

Diseño de una Base de Datos Sismológica de la REDNE

Carlos Humberto Díaz Salazar

Trabajo de Grado para optar al título de Ingeniero Electrónico

Director

Dr.-Ing. Sergio Alberto Abreo Carrillo

Co-Directora

Ph.D Ana Beatriz Ramírez Silva

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Programa Ingeniería Electrónica

Bucaramanga

2026

## Tabla de Contenido

Introducción	10
1 Objetivos	12
1.1 Objetivo general	12
1.2 Objetivos específicos	12
2 Marco Teórico	13
3 Metodología	20
3.1 Datos	20
3.2 Preprocesamiento	22
3.2.1 Acondicionamiento de Señales y Depuración	23
3.2.2 Automatización de la Detección (Pipeline Adaptativo)	23
3.2.3 Generación y Validación del Dataset de Ruido	27
3.3 Análisis Energético y Calibración de Parámetros	28
3.4 Base de datos local	29
3.4.1 MongoDB	30
3.5 Metadatos	33
4 Conclusiones	35
5 Recomendaciones	37
Referencias Bibliográficas	37
Apéndices	41

### Lista de Figuras

Figura 1	Ubicación de las estaciones REDNE	15
Figura 2	Esquema del método STA/LTA para detección de eventos sísmicos	17
Figura 3	Arquitectura general del pipeline de procesamiento	20
Figura 4	Flujo de adquisición de trazas sísmicas	21
Figura 5	Etapas secuenciales del preprocesamiento y limpieza de señales	22
Figura 6	Visualización del funcionamiento del detector en diferentes escenarios. Las filas muestran desde ausencia de eventos (arriba) hasta detecciones múltiples (abajo). Las áreas sombreadas en rojo indican el momento exacto donde el algoritmo "decidió" que había un sismo.	26
Figura 7	Flujo de operaciones de la base de datos local	30
Figura 8	Esquema relacional de colecciones en MongoDB	32
Figura 9	Visualización de un evento sísmico con sus metadatos asociados	33
Figura 10	Eventos a través del tiempo, estación UIS01 y UIS03	42
Figura 11	Eventos a través del tiempo, estación UIS04 y UIS05	43
Figura 12	Eventos a través del tiempo, estación UIS09 y UIS11	44
Figura 13	Gráficas Comparativas medianas y cantidades	45
Figura 14	Diagrama de flujo algoritmo STA/LTA	50

### Lista de Tablas

Tabla 1	Estaciones, ubicaciones, dispositivos y canales registrados	14
Tabla 2	Análisis de sensibilidad: Cantidad de eventos detectados según el tamaño de la ventana temporal	24
Tabla 3	Resultados de la clasificación automática con ventana de 20 segundos	25
Tabla 4	Resultados de la clasificación automática con ventana de 30 segundos (Configuración óptima)	25
Tabla 5	Resumen de detección de eventos en archivos de RUIDO analizados	28
Tabla 6	Versiones de librerías utilizadas en el entorno Python 3.9.21	49

**Lista de Apéndices**

	<b>pág.</b>
Apéndice A Análisis Energético y Depuración STA/LTA	41
Apéndice B Otros metadatos	46
Apéndice C Entorno y repositorio	49
Apéndice D Diagrama de flujo algoritmo STA/LTA	50

## Glosario

**JSON:** Formato de intercambio de datos binario utilizado principalmente por MongoDB.

**Colección:** Grupo de documentos en una base de datos MongoDB.

**Dataset:** Conjunto de datos estructurado.

**Documento:** Unidad básica para los datos en MongoDB.

**HDF5:** Formato de datos jerárquico para almacenar grandes cantidades de datos.

**MiniSEED (mseed):** Formato estándar de datos sísmológicos para almacenamiento de series temporales.

**MongoDB:** Gestor de bases de datos NoSQL, diseñado para contener datos en estructuras flexibles.

**Pipeline** Flujo de trabajo automatizado que conecta diversas etapas de procesamiento de datos, desde la recolección y limpieza hasta el almacenamiento final.

**REDNE:** Red sísmológica del Nororiente Colombiano, infraestructura de estaciones sísmológicas ubicadas estratégicamente para monitorear la actividad sísmica en la región.

**Ruido sísmico** Vibraciones permanentes del suelo registradas por las estaciones sísmológicas que no corresponden a eventos sísmicos (terremotos), sino a fuentes ambientales o antrópicas.

**STA/LTA:** Algoritmo de disparo que procesa señales mediante dos ventanas temporales (promedio de corto plazo y promedio de largo plazo) para detectar eventos sísmicos basándose en cambios de amplitud.

**STEAD:** Stanford Earthquake Dataset es el conjunto de datos de referencia global de alta calidad.

**Traza sísmica:** Registro de la señal de movimiento del suelo capturada por un sismómetro o acelerómetro durante un periodo de tiempo.

**Trigger:** Mecanismo de disparo que se activa cuando la señal de una traza supera un criterio predefinido.

**Umbral (Threshold):** Valor definido de activación o desactivación para determinar el principio o final de un evento sísmico.

**UTC:** Tiempo Universal Coordinado estándar principal usado para registrar fechas y horas de los eventos sísmicos.

## Resumen

**Título:** Diseño de una Base de Datos Sismológica de la REDNE \*

**Autor:** Carlos Humberto Díaz Salazar \*\*

**Palabras Clave:** Sismología, REDNE, STEAD, MongoDB, STA/LTA, Inteligencia Artificial.

**Descripción:** Hoy en día, con el rápido desarrollo de ciencias como la sismología y su generación masiva de datos es necesario procesar eficazmente estas enormes cantidades de información para poder usarla en el desarrollo de aplicaciones con inteligencia artificial. Esta tesis describe el diseño y desarrollo de una base de datos sismológica basada en el modelo del Conjunto de Datos Sísmicos de Stanford (STEAD) para la Red Sismológica del Nororiente Colombiano (REDNE). El proyecto tiene como objetivo el diseño de una base de datos que almacena y organiza los metadatos de eventos sismológicos y de ruido, además, también procesa y analiza automáticamente trazas sísmicas para el crecimiento futuro y la robustez del conjunto de datos y así mismo poder entrenar modelos de aprendizaje automático.

La metodología incluye el procesamiento de trazas MiniSEED (mseed), Se creó un pipeline basado en Jupyter para el preprocesamiento de señales y se aplicó el algoritmo STA/LTA para detectar y recortar eventos sísmicos. Para el almacenamiento, se adoptó un enfoque NoSQL y se utilizó MongoDB aplicando patrones de diseño basados en atributos y control de versiones con el fin de que el sistema fuera escalable y flexible. Esto se combinó en un conjunto de datos local con eventos sismológicos y registros de ruido, y una base de datos con metadatos en un formato estandarizado internacional, este sistema no solo optimiza la gestión de la información, sino que también ofrece una infraestructura importante para el análisis avanzado de la sismicidad local, nacional e internacional.

---

\* Trabajo de grado

\*\* Facultad de Ingenierías Fisicomecánicas. Escuela de Ingenierías de Electrica, Electrónica y de Telecomunicaciones. Programa Ingeniería Electrónica. Director: Dr.-Ing. Sergio Alberto Abreo Carrillo. Codirectora: Ph.D Ana Beatriz Ramírez Silva

## Abstract

**Title:** Design of a Seismological Database for the REDNE \*

**Author:** Carlos Humberto Díaz Salazar \*\*

**Keywords:** Seismology, REDNE, STEAD, MongoDB, STA/LTA, Artificial Intelligence.

**Description:** Today, with the rapid development of sciences such as seismology and the massive generation of data, it is necessary to process these enormous amounts of information efficiently in order to use it in the development of artificial intelligence applications. This thesis describes the design and development of a seismological database based on the Stanford Seismic Data Set (STEAD) model for the Seismological Network of Northeast Colombia (REDNE). The project aims to design a database that stores and organizes metadata from seismological and noise events, as well as automatically processing and analyzing seismic traces for future growth and robustness in order to train machine learning models.

The methodology includes MiniSEED (mseed) trace processing. A Jupyter-based pipeline was created for signal pre-processing, and the STA/LTA algorithm was applied to detect and trim seismic events. For storage, a NoSQL approach was adopted and MongoDB was used, applying attribute-based design patterns and version control to make the system scalable and flexible. This was combined into a local dataset with seismological events and noise records, and a database with metadata in an internationally standardized format. This system not only optimizes information management, but also provides an important infrastructure for advanced analysis of local, national, and international seismicity.

---

\* Degree Work

\*\* Faculty of Physicomechanical Engineering. School of Electrical, Electronic and Telecommunications Engineering. Electronic Engineering Program. Director: Dr.-Ing. Sergio Alberto Abreo Carrillo. Co-director: Ph.D. Ana Beatriz Ramírez Silva

## Introducción

La integración de modelos de aprendizaje automático en la sismología moderna ha transformado la manera en que comprendemos los movimientos sísmicos, cambiando de la interpretación manual de sismogramas a un análisis múltiple y masivo de datos. Sin embargo, la confiabilidad y alcance de estos modelos predictivos no solo dependen de complejos algoritmos, sino de la calidad y estructuración de los datos que los alimentan. Trabajos como el *Stanford Earthquake Dataset* (STEAD) Mousavi, Sheng, Zhu, y Beroza (2019) han demostrado que estandarizar la información es el primer paso para lograr avances; no obstante, en el contexto local de Colombia, al existir una preocupante brecha tecnológica, la abundancia de datos crudos sin catalogar no se traduce automáticamente en datos útiles para una investigación a profundidad.

En Santander existe el nido sísmico de la Mesa de los Santos, una de las fuentes de actividad sismológica más activas del mundo. La Red sismológica del Nororiente Colombiano (REDNE) captura continuamente la actividad del nido, generando un volumen masivo de información. Sin embargo, el almacenamiento convencional de estos datos (dispersos en archivos crudos, sin procesamiento automático y carentes de metadatos estandarizados) representa una barrera que impide el aprovechamiento su potencial científico. Actualmente, la región es rica en datos sísmicos, pero deficiente en conjuntos de datos preparados para el entrenamiento de modelos de aprendizaje automático (*Machine Learning*).

Para superar esta limitación y brindar un aporte significativo al entrenamiento de modelos

de ML, este trabajo presenta el diseño y desarrollo de una base de datos sismológica automatizada para la REDNE, estructurada bajo estándares internacionales que maneja STEAD. La propuesta no se limita al almacenamiento; se centra en la automatización del ciclo de vida del dato. Se ha diseñado una arquitectura centralizada basada en tecnologías NoSQL (MongoDB), que permite gestionar de manera flexible y escalable tanto las trazas sísmicas como sus metadatos asociados, garantizando la trazabilidad y el control de versiones históricos.

Para materializar esta infraestructura, se implementó un pipeline de procesamiento en Python que automatiza el flujo de trabajo completo: desde la descarga sistemática de datos del Servicio Geológico Colombiano (SGC) y el OSSO, hasta la detección y recorte de eventos mediante el algoritmo STA/LTA. Este enfoque garantiza que la información cruda se transforme en un conjunto de datos listo para usar, validado y preprocesado.

La estandarización de estos datos según los lineamientos de SeisBench (Woollam y cols., 2022) y STEAD (Mousavi y cols., 2019) facilita la interoperabilidad científica y el intercambio de información. Socialmente, disponer de una base de datos robusta potencia la capacidad de vigilancia sísmica en Santander, beneficiando directamente al grupo de investigación CPS y a las organizaciones de gestión del riesgo, quienes contarán con insumos de alta calidad para el desarrollo de futuros modelos predictivos y de clasificación basados en Inteligencia Artificial.

## 1. Objetivos

### 1.1. Objetivo general

Diseñar una base de datos sísmológica a partir de la información de la REDNE, que permita almacenar, organizar y procesar automáticamente trazas sísmicas y sus metadatos, con el propósito de apoyar el entrenamiento de modelos de aprendizaje automático en el ámbito de la sísmología.

### 1.2. Objetivos específicos

1. Desarrollar un procedimiento sistemático de recolección y depuración de trazas sísmicas provenientes de la REDNE.
2. Implementar y automatizar un *pipeline* de procesamiento de datos sísmológicos.
3. Diseñar y estructurar una base de datos sísmológica inspirada en la organización del Stanford Earthquake Dataset (STEAD).
4. Integrar y escalar un sistema de automatización para la actualización continua del dataset.

## 2. Marco Teórico

La sismología se encuentra abastecida de redes y estaciones sismológicas repartidas por el mundo, las cuales capturan y monitorean los movimientos del subsuelo, nidos sísmicos, cantidad de energía elástica liberada por las placas tectónicas y demás movimientos terrestres. Estas estaciones proporcionan una enorme cantidad de datos diariamente, dando lugar a que el campo sea rico en datos en crudo para poder ser tratados, procesados, clasificados y definidos con sus características propias como: ruido sísmico, magnitudes de ondas y cálculos de posicionamiento.

El desarrollo y aplicación de modelos de aprendizaje automático en este entorno contribuye al avance tecnológico, fomenta la creación de nuevos modelos y algoritmos robustos para procesar y analizar los datos sismológicos (Mousavi y cols., 2019). En la vastedad de los datos recopilados, es esencial asignarles etiquetas para la misma construcción de modelos supervisados. Su fiabilidad es variable y dependiente de factores como la experticia humana seleccionando las mismas etiquetas. Por esta misma razón, la cantidad de datos etiquetados es menor a los terabytes de datos de ondas sísmicas.

STEAD (Stanford Earthquake Dataset) se presenta con la idea de generar un conjunto de datos de alta calidad y fiabilidad con el cual se logren entrenar nuevos modelos de aprendizaje automático, seccionado en 2 categorías: señales sísmicas con formas de onda registradas localmente dentro de 350 km de los terremotos y no sísmicas con formas de onda de ruido. STEAD se constituye aproximadamente de 1.2 millones de series temporales, es decir, unas 20 mil horas de

grabación de señales sísmicas (Mousavi y cols., 2019).

Son 1.050.000 trazas locales y aproximadamente 450.000 trazas grabadas desde 1984 a 2018, ambas de 60 segundos de duración descritas como matrices Numpy de 3 canales y 6000 muestras capturadas por 2.613 receptores, 144 redes sísmicas a escala local, regional y global, 7 tipos de instrumentos donde el 99.5% son BH, HH y EH. Cada traza cuenta con 35 metadatos que describen la información del instrumento, del sismo y de la señal, mientras que las señales no sísmicas son aproximadamente 100.000 trazas de 60 segundos de duración y cuentan con 8 metadatos que describen la información del instrumento que capturó el ruido. Desde el año 2000, la mayoría de los sismos del dataset se encuentran en USA y Europa, debido a su densa cobertura de estaciones sísmológicas (Mousavi y cols., 2019).

**Tabla 1**

*Estaciones, ubicaciones, dispositivos y canales registrados*

#	Estación	Ubicación	Dispositivo	Canales
1	UIS01	Los Santos	ETNA 2	HNE HNN HNZ
2	UIS03	Málaga	ETNA 2	HNE HNN HNZ
3	UIS04	Pamplona	ETNA 2	HNE HNN HNZ
4	UIS05	La Esmeralda	RS 4D	ENE ENN ENZ EHZ
5	UIS06	San Alberto	RS 3D	OFF o en mantenimiento
6	UIS09	Zapatoca	RS 3D	EHE EHN EHZ
7	UIS010	Macaravita	RS 3D	OFF o en mantenimiento
8	UIS11	Chitagá	RS 3D	EHE EHN EHZ

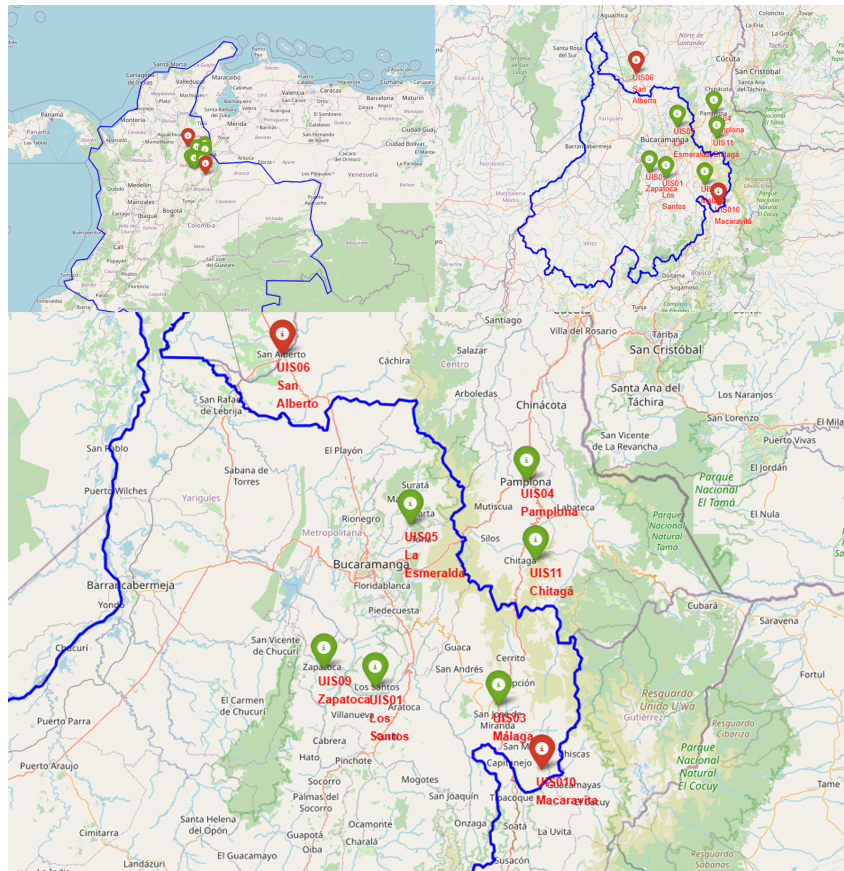
Nota. Información correspondiente a las estaciones sísmológicas de la REDNE.

REDNE está constituida por 8 estaciones sísmológicas (con 3 distintos tipos de dispositivos)

(Ver Tabla 1) ubicadas estratégicamente para encerrar el nido sísmico que se encuentra en la Mesa de los Santos (ver figura 1).

### Figura 1

#### Ubicación de las estaciones REDNE



*Nota.* Se resaltan en color verde las estaciones activas y en rojo las que se encuentran en mantenimiento, cercanas y dentro del perímetro del departamento de Santander.

El algoritmo STA/LTA procesa las señales sísmicas filtradas mediante dos ventanas temporales móviles: una ventana de promedio de corto plazo (*Short-Term Average, STA*) y una ventana de promedio de largo plazo (*Long-Term Average, LTA*). La ventana STA mide la

amplitud instantánea de la señal sísmica y permite la detección de terremotos, mientras que la ventana LTA representa la amplitud promedio del ruido sísmico de fondo.

Inicialmente, se calcula la amplitud absoluta de cada muestra de la señal entrante. Posteriormente, se obtiene el promedio de dichas amplitudes absolutas dentro de ambas ventanas temporales. En un paso posterior, se calcula la relación entre estos dos valores, conocida como la relación STA/LTA. Esta relación se compara continuamente con un valor umbral definido por el usuario, denominado umbral de activación STA/LTA. Cuando la relación STA/LTA supera dicho umbral, se declara un disparo de canal.

Un disparo de canal no implica necesariamente que un registrador multicanal o una red sísmica comiencen a registrar datos. En la práctica, las redes sísmicas y la mayoría de los registradores incorporan un mecanismo de votación de disparos, el cual define cuántos y cuáles canales deben activarse antes de iniciar el registro de datos. Para simplificar la explicación, se considera únicamente un canal de señal, asumiendo que un disparo de canal equivale a un disparo de red o de registrador.

Tras la disminución progresiva de la señal sísmica, el canal se desactiva cuando la relación STA/LTA cae por debajo de un segundo parámetro definido por el usuario, denominado umbral de desactivación STA/LTA. Este umbral debe ser inferior (o, en casos excepcionales, igual) al umbral de activación STA/LTA (CUSeisTut Project, 2025).

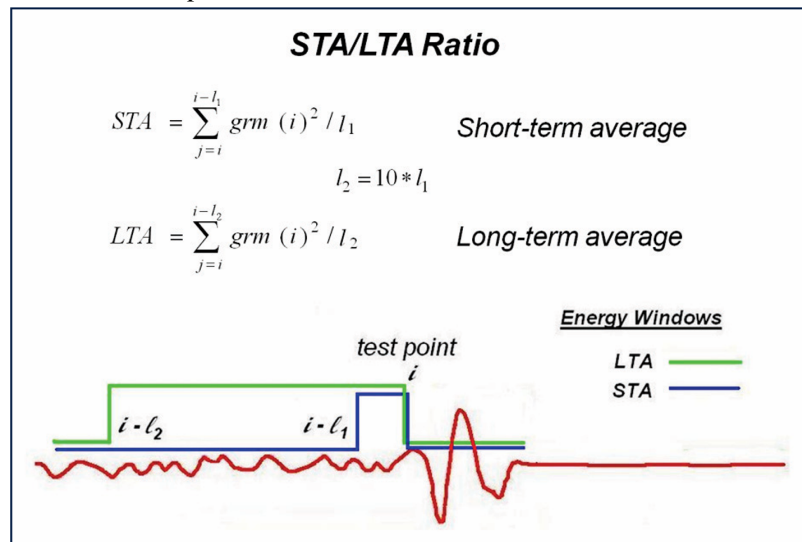
Las ecuaciones asociadas al método calculan dos promedios de energía de la señal: uno sobre una ventana corta (STA) y otro sobre una ventana más extensa (LTA). La ventana STA representa la energía instantánea de la señal y es sensible a cambios bruscos, como los producidos

por la llegada de una onda sísmica. Por el contrario, la ventana LTA describe el nivel de energía promedio del ruido de fondo, proporcionando una referencia estable del comportamiento normal de la señal.

Al evaluar el cociente entre ambos promedios en cada punto del registro, se obtiene una medida adimensional que resalta incrementos súbitos de energía. Cuando este cociente supera un umbral predefinido, se interpreta como la posible ocurrencia de un evento sísmico. La Figura 2 ilustra gráficamente este proceso, mostrando la relación temporal entre las ventanas STA y LTA y su evaluación sobre la señal sísmica.

**Figura 2**

*Esquema del método STA/LTA para detección de eventos sísmicos*



*Nota.* Representación del cálculo de los promedios a corto plazo (STA) y largo plazo (LTA) sobre una señal sísmica, y su relación con el punto de evaluación. Adaptado de la documentación de ObsPy Development Team (2024).

SeisBench es una herramienta de código abierto de Python para el aprendizaje automático en sismología. Proporciona una API unificada para aplicar modelos de aprendizaje profun-

do a formas de onda sísmicas, así como para acceder y entrenar algoritmos de aprendizaje automático en conjuntos de datos sísmicos. SeisBench se ha desarrollado para reducir los obstáculos habituales al aplicar técnicas de aprendizaje automático a datos sísmicos, en particular en las etapas de preparación, recopilación y etiquetado de datos. Asimismo, busca fomentar la interoperabilidad, la intercambiabilidad y la comparabilidad de diferentes modelos y conjuntos de datos (Woollam y cols., 2022).

Un sistema de gestión de bases de datos (SGBD) es un software el cual permite manipular y administrar datos de forma eficiente procurando la integridad, la seguridad y la disponibilidad de la información. Las estructuras más conocidas son SQL (Structured Query Language) y NoSQL.

Las bases de datos NoSQL (Not Only SQL) manejan características principales como un esquema flexible, una escalabilidad horizontal y un alto rendimiento con grandes volúmenes de datos. MongoDB, Inc. (2025c) es un sistema de gestión de bases de datos NoSQL orientado a documentos, diseñado para almacenar datos en estructuras flexibles tipo BSON, permitiendo esquemas dinámicos y una alta escalabilidad.

La relación entre conceptos básicos de SQL y MongoDB en una base de datos, consiste en que una tabla es llamada colección y una fila o registro es un documento, esto facilita entender MongoDB desde la perspectiva de SQL.

A diferencia de los modelos relacionales tradicionales, los documentos en MongoDB no requieren un esquema rígido y pueden variar en su estructura dentro de una misma colección. Esta característica facilita el almacenamiento de datos semiestructurados y heterogéneos, permitiendo una mayor flexibilidad en el diseño de la base de datos y reduciendo la necesidad de migraciones

estructurales ante cambios en los datos.

MongoDB posee la características de esquemas flexibles, en los que se pueden añadir, modificar o eliminar campos dentro de los documentos sin dañar la integridad de los datos, esta flexibilidad proporciona una ventaja para los sistemas en los que la estructura de la información puede cambiar con el tiempo.

Se admiten modelos dinámicos como el control de versiones de los datos con fines históricos o cambios en los campos que hacen referencia a nuevas versiones de los datos, de este modo, MongoDB puede gestionar fácilmente los cambios en la información sin sobrecargar el sistema con cambios complejos sobre las bases de datos.

La estructura de documentos de MongoDB permite realizar consultas flexibles que sean eficientes sobre estos, por ejemplo, el filtrado basado en valores de campos. En este apartado los índices se emplean para mejorar el rendimiento de las consultas y el tiempo de acceso a los grandes volúmenes de datos.

En cuanto a la seguridad, MongoDB incorpora mecanismos de autenticación y autorización basados en roles, los cuales permiten definir distintos niveles de acceso para los usuarios del sistema, esta característica contribuye a garantizar la integridad, la confidencialidad y la disponibilidad de la información.

MongoDB shell es un entorno Read Eval Print Loop (REPL) JavaScript y Node.js, para interactuar con la información de las bases de datos por medio de peticiones o consultas (queries) y el Graphical User Interface (GUI) es MongoDB Compass, una herramienta interactiva gratuita para consultar, optimizar y analizar datos almacenados con MongoDB.

### 3. Metodología

Con el objetivo de automatizar la construcción de una base de datos sísmológica a partir de la información de la REDNE, se diseñó un flujo de trabajo que integra de forma sistemática las etapas de recolección, depuración, procesamiento y almacenamiento de trazas sísmicas y sus metadatos. Este enfoque permite organizar los registros de manera coherente y escalable, facilitando su uso posterior en aplicaciones de aprendizaje automático. La arquitectura general del proceso se presenta en la Figura 3.

**Figura 3**

*Arquitectura general del pipeline de procesamiento*



*Nota.* El diagrama ilustra el flujo de datos desde la consulta del catálogo SGC y descarga del OSSO, hasta las etapas de preprocesamiento y almacenamiento en la base de datos.

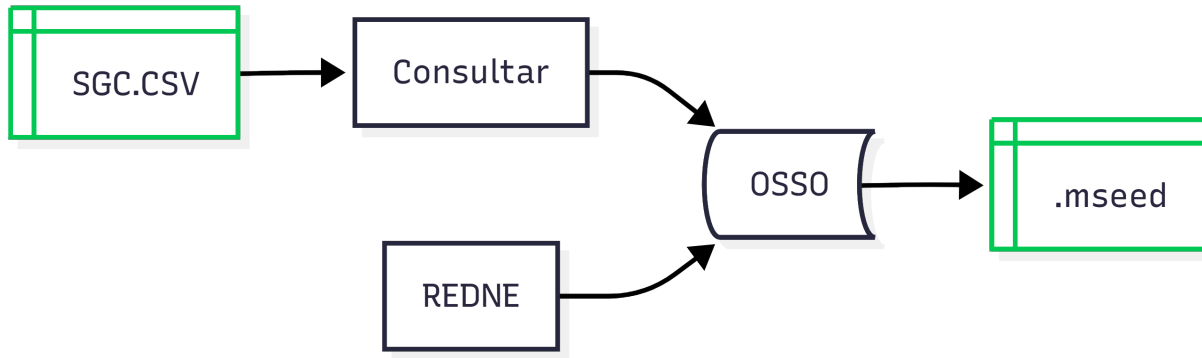
#### 3.1. Datos

Para dar cumplimiento al **primer objetivo específico**, referente al **desarrollo de un procedimiento sistemático de recolección**, se diseñó una estrategia de procesamiento bajo un esquema de etapas secuenciales. Este diseño garantiza la trazabilidad e integridad de la información, asegurando que los datos originales no sean sobrescritos, sino que evolucionen a través de directorios específicos a medida que son transformados. El flujo abarca desde la obtención de datos crudos,

pasando por la validación de canales, hasta la detección de eventos.

#### Figura 4

*Flujo de adquisición de trazas sísmicas*



*Nota.* Describe el proceso automatizado donde el script lee fechas del catálogo SGC y realiza peticiones al servicio web del OSSO para descargar los archivos MiniSEED.

El proceso de adquisición inicia con una consulta al catálogo de sismicidad del Servicio Geológico Colombiano SGC (2025). Se delimitó la búsqueda a la región de Santander, abarcando el periodo del 1 de abril de 2024 al 14 de julio de 2025, resultando en la recopilación de 1496 sismos con magnitudes superiores a 2.5 en diferentes escalas como: magnitud local y magnitud de momento. La información temporal de estos eventos (Fecha-Hora UTC) contenida en el archivo .csv base, alimenta un algoritmo desarrollado en Python encargado de gestionar las peticiones de descarga.

Como se esquematiza en la Figura 4, el sistema transforma los registros del catálogo en solicitudes al servicio web del OSSO. Para cada evento, se descargan ventanas de tiempo de 2 minutos (6 segundos de pre-evento y 114 segundos de post-evento), generando un directorio local de archivos mseed. Este procedimiento automatizado permitió consolidar un total de 5061 archivos

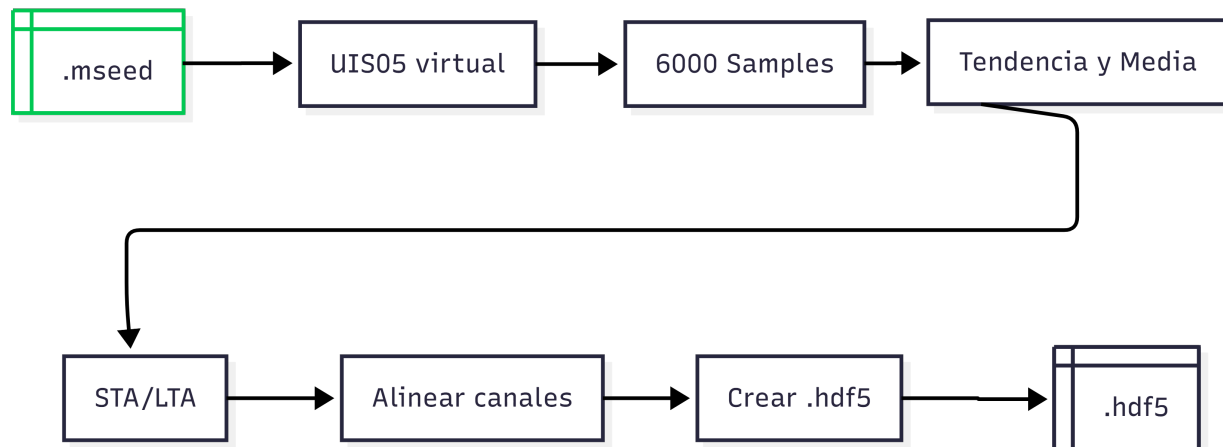
mseed, estableciendo así la base de datos cruda necesaria para las siguientes etapas y cumpliendo con el **desarrollo de un procedimiento sistemático de recolección**.

### 3.2. Preprocesamiento

La etapa de preprocesamiento tiene como finalidad asegurar la calidad de las señales y estandarizar los datos para su posterior análisis, abordando así la componente de **depuración de trazas sísmicas provenientes de la REDNE** planteada en los objetivos.

**Figura 5**

*Etapas secuenciales del preprocesamiento y limpieza de señales*



*Nota.* Esquema de las operaciones de limpieza: gestión de fallos instrumentales, corrección de señales y detección inteligente de eventos.

Como se resume en la Figura 5, el flujo de trabajo integra la gestión de canales, la normalización de la señal y la detección automatizada de eventos.

### ***3.2.1. Acondicionamiento de Señales y Depuración***

Un desafío técnico importante resuelto en esta fase fue el ruido capturado en los canales de la estación UIS05 ante sismos leves. Para evitar la pérdida de información valiosa, se diseñó una estrategia: el sistema descarta las trazas de los canales de aceleración que contienen solamente ruido al momento de los sismos y son mejor capturados por el canal de velocidad que registran el evento con mayor fidelidad al ser de menor magnitud.

Adicionalmente, se aplicaron filtros de cantidad de muestras para descartar registros incompletos (producidos por autocalibraciones del equipo) y se realizaron correcciones estándar de línea base eliminando niveles de offset de los equipos de sensado para normalizar las señales. De los miles de archivos iniciales, el sistema conservó únicamente aquellas trazas que cumplían con los estándares de calidad manejados por STEAD para el análisis científico.

La implementación de este flujo organizado de descarga, corrección instrumental y limpieza de señal constituye el cumplimiento del **primer objetivo específico**, al entregar un conjunto de datos **sistemáticamente recolectado y depurado**.

### ***3.2.2. Automatización de la Detección (Pipeline Adaptativo)***

Para materializar el **pipeline de procesamiento automatizado del segundo objetivo específico**, se implementó un sistema de detección que evalúa el comportamiento de la señal. Este módulo utiliza el método clásico de relación de energías (STA/LTA), optimizado con una lógica que se adapta dinámicamente al nivel de los eventos simológicos débiles o fuertes.

En una primera fase, se realizó un estudio de sensibilidad para determinar la ventana de

tiempo ideal para la red sismológica. Se evaluaron distintos intervalos para encontrar el equilibrio entre capturar sismos de menor magnitud y no confundirlos con ruido de fondo. La Tabla 2 resume este comportamiento, evidenciando que una ventana de 30 segundos ofrece la mejor estabilidad para este caso de estudio al seleccionar la mayor cantidad de trazas para la construcción robusta del dataset.

**Tabla 2**

*Análisis de sensibilidad: Cantidad de eventos detectados según el tamaño de la ventana temporal*

Ventana LTA (s)	0 eventos	1 evento	2 eventos	3 eventos	$\geq 4$ eventos
60	5835	326	246	40	24
50	4069	1600	688	83	31
40	1884	2892	1448	195	52
30	1628	556	3922	297	68
20	1603	650	3829	304	85

*Nota.* La columna (0 eventos) refleja mayoritariamente los canales con solo ruido de la estación UIS05, los cuales el algoritmo logró filtrar correctamente.

A diferencia de los sistemas tradicionales que usan un único valor de disparo fijo, el algoritmo desarrollado aplica una estrategia de re-evaluación: primero realiza una exploración con un umbral estándar y, dependiendo de lo que encuentre, ajusta su criterio automáticamente:

- **Modo de Alta Sensibilidad:** Si el sistema detecta un evento único pero muy débil, reduce automáticamente la barrera de detección. Esto permite rescatar sismos tenues que un sistema rígido ignoraría.
- **Modo de Rechazo de Ruido:** Por el contrario, si el sistema detecta múltiples disparos desordenados (típico del tráfico o actividad urbana), aumenta la exigencia del umbral. Esto

endurece el criterio para asegurar que solo se registren eventos sísmicos reales y no falsas alarmas.

**Tabla 3**

*Resultados de la clasificación automática con ventana de 20 segundos*

Umbral Bajo	Umbral Alto	0 eventos	1 evento	2 eventos	3 eventos	$\geq 4$ eventos
3	6	1603	650	3829	304	85
3	8	1698	748	3781	191	53
2	8	1698	302	3545	382	544
2	6	1603	204	3593	495	576

*Nota.* Resultados obtenidos variando los umbrales de activación (TH\_ON\_LOW y TH\_ON\_HIGH). Se analizaron 6471 archivos en total.

Esta capacidad de adaptación fue validada mediante pruebas masivas cuyos resultados cuantitativos se detallan en las Tablas 3 y 4. En ellas se observa cómo la variación de los umbrales (inferior y superior) que impactan en la clasificación final.

**Tabla 4**

*Resultados de la clasificación automática con ventana de 30 segundos (Configuración óptima)*

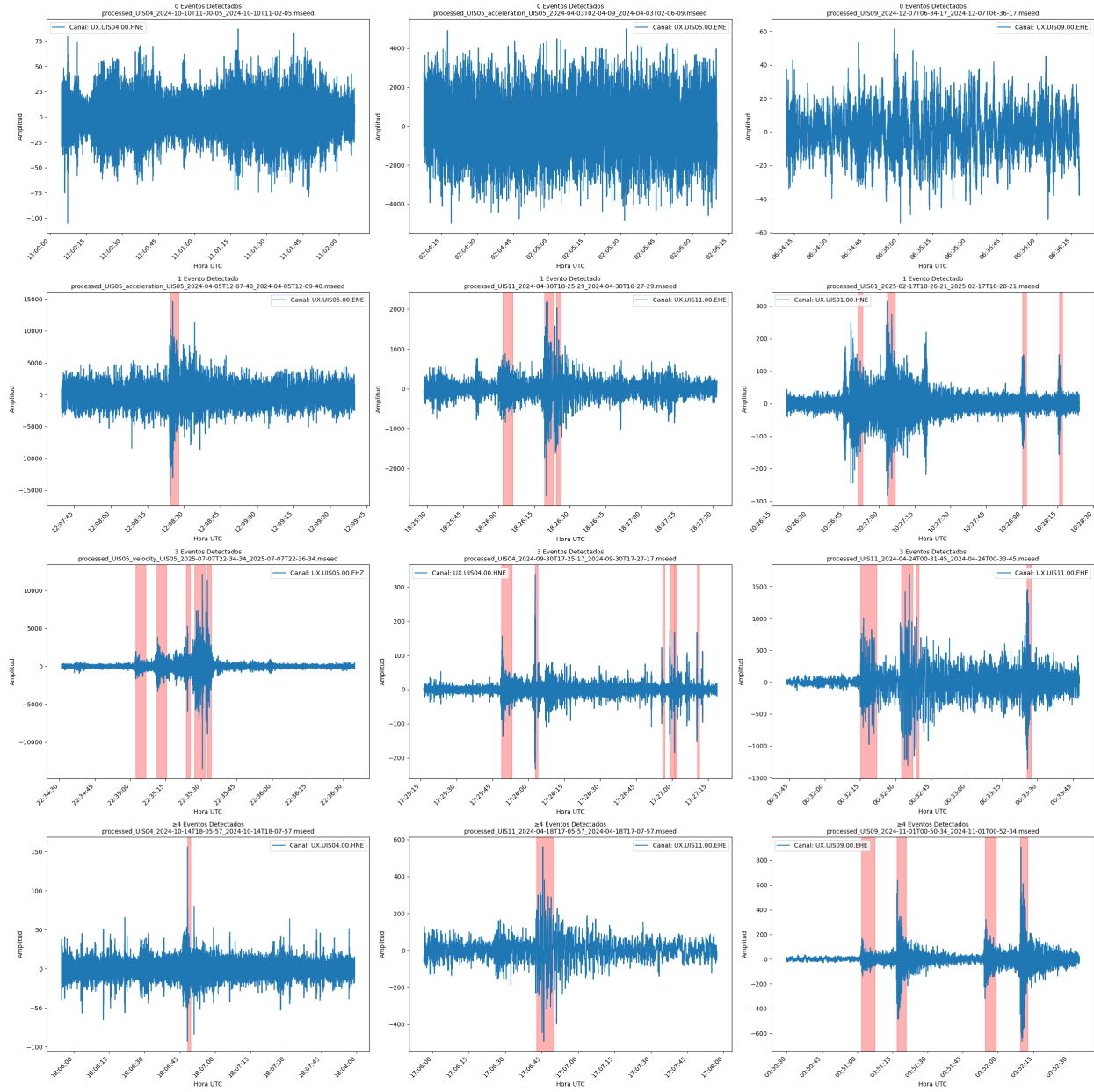
Umbral Bajo	Umbral Alto	0 eventos	1 evento	2 eventos	3 eventos	$\geq 4$ eventos
3	6	1628	556	3922	297	68
3	8	1702	660	3886	192	31
2	8	1702	362	3730	289	388
2	6	1628	258	3766	394	425

*Nota.* Resultados obtenidos variando los umbrales de activación (TH\_ON\_LOW y TH\_ON\_HIGH). Se analizaron 6471 archivos en total.

La efectividad de esta lógica se ilustra visualmente en la Figura 6. Allí se aprecia cómo

Figura 6

Visualización del funcionamiento del detector en diferentes escenarios. Las filas muestran desde ausencia de eventos (arriba) hasta detecciones múltiples (abajo). Las áreas sombreadas en rojo indican el momento exacto donde el algoritmo "decidió" que había un sismo.



Nota. Se visualiza el primer canal disponible de cada registro para verificar la precisión de los tiempos de corte.

el algoritmo discrimina correctamente entre una señal vacía, un sismo limpio y registros con ruido complejo. Una vez validadas, las señales se recortan a una duración estándar de 60 segundos, siguiendo las normas internacionales (Mousavi y cols., 2019).

El proceso finaliza consolidando más de 11,000 trazas individuales en un archivo unificado y estructurado (.hdf5), listo para el análisis avanzado (El diagrama de flujo del algoritmo se encuentra en el Anexo 4).

La implementación exitosa de este detector autónomo, capaz de autocalibrarse según el ruido, materializa el **segundo objetivo específico**, entregando una herramienta robusta y escalable para la gestión de datos sísmológicos a partir de **implementar y automatizar un pipeline de procesamiento de datos sísmológicos**.

### ***3.2.3. Generación y Validación del Dataset de Ruido***

Para entrenar modelos de aprendizaje automático robustos, es fundamental contar no solo con ejemplos de sismos, sino también con ejemplos claros de no-sismos (ruido). Para conformar este conjunto de datos, se reutilizó la infraestructura de adquisición automatizada, aplicando una estrategia de *desplazamiento temporal*.

La lógica implementada consiste en extraer ventanas de tiempo correspondientes a los minutos previos al inicio de cada evento sísmico catalogado por el SGC SGC (2025). Esta ventana temporal garantiza la obtención de registros de ruido ambiental natural, teóricamente libres de ondas sísmicas significativas.

Para asegurar la pureza de estos datos, se aplicó un estricto control de calidad utilizando el mismo algoritmo detector (STA/LTA) descrito anteriormente. Este actuó como un filtro de

validación:

- Se procesaron los archivos potenciales buscando cualquier activación o disparo.
- Se descartaron automáticamente aquellos registros donde el algoritmo detectó alguna anomalía o evento imprevisto.
- Se conservaron únicamente las trazas clasificadas con **cero eventos**.

Como resultado de este proceso de depuración, se consolidó un subconjunto final de 5397 trazas de ruido validadas, asegurando que el *dataset* de entrenamiento cuente con ejemplos negativos de alta fidelidad (Ver Tabla 5).

**Tabla 5**

*Resumen de detección de eventos en archivos de RUIDO analizados*

<b>Total archivos</b>	<b>0 eventos</b>	<b>1 evento</b>	<b>2 eventos</b>	<b>3 eventos</b>	<b>≥ 4 eventos</b>
6463	5397	537	452	62	15

*Nota.* Para el dataset de ruido se consideran únicamente las trazas con cero eventos detectados.

### 3.3. Análisis Energético y Calibración de Parámetros

Para garantizar la efectividad del detector STA/LTA, no se aplicaron umbrales genéricos, sino que se realizó una caracterización previa de la energía sísmica específica de la red REDNE. Utilizando las trazas del directorio `D_REDNE_mseed_tendencia_media`, se analizó la distribución de amplitudes calculando el valor promedio del valor absoluto de cada evento y su mediana por estación.

Este estudio cuantitativo se presenta detalladamente en el **Anexo 1**, donde las gráficas permiten visualizar el comportamiento del ruido frente a los eventos reales. Dicho análisis fue determinante para establecer los niveles de disparo, evidenciando que los parámetros teóricos sugeridos por Trnkoczy (1999) requerían ajustes específicos según la relación señal-ruido de cada estación.

En consecuencia, las conclusiones derivadas del Anexo 1 justificaron la selección del esquema de umbrales adaptativos (3, 4 y 6) descrito en la metodología, asegurando así un balance óptimo entre la sensibilidad de detección y el rechazo de falsos positivos.

### **3.4. Base de datos local**

La organización jerárquica de los archivos procesados, junto con la definición de metadatos asociados a cada traza, permite **estructurar una base de datos sísmológica inspirada en el Stanford Earthquake Dataset (STEAD)**, cumpliendo así con el **tercer objetivo específico** planteado.

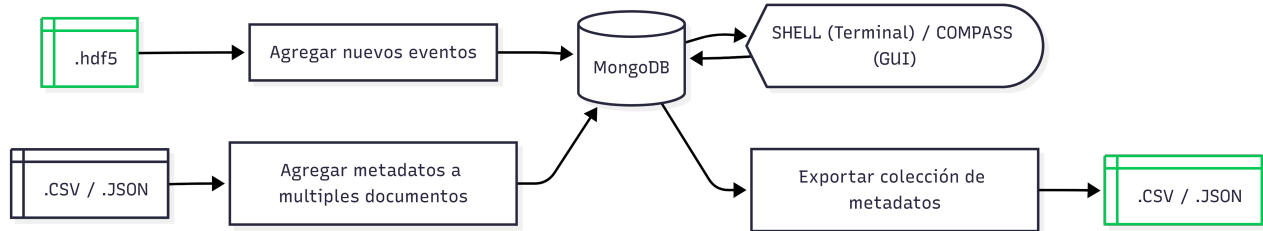
La gestión de la información generada en este proyecto no se concibe como un proceso estático, sino como un sistema dinámico que permite la incorporación continua de nuevos datos, su actualización y posterior consulta. En este contexto, la base de datos local actúa como un componente central que organiza y preserva los metadatos asociados a los eventos sísmicos y a las trazas de ruido procesadas.

La Figura 7 presenta de manera general el flujo de operaciones de la base de datos, mostrando cómo los eventos procesados en formato HDF5 son incorporados al sistema, almacenados de forma estructurada y posteriormente consultados o exportados para su análisis. Este enfoque facilita el acceso a la información tanto a través de herramientas gráficas como mediante interfaces de línea de comandos, así como su interoperabilidad con otros formatos estándar utilizados en el

análisis de datos.

**Figura 7**

*Flujo de operaciones de la base de datos local*



*Nota.* Representa el ciclo de vida de los metadatos, desde la incorporación de nuevos eventos en formato HDF5 hasta su consulta y exportación en formatos estándar.

### 3.4.1. MongoDB

Para el almacenamiento y gestión de los metadatos se seleccionó MongoDB como sistema gestor de bases de datos, debido a su flexibilidad y a su capacidad para manejar información semi-estructurada. Estas características resultan especialmente adecuadas para datos sísmológicos, cuyo conjunto de atributos puede ampliarse o modificarse a medida que evoluciona el proyecto.

El diseño de la base de datos se apoya en el uso de patrones de organización que permiten estructurar de manera coherente los documentos almacenados. En particular, se adoptó un enfoque basado en atributos, el cual facilita la organización y consulta de documentos que comparten características comunes, como los metadatos asociados a eventos sísmicos o a registros de ruido.

Adicionalmente, se incorporó un esquema de versionado de documentos que permite conservar cambios históricos en los metadatos más relevantes. Este mecanismo proporciona trazabilidad sobre las modificaciones realizadas a lo largo del tiempo y contribuye a mantener un control

adecuado de la información almacenada. En el contexto de este proyecto, el versionado se diseñó bajo dos consideraciones principales:

- **Los documentos presentan un número limitado de versiones**, dado que los metadatos principales se mantienen estables y solo se actualizan de forma ocasional.
- **Las consultas se realizan principalmente sobre la información más reciente**, lo que garantiza un acceso eficiente a los datos actualizados sin perder el historial de cambios.

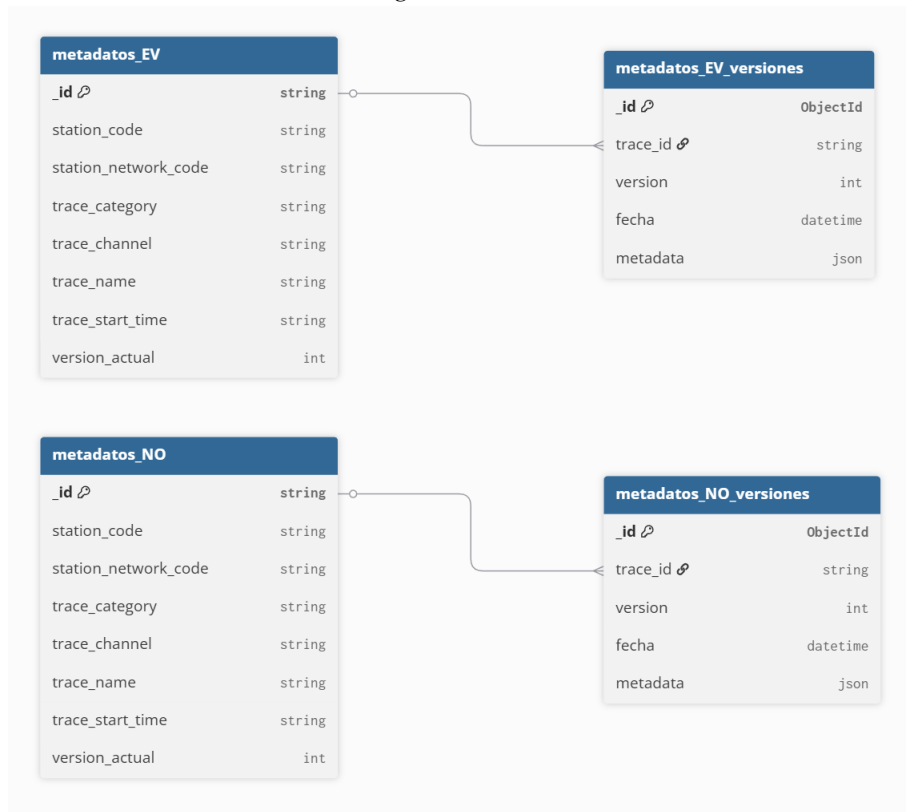
Con el fin de preservar la integridad y seguridad de la información, se implementó un esquema de control de acceso basado en roles. Este esquema separa las funciones administrativas del uso operativo de la base de datos, permitiendo restringir las operaciones de escritura a los procesos automáticos de ingestión de datos, mientras que las consultas quedan disponibles para usuarios autorizados. Esta estrategia reduce el riesgo de modificaciones accidentales y mejora la trazabilidad del sistema.

MongoDB actúa como gestor de la base de datos local **REDNE**, en la cual se almacenan los metadatos extraídos de los archivos `REDNE_plantilla.hdf5` y `REDNE_plantilla_noise.hdf5`. Estos metadatos se organizan en dos colecciones principales, correspondientes a eventos sísmicos y a trazas de ruido, respectivamente.

De manera complementaria, la base de datos incluye colecciones adicionales destinadas al almacenamiento de versiones históricas de los documentos. Esta estructura permite separar la información actual de los registros históricos, manteniendo la coherencia del sistema y facilitando su mantenimiento. La relación entre estas colecciones se ilustra en la Figura 8.

**Figura 8**

*Esquema relacional de colecciones en MongoDB*



*Nota.* Esquema general de la organización de los documentos para eventos y ruido, así como su relación con las colecciones destinadas al control de versiones.

El uso de mecanismos de indexación permite optimizar el acceso a la información más consultada, mejorando el rendimiento de las búsquedas sin afectar la flexibilidad del modelo de datos. Estas optimizaciones resultan especialmente relevantes a medida que el conjunto de datos crece en volumen y complejidad.

Finalmente, el diseño modular del pipeline de procesamiento y la estructura de la base de datos diseñada facilitan la incorporación sistemática de nuevos registros y la actualización conti-

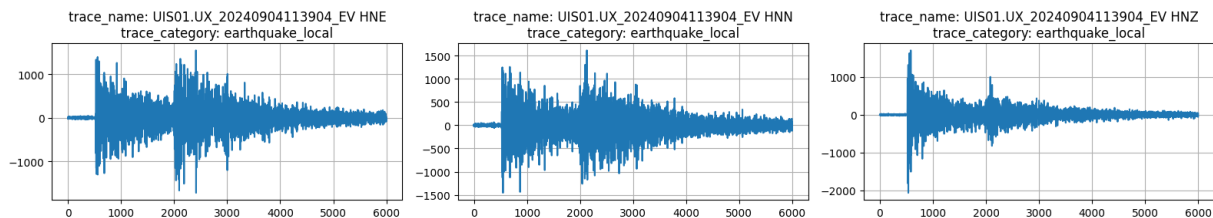
nua del conjunto de datos. Esta arquitectura asegura la reproducibilidad del proceso, favorece la escalabilidad del sistema y contribuye al cumplimiento del **cuarto objetivo específico, integrar y escalar un sistema automatizado para la actualización continua del dataset.**

### 3.5. Metadatos

Los metadatos describen propiedades externas de los eventos sísmológicos: la estación de registro, la fecha y hora de adquisición y el tipo de sensor o canal de medición, lo que permite su correcta interpretación y clasificación (ver Figura 9; otros metadatos se presentan en el Anexo 2).

#### Figura 9

*Visualización de un evento sísmico con sus metadatos asociados*



*Nota.* Visualización de las tres componentes de un evento sísmico de la estación UIS01. Cada traza se encuentra etiquetada con su identificador (`trace_name`) y su categoría, evidenciando la vinculación directa entre la señal registrada y sus metadatos almacenados en la base de datos.

- **Trace\_name:** Se considera como un identificador único para la traza con la siguiente nomenclatura: `<station_code>.<station_network_code>_YYYYMMDDhhmmss_<EV o NO>`
- **Trace\_start\_time:** Se establece el inicio de la traza siguiendo la norma ISO 8601:2004.
- **Trace\_channel** Canal del que se obtienen los datos sin el identificador del componente.
- **Trace\_category:** Se refiere a la categoría que pertenece la traza: terremoto o ruido. La base

de datos los eventos serán catalogados como (earthquake\_local) y el ruido como (noise).

- **Station\_code** Código de la estación referida al instrumento (ver columna Estación tabla 1).
- **Station\_network\_code:** Código de la red de los sensores o instrumentos .
- **Station\_latitude\_deg:** Latitud del sensor o instrumento en grados.
- **Station\_longitude\_deg:** Latitud del sensor o instrumento en grados.
- **Station\_elevation\_m:** Latitud del sensor o instrumento en metros.

#### 4. Conclusiones

Se desarrolló un proceso metódico para la recopilación y limpieza de datos sísmicos provenientes de la REDNE. Este procedimiento abarcó la conexión a la fuente de almacenamiento de los datos de la REDNE, la adquisición de trazas en formato MiniSEED y la verificación de los metadatos asociados. Como resultado se obtuvo un conjunto de datos con metadatos en un archivo hdf5.

Se diseñó una base de datos sísmológica inspirada en la configuración del Stanford Earthquake Dataset (STEAD), ajustando su esquema conceptual a las necesidades de una base de datos para la REDNE, esta base incluye metadatos de trazas sísmicas y ruido, utilizando colecciones y relaciones entre documentos que poseen consistencia, trazabilidad y escalabilidad. La estructura final permite una fácil consulta, almacenamiento y organización de la información sísmica.

Se estableció un flujo de trabajo automatizado para el procesamiento que combina la descarga de trazas, la conversión de formatos, la normalización y el almacenamiento final. Este sistema automatizado reduce el tiempo de procesamiento manual y consolida el tratamiento de los datos sísmicos, además del funcionamiento automático, ofrece la reproducibilidad del proceso.

Se incorporó un sistema automatizado que puede actualizar de manera continua el conjunto de datos a medida que se almacenan nuevos eventos sísmológicos en la REDNE. Esta automatización incluye la detección regular de eventos nuevos, gestión interna de versiones y actualizaciones incrementales, asimismo, se aseguró que el sistema pueda escalar permitiendo la inclusión de nue-

vas estaciones, canales o formatos sin grandes modificaciones estructurales; esto garantiza que el conjunto de datos se mantenga actualizado y en crecimiento, alcanzando así los objetivos específicos planteados.

Durante la aplicación del método STA/LTA, se emplearon parámetros de referencia comunes, sin embargo, por medio de la experiencia y un análisis cuantitativo de promedios y medianas, se determinó que la elección adecuada de las ventanas temporales y umbrales está influenciada por diversos factores propios de cada estación como el nivel de ruido, el tipo de instrumento, la distancia a la fuente y las condiciones ambientales. Dado que estas variables varían con el tiempo, resulta desafiante establecer valores universales y permanentes para todos los casos, por lo que la calibración precisa del método STA/LTA debe tratarse como un proceso dinámico y específico para cada estación.

El proyecto permitió la creación de una base de datos sísmológica organizada y un flujo de trabajo automatizado para el procesamiento y actualización constante de trazas con eventos y ruido, junto con un almacenamiento estandarizado de metadatos. Estos logros refuerzan el impacto académico del proyecto al facilitar investigaciones futuras y fomentar el desarrollo de herramientas sísmológicas basadas en aprendizaje automático. Esta solución moderniza la gestión de información sísmica en el grupo de investigación CPS, estandariza procesos y mejora la eficiencia y trazabilidad del sistema; contribuye al análisis de la sismicidad regional en Santander con datos de la REDNE y su alineación con estándares como STEAD y SeisBench promueven la interoperabilidad y posibles colaboraciones científicas internacionales.

## 5. Recomendaciones

1. Avanzar hacia una arquitectura de adquisición de datos que no dependa exclusivamente del Servicio Geológico Colombiano (SGC), sino que provenga directamente de la infraestructura instrumental propia de la REDNE, incorporando el uso de los triggers y thresholds en los sensores para la detección y almacenamiento automático de señales.
2. Explorar e implementar nuevas metodologías de análisis y clasificación automática de eventos sísmicos, complementando el uso del algoritmo STA/LTA con enfoques basados en filtrado adaptativo y aprendizaje automático.
3. Establecer ciclos de actualización mensual para la consulta y procesamiento de datos externos, considerando que en Santander se registran en promedio entre 80 y 100 eventos sísmicos al mes, lo que permite optimizar recursos y mejorar el rendimiento del pipeline.
4. Integrar el análisis de trazas de ruido con el fin de mejorar la calidad de las trazas con eventos sísmicos, aplicar técnicas de limpieza de ruido y fortalecer la detección de eventos sísmicos de baja energía.
5. Ampliar el conjunto de metadatos incorporando parámetros como mecanismos focales, magnitudes y localización detallada de eventos, para enriquecer el valor científico de la base de datos y fortalecer su aplicabilidad académica y profesional.

## Referencias

- Coupal, D., y Alger, K. W. (2023, octubre). *Construyendo con patrones: un resumen*. <https://www.mongodb.com/blog/post/building-with-patterns-a-summary-es>. (Accedido en noviembre de 2025)
- CUSEisTut Project. (2025). *Stalita trigger algorithm*. <https://cuseistut.readthedocs.io/en/latest/stalita/index.html>. (Proyecto “Developing hands-on micro modules of seismological data processing”, desarrollado por ABBAS Aqeel et al., soportado por la Teaching Development and Language Enhancement Grant (TDLEG) de CUHK. Accedido en octubre de 2025)
- Folk, M., Heber, G., Koziol, Q., Pourmal, E., y Robinson, D. (2011, 03). An overview of the hdf5 technology suite and its applications. *ACM International Conference Proceeding Series*, 36-47. doi: 10.1145/1966895.1966900
- Kinematics, Inc. (2017). *Etna 2 accelerograph datasheet*. <https://kinematics.com/wp-content/uploads/2017/04/datasheet-etna2-accelerograph-kinematics.pdf>. (Accedido en febrero de 2025)
- MongoDB, Inc. (2025a). *Authorization — mongodb manual*. <https://www.mongodb.com/docs/manual/core/authorization/>. (Accedido en noviembre de 2025)
- MongoDB, Inc. (2025b). *Indexes — mongodb manual*. <https://www.mongodb.com/docs/manual/indexes/>. (Accedido en noviembre de 2025)

- MongoDB, Inc. (2025c). *Mongodb official website*. <https://www.mongodb.com/>. (Accedido en noviembre de 2025)
- Mousavi, S. M., Sheng, Y., Zhu, W., y Beroza, G. C. (2019). Stanford earthquake dataset (stead): A global data set of seismic signals for ai. *IEEE Access*, 7, 179464-179476. doi: 10.1109/ACCESS.2019.2947848
- Observatorio Sismológico del Suroccidente Colombiano (OSSO). (2025). *Servicio web fdsnws dataselectwha*. <https://osso.univalle.edu.co/fdsnws/dataselect/1/builder>. (Accedido en febrero de 2025)
- ObsPy Development Team. (2024). *Trigger tutorial*. [https://docs.obspy.org/archive/master/tutorial/code\\_snippets/trigger\\_tutorial.html](https://docs.obspy.org/archive/master/tutorial/code_snippets/trigger_tutorial.html). (Accedido en 2025)
- Rasel, R. I., Sultana, N., Islam, G., Islam, M., y Meesad, P. (2019, 12). Spatio-temporal seismic data analysis for predicting earthquake: Bangladesh perspective. En *2019 research, invention, and innovation congress (ri2c)* (p. 1-5). doi: 10.1109/RI2C48728.2019.8999880
- Raspberry Shake Community. (2025). *Raspberry shake manual*. <https://manual.raspberrypi.org/>. (Accedido en febrero de 2025)
- SGC. (2025). *Catálogo de sismicidad*. [https://bdrsnc.sgc.gov.co/paginas1/catalogo/Consulta\\_Experta\\_Seiscomp/consultaexperta.php](https://bdrsnc.sgc.gov.co/paginas1/catalogo/Consulta_Experta_Seiscomp/consultaexperta.php). (Accedido en julio de 2025)
- Trnkoczy, A. (1999). *Understanding and parameter setting of sta/lta trigger algorithm*. Documento técnico, Kinemetrics SA. (Accedido en septiembre de 1999)
- Woollam, J., Münchmeyer, J., Tilmann, F., Rietbrock, A., Lange, D., Bornstein, T., ... Soto, H. (2022). Seisbench — a toolbox for machine learning in seismology. *Seismological Research*

*Letters*, 93(3), 1695–1709. Descargado de <https://seisbench.readthedocs.io/> (Accedido en octubre de 2025) doi: 10.1785/0220210324

## Apéndices

### Apéndice A. Análisis Energético y Depuración STA/LTA

En este apartado se presenta el análisis comparativo del conjunto de datos antes y después de la aplicación del algoritmo de disparo. Las Figuras 10, 11 y 12 muestran, en el panel izquierdo, la totalidad de las trazas tras la eliminación de tendencia y media; y en el panel derecho, el subconjunto resultante tras el filtrado STA/LTA. Visualmente, se evidencia una reducción en la densidad de puntos, lo que corresponde a la eliminación efectiva de trazas que contenían predominantemente ruido de fondo o señales no sísmicas.

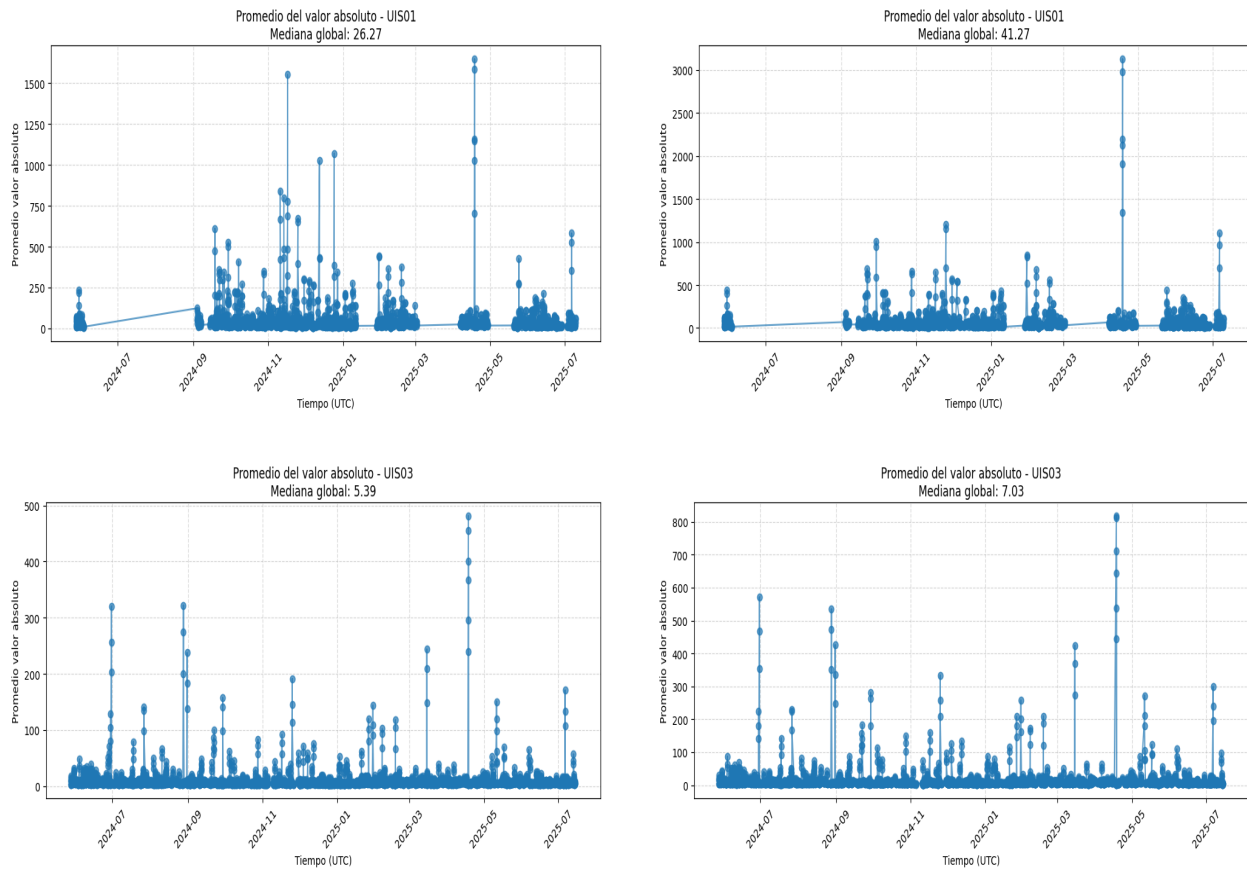
Este comportamiento se cuantifica en la **Figura 13**, donde se comparan las medianas globales del promedio de las amplitudes absolutas y la cantidad de trazas. Se observa un fenómeno clave: al aplicar el algoritmo, la **mediana de energía aumenta** mientras que la **cantidad de eventos disminuye**. Esto se debe a que el algoritmo descarta las señales de baja energía (ruido), conservando únicamente los eventos sísmicos claros que poseen amplitudes mayores. Por tanto, el aumento en la mediana es un indicador directo de la mejora en la calidad del conjunto de datos (mayor relación señal-ruido).

Un caso particular de análisis es la estación **UIS05** (ver Figura 11 y Figura 13). Los canales de aceleración (*UIS05\_acceleration*) muestran una reducción drástica en la cantidad de eventos detectados tras el procesamiento. Esto confirma que dichos canales registran un alto nivel de ruido ambiental o saturación, el cual es exitosamente filtrado por el algoritmo. En contraste, el canal de

velocidad (*UIS05\_velocity*) conserva una proporción mayor de eventos, demostrando una mayor sensibilidad y eficacia para el registro de sismos en comparación con los acelerómetros de la misma estación.

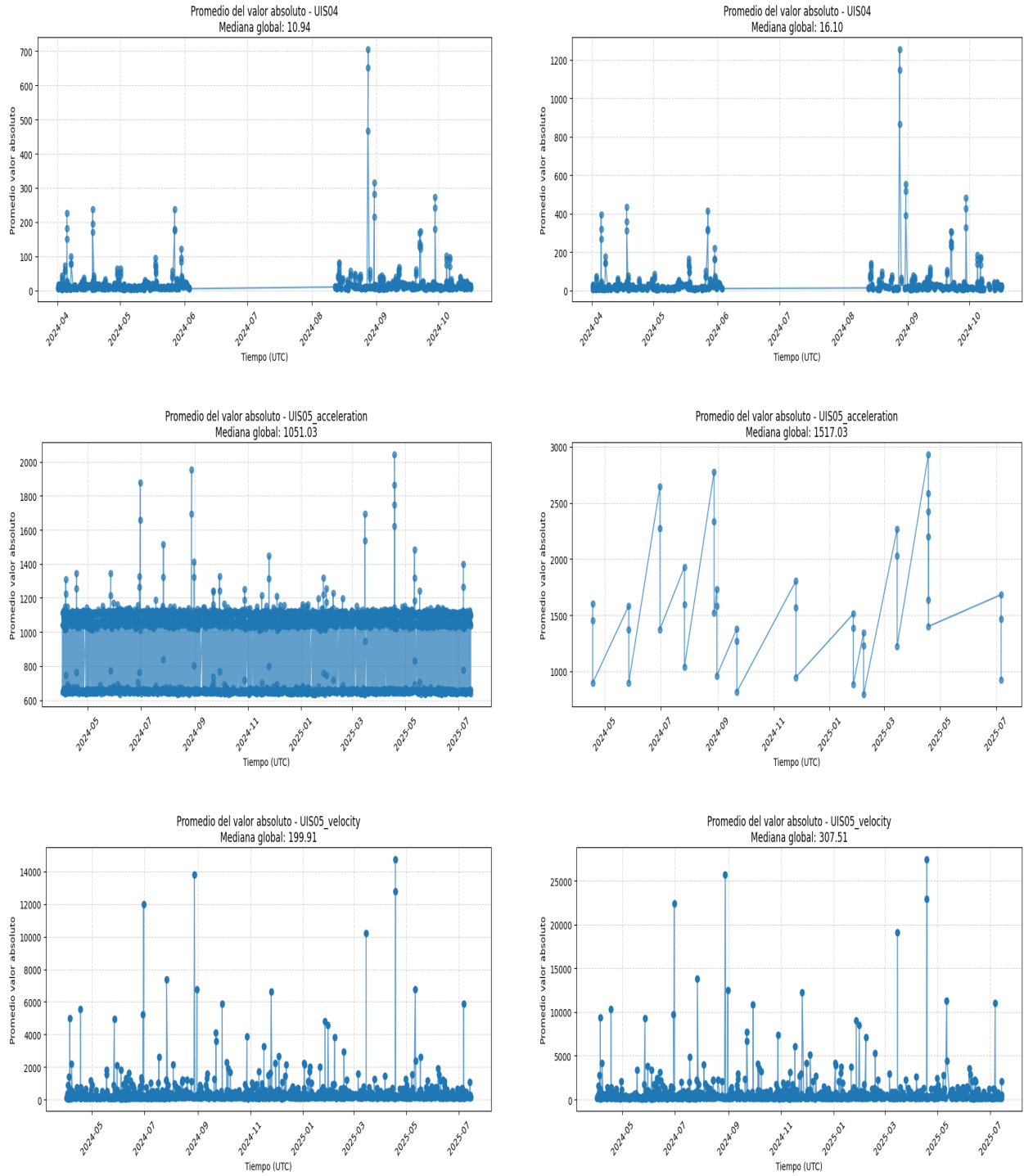
**Figura 10**

*Eventos a través del tiempo, estación UIS01 y UIS03*



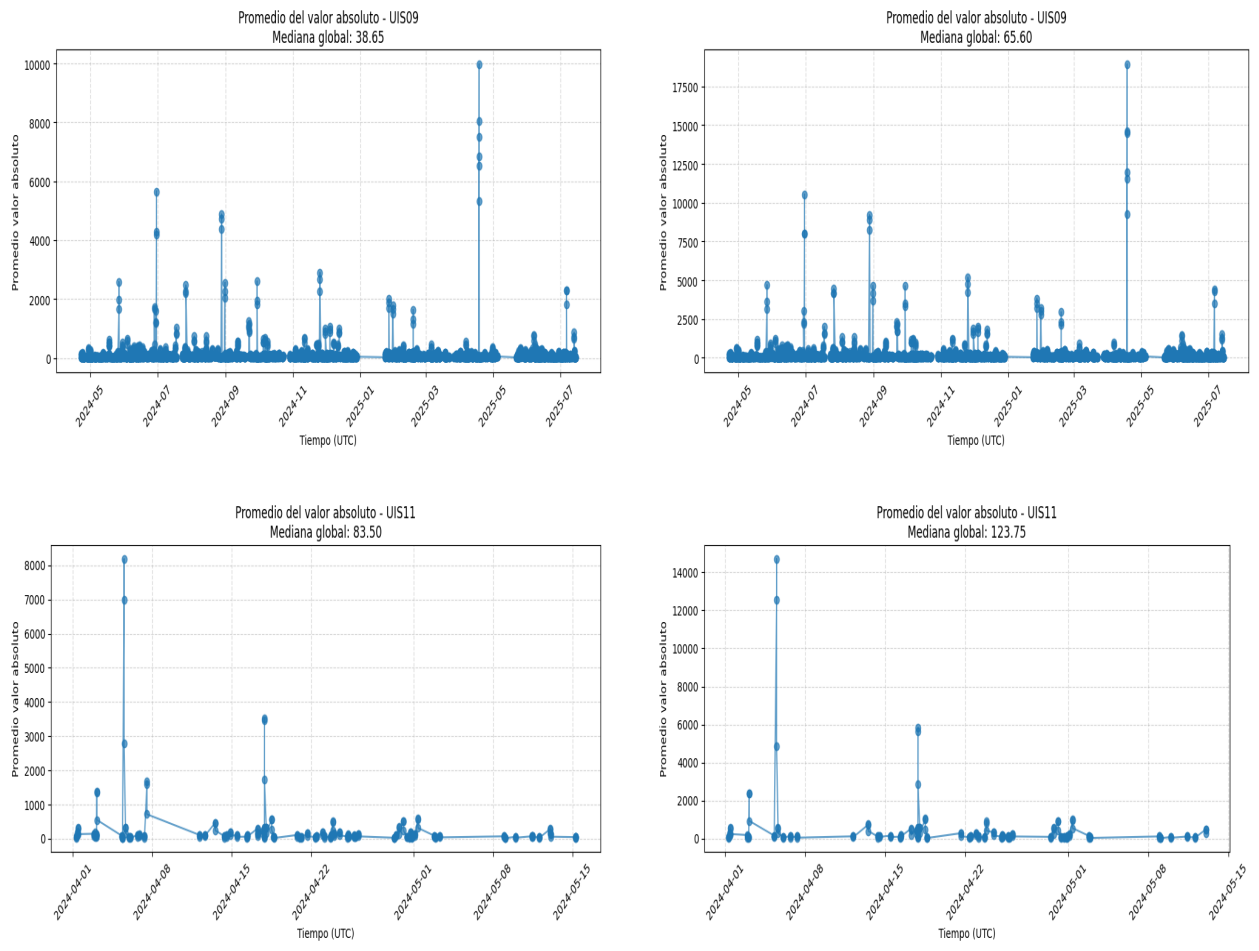
**Figura 11**

*Eventos a través del tiempo, estación UIS04 y UIS05*



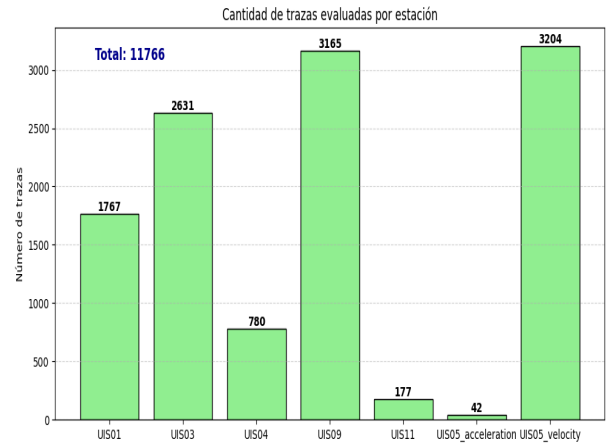
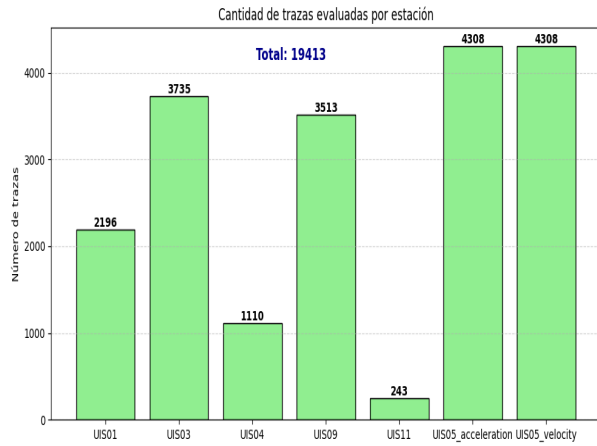
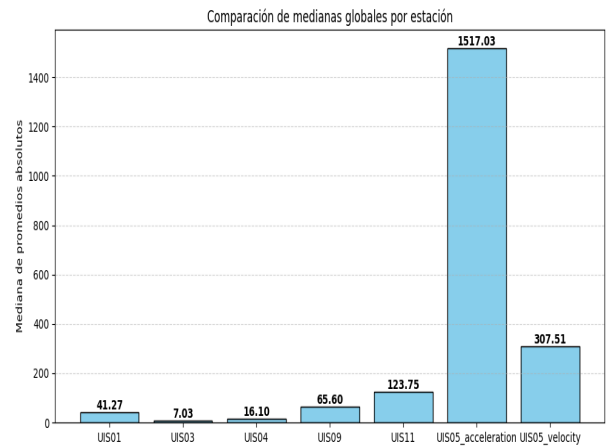
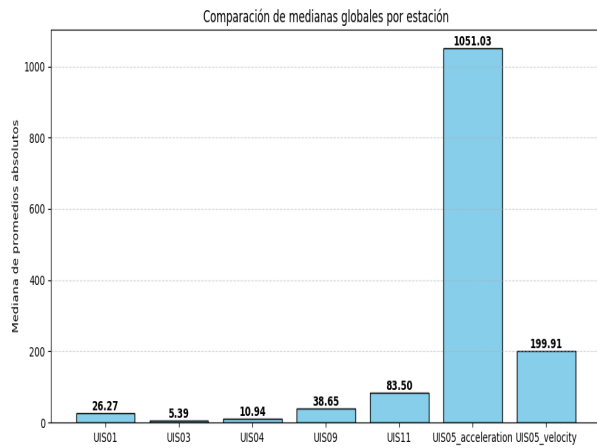
**Figura 12**

*Eventos a través del tiempo, estación UIS09 y UIS11*



**Figura 13**

*Gráficas Comparativas medianas y cantidades*



## Apéndice B. Otros metadatos

- **trace\_p\_arrival\_sample** Se muestra el momento en el que llega la fase P en la traza (en muestras).
- **trace\_p\_status** Evidencia la manera en que la fase P fué seleccionada, p. ej: manual/automático. El automatico está dado por un algoritmo de aprendizaje automático.
- **trace\_p\_weight** Es el factor de ponderación asignado a la selección de la fase P que oscila entre 0 y 1 donde 1 es el 100 %
- **trace\_s\_arrival\_sample** Se muestra el momento en el que llega la fase S en la traza (en muestras).
- **trace\_s\_status** Evidencia la manera en que la fase S fué seleccionada, p. ej: manual/automático. El automatico está dado por un algoritmo de aprendizaje automático.
- **trace\_s\_weight** Es el factor de ponderación asignado a la selección de la fase S que oscila entre 0 y 1 donde 1 es el 100 %
- **trace\_coda\_end\_sample** Se muestra el final de la señal sísmica (en muestras) basado en la envolvente de la serie temporal.
- **trace\_snr\_db** Relación señal-ruido de la traza en decibelios, se mide la snr por separado para cada componente como:

$$snr = 10 \log_{10} \left( \frac{\|S\|_2^2}{\|N\|_2^2} \right)$$

donde S y N son el percentil 95 de las amplitudes en una ventana corta antes de la hora de llegada de la fase P y después de la fase S (Mousavi y cols. (2019)).

- **source\_id** Es un número de identificación único proporcionado por la red de monitoreo que se puede utilizar para recuperar las formas de onda y los metadatos de los centros de datos sísmicos (Mousavi y cols. (2019)).
- **source\_origin\_time** Hora de origen de la fuente en formato UTC.
- **source\_origin\_uncertainty\_sec** Incertidumbre del tiempo de origen de la fuente dada en segundos.
- **source\_latitude\_deg** Coordenadas de latitud de origen dada grados.
- **source\_longitude\_deg** Coordenadas de longitud de origen dada grados.
- **source\_error\_sec** Error de asociación con la ubicación de origen dada en segundos.
- **source\_gap\_deg** Determinación de la brecha azimutal desde la fuente dada en grados.
- **source\_horizontal\_uncertainty\_km** Incertidumbre epicentral de la ubicación de la fuente en kilómetros.
- **source\_depth\_km** Profundidad de la fuente en kilómetros.
- **source\_depth\_uncertainty\_km** Incertidumbre de la profundidad de la fuente en kilómetros.
- **source\_magnitude** Asociación del valor de magnitud con la fuente.

- **source\_magnitude\_type** Tipo de cálculo de magnitud utilizado al asignar magnitud a la fuente.
- **source\_magnitude\_author** Autor del cálculo de magnitud.
- **source\_mechanism\_strike\_dip\_rake**

Mecanismo focal del evento sísmico calculado con base en el patrón del primer movimiento de ondas P, se expresa mediante Strike, Dip y Rake, describiendo la orientación de falla, ángulo de inclinación y dirección del deslizamiento respectivamente Mousavi y cols. (2019).
- **source\_distance\_deg** Se refiera la distancia calculada (el ángulo subtendido en el centro de la Tierra por el arco de la circunferencia máxima entre los dos puntos) (Mousavi y cols. (2019)) entre la fuente del evento sísmico y los instrumentos sismológicos en grados.
- **source\_distance\_km** Se refiera la distancia calculada (el ángulo subtendido en el centro de la Tierra por el arco de la circunferencia máxima entre los dos puntos) (Mousavi y cols. (2019)) entre la fuente del evento sísmico y los instrumentos sismológicos en kilómetros.
- **path\_p\_travel\_s** Tiempo que tarda en viajar la fase P desde su origen al instrumento en segundos.
- **path\_back\_azimuth\_deg** Azimut posterior de la trayectoria de fase desde la fuente al receptor en grados.

### Apéndice C. Entorno y repositorio

Se creó un entorno conda con librerías estándar y externas de python

**Tabla 6**

*Versiones de librerías utilizadas en el entorno Python 3.9.21*

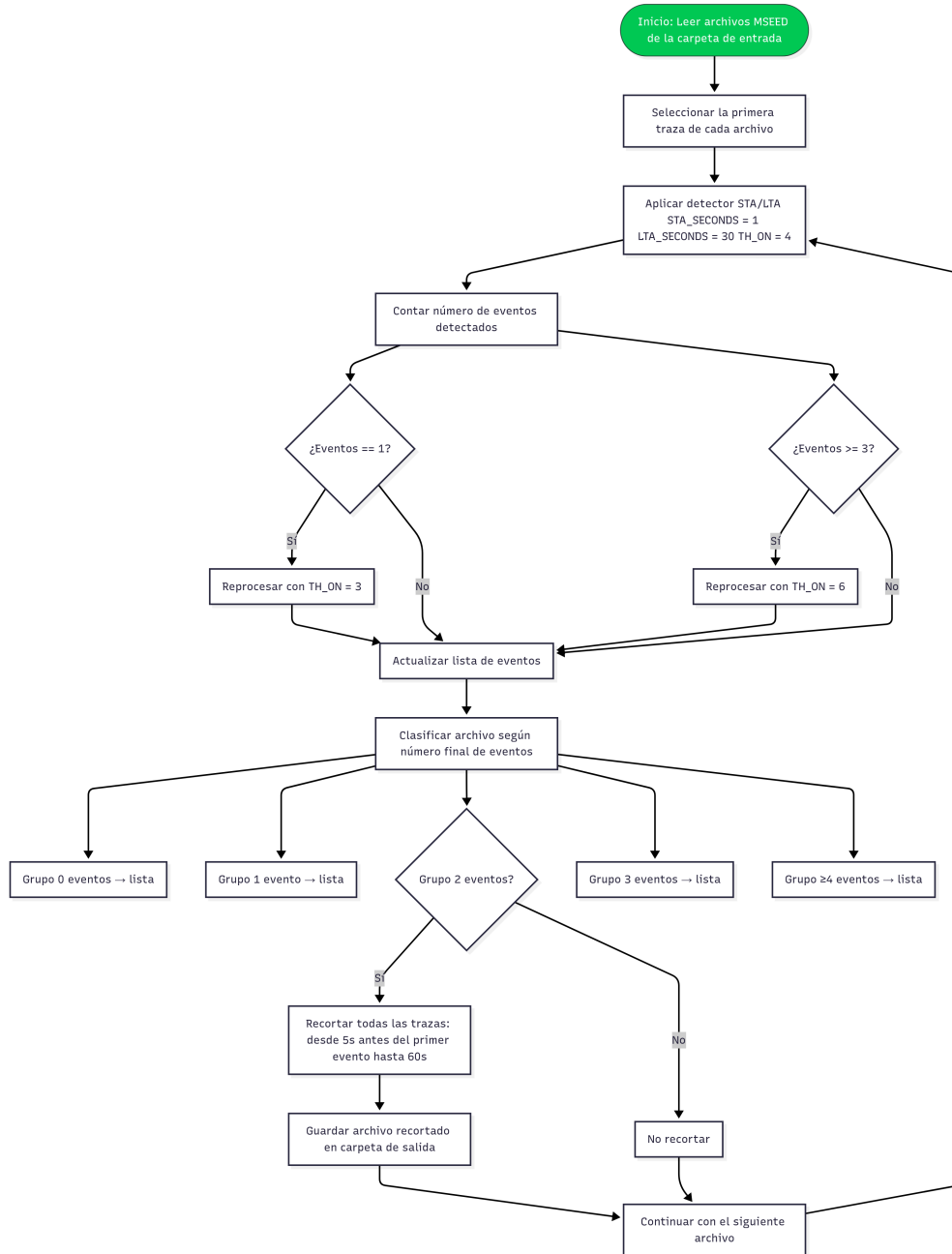
<b>Librería</b>	<b>Versión</b>	<b>Tipo</b>
numpy	2.0.2	Externa
pandas	2.3.3	Externa
obspy	1.4.2	Externa
h5py	3.14.0	Externa
requests	2.31.0	Externa
pymongo	4.13.0	Externa
os	–	Estándar Python
glob	–	Estándar Python
json	–	Estándar Python
re	–	Estándar Python
shutil	–	Estándar Python
datetime	–	Estándar Python
pathlib	–	Estándar Python
insertupdate	–	Módulo propio

*Nota.* Las librerías estándar de Python no poseen versión independiente, ya que dependen directamente de la versión del intérprete utilizado en este caso, todas corresponden a Python 3.7.16.

Repositorio github del trabajo: <https://github.com/Cahudisa/BADASI.git>

## Apéndice D. Diagrama de flujo algoritmo STA/LTA Figura 14

### Diagrama de flujo algoritmo STA/LTA



*Nota.* Se visualiza la evaluación de las trazas sísmológicas y su clasificación.