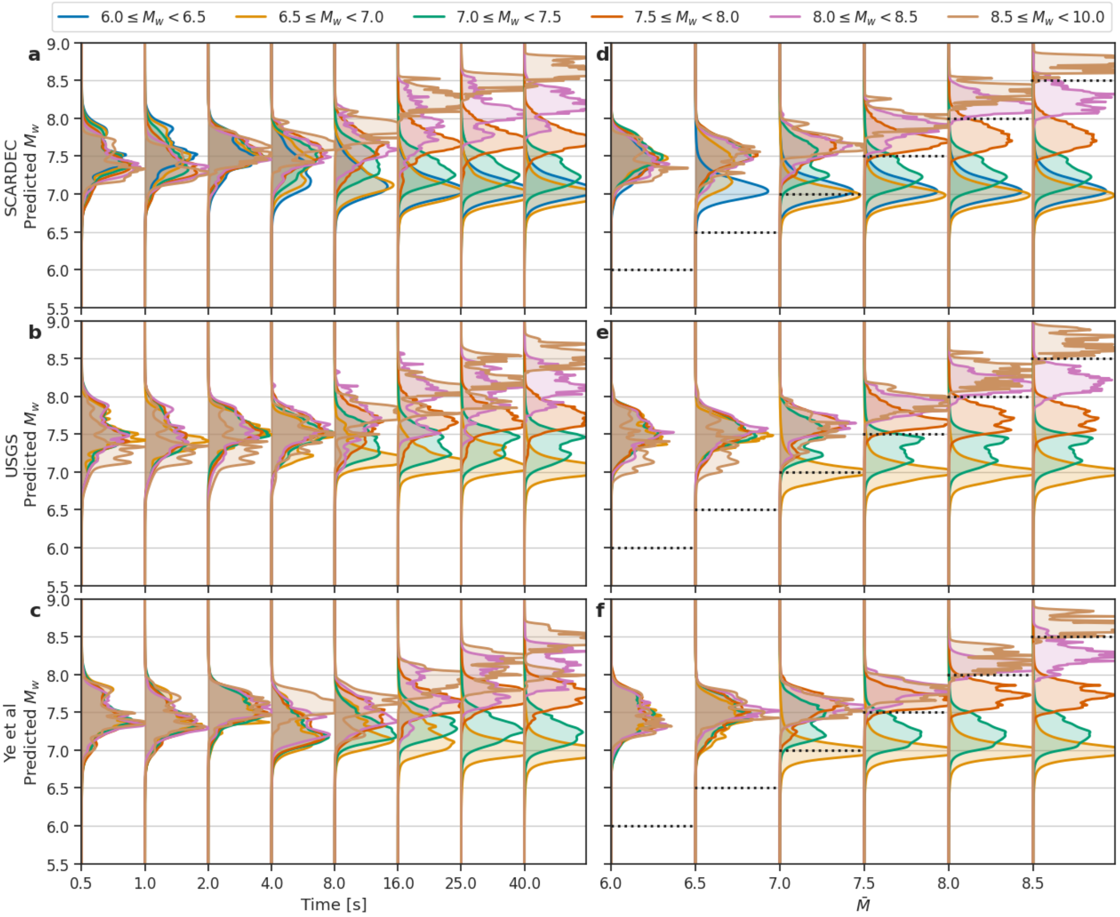


Figure E.2: Results similar to Figure 6.5, but using the USGS STFs instead of the SCARDEC ones for model training. For details see the description of Figure 6.5. Note that the marginal distribution of magnitudes in the USGS dataset is considerably different from the SCARDEC dataset, i.e., it is missing smaller events. This is clearly reflected in the results, in particular, in the overestimation of small SCARDEC events.

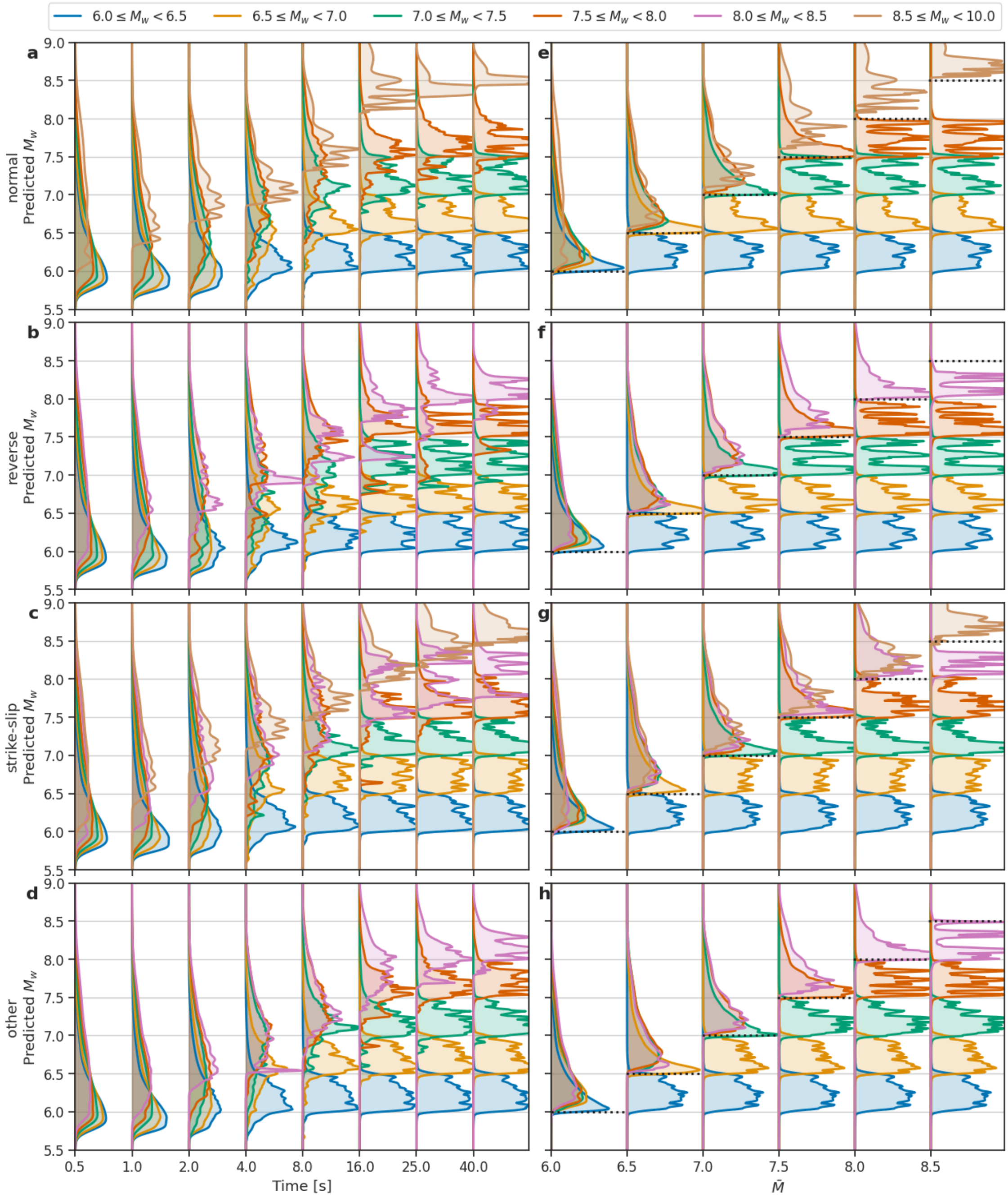


Figure E.3: Average $\mathbb{P}(M|O_t)$ (**a-d**) and $\mathbb{P}(M|O_{\bar{M}})$ (**e-h**) by magnitude bin for the SCARDEC dataset. This figure displays the same results as shown in Figure 6.5a, d but with the analysis split by focal mechanism type. Focal mechanism types were derived from the Global CMT solution using the principal axes of the moment tensors. If the n axis was within 30° of the horizontal, the event was classified as "normal" (t axis more vertical than p axis) or "reverse" (p axis more vertical than t axis). If the n axis was within 30° or the vertical axis, the event was classified as "strike-slip". All remaining events were classified as "other". PDFs were truncated to avoid overlap between different times/base magnitudes. Black dotted lines in **e-h** indicate the current base magnitude. For events shorter than the given time (**a-d**) or with final magnitudes below the base magnitude (**e-h**), the estimation from the final sample of the STF was used.

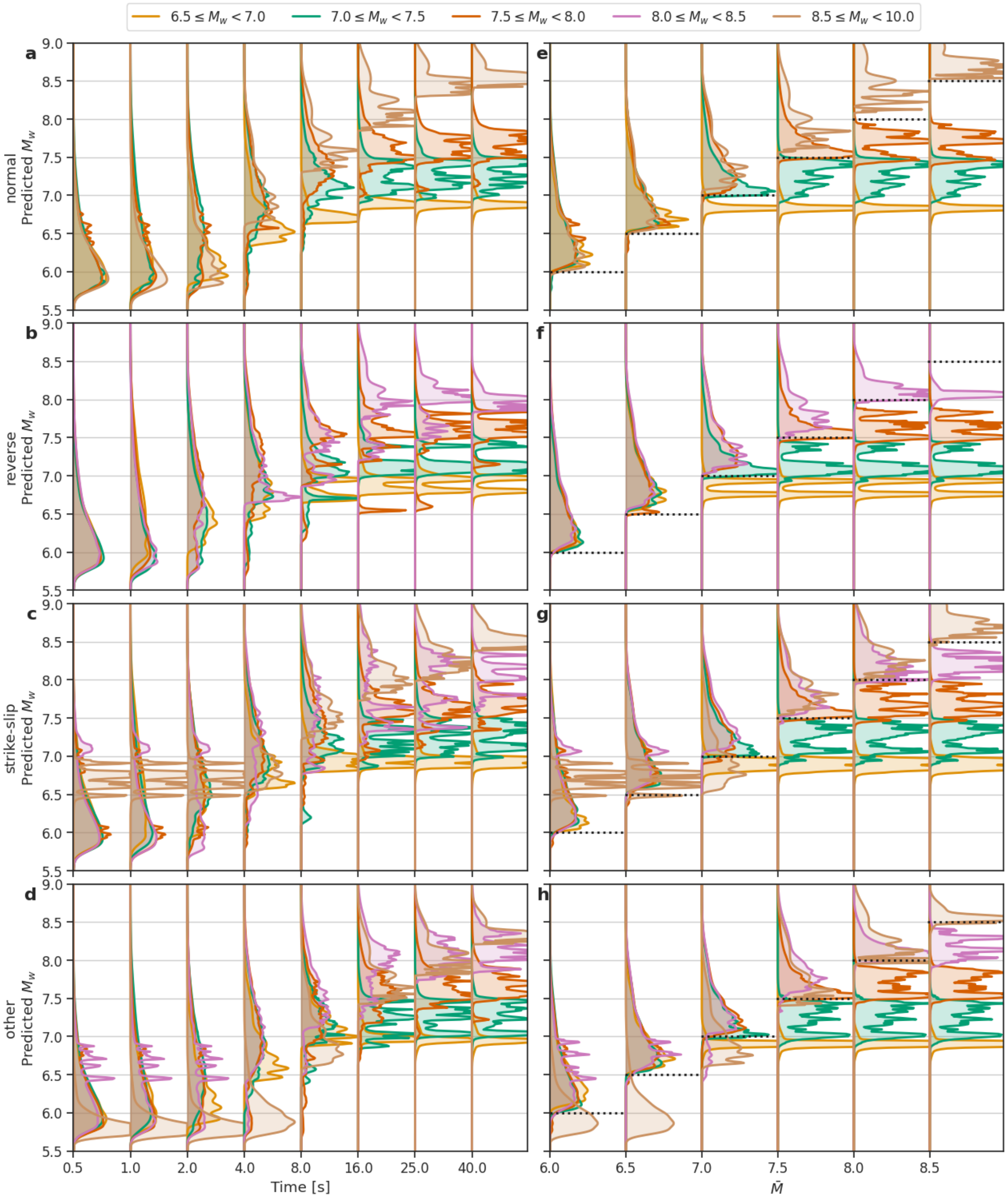


Figure E.4: Average $\mathbb{P}(M|O_t)$ (**a-d**) and $\mathbb{P}(M|O_{\bar{M}})$ (**e-h**) by magnitude bin for the USGS dataset. This figure displays the same results as shown in Figure 6.5b, e but with the analysis split by focal mechanism type. Otherwise, see caption of Figure E.3 for further explanations.

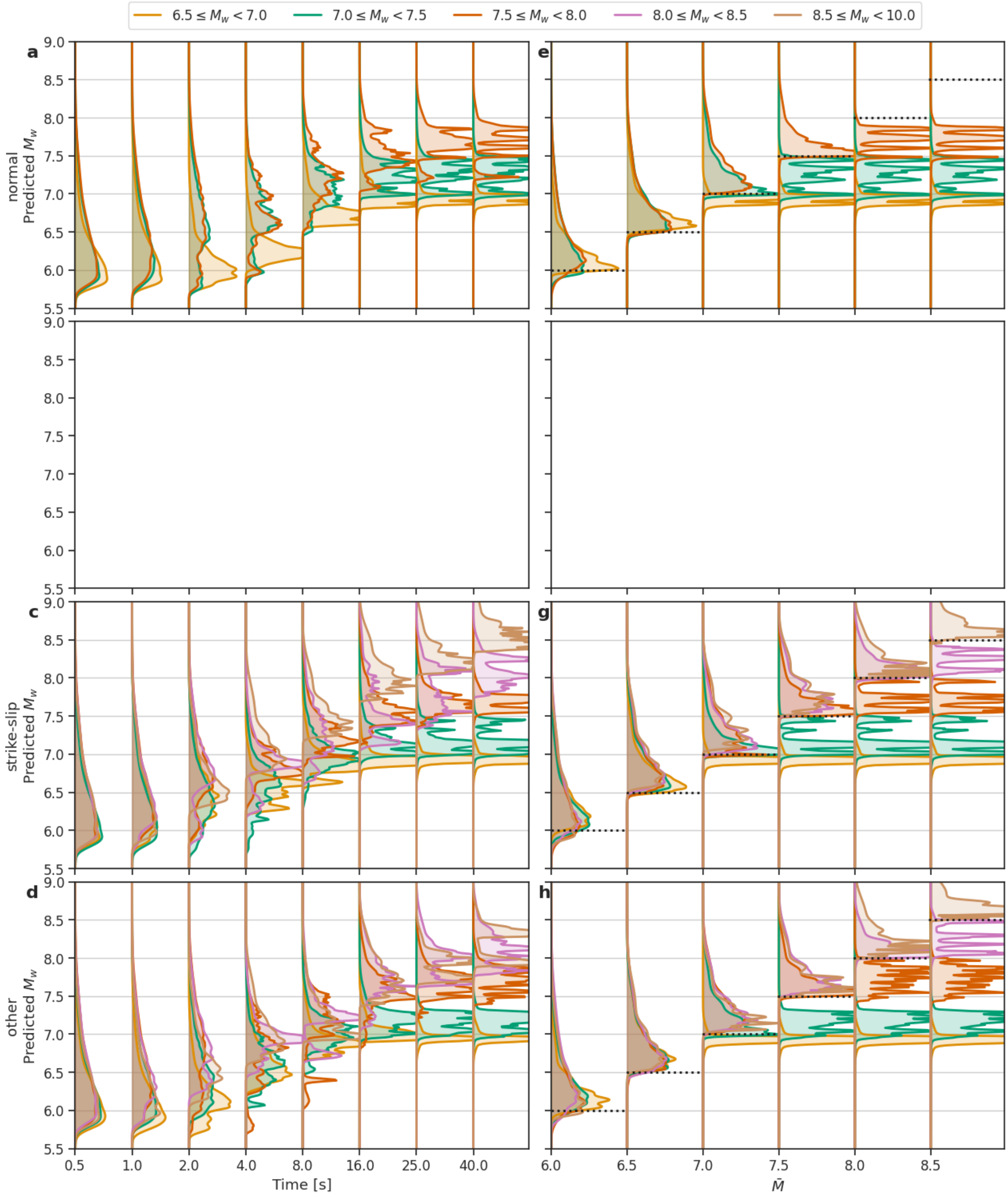


Figure E.5: Average $\mathbb{P}(M|O_t)$ (**a-d**) and $\mathbb{P}(M|O_{\bar{M}})$ (**e-h**) by magnitude bin for the Ye et al dataset. This figure displays the same results as shown in Figure 6.5c, f but with the analysis split by focal mechanism type. The dataset contains no examples of reverse faulting, therefore the corresponding panels are left empty. Otherwise, see caption of Figure E.3 for further explanations.

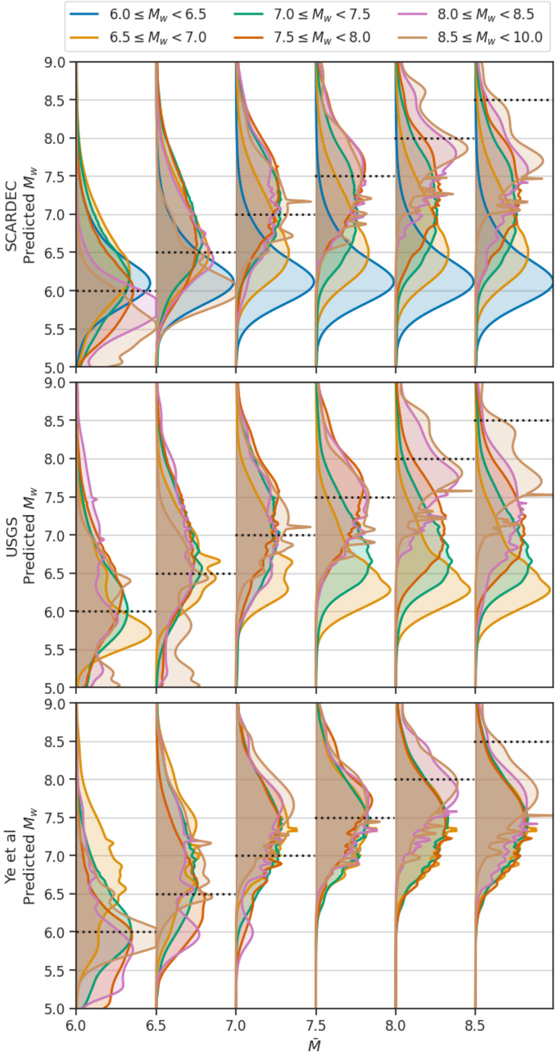


Figure E.6: $\mathbb{P}(M|O_{\bar{M}})$ binned by magnitude using the three STF datasets for determining $t_{\bar{M}}$. The figure is otherwise equivalent to Figure 6.7b; for more details of the figures format also see the caption of Figure 6.7. Note that the events shown differ between the panels, as only those events included in the respective STF datasets can be shown.

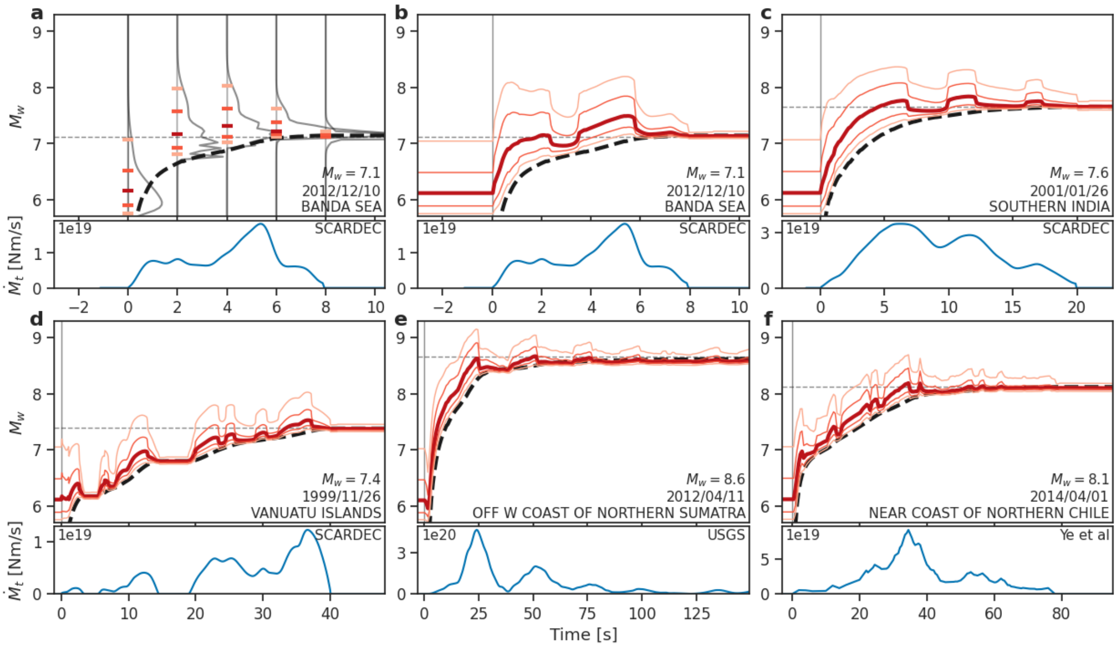


Figure E.7: Results similar to Figure 6.4, but using the optimal SCARDEC STFs instead of the average ones for model training and evaluation on SCARDEC. For details see the description of Figure 6.4.

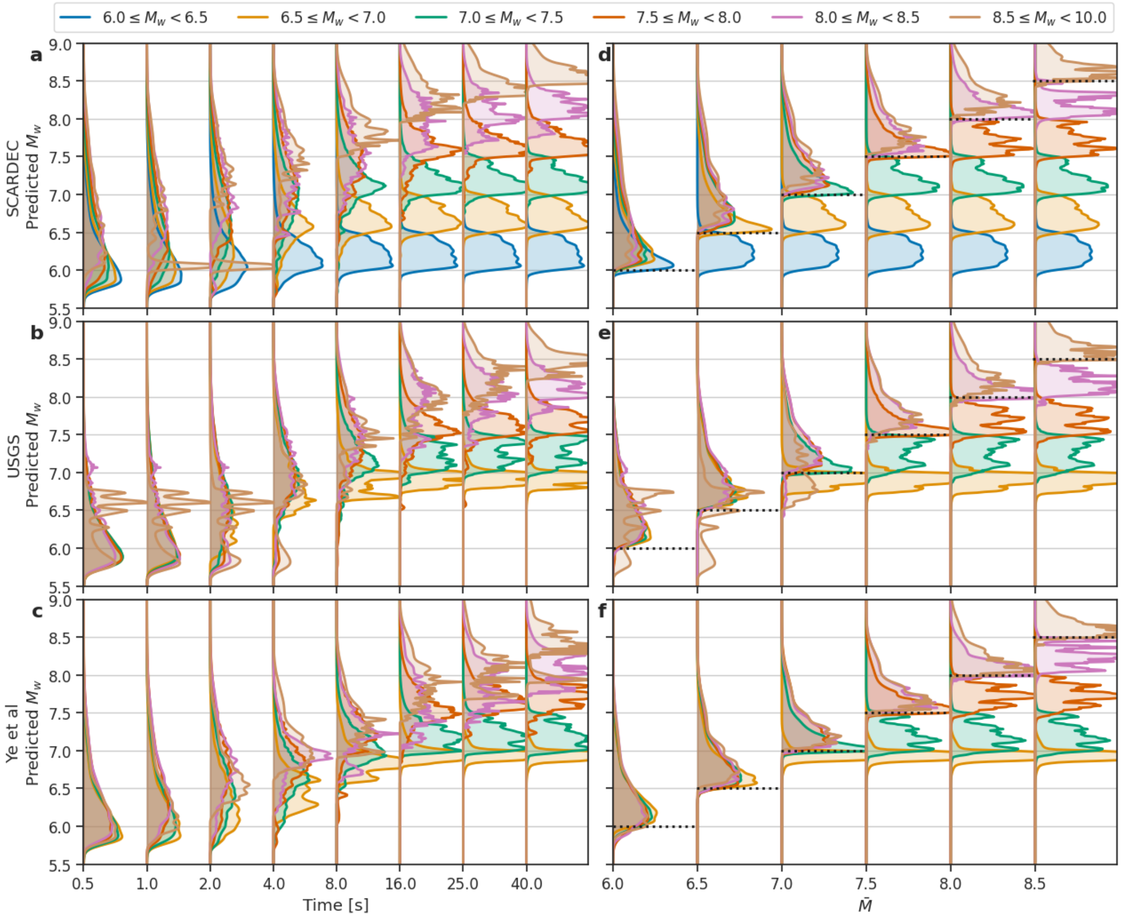


Figure E.8: Results similar to Figure 6.5, but using the optimal SCARDEC STFs instead of the average ones for model training and evaluation on SCARDEC. For details see the description of Figure 6.5.

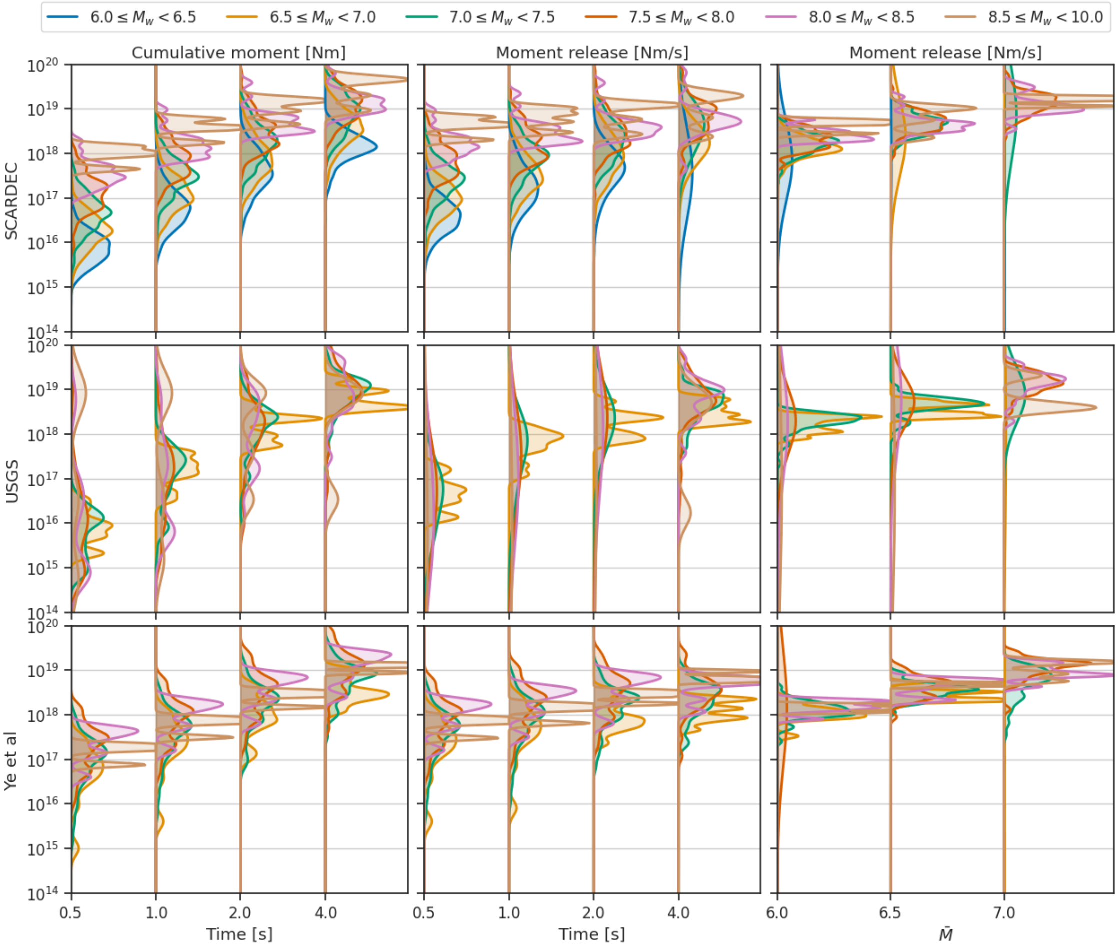


Figure E.9: Comparative analysis of the early moment release for the three STF datasets binned by magnitude. Each row represents one STF dataset. The left column shows cumulative moment release at time t, the middle column current moment release at time t, the right column moment release at the time when a magnitude $\bar{M}$ is reached. Notably, while all three measures differ between the magnitude buckets for SCARDEC, no such behavior is visible for the USGS or Ye et al datasets. This points at a processing artefact in SCARDEC rather than a physical explanation.

Selbstständigkeitserklärung

Ich erkläre, dass ich die Dissertation selbständig und nur unter Verwendung der von mir gemäß § 7 Abs. 3 der Promotionsordnung der Mathematisch-Naturwissenschaftlichen Fakultät, veröffentlicht im Amtlichen Mitteilungsblatt der Humboldt-Universität zu Berlin Nr. 42/2018 am 11.07.2018, angegebenen Hilfsmittel angefertigt habe.

Ort, Datum, Unterschrift