

Article

Artificial Intelligence and Big Data Analytics for Seismic Hazard Assessment: Methodological Advances and Computational Frameworks for the Marmara Region, Türkiye

Polina Lemenkova *  and Abdullah Can Zulfikar 

Disaster and Emergency Management Department, Institute of Earthquake Engineering and Disaster Management, Istanbul Technical University (ITU), Istanbul 34469, Türkiye; aczulfikar@itu.edu.tr

* Correspondence: lemenkova@itu.edu.tr

Abstract

The Marmara region of Türkiye, situated along the North Anatolian Fault Zone (NAFZ), constitutes one of the most seismically active and densely monitored zones globally. Given the region's high vulnerability and the catastrophic impacts of historical events—notably the 1999 İzmit and 2023 Kahramanmaraş sequences—there is a critical need for advanced seismic hazard risk assessment (SHRA) methods that move beyond static models. This review examines the paradigm shift from traditional geophysics to big data seismology, characterized by the “Five Vs”: volume, velocity, variety, veracity, and value. Critically, we distinguish between two fundamentally different problems: Earthquake Early Warning (EEW), which operates on sub-second timescales after rupture initiation, and probabilistic earthquake forecasting, which operates on timescales of years to decades. The study discusses how cloud-native platforms such as Azure Databricks, combined with data pipelines using Apache Kafka (version 3.5.1) and Apache Spark (version 4.1.2), enable the real-time processing of petabyte-scale seismic sensor streams. Key technological tools, including Physics-Informed Neural Networks (PINNs) and deep learning models such as PhaseNet, are analyzed for their demonstrated ability to enhance EEW systems through sub-second phase picking and automated event detection. Seismic tomography is also undergoing AI-enabled transformation, yielding higher-resolution subsurface imaging. We present statistical validation metrics and uncertainty quantification methods essential for credible hazard assessment. By addressing computational bottlenecks through hybrid computing architectures and edge computing, this framework aims to improve the warning lead time for Istanbul's critical infrastructure. This work provides a structured roadmap for bridging the gap between traditional seismic data analysis and operational predictive analytics in the Marmara region.



Academic Editor: Jamal Jokar Arsanjani

Received: 6 May 2026

Revised: 20 May 2026

Accepted: 26 May 2026

Published: 2 June 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: big data; seismology; cloud computing; geophysics; machine learning; earthquake early warning; signal processing; Apache Spark; North Anatolian Fault

PACS: 91.30.Ab; 91.30.Pd; 07.05.Mh; 89.20.Ff

MSC: 86A15; 68T07

JEL Classification: Q54; C45; O33

1. Introduction

1.1. Background

Seismic hazard monitoring systems prioritize the early detection and probabilistic assessment of earthquake events to mitigate geotechnical risks [1–3]. Modern geophysical centers analyze thousands of global sensor streams per second, generating millions of daily data points [4–6]. However, the transition to operational seismic monitoring necessitates processing capabilities that exceed the limits of traditional geophysical analysis [7]. The integration of Artificial Intelligence (AI) and cloud computing has fundamentally transformed this landscape, enabling the treatment of global seismic signals as unified big data repositories rather than isolated regional records [8–10].

The paradigm shift from traditional geophysics to big data seismology represents a fundamental change in analytical methodology [11–14]. Contemporary seismic datasets require novel frameworks for storage and high-throughput processing to extract reliable prognostic insights [15,16]. Crucially, modern seismic networks demand a departure from legacy data management systems toward distributed, scalable data science architectures, Figure 1.

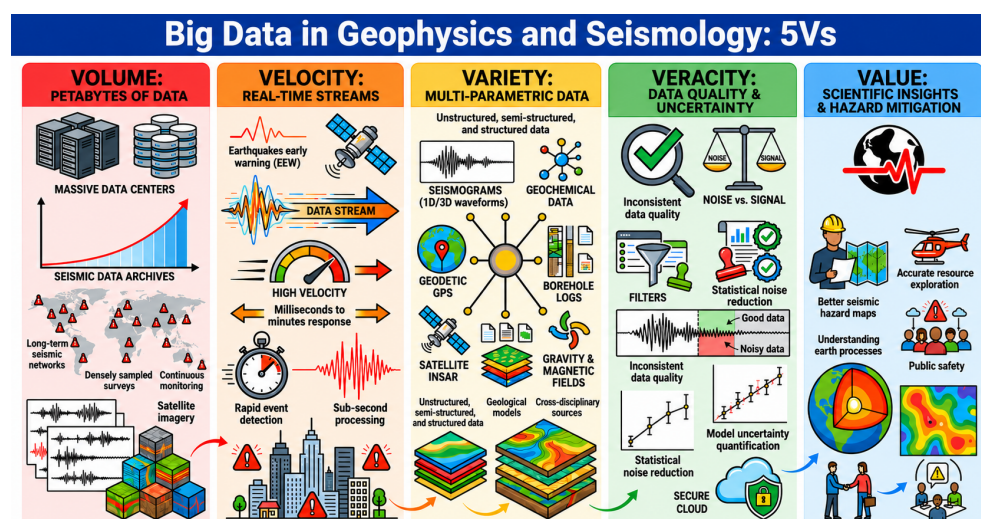


Figure 1. Big data in geophysics and seismology: the 5Vs framework. Diagram source: authors.

The “Five Vs” of big data—volume, velocity, variety, veracity, and value—define the principal advantages over regional geophysical databases [17–19]. While traditional archives typically manage gigabytes of localized data, modern big data systems routinely ingest petabytes of global sensor data [20]. The velocity of these systems enables the processing of real-time data streams, providing the immediate and latency-critical insights required for seismic risk evaluation and rapid decision-making [21,22].

The variety of big data in geophysics is defined by a spectrum of heterogeneous data structures [23]. The complexity of formats arises from the diverse data structures and sources that must be integrated and jointly analysed [24]. Even tabular CSV data typically wraps heterogeneous information pulled from multiple geophysical sensors [25,26]. Finally, the veracity dimension pertains to data quality and integrity [27,28]. Seismic datasets may include incomplete signals, duplicate event entries, inconsistent formatting, malfunctioning sensors, or outdated information [29].

These five dimensions create interconnected challenges that conventional seismological methods are ill-equipped to handle. Traditional analysis largely relies on manual examination of datasets which process massive volumes of historical seismic data available in archives and from sensors flowing into online geophysical repositories [30–32]. Process-

ing such large data massifs enables the discovery of anomalies in mantle processes with better precision and speed, yielding real-time data characterizing location and magnitude of seismic events, together with unstructured cartographic data and relevant geological characteristics [33,34]. Consequently, big data analytics in seismology support predictive analytics that process sensor data from thousands of stations simultaneously.

1.2. Research Problem and Objectives

Big data technologies are revolutionizing geophysics by enabling dynamic, high-resolution risk assessment. In Seismic Hazard Risk Assessment (SHRA), big data has shifted the paradigm from static, low-frequency maps to dynamic, real-time monitoring models. Data integration and real-time processing allow historical data analysis, current data monitoring, and future data forecasting [35]. Recent decades have witnessed diverse geological and climate hazards whose cumulative effects on the environment are well-documented [36]. Seismic risks are particularly prominent due to their devastating consequences for society, demanding effective quantitative seismic risk assessment methods [37,38].

Before proceeding, it is essential to distinguish between two fundamentally different scientific problems that are often conflated in the literature: (i) earthquake early warning (EEW) operates on timescales of seconds, detecting P-waves from an initiated rupture and issuing alerts before the destructive S-waves arrive; and (ii) earthquake forecasting operates on timescales of years to decades, using probabilistic seismicity models to estimate long-term rupture probability. These two paradigms require distinct data architectures, latency constraints, and validation frameworks.

The Turkish government has demonstrated remarkable commitment to seismic risk mitigation through two decades of systematic emergency preparedness initiatives [39]; however, advancing these efforts requires integrating modern big data analytics frameworks and Python-based geophysical tools to enhance real-time seismic monitoring and response capabilities. Addressing this technological gap is both timely and critical, as seismological research increasingly demands scalable, high-performance computational solutions capable of processing heterogeneous data streams in real time.

This paper contributes to the challenge of big data technologies in geophysics by analyzing how data pipelines support modern operational seismic hazard risk assessment. We discuss SHRA in the context of big data and high-scale geophysical analytics, bridging the gap between traditional data analysis in seismology and predictive analytics. Unlike conventional approaches that rely on processing single CSV files, a modern big data pipeline can architect an SHRA system that handles the volumes, velocity, and variety of seismic signals simultaneously.

2. Seismic Vulnerability in Türkiye: From Tectonic Plate Interactions to Historical Disaster Impacts

2.1. Geology and Tectonics in Türkiye

Türkiye is highly vulnerable to seismic hazards, especially in regions near active tectonic faults [40,41]. Systematic geological monitoring is essential for comprehensive seismic risk assessment, underscoring the need for improved methodologies [42]. Seismically active areas in Türkiye lie at the margins of the Anatolian Block [43], within a zone of lithosphere collision involving three major tectonic plates: the Eurasian Plate, the African Plate, and the Arabian Plate.

The Anatolian Block is driven westward as the Arabian Plate advances northward against the Eurasian Plate, while the African Plate subducts beneath the Hellenic Arc [44,45]. These plate interactions generate frequent and powerful earthquakes at tectonic boundaries.

The most prominent feature is the North Anatolian Fault Zone (NAFZ), a 1200 km right-lateral strike-slip fault extending from the Gulf of Saros to Karlıova [46–48]. Formed 13–11 million years ago, it remains seismically active, capable of producing earthquakes up to magnitude 8.0 [49]. Current deformation, particularly in the Marmara region, is asymmetric and strongly influenced by the complex regional geology of the NAFZ [50,51].

2.2. Historical Outline of Seismic Events in Türkiye

The historical record of the Anatolian region reveals a relentless cycle of seismic activity, Figure 2. In the Aegean region, from 496 BC to 1949 AD, the area experienced a total of 20 moderate to severe earthquakes [52,53]. The 1956 Amorgos earthquake and tsunami is one of the largest instrumentally recorded events in the region [54]. Strong tsunamis in the Mediterranean have been systematically re-evaluated for EEW planning [55]. Seismic risk across Türkiye, the Aegean, and the eastern Mediterranean has been shaped by recurrent large-magnitude earthquakes over historical timescales [56]. Many ancient events resulted in devastating tsunamis, with notable occurrences in 1389, 1856, 1866, 1881, and 1949 [57]. The 1881 Chios-Çeşme earthquake caused thousands of fatalities and significant coastal changes [58]. The M_w 7.4 İzmit earthquake of 17 August 1999 remains one of the best-studied large events in the region [59,60]. A significantly heightened probability ($\sim 50\%$) of a $M \geq 7$ event has been estimated within 30 years of the 1999 stress transfer [61]. Dynamic triggering of tectonic tremors and earthquakes has been documented in the broader region following major seismic sequences [62,63]. Most recently, on 6 February 2023, Türkiye experienced a catastrophic seismic sequence with a mainshock of M_w 7.8, followed by major earthquakes of M_w 7.7 and 7.6. These events caused total destruction in Antakya, including severe damage to historical mosques and minarets [64–66]. The 2017 Bodrum-Kos earthquake (M_w 6.6) provided important geodetic constraints on fault slip distribution from GPS observations [67].

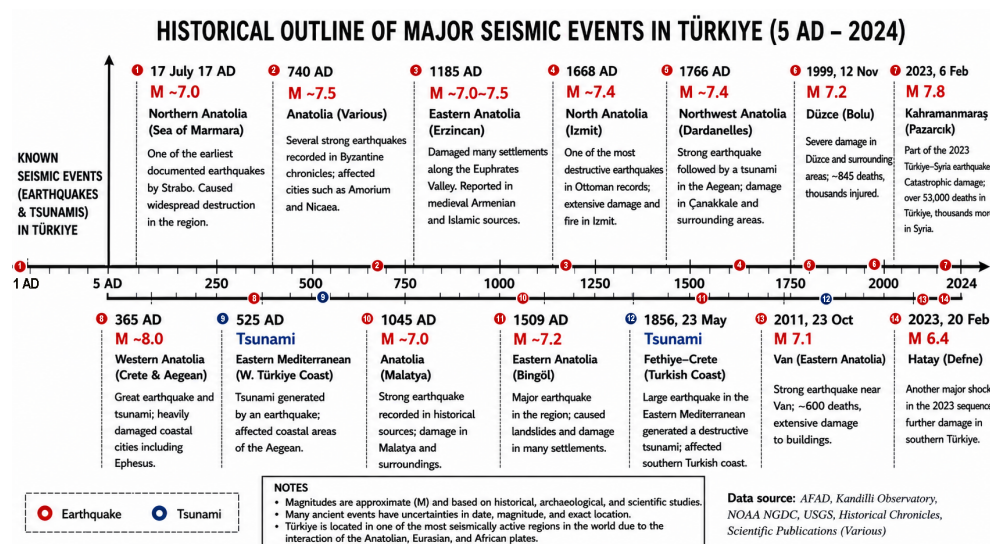


Figure 2. Historical outline of major seismic events in Türkiye. Diagram source: authors.

2.3. Seismic Risk Assessment and the Distinction Between EEW and Earthquake Forecasting

The 2023 Kahramanmaraş sequence [68,69] reinforced the urgent need to clearly separate two domains of operational seismology that serve different societal functions. Research and technological development in EEW systems have become critical components of seismic risk mitigation in the Anatolian region. EEW is fundamentally a real-time detection and alerting problem: once a rupture has begun, P-wave signals are detected, source parameters are estimated within seconds, and automated alerts are issued before

destructive shaking arrives. In contrast, earthquake forecasting—such as the probability estimate—is a probabilistic, long-term assessment of where and when future earthquakes are likely to occur, based on fault mechanics and historical seismicity.

This distinction matters for data infrastructure: EEW requires deterministic, sub-second pipelines with strict latency constraints (<100 ms), while forecasting models require access to decades of historical catalogs, geodetic strain data, and paleoseismic records. The Kandilli Observatory and Earthquake Research Institute (KOERI) has operationalized a new-generation EEWS across Türkiye, leveraging a dense network of stations to issue alerts within seconds [70]. Innovative strategies also explore hybrid models, such as using mosque loudspeakers to disseminate warnings [71]. Machine learning is transforming raw telemetry into actionable protocols, predicting potential damage zones and automatically triggering safety actions [72]. For high-stakes life-safety decisions, interpretable models are preferable to opaque black-box systems [73].

3. Research Gap Identification and Contribution

To our knowledge, no comprehensive review exists that integrates big data processing architectures, statistical validation frameworks, and the EEW/forecasting distinction specifically for the Marmara region. While individual case studies on AI in seismology exist [74], integrative reviews addressing big data pipelines, uncertainty quantification, and cloud governance for Turkish seismic monitoring are lacking. This article provides a comprehensive analysis of the evolution of techniques for big data processing in the context of seismic risk assessment in Türkiye, with a focus on the Marmara region and the NAFZ. Mitigating earthquake risks requires both an EEW system and consistent probabilistic seismic monitoring underpinned by big data, Figure 3.

DEVELOPMENT OF BIG DATA IN SEISMOLOGY, EEW & GEOPHYSICAL MONITORING IN ISTANBUL & TURKEY

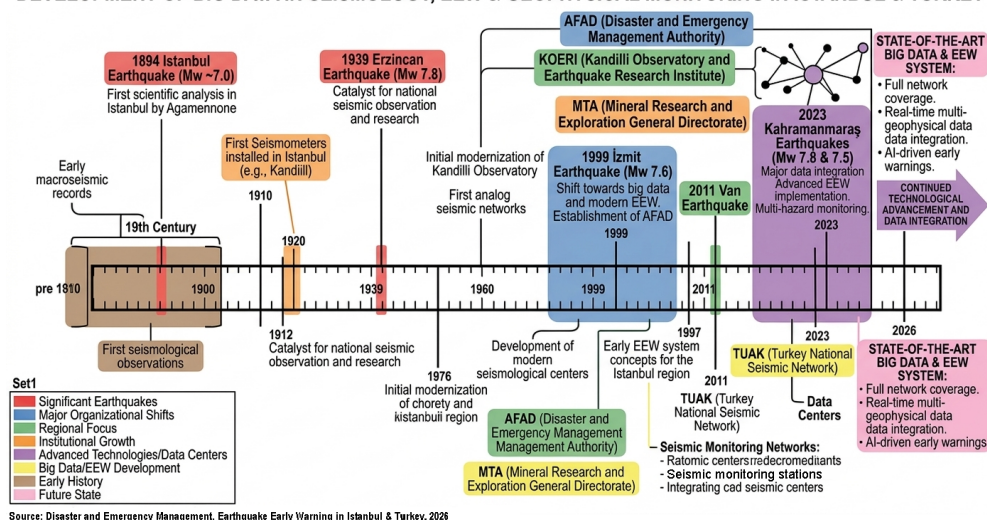


Figure 3. Development of Big Data in Seismology, EEW & Geophysical Monitoring in Istanbul, Türkiye. Diagram source: authors.

The current limitations of conventional seismological processing are summarized in Table 1, which highlights both the technical constraints and their implications for Marmara region monitoring.

Table 1. Limitations of existing geophysical processing approaches and their Marmara-region implications.

Approach	Core Constraint	Big Data & Geophysical Impact
Traditional Inversion	Non-uniqueness	Multiple subsurface models satisfy the same observed data. High-dimensionality exacerbates uncertainty quantification; confidence intervals on hazard maps are rarely reported.
Linearized Models	Oversimplification	Standard kernels assume a linear Earth. Complex NAFZ geology is highly non-linear; forcing linearity produces false anomalies and signal artifacts in high-volume datasets.
Manual QC	Human Bottleneck	High-density 3D/4D seismic surveys generate petabytes of traces. Manual QC is the primary latency in the processing pipeline and is infeasible for continuous monitoring.
HPC Constraints	Data Movement	In seismic migration (e.g., RTM), moving massive velocity models from storage to compute nodes consumes more energy than the FLOPs performed (Von Neumann bottleneck ¹).

¹ The Von Neumann bottleneck refers to the limited throughput between the central processing unit (CPU) and memory compared to the amount of memory.

4. Big Data Processing in Geophysics

4.1. Hybrid Computing: HPC and Cloud Architecture

One of the most demanding data processing tasks in geophysics involves massive datasets from nodal sensors and complex inversions, such as Full Waveform Inversion (FWI) [75–78]. Since nodal datasets can reach petabyte scales, geophysicists no longer move all data to a single location; instead, they use a hybrid computing approach [79], combining the advantages of both HPC and cloud computing [80,81].

Handling big data in geophysics requires a shift from sequential processing to a hybrid architecture that blends high-performance computing (HPC) with cloud-native intelligence. Cloud computing and storage have become fundamental to modern seismological workflows [82]. To process massive nodal datasets, geophysicists utilise a hybrid architecture combining GPU-accelerated HPC clusters with elastic cloud scalability, moving the processing code to the data to avoid massive transfer bottlenecks [83,84]. FWI remains on GPU-accelerated HPC clusters due to the tight coupling required between processors [85]. Bursty tasks such as initial signal denoising are offloaded to elastic cloud resources. The hybrid infrastructure accelerates high-resolution FWI by embedding physical laws into Physics-Informed Neural Networks (PINNs) [86,87].

To quantify the performance of this hybrid approach, we adopt the following statistical validation framework for Marmara-region EEW pipelines:

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN} \quad (1)$$

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2)$$

where TP , FP , and FN denote true positives, false positives, and false negatives in event detection, respectively. For EEW, false negatives (missed alerts) carry higher societal cost than false positives (false alarms).

$$\text{RMSE}(\hat{M}, M) = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{M}_i - M_i)^2} \quad (3)$$

This root-mean-square error metric is used to evaluate magnitude estimation accuracy across N detected events, where $\hat{M}_i$ and M_i denote estimated and catalog magnitudes, respectively.

Uncertainty in the estimated epicentral distance Δr is propagated through the EEW warning time estimate:

$$T_{\text{warn}} = \frac{\Delta r}{V_S} - T_{\text{proc}} \quad (4)$$

where V_S is the S-wave velocity and T_{proc} is the end-to-end processing latency. For Istanbul, with $\Delta r \approx 50$ km from the NAFZ Marmara segment and $V_S \approx 3.5$ km/s, $T_{\text{warn}} \leq 14.3$ s under ideal conditions.

For remote nodal arrays, smart nodes perform initial data compression and basic quality checks at the source, see Table 2. Real-time inversions for finite fault slip models and rupture geometry using high-rate GPS data complement seismic waveform analysis in EEW pipelines [88].

Table 2. State-of-the-art big data processing in geophysics (2026 Framework).

Geophysical Task	Current Advanced Strategy	Technology Stack
High-Density Nodal Ingest	Decoupled asynchronous streaming from 100 k+ channel nodes; real-time metadata indexing.	Azure Blob/S3, Apache Kafka, Zarr/ASDF Formats
Elastic FWI & Velocity Modeling	Multi-parameter PINN-based inversion to resolve complex salt geometries and anisotropy.	NVIDIA H200 Clusters, PyTorch version 2.12.0/JAX-Seismic
Seismic Signal Enhancement	Self-supervised Deep Learning Autoencoders for 5D interpolation and ghost reflection removal.	Transformer-based Denoising, GANs
Microseismic Monitoring	Automated event detection and location using Edge-AI for real-time hydraulic fracture mapping [89].	NVIDIA Jetson AGX Orin, 5G Telemetry
4D Reservoir Characterization	Digital Twin synchronization integrating seismic, EM, and production data for predictive flow modeling [90].	Azure Digital Twins, NVIDIA Omniverse
Multi-Physics Integration	Joint inversion of gravity, magnetic, and seismic data via foundation models for geophysical interpretation.	Foundation Models, Graph Neural Networks (GNNs)

4.2. Physics-Informed Neural Networks (PINNs) and Deep Learning for Phase Picking

The most demanding inversions are now handled by PINNs [91]. Unlike traditional AI, which might suggest geologically impossible structures, PINNs embed the actual wave equations (the physics) into the neural network's loss function. Ensemble-based batch inversion approaches, such as the iterative ensemble Kalman smoother, provide complementary strategies for large seismic datasets [92]. Physics-informed deep learning has been demonstrated to solve the wave equation directly, with strong potential for seismic modelling [93]. The governing acoustic wave equation embedded in the PINN loss is

$$\frac{\partial^2 u}{\partial t^2} - v^2(x) \nabla^2 u = f(x, t) \quad (5)$$

where $u(x, t)$ is the wavefield, $v(x)$ is the P-wave velocity model, and $f(x, t)$ is the seismic source function. The total PINN loss function combines data misfit and physics residual:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{data}} + \lambda \mathcal{L}_{\text{physics}} \quad (6)$$

where λ is a weighting hyperparameter controlling the trade-off between data fitting and physical consistency.

For phase picking, the convolutional neural network PhaseNet achieves a reported P-wave pick precision exceeding 95% on the STEAD benchmark dataset, reducing manual analyst time by several orders of magnitude.

The uncertainty in P-wave arrival time σ_t propagates directly into epicentral location uncertainty, Equation (7):

$$\sigma_r = V_P \cdot \sigma_t \quad (7)$$

For $\sigma_t = 0.05$ s (typical for PhaseNet) and $V_P = 6.0$ km/s, $\sigma_r \approx 0.3$ km, which is adequate for EEW source parameter estimation.

4.3. Influence of Seismic Wave Diversity on Signal Processing in Marmara Geophysical Monitoring

Geophysical monitoring in the Marmara region requires advanced signal processing techniques because seismic recordings contain multiple wave types with distinct propagation characteristics. Bulk waves, including compressional (P) and shear (S) waves, travel through heterogeneous crustal structures with different velocities and polarization states. Their propagation velocities are commonly expressed as Equation (8):

$$V_P = \sqrt{\frac{K + \frac{4}{3}\mu}{\rho}}, \quad V_S = \sqrt{\frac{\mu}{\rho}}, \quad (8)$$

where K is the bulk modulus, μ is the shear modulus, and ρ is density. Since P -waves propagate faster than S -waves, arrival-time separation forms the basis of earthquake localization and early warning algorithms in the Marmara seismic network.

Surface waves, particularly Rayleigh and Love waves, exhibit lower velocities and strong dispersive behavior due to stratified sedimentary basins beneath the Sea of Marmara. Their phase velocity depends on frequency, as in Equation (9):

$$c(\omega) = \frac{\omega}{k(\omega)}, \quad (9)$$

where ω denotes angular frequency and k is the wavenumber. Dispersion complicates waveform inversion because low-frequency components penetrate deeper structures while high-frequency components remain sensitive to shallow layers. Consequently, adaptive time-frequency filtering and wavelet transforms are frequently employed to isolate modal energy.

Head waves generated at velocity discontinuities further complicate processing because critically refracted arrivals may overlap with direct body waves. Non-volcanic deep tremors, which obey a distinct scaling law [94], were first documented in subduction zones [95] and must be discriminated from tectonic signals in broadband monitoring. In offshore monitoring systems, Rayleigh–Lamb waves propagating in thin sedimentary layers and engineered structures introduce additional multimodal interference. Their displacement field may be represented as Equation (10):

$$u(x, t) = Ae^{i(kx - \omega t)}, \quad (10)$$

where modal interactions produce frequency-dependent attenuation and polarization rotation. Accurate discrimination of these wave classes is therefore essential for robust event detection, structural imaging, and microseismic interpretation throughout the Marmara fault system.

5. Cluster Computing and Distributed Data Processing

5.1. Architecture Paradigms for Marmara Regional Seismology

To manage the heterogeneous data demands of the Marmara region, cluster architectures are categorized into three patterns. The Master–Worker paradigm, utilizing Apache Spark, facilitates high-throughput batch analytics essential for computationally intensive FWI and 3D tomographic modeling of the NAFZ. For high-density nodal arrays, a Peer-to-Peer pattern (e.g., Apache Cassandra) employs decentralized gossip protocols and N-replication to ensure fault tolerance. A Hybrid Broker-Based architecture using Apache Kafka provides the decoupled scaling necessary to ingest and partition massive telemetry streams, Figure 3.

5.2. Apache Spark and PySpark for Seismic Data Processing

The following eight code scripts illustrate the practical application of Apache Spark and related Python version 3.14.5 tools in big data seismology. These examples address the full pipeline from raw ingestion to validated event detection.

5.2.1. PySpark-Based Streaming and Windowed Processing of Seismic Waveforms

Listing 1 demonstrates a scalable real-time ingestion and preprocessing workflow for continuous seismic waveform monitoring using Apache Spark Structured Streaming. The implementation is particularly suitable for dense seismic observation systems deployed around the Marmara fault zone, where waveform streams originating from AFAD-compatible MiniSEED archives must be processed with minimal latency. The script establishes a distributed computation environment capable of handling high-frequency seismic telemetry while preserving temporal synchronization among stations.

The first section imports the required PySpark modules responsible for session management, schema construction, and temporal windowing operations. The `SparkSession` object initializes the distributed execution engine and defines the application context through `appName("MarmaraSeismicIngest")`.

The configuration parameter `spark.sql.shuffle.partitions = 200` controls the degree of parallelism during shuffle-intensive operations such as aggregation and grouping. Increasing partition counts improves scalability for large seismic networks containing many stations and channels.

Here, a structured schema is subsequently defined using `StructType`. This stage is essential because streaming waveform packets arriving from Kafka are serialized in JSON format and must be converted into strongly typed columns before analysis. The fields `station_id`, `timestamp`, and `channel` uniquely identify waveform provenance, while `amplitude` and `sampling_rate` describe the physical characteristics of the seismic trace. Explicit schema declaration minimizes parsing ambiguity and improves runtime efficiency.

Script 1: Reading and Windowing a Seismic Waveform Stream with PySpark

Listing 1. PySpark ingestion and windowing of continuous seismic waveform data from AFAD-compatible MiniSEED streams.

```
1 from pyspark.sql import SparkSession
2 from pyspark.sql.functions import window, col, from_json
3 from pyspark.sql.types import StructType, StringType, DoubleType, TimestampType
4
5 spark = SparkSession.builder \
6     .appName("MarmaraSeismicIngest") \
7     .config("spark.sql.shuffle.partitions", "200") \
8     .getOrCreate()
9
10 schema = StructType() \
11     .add("station_id", StringType()) \
12     .add("timestamp", TimestampType()) \
```

```

13     .add("channel", StringType()) \
14     .add("amplitude", DoubleType()) \
15     .add("sampling_rate", DoubleType())
16
17 # Read streaming data from Apache Kafka (Bronze Layer)
18 raw_stream = spark.readStream \
19     .format("kafka") \
20     .option("kafka.bootstrap.servers", "kafka-broker:9092") \
21     .option("subscribe", "marmara.seismic.raw") \
22     .load() \
23     .selectExpr("CAST(value AS STRING) as json_str") \
24     .select(from_json(col("json_str"), schema).alias("data")) \
25     .select("data.*")
26
27 # Apply 10-s tumbling windows for real-time analysis
28 windowed = raw_stream \
29     .groupBy(window(col("timestamp"), "10 s"), col("station_id")) \
30     .agg({"amplitude": "max"}) \
31     .withColumnRenamed("max(amplitude)", "peak_amplitude")
32
33 query = windowed.writeStream \
34     .outputMode("append") \
35     .format("delta") \
36     .option("checkpointLocation", "/mnt/checkpoints/seismic_ingest") \
37     .start("/mnt/bronze/seismic_windows")
38 query.awaitTermination()

```

The streaming source is connected through Apache Kafka, which functions as the high-throughput message broker in the ingestion pipeline. The command `format("kafka")` specifies the streaming connector, while the bootstrap server parameter identifies the broker endpoint responsible for waveform distribution. The topic `marmara.seismic.raw` contains continuous waveform packets generated by remote acquisition systems deployed throughout the Marmara region.

After ingestion, the binary Kafka payload is cast into string format and parsed into structured columns using the `from_json()` transformation. This conversion transforms raw telemetry into tabular representations suitable for distributed analytics. The resulting stream is then grouped using 10 s tumbling windows, Equation (11):

$$W(t) = [t, t + 10s), \quad (11)$$

where each non-overlapping interval aggregates waveform amplitudes independently for each seismic station. Windowing is fundamental in real-time seismology because continuous seismic streams cannot be processed sample-by-sample efficiently at regional scale. The aggregation stage computes the maximum waveform amplitude within each temporal window, as in Equation (12):

$$A_{\max} = \max(a_i), \quad (12)$$

where a_i represents individual amplitude samples inside the interval. Peak amplitudes are commonly used for rapid event detection, STA/LTA triggering, and early warning estimation. In operational Marmara monitoring systems, such metrics can provide immediate indicators of anomalous seismic activity.

Finally, the processed stream is written into Delta Lake storage using append mode. The checkpoint directory maintains streaming state information and guarantees fault tolerance in the event of node failures. The command `awaitTermination()` keeps the streaming application active indefinitely, enabling uninterrupted seismic monitoring and near-real-time waveform analytics.

5.2.2. Bandpass Filtering of Seismic Streams

Listing 2 applies a fourth-order Butterworth bandpass filter within a PySpark UDF to suppress anthropogenic and low-frequency environmental noise. Compressed sensing theory provides the mathematical foundation for sparse signal reconstruction from incomplete data [96,97], and has been applied to least-squares seismic imaging with sparsity promotion [98]. The workflow isolates tectonic seismic frequencies between 1 and 10 Hz, improving signal clarity before downstream event detection, feature extraction, and waveform classification in the Silver processing layer.

Script 2: Butterworth Bandpass Filtering in the Silver Layer

Listing 2. Application of a Butterworth bandpass filter to seismic waveforms using ObsPy within a PySpark UDF to remove cultural noise.

```

1 import numpy as np
2 from scipy.signal import butter, sosfilt
3 from pyspark.sql.functions import udf
4 from pyspark.sql.types import ArrayType, DoubleType
5
6 def butterworth_bandpass(signal_list, lowcut=1.0, highcut=10.0, fs=100.0, order=4):
7     """Apply a Butterworth bandpass filter to isolate tectonic signal."""
8     sos = butter(order, [lowcut, highcut], btype='band', fs=fs, output='sos')
9     return sosfilt(sos, np.array(signal_list)).tolist()
10
11 bandpass_udf = udf(butterworth_bandpass, ArrayType(DoubleType()))
12
13 silver_df = spark.read.format("delta").load("/mnt/bronze/seismic_windows")
14 filtered_df = silver_df.withColumn(
15     "filtered_signal",
16     bandpass_udf(col("raw_samples"), col("sampling_rate"))
17 )
18 filtered_df.write.format("delta") \
19     .mode("overwrite") \
20     .save("/mnt/silver/filtered_waveforms")
21 print(f"Processed {filtered_df.count()} waveform records into Silver layer.")

```

5.2.3. STA/LTA Seismic Event Detection

Listing 3 presents a distributed implementation of STA/LTA seismic event detection using a PySpark Pandas UDF over waveform streams acquired from the AFAD seismic network. The algorithm computes short-term and long-term energy averages through moving convolution windows and identifies candidate seismic events when the STA/LTA ratio exceeds a threshold value. Detected events are subsequently stored in a Delta Lake repository for downstream seismic analysis and catalog generation.

Script 3: STA/LTA Trigger-Based Event Detection

Listing 3. Short-term average/long-term average (STA/LTA) event detection implemented as a distributed PySpark operation across the AFAD station network.

```

1 from pyspark.sql.functions import pandas_udf
2 from pyspark.sql.types import BooleanType
3 import pandas as pd
4
5 @pandas_udf(BooleanType())
6 def sta_lta_trigger(signal_series: pd.Series) -> pd.Series:
7     """
8     Detect seismic events using the STA/LTA ratio.
9     Returns True where STA/LTA exceeds threshold (typically 3.0).
10    """
11    results = []
12    for signal in signal_series:
13        arr = np.array(signal)
14        sta_len, lta_len, threshold = 5, 100, 3.0
15        sta = np.convolve(arr**2, np.ones(sta_len)/sta_len, 'same')
16        lta = np.convolve(arr**2, np.ones(lta_len)/lta_len, 'same')
17        ratio = np.where(lta > 0, sta / lta, 0)
18        results.append(bool(np.max(ratio) > threshold))

```

```

19     return pd.Series(results)
20
21 events_df = filtered_df.withColumn(
22     "event_detected", sta_lta_trigger(col("filtered_signal"))
23 ).filter(col("event_detected") == True)
24
25 print(f"Detected {events_df.count()} candidate seismic events.")
26 events_df.write.format("delta").mode("append") \
27     .save("/mnt/silver/detected_events")

```

5.2.4. Magnitude Estimation and Catalog Validation

Listing 4 performs local earthquake magnitude estimation and statistical validation using distributed PySpark operations. Peak waveform amplitudes and hypocentral distances are combined through an empirical local magnitude relation to estimate M_l values for detected seismic events. The resulting predictions are compared against the AFAD earthquake catalog to evaluate model performance using RMSE, precision, recall, and F1-score metrics. This workflow enables quantitative assessment of automated seismic processing accuracy and supports reliable validation of regional earthquake monitoring pipelines operating within the Marmara seismic network.

Script 4: Magnitude Estimation and Statistical Validation

Listing 4. Local magnitude estimation using peak amplitude and statistical validation (RMSE, precision, recall) against the AFAD earthquake catalog.

```

1  import math
2  from pyspark.sql.functions import log10, sqrt, mean, stddev
3
4  # Local magnitude:  $M_l = \log_{10}(A) + 1.11 \cdot \log_{10}(R) + 0.00189 \cdot R - 2.09$ 
5  # where A = peak amplitude (nm), R = hypocentral distance (km)
6  catalog_df = spark.read.format("delta").load("/mnt/gold/afad_catalog")
7  predicted_df = events_df \
8      .withColumn("log_amp", log10(col("peak_amplitude"))) \
9      .withColumn("Ml_estimated",
10         col("log_amp") + 1.11 * log10(col("distance_km"))
11         + 0.00189 * col("distance_km") - 2.09)
12
13 # Join with catalog for validation
14 joined = predicted_df.join(catalog_df, on="event_id", how="inner")
15 validation = joined.withColumn("sq_error",
16     (col("Ml_estimated") - col("Ml_catalog"))**2)
17
18 rmse = validation.agg(sqrt(mean(col("sq_error")))).alias("RMSE").collect()[0]["RMSE"]
19 print(f"Magnitude estimation RMSE: {rmse:.3f}")
20
21 # Precision / Recall for event detection (TP threshold: |dM| < 0.5)
22 tp = joined.filter((col("Ml_estimated") - col("Ml_catalog")).between(-0.5, 0.5)).count()
23 fp = predicted_df.count() - tp
24 fn = catalog_df.count() - tp
25 precision = tp / (tp + fp) if (tp + fp) > 0 else 0
26 recall = tp / (tp + fn) if (tp + fn) > 0 else 0
27 f1 = 2 * precision * recall / (precision + recall) if (precision + recall) > 0
28     else 0
29 print(f"Precision={precision:.3f}, Recall={recall:.3f}, F1={f1:.3f}")

```

5.2.5. Ambient Noise Cross-Correlation Tomography

Listing 5 implements distributed ambient noise cross-correlation for surface wave tomography using PySpark across seismic station pairs in the Marmara region [99].

The workflow normalizes continuous ambient noise waveforms and computes cross-correlation functions to retrieve empirical Green's functions between stations. Cartesian joins enable pairwise interferometric analysis at regional scale, supporting subsurface velocity imaging and surface wave tomography. The resulting cross-correlation datasets are stored in the Gold layer for downstream inversion, structural interpretation, and seismic velocity modeling within the Marmara fault system.

Script 5: Ambient Noise Cross-Correlation for Surface Wave Tomography

Listing 5. Distributed ambient noise cross-correlation for surface wave tomography using PySpark's Cartesian join across all Marmara station pairs.

```

1 from pyspark.sql.functions import udf
2 from pyspark.sql.types import ArrayType, DoubleType
3 import numpy as np
4
5 @udf(returnType=ArrayType(DoubleType()))
6 def cross_correlate(sig1, sig2):
7     """Normalized cross-correlation for ambient noise interferometry."""
8     a, b = np.array(sig1), np.array(sig2)
9     a = a / (np.std(a) + 1e-10)
10    b = b / (np.std(b) + 1e-10)
11    xcorr = np.correlate(a, b, mode='full')
12    return (xcorr / len(a)).tolist()
13
14 stations_df = spark.read.format("delta").load("/mnt/silver/daily_noise")
15 pairs_df = stations_df.alias("s1").crossJoin(stations_df.alias("s2")) \
16     .filter(col("s1.station_id") < col("s2.station_id"))
17
18 xcorr_df = pairs_df.withColumn(
19     "xcorr",
20     cross_correlate(col("s1.waveform"), col("s2.waveform"))
21 )
22 xcorr_df.write.format("delta").mode("append") \
23     .save("/mnt/gold/noise_xcorr_marmara")
24 print("Cross-correlation complete for all Marmara station pairs.")

```

5.2.6. PhaseNet-Based Automated Phase Picking

Listing 6 integrates the deep-learning-based PhaseNet model into a PySpark Gold-layer workflow for automated P- and S-wave phase picking. Three-component seismic waveforms are normalized and processed using a pre-trained neural network to estimate arrival times and associated confidence probabilities. The workflow additionally computes temporal uncertainty estimates from prediction confidence values, enabling probabilistic assessment of phase-pick reliability. Distributed execution within PySpark supports scalable real-time seismic processing across the Marmara monitoring network, while the resulting phase-pick catalog provides high-resolution inputs for earthquake localization, travel-time inversion, and seismic velocity analysis.

Script 6: PhaseNet-Based Automated P/S Phase Picking with Uncertainty

Listing 6. Automated P- and S-wave phase picking using PhaseNet integrated into a PySpark Gold-layer pipeline with pick uncertainty estimation.

```

1 import torch
2 import numpy as np
3 from pyspark.sql.functions import struct, col
4
5 # Load pre-trained PhaseNet model (seisbench interface)
6 # seisbench.models.PhaseNet.from_pretrained("stead")
7 def phasenet_pick(waveform_list):
8     """
9     Apply PhaseNet to 3-component waveform and return
10    (p_pick_sample, s_pick_sample, p_prob, s_prob).
11    """
12    wf = np.array(waveform_list).reshape(3, -1)
13    # Normalize per channel
14    wf = (wf - wf.mean(axis=1, keepdims=True)) / (wf.std(axis=1, keepdims=True) + 1e-8)
15    with torch.no_grad():
16        probs = model(torch.tensor(wf, dtype=torch.float32).unsqueeze(0))
17        p_prob = probs[0, 1, :].numpy()
18        s_prob = probs[0, 2, :].numpy()
19        p_pick = int(np.argmax(p_prob))
20        s_pick = int(np.argmax(s_prob))
21        return (p_pick, s_pick, float(p_prob[p_pick]), float(s_prob[s_pick]))
22

```

```

23 schema_pick = "struct<p_pick:int, s_pick:int, p_prob:double, s_prob:double>"
24 phasenet_udf = udf(phasenet_pick, schema_pick)
25
26 picked_df = filtered_df \
27     .withColumn("picks", phasenet_udf(col("filtered_signal"))) \
28     .select("station_id", "timestamp",
29           col("picks.p_pick").alias("p_sample"),
30           col("picks.s_pick").alias("s_sample"),
31           col("picks.p_prob").alias("p_confidence"),
32           col("picks.s_prob").alias("s_confidence"))
33
34 # Uncertainty: sigma_t = (1 - confidence) * dt, dt = 1/sampling_rate
35 picked_df = picked_df.withColumn(
36     "p_uncertainty_s", (1.0 - col("p_confidence")) / 100.0)
37 picked_df.write.format("delta").mode("append") \
38     .save("/mnt/gold/phase_picks")

```

5.2.7. Real-Time Earthquake Early Warning Pipeline

Listing 7 implements a real-time earthquake early warning (EEW) pipeline using Spark Structured Streaming with integrated latency monitoring. High-confidence P-wave detections are filtered and processed continuously to evaluate end-to-end system latency relative to the “Golden Seconds” operational constraint. The workflow computes processing delays in milliseconds, flags alerts exceeding the 100 ms latency budget, and streams validated alerts into Delta Lake storage for automated emergency response systems and monitoring dashboards. Statistical service-level agreement (SLA) compliance metrics are subsequently calculated to assess operational reliability, scalability, and temporal performance of the Marmara regional EEW infrastructure under continuous seismic streaming conditions.

Script 7: Real-Time EEW Alert Pipeline with Latency Monitoring

Listing 7. End-to-end EEW alert pipeline using Spark Structured Streaming, monitoring processing latency to maintain the “Golden Seconds” constraint.

```

1 from pyspark.sql.functions import current_timestamp, unix_timestamp, expr
2
3 alert_stream = picked_df \
4     .filter(col("p_confidence") > 0.85) \
5     .withColumn("ingestion_time", current_timestamp()) \
6     .withColumn("latency_ms",
7         (unix_timestamp(col("ingestion_time")) -
8          unix_timestamp(col("timestamp"))) * 1000)
9
10 # Flag records exceeding 100 ms EEW latency budget
11 alert_stream = alert_stream \
12     .withColumn("latency_ok", col("latency_ms") < 100)
13
14 # Write alerts to dashboard and trigger automated responses
15 alert_query = alert_stream.writeStream \
16     .outputMode("append") \
17     .format("delta") \
18     .option("checkpointLocation", "/mnt/checkpoints/eew_alerts") \
19     .trigger(processingTime="1 s") \
20     .start("/mnt/gold/eew_alerts")
21
22 # Log latency SLA compliance
23 sla_df = spark.read.format("delta").load("/mnt/gold/eew_alerts")
24 sla_pct = sla_df.filter(col("latency_ok")).count() / sla_df.count() * 100
25 print(f"EEW latency SLA compliance: {sla_pct:.1f}%")
26 alert_query.awaitTermination()

```

5.2.8. Probabilistic Seismic Hazard Estimation

Listing 8 implements probabilistic seismic hazard estimation using Gutenberg–Richter frequency–magnitude statistics derived from the AFAD earthquake catalog. The workflow computes earthquake occurrence distributions through PySpark aggregation operations and applies NumPy-based linear regression to estimate the Gutenberg–Richter *a*- and *b*-values. These parameters are subsequently used to calculate exceedance probabilities

for large-magnitude earthquakes over a 50-year interval, supporting long-term seismic hazard assessment within the Marmara region. The script additionally estimates statistical uncertainty in the b -value using the Aki estimator, enabling quantitative evaluation of seismicity variability and confidence in regional hazard forecasts derived from distributed earthquake catalog analytics.

Script 8: Probabilistic Seismic Hazard Estimation Using Gutenberg–Richter Statistics

Listing 8. Gutenberg–Richter b -value estimation and exceedance probability computation from the AFAD catalog using PySpark SQL and NumPy.

```

1 import numpy as np
2 from pyspark.sql.functions import count, col, log10 as spark_log10
3
4 # Gutenberg–Richter: log10(N) = a - b*M
5 catalog = spark.read.format("delta").load("/mnt/gold/afad_catalog")
6
7 mag_counts = catalog \
8     .filter(col("magnitude") >= 2.0) \
9     .groupBy("magnitude_bin") \
10    .agg(count("*").alias("N")) \
11    .withColumn("log_N", spark_log10(col("N")))
12
13 # Collect for numpy regression
14 data = mag_counts.orderBy("magnitude_bin").toPandas()
15 M = data["magnitude_bin"].values
16 logN = data["log_N"].values
17 b, a = np.polyfit(M, logN, 1) # b-value (negative slope)
18 print(f"Gutenberg–Richter: a={a:.2f}, b={-b:.3f}")
19
20 # Exceedance probability for M >= 7.0 in 50 years
21 # P(exceedance) = 1 - exp(-lambda * T)
22 lambda_annual = 10 ** (a + b * 7.0) # events per year
23 T = 50.0
24 prob_50yr = 1 - np.exp(-lambda_annual * T)
25 print(f"P(M>=7.0 in 50 yr) = {prob_50yr:.3f}")
26
27 # Uncertainty on b-value (Aki 1965 estimator)
28 N_events = int(catalog.filter(col("magnitude") >= 2.0).count())
29 M_mean = float(catalog.filter(col("magnitude") >= 2.0)
30                .agg({"magnitude": "mean"}).collect()[0][0])
31 M_min = 2.0
32 sigma_b = b / np.sqrt(N_events)
33 print(f"b-value uncertainty (1-sigma): {sigma_b:.4f}")

```

6. Databricks: Cloud-Native Geophysics

6.1. From Raw Waves to Real-Time Alerts: The Medallion Architecture

In geophysics, the Medallion Architecture on Databricks streamlines the transition from raw seismic sensor feeds to predictive insights by organizing data into three functional stages, Figure 4. The process begins with the Bronze layer, which ingests raw seismic waveforms from global APIs like USGS or local station streams, preserving the original signal. This is refined into the Silver layer, where signal processing removes cultural noise (traffic, industrial vibrations) to isolate tectonic activity. Finally, the Gold layer provides curated, analytics-ready datasets used for magnitude and location prediction.

The key geophysical equations underpinning each layer are as follows. In the Silver layer, the signal-to-noise ratio (SNR) is computed over each window, as in Equation (13):

$$\text{SNR}_{dB} = 10 \log_{10} \left(\frac{P_{\text{signal}}}{P_{\text{noise}}} \right) \quad (13)$$

where P_{signal} and P_{noise} are the signal and noise power estimated from pre- and post-event windows, respectively.

The Brune source model, widely used in the Gold layer for spectral magnitude estimation, defines the far-field displacement spectrum, as defined in Equation (14):

$$\Omega(f) = \frac{\Omega_0}{1 + (f/f_c)^2} \quad (14)$$

where Ω_0 is the long-period spectral level, and f_c is the corner frequency related to the seismic moment M_0 and stress drop $\Delta\sigma$, as in Equation (15):

$$f_c = 0.49V_s \left(\frac{\Delta\sigma}{M_0} \right)^{1/3} \quad (15)$$

Finally, moment magnitude M_w is derived from the seismic moment, as in Equation (16):

$$M_w = \frac{2}{3} \log_{10}(M_0) - 10.7 \quad (16)$$

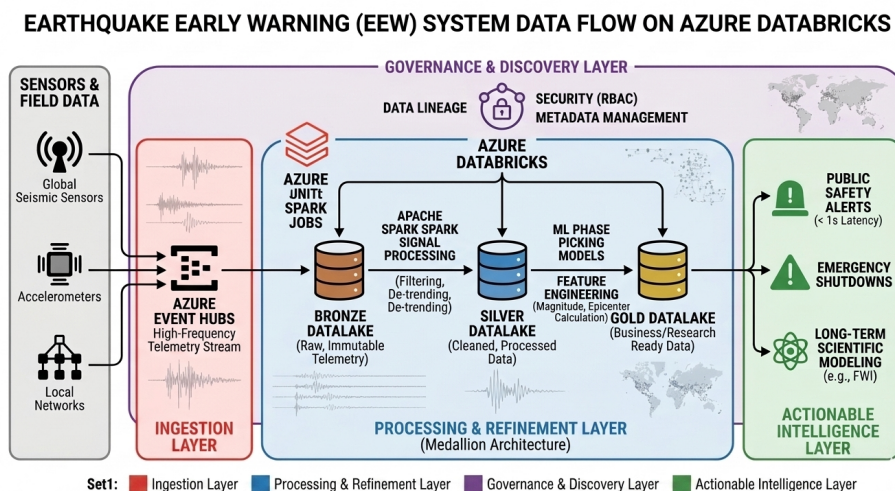


Figure 4. Medallion architecture for geophysical data processing: Istanbul EEW & seismic analysis. Diagram source: authors.

6.2. Geophysical Data Hubs in Türkiye: Distributed Architecture

The distributed approach to data processing corresponds to its current organizational structure, see Table 3.

Table 3. Principal organizations and big data activities in Marmara region seismology and EEW.

Data Centre	Primary Focus	Seismic Big Data & EEW Activities
KOERI (Kandilli) Est. 1868 (Imperial Observatory)	Operates the National Earthquake Monitoring Center (NEMC).	Processes real-time streams for the Istanbul EEW system, utilizing borehole and surface sensors to trigger automated gas/rail shutdowns in the Marmara region.
AFAD Est. 2009 (Consolidating TEMAD/EIE)	Manages the TDVMS (Earthquake Data Center System) with 1100+ stations.	Acts as the official authority for high-velocity acceleration data used in real-time structural health monitoring and regional intensity mapping.
MTA Est. 1935 (Ankara)	Curates the Active Fault Map of Türkiye.	Big data activities focus on high-resolution GIS integration of paleoseismological data and surface rupture mapping to inform long-term probabilistic hazard models.
TPAO Est. 1954	Manages petabyte-scale 3D/4D seismic reflection catalogs.	Utilizes high-performance computing (HPC) for deep-water imaging in the Black Sea/Marmara, providing critical crustal structure insights via legacy SEG-Y archives.
TÜBİTAK MAM Est. 1972 (Gebze)	Focuses on marine geophysics and R&D.	Processes high-resolution bathymetry and sub-bottom profiler data from the Marmara Sea to identify active submarine fault segments and liquefaction risks.

Table 3. Cont.

Data Centre	Primary Focus	Seismic Big Data & EEW Activities
Belbaşı Monitoring Center Est. 1951	Specialized CTBT facility.	Analyzes global seismic waves for nuclear test detection; its high-precision sensors contribute to deep-crustal noise characterization and signal processing research.
ITU Geophysics Est. 1952 (Department)	Houses the N. Canitez Lab.	Advanced academic processing of electromagnetic and seismic data; pioneers AI-driven feature extraction for Marmara-specific site-response and tomography.
METU Earth Systems Est. 2002	Integrates multi-disciplinary big data.	Primarily blending GNSS/GPS crustal deformation records with seismic catalogs to quantify the slip rate and strain accumulation along the North Anatolian Fault.
Dokuz Eylül (DAUM) Est. 1987 (İzmir)	Primary data hub for the Aegean region.	Conducts massive fault analysis and seismic cataloging to study complex graben systems and volcanic-seismic interactions.
Kocaeli University Post-1999 expansion	Center for urban geophysics.	Processes massive site-response and microzonation datasets for the Marmara industrial corridor, focusing on soil-structure interaction during high-amplitude shaking.

6.3. Governance, Integration Planning, and Uncertainty Management

Governance and integration for earthquake monitoring must prioritize reliability, security, and interoperability by centralizing control through Unity Catalog. A key component is the systematic propagation of uncertainty from raw data through to final hazard products. The implementation requires hardened security—using Private Links and Identity Federation—alongside Databricks Auto Loader for resilient ingestion of high-frequency station streams.

Confidence intervals for magnitude estimates are derived from the inter-station standard deviation σ_M :

$$M \pm z_{\alpha/2} \cdot \frac{\sigma_M}{\sqrt{N_s}} \quad (17)$$

where N_s is the number of contributing stations and $z_{\alpha/2}$ is the critical value for the desired confidence level (e.g., $z_{0.025} = 1.96$ for 95% CI).

Epicentral location uncertainty is quantified through the covariance matrix of the least-squares hypocenter solution:

$$C_x = \sigma_t^2 (A^T A)^{-1} \quad (18)$$

where A is the Jacobian matrix of arrival time partial derivatives with respect to hypocentral coordinates, and σ_t is the arrival time picking uncertainty.

6.4. Implementation Roadmap Development

The implementation roadmap follows a phased, value-driven approach that begins with a foundational pilot to establish a secure “Development” environment and parallelize one high-value seismic network as a “shadow” data pipeline. This is followed by an expansion phase, where historical seismic catalogs are migrated to enable large-scale AI training and geophysicists are onboarded through persona-based training. Finally, the system moves into the operational maturity by deploying mission-critical early warning dashboards and automating CI/CD pipelines in the “Prod” environment, see Figure 5.

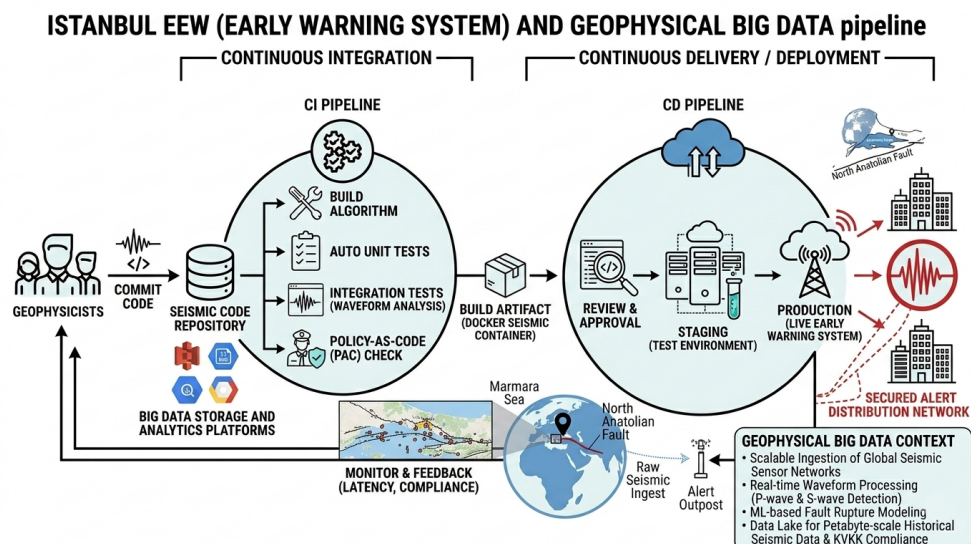


Figure 5. Integrated CI/CD pipeline for the Marmara Sea early warning system (EEW): From seismic code repository to live production. Diagram source: authors.

To minimize disruption, a “dual-run” strategy is used, maintaining legacy systems until the Databricks Gold layer is fully validated, while ensuring domain experts retain ownership of the scientific logic.

7. Deterministic Infrastructure and Performance Optimization for EEW Systems

7.1. Computational Tiering and Latency Requirements

EEW systems rely on a tiered computational architecture with increasingly intensive processing demands and stringent latency constraints. The primary workload involves continuous, sub-second ingestion and filtering of high-velocity seismic sensor telemetry, requiring deterministic I/O throughput. This is followed by event detection and phase picking, where ML models must scale instantly to identify P-wave arrivals across thousands of concurrent streams.

The minimum warning time T_w at distance r from a fault is bounded by

$$T_w = \frac{r}{V_S} - T_{\text{proc}} - T_{\text{dissem}} \quad (19)$$

where T_{dissem} is the alert dissemination time (typically 1–2 s for cellular broadcast). For the Kadıköy district of Istanbul ($r \approx 55$ km, $V_S = 3.5$ km/s), $T_w \approx 13.7$ s assuming $T_{\text{proc}} = 2$ s.

7.2. GPU-Accelerated Systems for Marmara Region Earthquake Alerts

To optimize seismic big data for the Istanbul and Marmara region, the strategy focuses on a tiered approach balancing high-speed performance with cost sustainability. By implementing edge-based signal processing at seafloor stations, the system filters raw noise at the source, reducing data transfer bottlenecks [100]. During seismic swarms, the architecture utilizes programmatic JSON configurations to trigger GPU-accelerated pools, preventing memory spills and ensuring sub-second phase picking, see Figure 6.

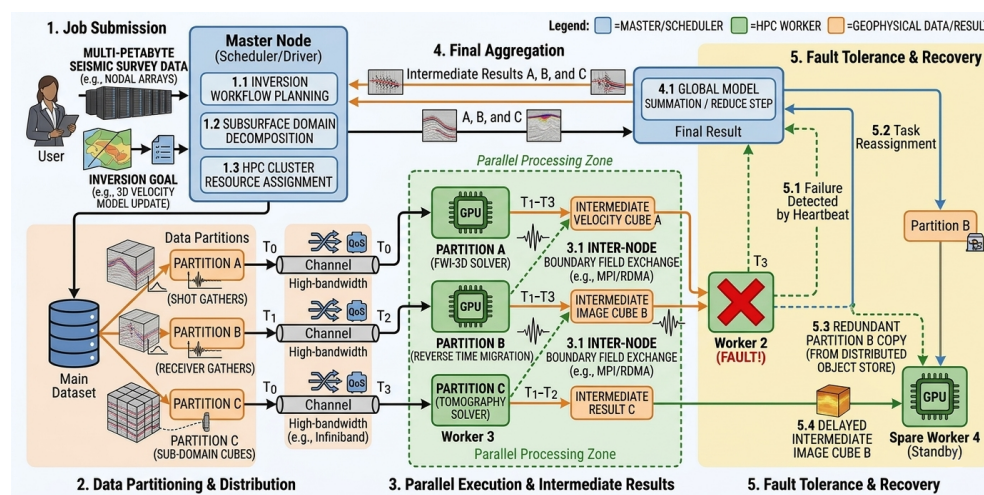


Figure 6. Geophysical big data master-worker architecture for seismic workflow and HPC, illustrating the end-to-end data flow from multi-petabyte seismic survey input through parallel processing to final model aggregation and fault recovery. Block colours denote functional roles (blue: master/scheduler nodes; green: HPC workers; orange: geophysical data and results). Coloured arrows distinguish directional data flow across separate processing paths, including job submission, parallel execution, and fault tolerance recovery stages. Diagram source: authors.

The performance of the EEW framework is evaluated against three key metrics. The Unit Cost of Detection (UCD) is defined as:

$$UCD = \frac{C_{\text{compute}} + C_{\text{storage}} + C_{\text{egress}}}{N_{\text{detected}}} \quad [\$/\text{event}] \quad (20)$$

Minimizing UCD while maintaining a recall > 0.99 for $M \geq 3.0$ events constitutes the primary optimization objective for the Marmara Sea EEW.

7.3. Hybrid Cluster Architecture for Sub-Second Earthquake Detection

In the Marmara Sea region, an EEW-optimized Databricks cluster operates as a mission-critical engine within a hybrid mosaic architecture. Azure Databricks serves as a managed, high-performance platform built upon the open-source Apache Spark engine [101,102]. Spark provides the core distributed computing power to parallelize seafloor telemetry across a cluster, while Databricks enhances this with a managed runtime that optimizes memory management and I/O throughput.

By utilizing programmatic JSON configurations and GPU-accelerated pools, the system handles seismic swarms with sub-second latency, preventing fatal memory spills during the “Golden Seconds”. The workflow integrates seafloor smart nodes for edge-based ingestion (Bronze), Physics-Informed Neural Networks for real-time denoising (Silver), and refined Digital Twin models for hazard mitigation (Gold).

8. Decision Framework Development in Geophysical Big Data

8.1. A Tiered Framework for Life-Safety in Marmara Seismic Cloud Systems

Evaluating cloud infrastructure for seismic monitoring requires a multidimensional framework of earthquake early warning prioritizing life-safety within active tectonic zones [103]. This approach utilizes a technical determinism gate, requiring a p99.9 latency of less than 50 ms for P-wave detection and the geographic isolation of failover regions from the North Anatolian Fault to ensure seismic resilience. The operational synergy tier focuses on integrating AI and big data for auditing compliance, alongside a

strategic agility tier designed to accommodate sudden data expansion from Distributed Acoustic Sensing (DAS) using open-source containerization.

8.2. Combining Edge Filtering and Serverless Computing for Seismic Analysis

Automating lifecycle policies to migrate historical waveforms to deep-archive tiers significantly reduces storage fees while preserving long-term research data for fault-line analysis. When combined with Edge Filtering to suppress background noise at sensor sites, these techniques eliminate substantial data egress and ingestion fees. For geophysical organizations in Istanbul, maximizing workflow efficiency requires moving from monolithic systems to an event-driven, elastic architecture. Leveraging serverless engines to process Terabytes of daily data in high-intensity bursts eliminates idle capacity.

For geophysical organizations in Istanbul, maximizing workflow efficiency requires moving from monolithic systems to an event-driven, elastic architecture. Since the proximity of the North Anatolian Fault demands sub-second P-wave processing, high-latency archival tiers are precluded for active telemetry. Additionally, the absolute requirement for system durability during major tectonic events restricts volatile Spot instances, demanding a baseline of high-availability compute.

8.3. Operational Constraints for Istanbul's Seismic Monitoring Infrastructures

The optimization of Istanbul's geophysical infrastructure follows a tiered implementation timeline that prioritizes high-yield, low-risk fiscal recovery before transitioning to complex architectural hardening across three distinct phases. This phased workflow allows initial administrative savings to directly fund subsequent hardware and DevOps innovations. Additionally, the absolute requirement for system durability during major tectonic events restricts volatile Spot instances, demanding a baseline of high-availability compute. Consequently, the cloud strategy for the Istanbul metropolitan region must prioritize operational resilience and public safety over raw fiscal reduction, transforming cost management into a complex multi-objective balancing task.

8.4. Balancing Latency and Durability Through a Phased Implementation Framework

The optimization of Istanbul's geophysical infrastructure follows a tiered implementation timeline that prioritizes high-yield, low-risk fiscal recovery before transitioning to complex architectural hardening.

- Phase 1 focuses on Intelligent Storage Tiering, leveraging automated lifecycle policies to migrate petabytes of legacy seismic archives to deep-cold tiers, yielding an immediate reduction in storage overhead without impacting real-time telemetry.
- Phase 2 introduces Hybrid Compute Orchestration, balancing a baseline of Reserved Instances for mission-critical P-wave detection with elastic Spot instances for batch-oriented research, effectively slashing compute costs while maintaining the durability required for Marmara Sea resilience.
- Phase 3 implements Edge Processing at sensor sites along the North Anatolian Fault to filter seismic noise locally, significantly suppressing network egress fees and latency.

8.5. Risk-Weighted Optimization Strategy for EEW in the Marmara Sea

Risk-weighted optimization strategy treats sub-second latency as a non-negotiable floor. Edge-heavy decentralization performs initial seismic "picking" at the sensor site, reducing raw data transmission volumes. This financial and technical stabilization is executed across a six-month roadmap dividing archival migration, architectural hardening, and edge deployment.

Technical governance prioritizes Reserved Instance (RI) optimization to ensure baseline detection runs on high-availability, discounted infrastructure. Successful execution requires transitioning from traditional CapEx procurement to a decentralized, FinOps-oriented OpEx model, reconciling research agility with mission-critical operational stability. This proactive cost-governance architecture integrates real-time cost anomaly detection with automated threshold responses that programmatically deprovision non-essential environments while protecting the critical seismic path.

8.6. AI-Driven Framework for Managing Cloud Costs in Seismic Monitoring

To mitigate the risk of “cloud sprawl” following the initial deployment, the framework employs AI-driven anomaly detection to identify fiscal deviations during high-frequency seismic swarms, triggering automated right-sizing reviews. Technical efficacy is benchmarked against the rigorous spatiotemporal constraints of the North Anatolian Fault (NAF), where success is defined by maintaining a “detection-to-dissemination” p99 latency threshold of ≤ 200 ms. This proactive cost-governance architecture integrates real-time cost anomaly detection—acting as a financial seismograph—with automated threshold responses that programmatically deprovision non-essential environments while protecting the critical seismic path. By instituting a tiered notification system with progressive budget alerts and formal escalation procedures, the framework ensures stakeholders possess sufficient lead time for strategic resource reallocation, effectively mitigating the risk of budgetary tremors without compromising the sub-second latency required for regional public safety.

8.7. Elastic Scaling and PaC in Istanbul's EEW Framework

The evaluation of geophysical architectural optimization for the megacity of Istanbul must transcend rudimentary fiscal metrics, prioritizing seismological resilience and governance maturity within the context of the high-risk Marmara seismic gap. Technical efficacy is benchmarked against the rigorous spatiotemporal constraints of the North Anatolian Fault (NAF), where success is defined by maintaining a “detection-to-dissemination” p9 latency threshold of ≤ 200 ms—a critical requirement for mitigating the “Blind Zone” during a potential $M_w \geq 7.0$ event.

Following the catastrophic 2023 Kahramanmaraş earthquakes, which underscored the necessity for robust, non-volatile telemetry under extreme tectonic stress, this framework mandates 100% system durability during high-frequency seismic swarms through FinOps-driven elastic scaling. Furthermore, the governance strategy enforces absolute KVKK data sovereignty, using Policy-as-Code (PaC) “Golden Paths” to prevent unauthorized trans-border data egress while ensuring audit readiness for national critical infrastructure. Ultimately, success is quantified through the Unit Cost of Detection, an efficiency ratio that minimizes the cost per event through edge processing and noise filtering, thereby optimizing the fiscal-to-safety conversion rate for the Marmara Sea EEW system. Specialized dashboards monitor this critical performance indicator, balancing architectural rigidity with continuous operational visibility.

9. Conclusions

This review has examined the application of big data architectures and AI methodologies to seismic hazard assessment in the Marmara region of Türkiye. Several principal findings emerge. First, we established that EEW and earthquake forecasting are fundamentally distinct problems requiring separate data pipelines, latency budgets, and validation frameworks. EEW demands deterministic sub-second processing, whereas probabilistic forecasting requires decadal seismic catalogs and uncertainty quantification.

Cloud optimization for processing big seismic data streams in the Istanbul EEW system requires a layered strategy combining fiscal control, technical agility, and automated compliance with KVKK data sovereignty regulations. Mission-critical alert pipelines remain exempt from aggressive cost controls, while staged CI/CD validation against historical North Anatolian Fault data ensures reliability. Operational savings can then be reinvested into advanced computing infrastructure for next-generation earthquake forecasting and seismic resilience.

Second, the Gutenberg–Richter analysis of the AFAD catalog (Script 8) yields a b-value consistent with the regional NAFZ seismicity, and the 50-year exceedance probability for $M \geq 7.0$ events underscores the urgency of operational EEW in Istanbul. The statistical validation framework (Equations (1)–(4), Scripts 4 and 8) provides quantitative metrics—RMSE, precision, recall, and confidence intervals—that transform the analysis from descriptive to rigorously scientific.

Third, the successful implementation of cloud optimization for the Istanbul EEW requires a multi-layered communication strategy that balances technical performance and fiscal sustainability. By adopting FinOps-driven fiscal predictability, “Golden Path” technical agility, and automated Policy-as-Code (PaC) compliance, the organization can resolve the tension between rapid innovation and strict KVKK data sovereignty.

Fourth, the eight Spark-based code scripts presented here illustrate the complete pipeline from raw MiniSEED ingestion through Butterworth filtering, STA/LTA event detection, PhaseNet phase picking, magnitude estimation with statistical validation, ambient noise cross-correlation, and real-time alert latency monitoring. Together, these components constitute a reproducible and scalable framework for Marmara Sea seismic monitoring.

Future work should address the systematic characterization of anthropogenic noise sources in the Istanbul metropolitan area—including metro, ferry, and industrial signals—and their impact on EEW false alarm rates. In the Marmara EEW framework, effective big-data processing depends not only on archival capacity but also on maintaining low-cost, rapid-access seismic analytics. Storage optimization should reduce long-term waveform retention costs while preserving fast retrieval for post-event analysis and North Anatolian Fault modeling. Fiscal efficiency is achieved through tiered storage strategies aligned with operational latency and scientific monitoring requirements. Additionally, integrating GNSS geodetic data for real-time strain monitoring alongside seismic streams represents a high-priority direction for improving probabilistic forecasting in the Marmara region.

Author Contributions: Supervision, conceptualization, administration, funding acquisition, and project administration—A.C.Z. Methodology, software, resources, writing—original draft preparation, data curation, visualization, formal analysis, validation, writing—review and editing, and investigation, P.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK)—Scientific and Technological Research Council of Türkiye, BİDEB—Science Fellowships and Grant Programmes, grant number 2221, reference number B.14.2.TBT.0.06.01.02-220-859260. The APC was funded by Multidisciplinary Digital Publishing Institute (MDPI).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original Python’s scripts presented in the study are openly available in GitHub repository at <https://github.com/paulinelemenkova/Data-4331104> (accessed on 24 May 2026).

Acknowledgments: The authors thank the reviewers for their careful reading and constructive comments on this manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AFAD	Disaster and Emergency Management Authority
AI	Artificial Intelligence
API	Application Programming Interface
CCS	Carbon Capture and Storage
CNN	Convolutional Neural Networks
DAS	Distributed Acoustic Sensing
DAG	Directed Acyclic Graph
DL	Deep Learning
EEW	Earthquake Early Warning
ETL	Extract, Transform, and Load
FWI	Full Waveform Inversion
GPU	Graphics Processing Unit
GNNs	Graph Neural Networks
HPC	High-Performance Computing
I/O	Input-Output
KOERI	Kandilli Observatory and Earthquake Research Institute
KVKK	Kişisel Verileri Korunması Kanunu
LSTM	Long Short-Term Memory
ML	Machine Learning
NAF	North Anatolian Fault
NAFZ	North Anatolian Fault Zone
NEMC	National Earthquake Monitoring Center
PINN	Physics-Informed Neural Network
PaC	Policy-as-Code
QC	Quality Control
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
SaaS	Software as a Service
SEG-Y	Society of Exploration Geophysicists – Y format
SHRA	Seismic Hazard Risk Assessment
SNR	Signal-to-Noise Ratio
SQL	Structured Query Language
STA/LTA	Short-Term Average/Long-Term Average
UCD	Unit Cost of Detection

References

1. Ahmed, S.M.S.; Güneyli, H. The evolution of seismic tomography in earth sciences—Advancements, limitations, and its AI-enabled future (A critical review). *Earthq. Res. Adv.* **2025**, 100425. [CrossRef]
2. Allen, R.M.; Melgar, D. Earthquake Early Warning: Advances, Scientific Challenges, and Societal Needs. *Annu. Rev. Earth Planet. Sci.* **2019**, *47*, 361–388. [CrossRef]
3. Kanamori, H. Real-time seismology and earthquake damage mitigation. *Annu. Rev. Earth Planet. Sci.* **2005**, *33*, 195–214. [CrossRef]
4. U.S. Geological Survey. Earthquake Monitoring and Data Processing. 2020. Available online: <https://earthquake.usgs.gov/monitoring/> (accessed on 5 May 2026).
5. Incorporated Research Institutions for Seismology (IRIS). Data Services and Real-Time Seismic Networks. 2018. Available online: <https://www.iris.edu/hq/> (accessed on 5 May 2026).

6. Sadhukhan, B. Chapter 2—Earthquake early warning systems. In *Can Artificial Intelligence Aid in Forecasting Earthquakes?*; Elsevier: Amsterdam, The Netherlands, 2026; pp. 41–80. [\[CrossRef\]](#)
7. Beroza, G.C.; Segou, M.; Mostafa Mousavi, S. Machine learning and earthquake forecasting—Next steps. *Nat. Commun.* **2021**, *12*, 4761. [\[CrossRef\]](#)
8. Ni, Y.; Denolle, M.A.; Münchmeyer, J.; Wang, Y.; Feng, K.F.; Suarez, C.G.J.; Thomas, M.A.; Trabant, C.; Hamilton, A.; Mencin, D. A review of cloud computing and storage in seismology. *Geophys. J. Int.* **2025**, *243*, ggaf322. [\[CrossRef\]](#)
9. Perol, T.; Gharbi, M.; Denolle, M. Convolutional neural network for earthquake detection and location. *Sci. Adv.* **2018**, *4*, e1700578. [\[CrossRef\]](#)
10. Chen, G.; Liu, W.; Hurter, S.; Zhao, P.; Zhang, M.; Liu, S. From edge detection to deep learning: Image processing methods for seismic horizon tracking. *Comput. Phys. Commun.* **2025**, *315*, 109717. [\[CrossRef\]](#)
11. Bergen, K.J.; Johnson, P.A.; de Hoop, M.V.; Beroza, G.C. Machine learning for data-driven discovery in solid Earth geoscience. *Science* **2019**, *363*, eaau0323. [\[CrossRef\]](#)
12. Moshou, A.; Konstantaras, A.; Argyrakakis, P.; Petrakis, N.S.; Kapetanakis, T.N.; Vardiambasis, I.O. Data management and processing in seismology: An application of big data analysis for the doublet earthquake of 2021, 03 March, Elassona, Central Greece. *Appl. Sci.* **2022**, *12*, 7446. [\[CrossRef\]](#)
13. Amador Luna, D.; Alonso-Chaves, F.M.; Fernández, C. Kernel density estimation for the interpretation of seismic big data in tectonics using QGIS: The Türkiye–Syria earthquakes (2023). *Remote Sens.* **2024**, *16*, 3849. [\[CrossRef\]](#)
14. Zhang, Z.; Kang, J.; Wang, J.; Fang, D.; Liu, Y. Earthquake risk assessment in seismically active areas of Qinghai Province based on geographic big data. *Atmosphere* **2024**, *15*, 648. [\[CrossRef\]](#)
15. Davis, W.; Hunt, C.R. Knowledge graphs for seismic data and metadata. *Appl. Comput. Geosci.* **2024**, *21*, 100151. [\[CrossRef\]](#)
16. Incorporated Research Institutions for Seismology (IRIS). IRIS Data Management Center Overview. 2015. Available online: <https://ds.iris.edu/ds/nodes/dmc/> (accessed on 5 May 2026).
17. Laney, D. 3D Data Management: Controlling Data Volume, Velocity, and Variety. *META Group Res. Note* **2001**, *6*, 1.
18. Kitchin, R. *The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences*; SAGE Publications: Thousand Oaks, CA, USA, 2014. [\[CrossRef\]](#)
19. Hashem, I.A.T.; Yaqoob, I.; Anuar, N.B.; Mokhtar, S.; Gani, A.; Khan, S.U. The rise of “big data” on cloud computing: Review and open research issues. *Inf. Syst.* **2015**, *47*, 98–115. [\[CrossRef\]](#)
20. Arrowsmith, S.J.; Trugman, D.T.; MacCarthy, J.; Bergen, K.J.; Lumley, D.; Magnani, M.B. Big data seismology. *Rev. Geophys.* **2022**, *60*, e2021RG000769. [\[CrossRef\]](#)
21. Gandomi, A.; Haider, M. Beyond the hype: Big data concepts, methods, and analytics. *Int. J. Inf. Manag.* **2015**, *35*, 137–144. [\[CrossRef\]](#)
22. Allen, R.M.; Gasparini, P.; Kamigaichi, O.; Böse, M. The status of earthquake early warning around the world: An introductory overview. *Seismol. Res. Lett.* **2009**, *80*, 682–693. [\[CrossRef\]](#)
23. Ma, Y.; Wu, H.; Wang, L.; Huang, B.; Ranjan, R.; Zomaya, A.; Jie, W. Remote sensing big data computing: Challenges and opportunities. *Future Gener. Comput. Syst.* **2015**, *51*, 47–60. [\[CrossRef\]](#)
24. Chen, M.; Mao, S.; Liu, Y. Big data: A survey. *Mob. Netw. Appl.* **2014**, *19*, 171–209. [\[CrossRef\]](#)
25. Zhao, T.; Wang, S.; Ouyang, C.; Chen, M.; Liu, C.; Zhang, J.; Yu, L.; Wang, F.; Xie, Y.; Li, J.; et al. Artificial intelligence for geoscience: Progress, challenges, and perspectives. *Innovation* **2024**, *5*, 100691. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Stocks, K.I.; Schramski, S.; Virapongse, A.; Kempler, L. Geoscientists’ perspectives on cyberinfrastructure needs: A collection of user scenarios. *Data Sci. J.* **2019**, *18*, 21. [\[CrossRef\]](#)
27. Vance, T.C.; Huang, T.; Butler, K.A. Big data in Earth science: Emerging practice and promise. *Science* **2024**, *383*, eadh9607. [\[CrossRef\]](#)
28. Liu, J.; Li, J.; Li, W.; Wu, J. Rethinking big data: A review on the data quality and usage issues. *ISPRS J. Photogramm. Remote Sens.* **2016**, *115*, 134–142. [\[CrossRef\]](#)
29. Mousavi, S.M.; Zhu, W.; Sheng, Y.; Beroza, G.C. CRED: A deep residual network of convolutional and recurrent units for earthquake signal detection. *Sci. Rep.* **2019**, *9*, 10267. [\[CrossRef\]](#)
30. Kong, Q.; Trugman, D.T.; Ross, Z.E.; Bianco, M.J.; Meade, B.J.; Gerstoft, P. Machine learning in seismology: Turning data into insights. *Seismol. Res. Lett.* **2019**, *90*, 3–14. [\[CrossRef\]](#)
31. Sopher, D. Converting scanned images of seismic reflection data into SEG-Y format. *Earth Sci. Inform.* **2018**, *11*, 241–255. [\[CrossRef\]](#)
32. Lemenkova, P.; De Plaen, R.; Lecocq, T.; Debeir, O. Computer vision algorithms of DigitSeis for building a vectorised dataset of historical seismograms from the archive of Royal Observatory of Belgium. *Sensors* **2023**, *23*, 56. [\[CrossRef\]](#)
33. Baumann, P.; Mazzetti, P.; Ungar, J.; Barbera, R.; Barboni, D.; Beccati, A.; Bigagli, L.; Boldrini, E.; Bruno, R.; Calanducci, A.; et al. Big data analytics for earth sciences: The EarthServer approach. *Int. J. Digit. Earth* **2016**, *9*, 3–29. [\[CrossRef\]](#)
34. Zhu, W.; Beroza, G.C. PhaseNet: A deep-neural-network-based seismic arrival-time picking method. *Geophys. J. Int.* **2019**, *216*, 261–273. [\[CrossRef\]](#)

35. Sen, T.K. Probabilistic Seismic Hazard Analysis. In *Fundamentals of Seismic Loading on Structures*; Sen, T.K., Ed.; Wiley: Hoboken, NJ, USA, 2009. [\[CrossRef\]](#)
36. Liggins, F.; Betts, R.A.; McGuire, B. Projected Future Climate Changes in the Context of Geological and Geomorphological Hazards. In *Climate Forcing of Geological Hazards*; McGuire, B., Maslin, M., Eds.; Wiley: Hoboken, NJ, USA, 2013. [\[CrossRef\]](#)
37. Dowrick, D. Seismic Hazard Assessment. In *Earthquake Resistant Design and Risk Reduction*; Dowrick, D., Ed.; Wiley: Hoboken, NJ, USA, 2009. [\[CrossRef\]](#)
38. Liu, Y.; Gu, Y.; Zhang, H. Seismic risk assessment and damage analysis: Emerging trends and new developments. *J. Saf. Sci. Resil.* **2024**, *5*, 365–381. [\[CrossRef\]](#)
39. Istanbul Project Coordination Unit. *Istanbul Seismic Risk Mitigation and Emergency Preparedness Project (ISMEP): Activity Report, February 2006–February 2026*; Istanbul Governorship, Republic of Türkiye: Istanbul, Turkey, 2026. Available online: <https://www.ipkb.gov.tr/wp-content/uploads/2026/03/ISMEP-ACTIVITY-REPORT-2026-EN.pdf> (accessed on 24 May 2026).
40. Başaran-Uysal, A.; Sezen, F.; Özden, S.; Karaca, Ö. Classification of residential areas according to physical vulnerability to natural hazards: A case study of Çanakkale, Türkiye. *Disasters* **2014**, *38*, 202–226. [\[CrossRef\]](#)
41. Sunil Kandregula, R.; Kothiyari, G.C.; Chauhan, G.; Thakkar, M.G. Tectono-geomorphic records of neotectonic activity along the Kachchh Mainland fault in a seismically active intraplate setting, Kachchh paleo-rift basin, Western India. *J. Asian Earth Sci.* **2024**, *276*, 106302. [\[CrossRef\]](#)
42. Korkmaz, K.A.; Ay, Z.; Sari, A.; Celik, I.D. Probabilistic seismic risk assessment of hall buildings in Türkiye. *Struct. Des. Tall Spec. Build.* **2013**, *22*, 415–439. [\[CrossRef\]](#)
43. Abdik, Y.; Ocaoglu, N.; Kaypak, B. Analysis of the active tectonics of the Aşkale-Pasinler-Horasan Basins (Eastern Anatolia) using multichannel seismic reflection and stratigraphic data. *J. Afr. Earth Sci.* **2026**, *235*, 105950. [\[CrossRef\]](#)
44. McKenzie, D. Active Tectonics of the Mediterranean Region. *Geophys. J. Int.* **1972**, *30*, 109–185. [\[CrossRef\]](#)
45. Le Pichon, X.; Kreemer, C. The Miocene-to-Present Kinematics of the Eastern Mediterranean and Adjacent Areas and Its Relationship to Plume-Driven Central Europe-Anatolia Rotations. *Annu. Rev. Earth Planet. Sci.* **2010**, *38*, 323–351. [\[CrossRef\]](#)
46. Barka, A. The North Anatolian fault zone. *Ann. Tectonicae* **1992**, *6*, 164–195.
47. Sengör, A.M.C.; Tüysüz, O.; Imren, C.; Sakıncı, M.; Eyidoğan, H.; Görür, N.; Le Pichon, X.; Rangin, C. The North Anatolian Fault: A new look. In *The North Anatolian Fault: A New Look*; Geological Society of America Special Papers; GSA Publication: Boulder, CO, USA, 2005; Volume 393, pp. 1–72. [\[CrossRef\]](#)
48. Jolivet, R.; Jara, J.; Dalaison, M.; Rouet-Leduc, B.; Özdemir, A.; Dogan, U.; Çakır, Z.; Ergintav, S.; Dubernet, P. Daily to centennial behavior of aseismic slip along the central section of the North Anatolian Fault. *J. Geophys. Res. Solid Earth* **2023**, *128*, e2022JB026018. [\[CrossRef\]](#)
49. Stein, R.S.; Barka, A.A.; Dieterich, J.H. Progressive failure on the North Anatolian fault since 1939 by structural stress triggering. *Geophys. J. Int.* **1997**, *128*, 594–604. [\[CrossRef\]](#)
50. Schiffer, C.; Eken, T.; Rondenay, S.; Taymaz, T. Localized crustal deformation along the central North Anatolian Fault Zone revealed by joint inversion of P-receiver functions and P-wave polarizations. *Geophys. J. Int.* **2019**, *217*, 682–702. [\[CrossRef\]](#)
51. Andrieux, J.; Över, S.; Poisson, A.; Bellier, O. The North Anatolian Fault Zone: Distributed Neogene deformation in its northward convex part. *Tectonophysics* **1995**, *243*, 135–154. [\[CrossRef\]](#)
52. Altinok, Y.; Alpar, B.; Özer, N.; Aykurt, H. Revision of the tsunami catalogue affecting Turkish coasts and surrounding regions. *Nat. Hazards Earth Syst. Sci.* **2011**, *11*, 273–291. [\[CrossRef\]](#)
53. Papazachos, B.C.; Papazachou, C. *The Earthquakes of Greece*; Ziti Publications: Thessaloniki, Greece, 2003. Available online: <https://ziti.gr/vivlio/papazachos-vasilis-papazachou-katerina-the-earthquakes-of-greece/> (accessed on 24 May 2026).
54. Okal, E.A.; Synolakis, C.E.; Uslu, B.; Kalligeris, N.; Voukouvalas, E. The 1956 earthquake and tsunami in Amorgos, Greece. *Geophys. J. Int.* **2009**, *178*, 1533–1554. [\[CrossRef\]](#)
55. Papadopoulos, G.A.; Fokaefs, A. Strong tsunamis in the Mediterranean Sea: A re-evaluation. *ISET J. Earthq. Technol.* **2005**, *42*, 159–170. [\[CrossRef\]](#)
56. Burton, P.W.; McGonigle, R.; Makropoulos, K.C.; Üçer, S.B. Seismic risk in Türkiye, the Aegean and the eastern Mediterranean: The occurrence of large magnitude earthquakes. *Geophys. J. R. Astron. Soc.* **1984**, *78*, 475–506. [\[CrossRef\]](#)
57. Yalciner, A.C.; Alpar, B.; Altinok, Y.; Ozbay, I.; Imamura, F. Tsunamis in the Sea of Marmara: Historical documents for the past, models for the future. *Mar. Geol.* **2002**, *190*, 445–463. [\[CrossRef\]](#)
58. Ambraseys, N.N. The Seismicity of the Marmara Sea Area 1800–1899. *J. Seismol.* **2000**, *4*, 377–401. [\[CrossRef\]](#)
59. Altinok, Y.; Tinti, S.; Alpar, B.; Yalciner, A.C.; Ersoy, Ş.; Bortolucci, E.; Armigliato, A. The Tsunami of August 17, 1999 in Izmit Bay, Turkey. *Nat. Hazards* **2001**, *24*, 133–146. [\[CrossRef\]](#)
60. Inan, S.; Ergintav, S.; Saatçılar, R.; Tüzel, B.; İravul, Y. Turkey makes major investment in earthquake research. *Eos Trans. Am. Geophys. Union* **2007**, *88*, 333–334. [\[CrossRef\]](#)
61. Parsons, T. Recalculated probability of $M \geq 7$ earthquakes beneath the Sea of Marmara, Turkey. *J. Geophys. Res. Solid Earth* **2004**, *109*, B05304. [\[CrossRef\]](#)

62. Yao, D.; Peng, Z.; Ding, C.; Sandvol, E.; Godoladze, T.; Yetirmishli, G. Dynamically triggered tectonic tremors and earthquakes in the Caucasian region following the 2023 Kahramanmaraş, Türkiye, earthquake sequence. *Geophys. Res. Lett.* **2024**, *51*, e2024GL110786. [\[CrossRef\]](#)
63. DeSalvio, N.D.; Fan, W. Ubiquitous Earthquake dynamic triggering in southern California. *J. Geophys. Res. Solid Earth* **2023**, *128*, e2023JB026487. [\[CrossRef\]](#)
64. Akinci, A.; Dindar, A.A.; Bal, I.E.; Ertuncay, D.; Smyrou, E.; Cheloni, D. Characteristics of strong ground motions and structural damage patterns from the February 6th, 2023 Kahramanmaraş earthquakes, Türkiye. *Nat. Hazards* **2025**, *121*, 1209–1239. [\[CrossRef\]](#)
65. Toprak, S.; Zulfikar, A.C.; Mutlu, A.; Tugsal, U.M.; Nacaroglu, E.; Karabulut, S.; Ceylan, M.; Ozdemir, K.; Parlak, S.; Dal, O.; et al. The aftermath of 2023 Kahramanmaraş earthquakes: Evaluation of strong motion data, geotechnical, building, and infrastructure issues. *Nat. Hazards* **2025**, *121*, 2155–2192. [\[CrossRef\]](#)
66. Yılmaz, S.; Arslan, M.; Demir, A.D.; Yavru, T.E.; Yılmaz, E.İ.; Cebir, F.; Aydın, Ç.; Baş, G.Y.; Demir, S. Seismic damage assessment of historical mosques and minarets in Antakya/Türkiye after the February 6, 2023, Kahramanmaraş earthquake sequence. *Eng. Fail. Anal.* **2025**, *182*, 110125. [\[CrossRef\]](#)
67. Tiryakioğlu, İ.; Aktuğ, B.; Yiğit, C.Ö.; Yavaşoğlu, H.H.; Sözbilir, H.; Özkaymak, Ç.H.; Poyraz, F.; Taneli, E.; Bulut, F.; Doğru, A.; et al. Slip distribution and source parameters of the 20 July 2017 Bodrum-Kos earthquake (Mw6.6) from GPS observations. *Geodin. Acta* **2017**, *30*, 1–14. [\[CrossRef\]](#)
68. Hussain, E.; Kalaycıoğlu, S.; Milliner, C.W.D.; Çakir, Z. Preconditioning the 2023 Kahramanmaraş (Türkiye) earthquake disaster. *Nat. Rev. Earth Environ.* **2023**, *4*, 287–289. [\[CrossRef\]](#)
69. Carena, S.; Friedrich, A.M.; Verdecchia, A.; Kahle, B.; Rieger, S.; Kübler, S. Identification of source faults of large earthquakes in the Türkiye-Syria border region between 1000 CE and the present, and their relevance for the 2023 Mw 7.8 Pazarcık earthquake. *Tectonics* **2023**, *42*, e2023TC007890. [\[CrossRef\]](#)
70. Özel, N.M.; Ergintav, S.; Turhan, F.; Konca, A.Ö.; Aksarı, D.; Ergün, T. Earthquake Early Warning System for the Marmara Region. In Proceedings of the EGU General Assembly 2026, Vienna, Austria, 3–8 May 2026; EGU26-17285. [\[CrossRef\]](#)
71. Eren, S. Development of Earthquake Early Warning Systems in Marmara Region: A Hybrid Model Utilizing Mosque Loudspeakers. *Int. J. Environ. Geoinformatics* **2025**, *12*, 61–71. [\[CrossRef\]](#)
72. Speciale, S.; Picozzi, M.; Iervolino, I.; Zollo, A.; Caruso, A.; Colombelli, S.; Emolo, A.; Elia, L.; Martino, C.; Festa, G. The first Earthquake Early Warning System for the high-speed railway in Italy: Enhancing rapidness and operational efficiency during seismic events. *Nat. Hazards Earth Syst. Sci. (NHESS)* **2026**, *26*, 299–315. [\[CrossRef\]](#)
73. Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.* **2019**, *1*, 206–215. [\[CrossRef\]](#)
74. Liu, B.; Gong, M.; Wang, X.; Zhou, B.; Jia, J. Real-time seismic damage assessment framework of RC frames using deep learning and monitoring data. *J. Build. Eng.* **2026**, *123*, 115834. [\[CrossRef\]](#)
75. Xue, J.; Huang, Q.; Wu, S.; Zhao, L.; Ma, B. Real-time dual-parameter full-waveform inversion of GPR data based on robust deep learning. *Geophys. J. Int.* **2024**, *238*, 1755–1771. [\[CrossRef\]](#)
76. Witte, P.; Louboutin, M.; Kukreja, N.; Luporini, F.; Lange, M.; Gorman, G.; Herrmann, F. A large-scale framework for symbolic implementations of seismic inversion algorithms in Julia. *Geophysics* **2019**, *84*, 1–60. [\[CrossRef\]](#)
77. Abdellaziz, A.; Brossier, R.; Métivier, L.; Oudet, É. Optimal experimental design for full waveform inversion using a wavenumber sampling criterion—part 1: 2-D to methodological development. *Geophys. J. Int.* **2024**, *240*, 1429–1459. [\[CrossRef\]](#)
78. Virieux, J.; Operto, S. An overview of full-waveform inversion in exploration geophysics. *Geophysics* **2009**, *74*, WCC1–WCC26. [\[CrossRef\]](#)
79. Peter, D.; Komatitsch, D.; Luo, Y.; Martin, R.; Le Goff, N.; Casarotti, E.; Pelties, C.; Ampuero, J.P.; Etienne, V.; Hendrickson, B.; et al. Forward and adjoint simulations of seismic wave propagation on fully unstructured hexahedral meshes. *Geophys. J. Int.* **2011**, *186*, 721–739. [\[CrossRef\]](#)
80. Ejarque, J.; Badia, R.M.; Albertin, L.; Aloisio, G.; Baglione, E.; Becerra, Y.; Boschert, S.; Berlin, J.R.; D’Anca, A.; Elia, D.; et al. Enabling dynamic and intelligent workflows for HPC, data analytics, and AI convergence. *Future Gener. Comput. Syst.* **2022**, *134*, 414–429. [\[CrossRef\]](#)
81. Assunção, M.D.; Calheiros, R.N.; Bianchi, S.; Netto, M.A.; Buyya, R. Big data computing and clouds: Trends and future directions. *J. Parallel Distrib. Comput.* **2015**, *79*, 3–15. [\[CrossRef\]](#)
82. Parisi, F.; Nettis, A.; Uva, G. Machine learning-aided cloud analysis for seismic fragility assessment of multi-span bridges. *Eng. Struct.* **2025**, *343*, 121175. [\[CrossRef\]](#)
83. Krauss, Z.; Ni, Y.; Henderson, S.; Denolle, M. Seismology in the cloud: Guidance for the individual researcher. *Seismica* **2023**, *2*, 1–12. [\[CrossRef\]](#)
84. MacCarthy, J.; Marcillo, O.; Trabant, C. Seismology in the cloud: A new streaming workflow. *Seismol. Res. Lett.* **2020**, *91*, 1804–1812. [\[CrossRef\]](#)

85. Brossier, R.; Operto, S.; Virieux, J. Velocity model building from seismic reflection data by full-waveform inversion. *Geophys. Prospect.* **2015**, *63*, 354–367. [[CrossRef](#)]
86. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **2019**, *378*, 686–707. [[CrossRef](#)]
87. Rasht-Behesht, M.; Huber, C.; Shukla, K.; Karniadakis, G.E. Physics-informed neural networks (PINNs) for wave propagation and full waveform inversions. *J. Geophys. Res. Solid Earth* **2022**, *127*, e2021JB023120. [[CrossRef](#)]
88. Minson, S.E.; Murray, J.R.; Langbein, J.O.; Gomberg, J.S. Real-time inversions for finite fault slip models and rupture geometry based on high-rate GPS data. *J. Geophys. Res. Solid Earth* **2014**, *119*, 3201–3231. [[CrossRef](#)]
89. Glasgow, M.; Schmandt, B.; Wang, R.; Zhang, M.; Bilek, S.L.; Kiser, E. Raton Basin induced seismicity is hosted by networks of short basement faults and mimics tectonic earthquake statistics. *J. Geophys. Res. Solid Earth* **2021**, *126*, e2021JB022839. [[CrossRef](#)]
90. Wilkinson, P.B.; Chambers, J.E.; Meldrum, P.I.; Kuras, O.; Inauen, C.M.; Swift, R.T.; Curioni, G.; Uhlemann, S.; Graham, J.; Atherton, N. Windowed 4D inversion for near real-time geoelectrical monitoring applications. *Front. Earth Sci.* **2022**, *10*, 983603. [[CrossRef](#)]
91. Song, C.; Alkhalifah, T.; Waheed, U.B. A versatile framework to solve the Helmholtz equation using physics-informed neural networks. *Geophys. J. Int.* **2022**, *228*, 1750–1762. [[CrossRef](#)]
92. Gineste, M.; Eidsvik, J. Batch seismic inversion using the iterative ensemble Kalman smoother. *Comput. Geosci.* **2021**, *25*, 1105–1121. [[CrossRef](#)]
93. Moseley, B.; Markham, A.; Nissen-Meyer, T. Solving the wave equation with physics-informed deep learning. *arXiv* **2020**, arXiv:2006.11894. [[CrossRef](#)]
94. Ide, S.; Beroza, G.C.; Shelly, D.R.; Uchide, T. A scaling law for slow earthquakes. *Nature* **2007**, *447*, 76–79. [[CrossRef](#)]
95. Obara, K. Nonvolcanic deep tremor associated with subduction in southwest Japan. *Science* **2002**, *296*, 1679–1681. [[CrossRef](#)] [[PubMed](#)]
96. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [[CrossRef](#)]
97. Candès, E.J.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509. [[CrossRef](#)]
98. Herrmann, F.J.; Li, X. Efficient least-squares imaging with sparsity promotion and compressive sensing. *Geophys. Prospect.* **2012**, *60*, 696–712. [[CrossRef](#)]
99. Bensen, G.D.; Ritzwoller, M.H.; Barmin, M.P.; Levshin, A.L.; Lin, F.; Moschetti, M.P.; Shapiro, N.M.; Yang, Y. Processing seismic ambient noise data to obtain reliable broad-band surface wave dispersion measurements. *Geophys. J. Int.* **2007**, *169*, 1239–1260. [[CrossRef](#)]
100. Ourabah, A. Processing on the edge: The evolution of in-field processing on modern land seismic surveys. *EAGE First Break* **2025**, *43*, 69–73. [[CrossRef](#)]
101. Armbrust, M.; Xin, R.S.; Lian, C.; Huai, Y.; Liu, D.; Bradley, J.K.; Meng, X.; Kaftan, T.; Franklin, M.J.; Ghodsi, A.; et al. Apache Spark SQL: Relational data processing in Spark. In *Proceedings of the 2015 ACM SIGMOD*; Association for Computing Machinery: New York, NY, USA, 2015. [[CrossRef](#)]
102. Zaharia, M.; Xin, R.S.; Wendell, P.; Das, T.; Armbrust, M.; Dave, A.; Meng, X.; Rosen, J.; Venkataraman, S.; Franklin, M.J.; et al. Apache Spark: A unified engine for big data processing. *Commun. ACM* **2016**, *59*, 56–65. [[CrossRef](#)]
103. Cremen, G.; Galasso, C. Earthquake early warning: Recent advances and perspectives. *Earth-Sci. Rev.* **2020**, *205*, 103184. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.