



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
MAESTRA EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

IMPLEMENTACIÓN DE TÉCNICAS DE APRENDIZAJE
AUTOMATIZADO PARA CLASIFICAR SISMOS
TECTÓNICOS Y EXPLOSIONES MINERAS EN EL
ESTADO DE ZACATECAS.

Tesis
QUE PARA OPTAR POR EL GRADO DE:
MAESTRIA EN MAESTRA EN CIENCIA E INGENIERIA DE LA COMPUTACIÓN

PRESENTA:
DELMA DÍAZ ALCÁNTARA

DR. LUIS MIGUEL DE LA CRUZ SALAS
Instituto de geofísica

Ciudad Universitaria, CDMX. Julio 2023



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

*Dedico este proyecto de tesis a mi familia y a todas las personas con quienes coincidí
alguna vez en la vida porque son un poquito de lo que hoy me tiene aquí.
Gracias...*

AGRADECIMIENTOS

Me gustaría empezar agradeciendo a mi asesor de tesis, el Dr. Luis Miguel de la Cruz Salas por brindarme su apoyo durante el desarrollo de este proyecto de tesis, que ha representado un reto para mi.

A la Dra. Xyoli Pérez Campos, al Dr. Víctor Hugo Espíndola Castro y al personal del Servicio Sismológico Nacional, quienes me apoyaron con la clasificación de los datos y discusión del proyecto.

Finalmente, agradezco al Conacyt por el apoyo económico y al Servicio Sismológico Nacional quién me facilitó el acceso a los datos utilizados en este proyecto. “Los datos del SSN fueron obtenidos por el Servicio Sismológico Nacional (Mexico), mantenimiento de las estaciones, adquisición de los datos y distribución son gracias a su personal”

RESUMEN

En los últimos años, la cantidad de información disponible ha crecido de manera exponencial y el campo de la sismología no es la excepción. Uno de los principales problemas también es la selección de información útil para los investigadores, por ejemplo, la discriminación de eventos con fuentes tectónicas y no tectónicas.

Atendiendo este problema, expertos en sismología han propuesto diversas técnicas estadísticas y cualitativas basadas en las características físicas del evento como los arribos de onda P y S. Actualmente, las técnicas de Aprendizaje Automático se han convertido en una de las mayores tendencias en el tratamiento de grandes volúmenes de información y buscan aprovechar la experiencia del analista.

Este trabajo de tesis propone clasificar una base de datos de sismos tectónicos y explosiones, utilizando modelos de aprendizaje automático para revisar las ventajas y desventajas.

Índice general

Índice de figuras	VII
1. Introducción	1
1.1. Objetivo general	2
1.1.1. Objetivos particulares	2
2. Marco teórico	3
2.1. Caracterización de la zona de Estudio	3
2.1.1. Sismicidad en el estado de Zacatecas	4
2.1.2. Actividad minera	5
2.2. Fundamentos del aprendizaje automatizado en la sis- mología	6
2.2.1. Antecedentes	8
2.2.2. Sismograma	9
2.2.3. Espectrograma	9
2.2.4. Clasificación de imágenes y aprendizaje au- tomático	11
2.3. Redes convolucionales	12
2.4. Elementos de una red convolucional	13
2.4.1. Operación de convolución	13
2.4.2. Capa convolucional	15
2.4.3. Capa de submuestreo	18
2.4.4. Capa <i>flatten</i>	18
2.4.5. Capa completamente conectada	19

2.4.6.	Función de activación	19
2.5.	Métodos de aprendizaje	21
2.5.1.	Descenso por gradiente	24
2.5.2.	Función de costo	26
2.5.2.1.	Entropía cruzada	27
2.5.2.2.	Error cuadrático	30
3.	Datos y metodología	31
3.1.	Datos	32
3.2.	Metodología	33
3.2.1.	Procesamiento de datos	33
3.2.1.1.	Prueba 1	35
3.2.1.2.	Prueba 2	36
3.2.1.3.	Prueba 3	36
3.2.1.4.	Prueba 4	37
3.2.2.	Arquitectura del modelo	37
4.	Resultados	39
4.1.	Prueba 1	39
4.2.	Prueba 2	40
4.2.1.	Prueba 3	42
4.2.2.	Prueba 4	43
4.3.	Análisis de los resultados	45
5.	Conclusiones y trabajo futuro	47

Índice de figuras

2.1. Mapa de ubicación del estado de Zacatecas.	4
2.2. Mapa de las principales minas en el estado de Zacatecas	7
2.3. Componentes de un evento sísmico registrado por la estación sismológica ZAIG, en el estado de Zacatecas el 16 de Julio del 2020.	10
2.4. Ejemplos de espectrogramas	11
2.5. Arquitectura básica de una red convolucional	14
2.6. a) Funciones originales, b) Operación de convolución y c) correlación cruzada	15
2.7. Operación de convolución entre una matriz 2D, cuadrícula en color rojo y un filtro con un paso igual a 1, en color amarillo	17
2.8. Proceso de aplanamiento de una matriz	19
2.9. Principales funciones de activación	21
2.10. Propagación hacia delante y hacia atrás	22
2.11. Propagación hacia delante	23
2.12. Propagación hacia atrás. En esta imagen E6 corres- ponde al error entre los valores predichos en la última capa a_6 y los esperados y_6 , E_{56} representa el error en- tre los valores de la capa 5 y 6, mientras que δ_6 hace referencia al gradiente del error E_{56}	25
3.1. Componentes de un evento sísmico registrado por la estación sismológica ZAIG, en el estado de Zacatecas el 16 de Julio del 2020.	32

3.2.	Espectrogramas de sismos.	34
3.3.	Espectrogramas de explosiones mineras.	34
4.1.	Matriz de confusión para el conjunto de prueba, y curvas de exactitud y pérdida, donde la curva color azul corresponde al conjunto de entrenamiento y la color negro al conjunto de validación. Prueba 1: Utilizando los espectrogramas de las tres componentes de cada evento	40
4.2.	Matriz de confusión para el conjunto de prueba, y curvas de exactitud y pérdida, donde la curva color rosa corresponde al conjunto de entrenamiento y la color naranja al conjunto de validación. Prueba 2: Utilizando los espectrogramas de las componentes N-S de cada evento	41
4.3.	Matriz de confusión para el conjunto de prueba, y curvas de exactitud y pérdida, donde la curva color morado corresponde al conjunto de entrenamiento y la color verde al conjunto de validación. Prueba 3: Utilizando los espectrogramas de las componentes E-O de cada evento	43
4.4.	Matriz de confusión para el conjunto de prueba, y curvas de exactitud y pérdida, donde la curva color naranja corresponde al conjunto de entrenamiento y la color negro al conjunto de validación. Prueba 4: Utilizando los espectrogramas de las componentes verticales de cada evento	44

Introducción

Uno de los principales retos en la sismología es clasificar correctamente cuál es el tipo de fuente sísmica que generó un evento, en dos tipos: natural o artificial. Se habla de una fuente natural cuando ocurre un mecanismo tectónico como deslizamientos o fracturas en la corteza y una fuente se dice artificial cuando agentes no tectónicos se ven involucrados; como es el caso de una explosión, actividad volcánica, entre muchos otros.

Sin embargo, identificar el tipo de fuente resulta una tarea muy compleja porque generalmente se realiza de manera visual, el porcentaje de sismos naturales suele ser mucho menor que el número de eventos de origen artificial y depende mucho de la experiencia humana.

Teniendo en cuenta lo anterior y hablando de los modelos computacionales que existen en la actualidad, podríamos pensar que es posible aprovechar la información disponible y la experiencia humana

para reducir el tiempo de clasificación. Un claro ejemplo son los trabajos de investigación realizados por Del Pezzo (2003); Yildirim et al., (2011); Linville et al., (2019); Tibi et al., (2019), quienes implementaron técnicas de aprendizaje automático como redes neuronales, análisis de componentes principales y *clustering* para clasificar eventos sísmicos con fuentes naturales o artificiales, logrando etiquetar correctamente más del 80 % de los eventos y al mismo tiempo una reducción en el tiempo de análisis.

1.1. Objetivo general

El objetivo principal de este proyecto de tesis es diseñar un 1 modelo con técnicas de aprendizaje automático que permitan clasificar un conjunto de registros sísmicos generados por fuentes naturales o artificiales en el estado de Zacatecas.

1.1.1. Objetivos particulares

- Crear una base de datos con espectrogramas de sismos y explosiones.
- Construir uno o varios modelos que clasifiquen el conjunto de datos, utilizando técnicas de aprendizaje automático.
- Comparar el rendimiento de los modelos y emitir recomendaciones.

Marco teórico

En este capítulo se realiza una introducción a los conceptos requeridos para el desarrollo de este proyecto de tesis.

2.1. Caracterización de la zona de Estudio

El estado de Zacatecas, se localiza en la parte central del norte de la República Mexicana, a una altitud media de 2,496 metros sobre el nivel del mar (msnm), entre las coordenadas $25^{\circ} 09'$ - $21^{\circ} 01'$ latitud norte y $100^{\circ} 48'$ - $104^{\circ} 20'$ longitud oeste. Tiene una superficie de $75,284 \text{ km}^2$ que equivale al 3.83% del territorio nacional, ocupando el 8vo lugar en extensión total. Colinda al norte con Coahuila, al noreste con Nuevo León, al este con San Luis Potosí, al sur con Aguascalientes y Jalisco y finalmente, al sureste con Nayarit, véase la figura 2.1. Está conformado por 57 municipios, un total de 4,882 localidades y la capital del estado es la ciudad de Zacatecas. Forma parte de las provincias: Sierra Madre Occidental, Sierra Madre

2. MARCO TEÓRICO

Oriental y Mesa del Centro.



Figura 2.1. Mapa de ubicación del estado de Zacatecas.

2.1.1. Sismicidad en el estado de Zacatecas

Un sismo ocurre con el rompimiento o deslizamiento de las placas tectónicas, debido al acumulamiento de la energía que se genera durante la interacción de las mismas. Por otro lado, el tamaño o magnitud de un sismo depende de cuánta energía se libera y es proporcional al tamaño de la zona de ruptura.

El estado de Zacatecas se ubica sobre la placa norteamericana,

mostrado en la figura 2.1, y se considera una zona de bajo potencial sísmico; dada su lejanía al contacto entre las placas del Pacífico y Cocos, donde ocurren la mayor cantidad de sismos importantes. Sin embargo, en este estado se tienen registros de eventos con magnitudes casi imperceptibles, conocidos como sismos intra placa porque ocurren en zonas lejanas a los límites de las placas tectónicas y en su mayoría son superficiales, menos de 35 kilómetros de profundidad.

El Servicio Sismológico Nacional (SSN) opera una estación sísmológica en la ciudad de Zacatecas (ZAIG), mostrada en la figura 2.1 y tiene por objetivo el registro continuo y permanente de la actividad sísmica de la zona.

2.1.2. Actividad minera

De acuerdo con el Servicio Geológico Mexicano, la Sierra de Zacatecas encierra grandes yacimientos de oro, plata, cobre, zinc y manganeso. Tradicionalmente es un estado minero desde la época de la colonia, cuando se fundaron las reales minas de San Martín, Sombrerete, Nieves y Fresnillo, permitiéndole tener un lugar dentro de los centros económicos más importantes de la época.

La minería en el estado representa una contribución enorme en el desarrollo económico, y aunado al incremento en la demanda de metales, se espera un crecimiento en la explotación minera.

De acuerdo con la Subdirección de Minería de la ciudad de Zacatecas, la zona se divide en 17 regiones mineras, delimitadas de acuerdo al tipo de mineralización y su ubicación. Actualmente, el estado tiene 12 minas activas de donde se extraen diferentes minerales como el oro. Así como, 17 minas inactivas que ya no se encuentran en funcionamiento y finalmente, 4 zonas mineralizadas en exploración, con potencial para extracción de minerales. La ubicación de estas minas se observa en la figura 2.2.

Algunas de las minas más importantes del estado son: Concepción del Oro, Mazapil, Fresnillo, Migue Auza, Ojocaliente, Sombrerete, Chalchihuites, Pinos y Zacatecas.

2.2. Fundamentos del aprendizaje automatizado en la sistmología

El aprendizaje automático es una rama de la inteligencia artificial que mediante la ingesta de grandes cantidades de datos, le permite a un sistema aprender y mejorar de manera autónoma. Este aprendizaje se puede dar de forma supervisada o no supervisada.

En medio de toda esta revolución tecnológica y viendo la enorme cantidad de información disponible, los expertos en ciencias de la Tierra comenzaron a implementar diversas técnicas de inteligencia artificial, en un intento por resolver problemas como reconocimiento

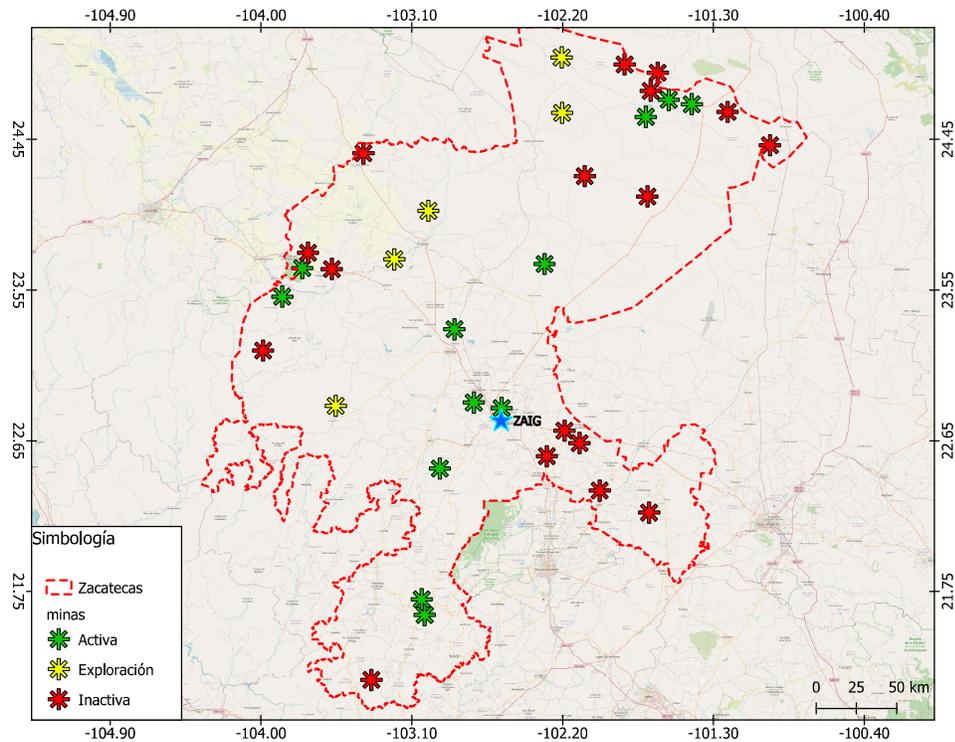


Figura 2.2. Mapa de las principales minas en el estado de Zacatecas

de arribos de onda P y S, clasificación de señales sísmicas, pronósticos del clima, procesamiento de imágenes satelitales, interpretación de propiedades geofísicas, entre muchos otros.

En este capítulo revisaremos conceptos que nos permiten entender cómo se clasifican señales sísmicas de origen natural y artificial, utilizando modelos de redes convolucionales y un conjunto de espectrogramas, que son imágenes generadas a partir de las series de tiempo registradas en un sismógrafo.

2.2.1. Antecedentes

En años recientes la inclusión de la inteligencia artificial en el campo de la sismología ha sido tema de investigación, debido a las diversas áreas de oportunidad. Una de estas es la clasificación de eventos sísmicos y explosiones por minería, lo cual ha sido explorado por diversos autores en distintas regiones del mundo. En estos trabajos se ha investigado la implementación de algunos modelos como redes convolucionales, redes recurrentes, entre otros.

Por ejemplo, Tibi et al., (2019) y Linville et al., (2019) diseñaron una red convolucional para clasificar más de 10,000 registros de sismos y explosiones mineras, en el estado de Utah. Utilizaron las tres componentes de un evento, juntas y por separado, logrando etiquetar correctamente más del 99% de los registros.

Por otro lado, Del Pezo et al., (2011) clasificó un conjunto de 280 sismos y explosiones, ocurridos bajo el agua. Utilizó una red de retropropagación y logró clasificar correctamente el 90% de los registros.

Finalmente, Yildirim et al., (2003) utilizó una red neuronal de retroalimentación, utilizando como entrada el cociente de los picos de amplitud de las fases espectrales, para clasificar eventos sísmicos y explosiones mineras. En sus resultados reportó haber etiquetado correctamente el 90% de los registros.

2.2.2. Sismograma

En la literatura, definen a un sismograma como el registro de la velocidad de un punto en suelo, en función del tiempo (Levinzon, (2017); el Servicio Geológico de Estados Unidos por sus siglas en inglés USGS)

Usualmente se representa en una gráfica donde el eje horizontal corresponde al tiempo que le toma a la onda llegar del punto de ruptura al sensor. Por otro lado, el eje vertical hace referencia a la velocidad del punto donde se ubica el sensor sísmico.

2.2.3. Espectrograma

El Servicio Geológico de los Estados Unidos, menciona que un espectrograma es una manera visual de representar el contenido en frecuencias de un sismograma en función del tiempo.

Para construir un espectrograma es necesario apilar los espectros de Fourier obtenidos de la serie temporal, al recorrer una ventana de muestra a lo largo del eje del tiempo. El eje horizontal corresponde al tiempo de registro y el vertical nos habla del rango de frecuencias que contiene la señal. La amplitud del espectro se muestra con una escala de colores.

Nos brinda información valiosa del tipo de fuente y medio en el cual ha viajado dicha onda, por esta razón es factible distinguir en-

2. MARCO TEÓRICO

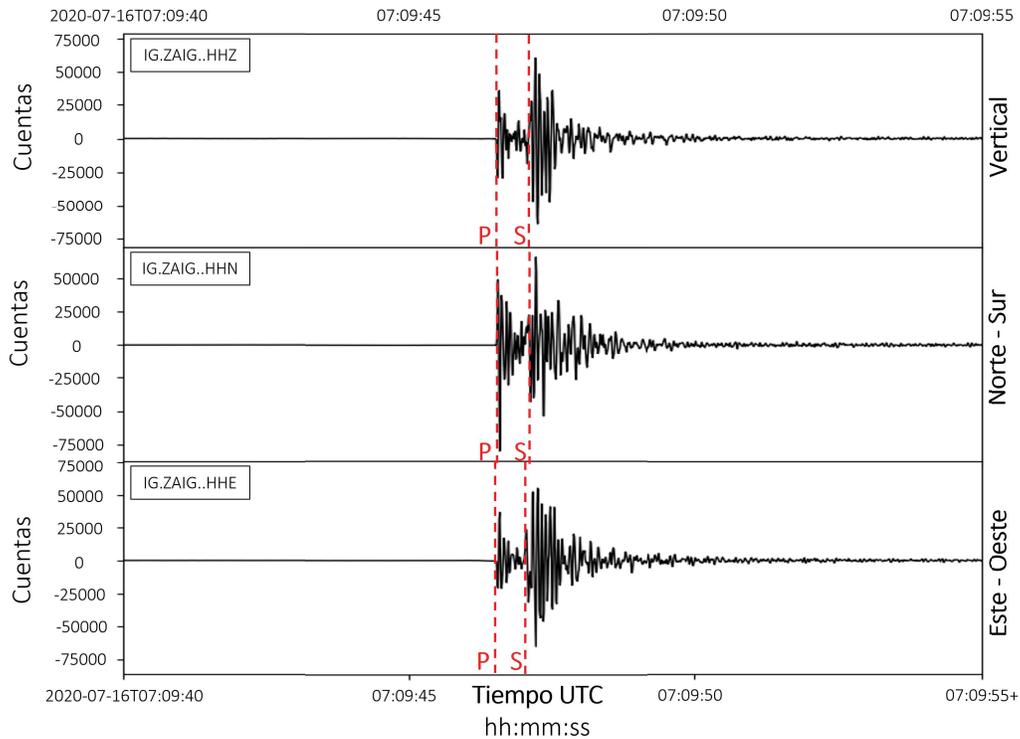


Figura 2.3. Componentes de un evento sísmico registrado por la estación sísmológica ZAIG, en el estado de Zacatecas el 16 de Julio del 2020.

tre un sismo generado por una fuente natural o una artificial.

En la figura 2.4 se observan dos espectrogramas. El primero corresponde a un sismo de origen natural y el segundo a una explosión minera, en ambos casos se aprecia el arribo de onda P (color amarillo intenso) y onda S (tonalidades azul claro-amarillo poco intenso).

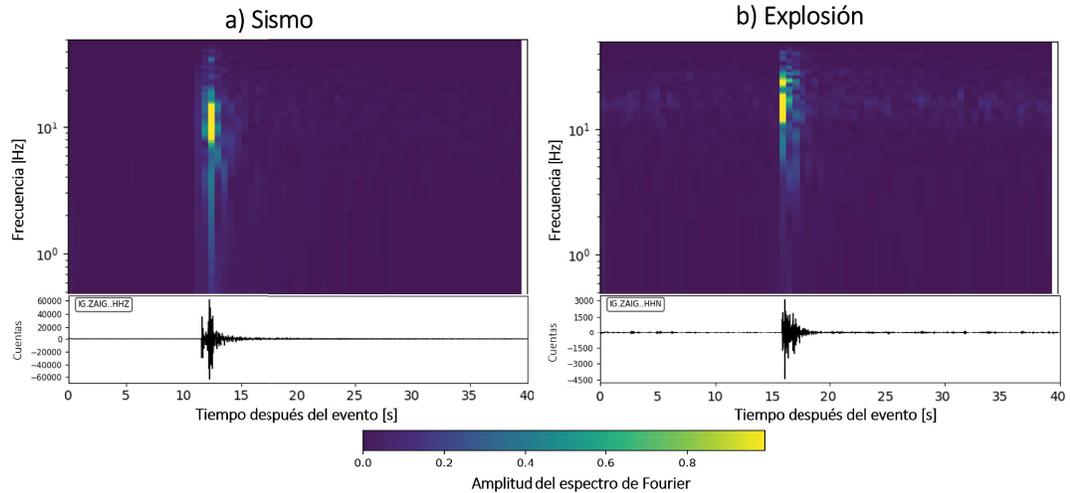


Figura 2.4. Ejemplos de espectrogramas

2.2.4. Clasificación de imágenes y aprendizaje automático

La clasificación de imágenes con técnicas de aprendizaje automático, consiste en el desarrollo de modelos capaces de identificar patrones, basados en ciertos grupos de píxeles que utiliza como ejemplos. Estas técnicas se dividen en dos categorías: aprendizaje supervisado y aprendizaje no supervisado.

Las técnicas de aprendizaje no supervisado utilizan un conjunto de imágenes para entrenar el modelo que no han sido etiquetadas previamente, es decir que estos modelos analizan y buscan patrones embebidos en el grupo de imágenes, sin ninguna intervención humana.

Por el contrario, las técnicas de aprendizaje supervisado emplean

un conjunto de entrenamiento previamente etiquetado, esto significa que el modelo aprende de una experiencia humana, extrae los patrones de las imágenes y etiqueta un nuevo conjunto de datos. Estas técnicas aprovechan el conocimiento de los analistas que clasifican, inicialmente, el conjunto de entrenamiento y le asignan una etiqueta a cada uno de los registros. Es en esta etapa donde la experiencia del analista se transmite al modelo para diseñar la estrategia de clasificación.

2.3. Redes convolucionales

Actualmente, el aprendizaje profundo es una de las técnicas más utilizadas en problemas de clasificación de imágenes, específicamente las redes convolucionales. Estos algoritmos están inspirados en la visión humana y es una de las técnicas de reconocimiento de patrones con mejores resultados. Su estructura principal se basa en capas o filtros que se ejecutan de manera consecutiva y conservan una estructura jerarquizada, buscando patrones más específicos en las capas de mayor profundidad mientras se realizan operaciones específicas sobre los mapas de características o cuadrículas de píxeles, como se puede observar en la figura [2.5](#).

Una de las principales ventajas con otras técnicas de aprendizaje automático, es que logran conservar la relación espacial entre

los grupos de píxeles, esto significa que una imagen reescalada, rotada o desplazada puede ser identificada fácilmente por una red convolucional porque logrará encontrar aquellos grupos que sean característicos de cada grupo de imágenes.

2.4. Elementos de una red convolucional

La arquitectura básica de una red convolucional consiste de al menos una capa convolucional, una capa de submuestreo (opcional), una función de activación y una capa completamente conectada (Haykin et al., 1998; Ketkar et al., 2017; Chollet et al., 2017; Manaswi et al., 2018; Nguyen et al., 2021). Las capas convolucionales y de submuestreo pueden repetirse varias veces y se utilizan para obtener los patrones más representativos del mapa de características o imagen. Por otro lado, en la figura 2.5, la capa de salida contiene las probabilidades de pertenecer a cada una de las clases objetivo.

2.4.1. Operación de convolución

La convolución (figura 2.6) es la integral del producto de dos funciones, y expresa que tanto se superponen ambas funciones. En esta operación es rotada con respecto al eje horizontal. Por ejemplo, cuando utilizamos una capa convolucional para identificar patrones, internamente deslizamos un *kernel* o filtro sobre un mapa

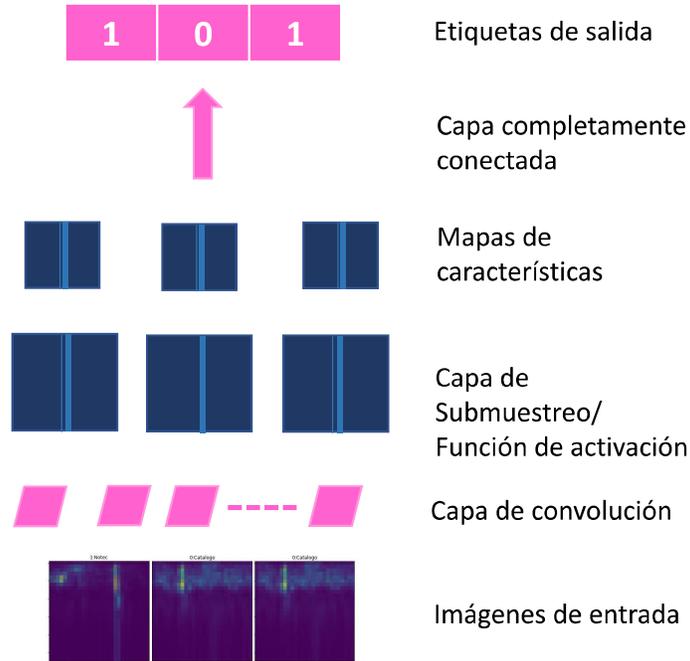


Figura 2.5. Arquitectura básica de una red convolucional

de características o cuadrícula, sumando el producto, elemento por elemento. Finalmente, los resultados formarán un nuevo mapa de características (Bracewell, R., 1965).

De manera formal podemos definir una matriz de entrada $I(a)$ y un *kernel* $K(a)$, donde a corresponde a un punto en el dominio de t . La operación convolución está dada por la siguiente ecuación (ec. 2.1):

$$s(t) = (I * K)(t) = \sum_a I(a) \cdot K(t - a). \quad (2.1)$$

La forma general de la operación convolución para dos dimensio-

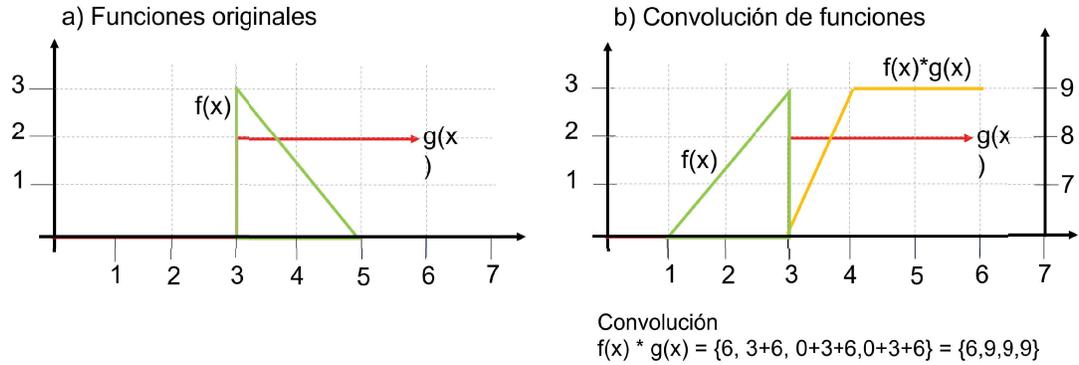


Figura 2.6. a) Funciones originales, b) Operación de convolución y c) correlación cruzada

nes m y n , donde a y b representan a un punto en dichos dominios. La ecuación quedaría de la siguiente manera (ec. 2.2):

$$s(t) = \sum_a \sum_b I(a, b) \cdot K(m - a, n - b). \quad (2.2)$$

2.4.2. Capa convolucional

Manaswi (2018) señala que la capa convolucional es el principal componente en este tipo de redes, se representa de la forma $y = f(s(x \cdot w))$, donde y corresponde al vector salida, x es el vector de entrada o conjunto de imágenes, w considera un conjunto de pesos derivados de los filtros, s denota la operación de convolución y f es la función de activación.

Estos filtros permiten mejorar la calidad de la imagen, nitidez, destacar bordes, entre muchas otras cosas. Suelen tener dimensiones más pequeñas que el mapa de características, a través del cual se

deslizan realizando una operación de convolución y resultando en nuevos mapas de características. La entrada y salida de cada una de las capas convolucionales es un volumen de mapas de características de tamaño $W \times H \times D$, donde W y H representan el largo y ancho, mientras que D es el número de canales de cada una.

Existen tres parámetros que controlan el tamaño y cantidad del volumen en la entrada y salida de una capa convolucional: 1) **La profundidad (N)**, que consiste en la cantidad de filtros implementados, el **tamaño del filtro (F)** y 2) **el paso (S)** que indica cómo será el movimiento del filtro sobre la imagen. Usualmente se utiliza un paso $F=1$ o $F=2$ para mover el filtro uno o dos píxeles. 3) El **relleno (P)**, establece la forma de manejar los bordes para conservar el tamaño de la imagen original. Si el valor del relleno es cero, las dimensiones podrían decrecer rápidamente y por ello normalmente se utiliza un valor de 1.

En la Figura 2.7 se presenta un ejemplo de la convolución de un espectrograma o matriz 2D y un *kernel* o filtro, en color amarillo.

Una vez declarados los tres parámetros, es importante revisar que la cuadratura entre parámetros sea correcta, utilizando la siguiente ecuación $((W - F + 2P) / S) + 1$. Si el resultado es un valor fraccional, podemos decir que los parámetros son incompatibles y, por tanto la conexión entre neuronas será incorrecta.

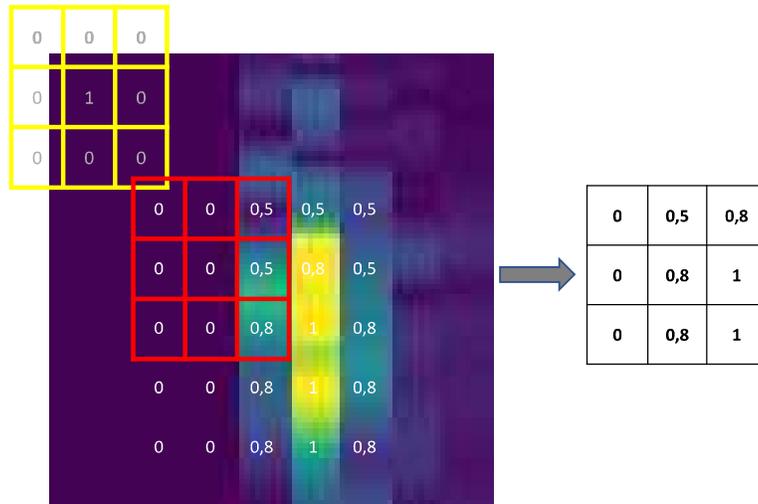


Figura 2.7. Operación de convolución entre una matriz 2D, cuadrícula en color rojo y un filtro con un paso igual a 1, en color amarillo

Por ejemplo, si pensamos en una imagen de 224×224 píxeles, con filtros de 11×11 , un paso de 4 y relleno igual a cero, tenemos que la ecuación $((224 - 11 + 2(0))/4) + 1 = 54.25$ y por tanto la conexión es incompatible pero si la imagen tiene 227×227 píxeles, la ecuación $((227 - 11 + 2(0))/4) + 1 = 55$ y esto nos dice que la conexión es correcta.

En resumen, una capa convolucional acepta un volumen de tamaño $W_{entrada} \times H_{entrada} \times D_{entrada}$; comúnmente se utiliza que W y H sean iguales. Los parámetros requeridos como entrada son: el número de filtros, tamaño del paso y relleno. Mientras que la salida será $W_{salida} \times H_{salida} \times D_{salida}$, donde:

- $W_{salida} = (W_{entrada} - F + 2P)/S + 1,$

- $H_{salida} = ((H_{entrada} - F + 2P)/S + 1,$
- $D_{salida} = N.$

2.4.3. Capa de submuestreo

Las capas de submuestreo permiten reducir la cantidad de información en la salida de una capa, dado que el costo computacional aumenta.

Existen distintas formas de realizar un submuestreo. La primera es utilizando un parámetro de paso mayor que 1 y de tal manera que el desplazamiento del filtro sea más grande; sin embargo, esto podría ignorar información importante.

La segunda manera, es mediante la implementación de capas de submuestreo que suelen reducir el tamaño de un *kernel* de atributos mediante la selección del máximo, mínimo o promedio. Lo anterior también es posible mediante la selección de atributos por su ubicación geográfica dentro del *kernel*.

2.4.4. Capa *flatten*

La información saliente de las capas convolucionales es un conjunto de cuadrículas o tensor que suele convertirse en un vector antes de ingresar en la regresión y obtener las probabilidades de pertenencia a las clases objetivo, mediante una capa de aplanamiento *flatten*. Una desventaja de utilizar este tipo de técnicas es que se destruye

la estructura espacial de los mapas.

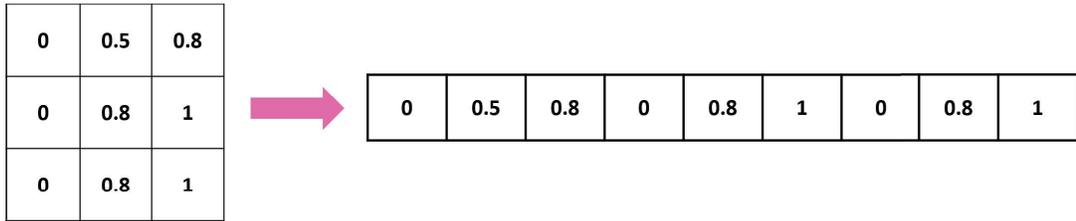


Figura 2.8. Proceso de aplanamiento de una matriz

2.4.5. Capa completamente conectada

Las capas completamente conectadas, suelen ser el último paso en una red convolucional y tienen el objetivo de clasificar cada imagen, mediante los atributos obtenidos de las capas anteriores. La salida es el conjunto de probabilidades de pertenecer a cada una de las categorías. El término “completamente conectada” hace referencia a la conexión entre cada uno de los nodos de la capa actual y la capa anterior.

2.4.6. Función de activación

El término “función de activación” tiene como base teórica el proceso de sinapsis que realizan las neuronas del cerebro humano, simulando un circuito que activa o desactiva una conexión, en función de un criterio previamente definido por el tipo de función y tiene el objetivo de reducir la cantidad de información que pasa a la siguiente capa (Wang et al., 2020; Lederer et al., 2021).

Son funciones no lineales y monótonas porque no es completamente creciente ni decreciente, como se observa en la figura 2.9. Este tipo de funciones permiten que el modelo se adapte mejor y trabaje con la mayor cantidad de datos.

1. Sigmoide o logística. Se llama así porque es simétrica con respecto a 0.5. En esta función la salida toma valores entre 0 y 1, de manera similar a una distribución Bernoulli en y , condicionado por x . Sin embargo el rango de valores no está centrado en el cero pero se resuelve utilizando una función tangente hiperbólica.
2. Función Softmax. Es una versión más generalizada y llamativa para resolver clasificaciones multiclase, donde la salida representa la probabilidad de pertenecer a cada una de las clases objetivo y comúnmente se emplea en la última capa de la red.
3. Función ReLu. Se utiliza en las capas ocultas, es medio lineal. Tiene un rango de 0 al infinito. El principal problema es que automáticamente convierte a todos los valores negativos en ceros, disminuyendo la habilidad de ajustar valores negativos de manera apropiada.

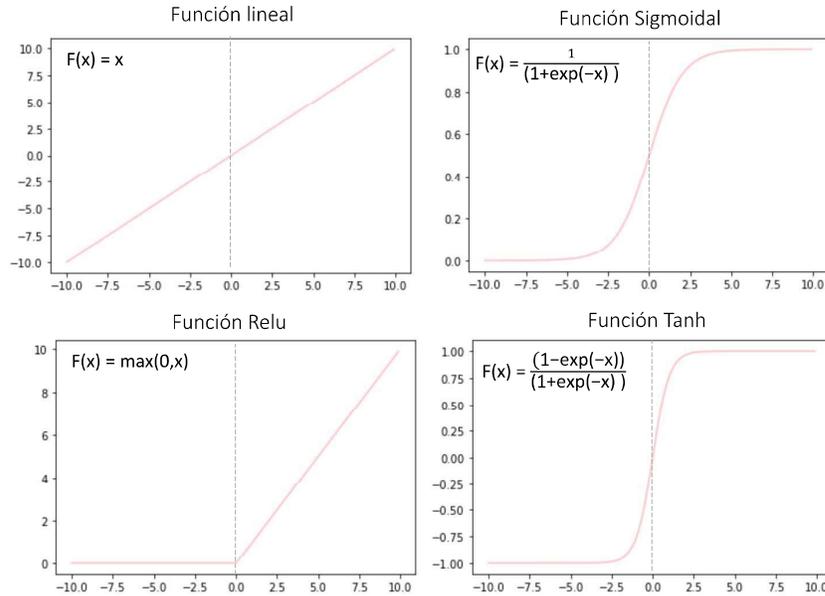


Figura 2.9. Principales funciones de activación

2.5. Métodos de aprendizaje

Diversos autores como Rojas, (1996), Haykin et al., (1998), Ketkar et al., (2017), Manaswi et al., (2018) y Nguyen et al., (2021) definen que un modelo de aprendizaje profundo consiste en un conjunto de capas (I), caracterizadas por una entrada (x), un conjunto de filtros (K), y sesgos (b) que ayudan a balancear las ecuaciones.

La entrada x_i^I , donde i se refiere a las neuronas en la capa I , queda representada por la siguiente ecuación 2.3:

$$x_i^I = \sum_j w_{i,j}^I * a_{I-1,j} + bi^I, \quad (2.3)$$

donde j se refiere a las neuronas de salida en la capa anterior ($I - 1$). Por lo tanto, la salida en la capa (I), está definida por la ecuación:

$$a_j^I = \sigma(x_j^I). \quad (2.4)$$

Los parámetros como los pesos w_{ij} y sesgos b_i se obtienen a partir de dos métodos matemáticos: propagación hacia adelante y propagación hacia atrás, representados en la figura 2.10

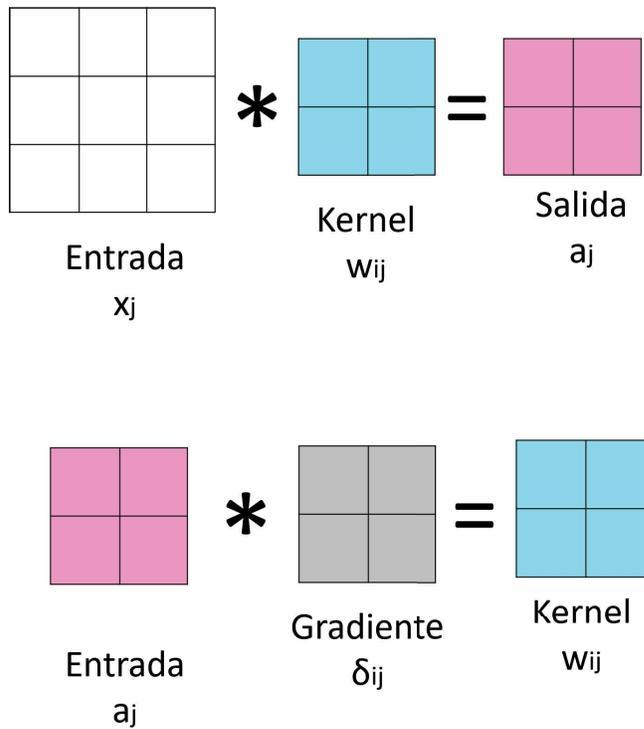


Figura 2.10. Propagación hacia adelante y hacia atrás

En la propagación hacia adelante, se inicializan los pesos y ses-

gos de manera aleatoria, en la primera capa, para estimar los nuevos valores que serán utilizados en la siguiente capa ($I + 1$). Cada nodo recibe la información de la capa anterior (x_i^{I-1}) y se realiza una multiplicación del valor en el nodo y los pesos que conecta, agregando el valor del sesgo en el nodo y aplicando una función de activación para obtener a_{ij} .

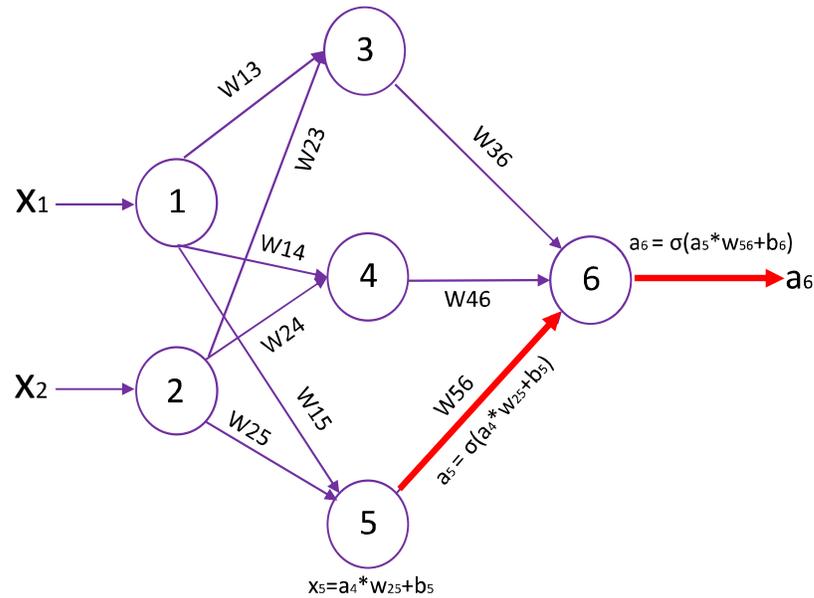


Figura 2.11. Propagación hacia delante

Matemáticamente resulta:

$$\begin{bmatrix} a_3 \\ a_4 \\ a_5 \end{bmatrix} = \sigma \left(\begin{bmatrix} w_{13} & w_{23} \\ w_{14} & w_{24} \\ w_{15} & w_{25} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} b_3 \\ b_4 \\ b_5 \end{bmatrix} \right) \quad (2.5)$$

Desarrollando lo anterior:

$$\begin{aligned} a_3 &= \sigma(w_{13}a_1 + w_{23}a_2 + b_3). \\ a_4 &= \sigma(w_{14}a_1 + w_{24}a_2 + b_4). \\ a_5 &= \sigma(w_{15}a_1 + w_{25}a_2 + b_5). \end{aligned} \tag{2.6}$$

Por otro lado, la propagación hacia atrás es un método que actualiza los pesos w_{ij} y sesgos b_i que deberían ser aplicados en los nodos n_i de la red neuronal, utilizando algoritmos de optimización numérica como descenso por gradiente. En otras palabras, permite que la red neuronal mejore minimizando la distancia entre la salida actual y la salida deseada, también conocida como función de costo o pérdida E .

Para ello se realizan dos actualizaciones: 1) La actualización de pesos, a partir de una regla de la cadena y 2) la actualización de deltas, conocidas como los gradientes de la función error E y se utiliza el mismo flujo que en el caso de los pesos.

2.5.1. Descenso por gradiente

Autores como Ketkar et al. (2017), Ruiz et al. (2016) y Chollet et al. (2017) definen que el descenso por gradiente es un algoritmo de optimización que trabaja con una función error, inicia en un punto aleatorio y se mueve en la dirección de mayor pendiente, buscando el mínimo de la función, donde la red neuronal es más precisa.

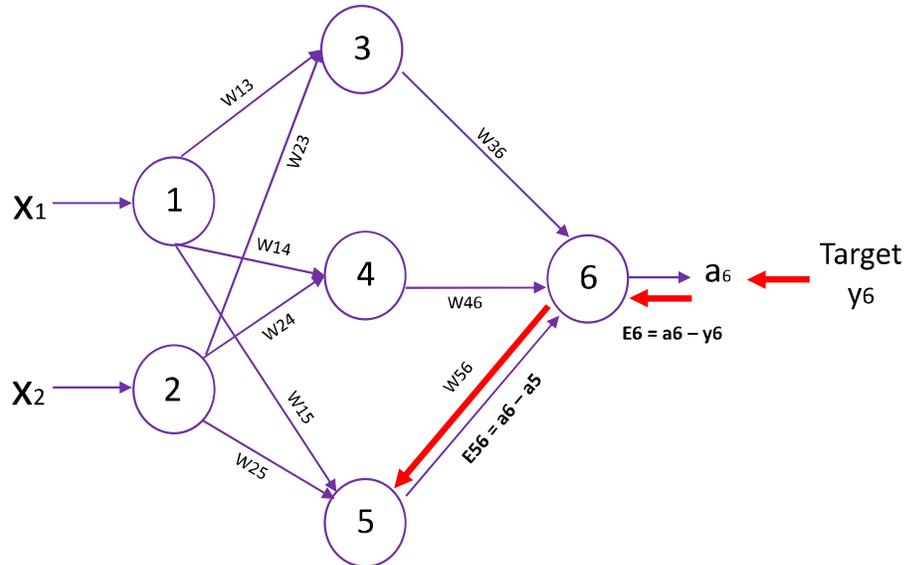


Figura 2.12. Propagación hacia atrás. En esta imagen E_6 corresponde al error entre los valores predichos en la última capa a_6 y los esperados y_6 , E_{56} representa el error entre los valores de la capa 5 y 6, mientras que δ_6 hace referencia al gradiente del error E_{56}

Existen distintas maneras para encontrar el mínimo de una función, uno de ellos es utilizando la primera derivada o pendiente que mide la rapidez con la que cambia una función. Si partimos de un punto aleatorio y seguimos la dirección negativa de máxima pendiente, llegaremos a un punto mínimo.

Ahora que ya sabemos cómo proceder para buscar ese punto mínimo donde la red tiene el mejor desempeño, debemos pensar en la velocidad de descenso (k), porque si es muy rápida, el algoritmo podría no encontrar el mínimo pero si es muy lenta entonces nuestra búsqueda podría tomar mucho tiempo.

El aprendizaje automático busca encontrar los parámetros W_i

que minimicen la función de pérdida E . Entonces podemos decir que $\hat{W} = W - k\nabla E(W)$. En esta formulación nosotros elegimos W , E y k , por lo cual solo es necesario calcular la función gradiente de la siguiente manera:

$$\nabla E = \left(\frac{\partial E}{\partial W_1}, \frac{\partial E}{\partial W_2}, \dots, \frac{\partial E}{\partial W_3} \right), \quad (2.7)$$

donde E es la función de pérdida y corresponde a la distancia entre el valor obtenido y el deseado, si utilizamos los pesos W_i .

En resumen, el algoritmo de descenso por gradiente nos permite obtener los pesos y sesgos que minimizan el error en el aprendizaje de nuestra red neuronal.

2.5.2. Función de costo

La función de costo nos ayuda a medir qué tan bueno es el funcionamiento de nuestro modelo para un conjunto de datos dado. Tiene por objetivo medir el error entre el valor real y el estimado, para optimizar los parámetros de la red neuronal. Las funciones más utilizadas en problemas de aprendizaje profundo son la función de entropía cruzada y el error cuadrático. Para mayor detalle revisar Ketkar (2017).

2.5.2.1. Entropía cruzada

Simula una clasificación binaria donde tenemos una entrada $D = \{(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)\}$ con $x \in R_n$ y un vector de salida $y \in \{0, 1\}$ que contiene a las clases objetivo.

Si suponemos un modelo capaz de predecir la probabilidad de y dado x , entonces podemos denotarlo como $f(x, \theta)$ donde θ representa los parámetros del modelo. De acuerdo con el principio de máxima verosimilitud, debemos encontrar los valores de θ que maximizan $P(D|\theta)$.

Ahora, asumiendo una distribución de Bernoulli que trabaja con casos de “éxito” o “no éxito”, cuando la hipótesis inicial es positiva o negativa, y dado que los eventos en $\{(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)\}$ son independientes, podemos decir que:

$$P(D) = \prod_{i=1}^n f(x, \theta)^{y_i} \cdot (1 - f(x, \theta))^{(1-y_i)}. \quad (2.8)$$

Aplicamos una función logaritmo en ambos lados de la ecuación para eliminar la potencia y_i y $(1 - y_i)$:

$$\log P(D) = \log \prod_{i=1}^n [f(x, \theta)^{y_i} \cdot (1 - f(x, \theta))^{(1-y_i)}]. \quad (2.9)$$

Luego entonces, podemos reducir la ecuación anterior como se muestra a continuación:

$$\log P(D) = \sum_{i=1}^n [y_i \log f(x, \theta) + (1 - y_i) \log(1 - f(x, \theta))]. \quad (2.10)$$

Si en lugar de maximizar la ecuación, minimizamos su valor negativo:

$$-\log P(D) = -\sum_{i=1}^n [y_i \log f(x, \theta) + (1 - y_i) \log(1 - f(x, \theta))]. \quad (2.11)$$

Finalmente, reemplazando y reduciendo elementos, tenemos que la entropía cruzada, en un contexto de clasificación binaria quedaría definida de la siguiente manera.

$$-\sum_{i=1}^n [y_i \log f(x, \theta) + (1 - y_i) \log(1 - f(x, \theta))]. \quad (2.12)$$

Generalizando el problema a una clasificación multiclase, reemplazamos la distribución Bernoulli por una distribución multinomial, donde:

$$P(D|\theta) = \frac{n!}{n_1! \cdot n_2! \cdot \dots \cdot n_n!} \prod_{i=1}^n f(x, \theta)^{y_i}. \quad (2.13)$$

Nuevamente, aplicando un logaritmo en ambas partes de la ecua-

ción,

$$\log P(D|\theta) = \log n! - \log n_1! \cdot n_2! \cdots n_k! + \log \prod_{i=1}^n f(x_i, \theta), \quad (2.14)$$

que simplificando términos, podemos reescribirla como:

$$\log P(D|\theta) = \log n! - \log n_1! \cdot n_2! \cdots n_k! + \sum_{i=1}^n y_i \log f(x_i, \theta). \quad (2.15)$$

Por otro lado, los términos $\log n!$ y $\log n_1! \cdot n_2! \cdots n_k!$ no están parametrizados por θ y por lo tanto podemos eliminarlos de la ecuación, dado que no buscamos cuáles son los valores de θ que maximizan la probabilidad $P(D|\theta)$,

$$\log P(D|\theta) = \sum_{i=1}^n y_i \log f(x_i, \theta). \quad (2.16)$$

De la misma manera que lo hicimos para una clasificación binaria, podemos sustituir la maximización por una minimización del negativo de la ecuación como se muestra a continuación.

$$-\log P(D|\theta) = -\sum_{i=1}^n y_i \log f(x_i, \theta). \quad (2.17)$$

La función entropía cruzada podemos obtenerla de la ecuación anterior y queda expresada como:

$$\text{Entropía} = -\sum_{i=1}^n y_i \log f(x_i, \theta). \quad (2.18)$$

2.5.2.2. Error cuadrático

El error cuadrático puede ser utilizado en un contexto de regresión usando el principio de máxima verosimilitud, donde asumimos que nuestro modelo predice el valor de y . Para aplicar el principio de máxima verosimilitud se asume que la diferencia entre el valor real y el calculado ($y - \hat{y}$) tiene una distribución Gaussiana o normal con media en cero y una varianza de σ_2 . Si ahora minimizamos la función $-\log P(D|\theta)$ podemos ver que el error cuadrático queda expresado como:

$$Error = \sum_{i=1}^n (y - \hat{y})^2. \quad (2.19)$$

Datos y metodología

Para el desarrollo de este proyecto se propone un modelo sencillo de una capa convolucional, utilizando espectrogramas de eventos sísmicos y explosiones mineras, con sus tres componentes: norte-sur (N-S), este-oeste (E-O) y vertical. De forma general, se realizaron cuatro pruebas: el primero incluye las tres componentes mencionadas anteriormente, en el segundo se utiliza únicamente la componente N-S, en el tercero la componente E-O y el cuarto considera la componente vertical. Siguiendo este mismo esquema, se construyen los subconjuntos de entrenamiento, validación y prueba.

El conjunto de datos se obtuvo del servidor del Servicio Sismológico Nacional (SSN) y la construcción de los espectrogramas se realizó con la biblioteca de Python llamada Obspy 1.4.0. La ejecución se realizó de manera local con un hardware: Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz 2.11 GHz y un sistema operativo Windows 64 bits.

3.1. Datos

El conjunto de datos está formado por 118 sismos tectónicos y 535 explosiones mineras registrados en el año 2020, en la estación sísmica ZAIG, ubicada en la ciudad de Zacatecas, gestionada por el SSN. Cada evento incluye tres componentes, registradas en tres direcciones: norte-sur, este-oeste y vertical. En la figura 3.1 se observan las tres componentes de un evento sísmico.

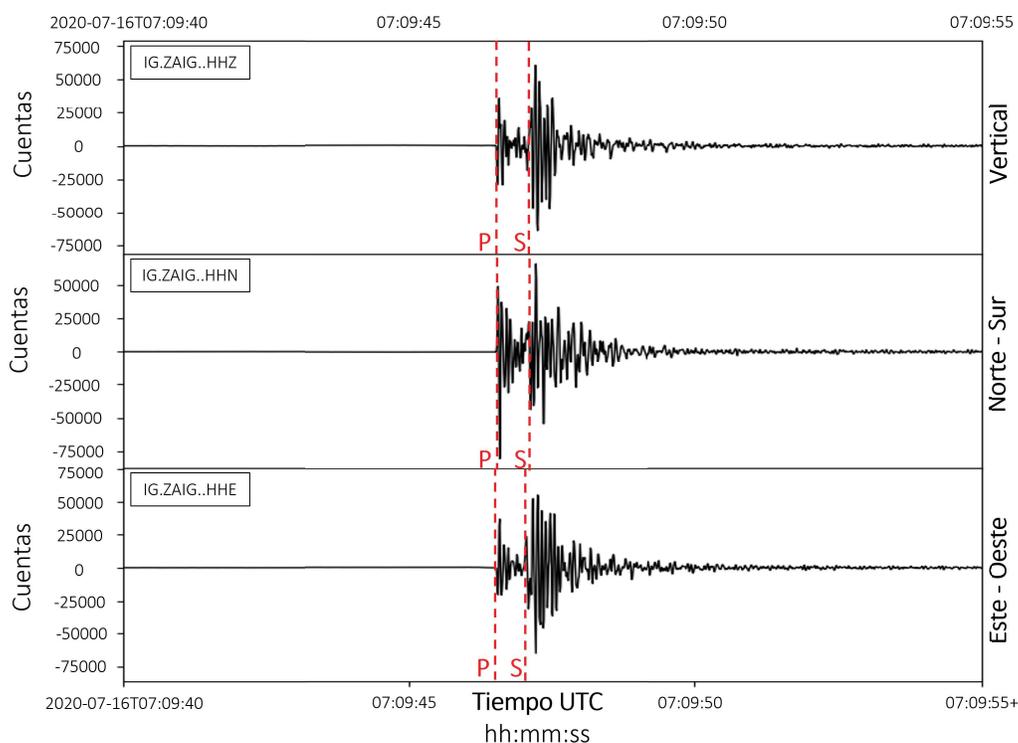


Figura 3.1. Componentes de un evento sísmico registrado por la estación sísmológica ZAIG, en el estado de Zacatecas el 16 de Julio del 2020.

El conjunto de datos se obtuvo del servidor del SSN, descargan-

do los sismogramas de 40s, iniciando 20s antes del arribo de onda P. Estos eventos fueron clasificados previamente por analistas del SSN en dos categorías, sismos y explosiones mineras. En la primera categoría se incluyen los eventos generados a partir de movimientos tectónicos, y la segunda, se refiere a eventos derivados de explosiones mineras cerca de la ciudad de Zacatecas.

3.2. Metodología

Para este trabajo, la construcción e implementación del modelo convolucional se utiliza Google Colab con una versión de Python 3, utilizando las siguientes bibliotecas: PyTorch 2.0, NumPy 1.2.3 y Pandas 1.5.2.

3.2.1. Procesamiento de datos

Como parte de la construcción del conjunto de espectrogramas, a partir de los sismogramas de las componentes N-S, E-O y vertical, descritas anteriormente, se aplicó un *detrend*, un *taper* del 1% y se utilizó *highpass* de 1Hz.

Posteriormente, los espectrogramas se construyeron aplicando la transformada rápida de Fourier, en un rango de frecuencias entre 1 y 50Hz, con una ventana de 1s y un traslape del 12%. En las figuras 3.2 y 3.3 se presentan algunos ejemplos de los espectrogramas, con sus respectivos sismogramas, de eventos sísmicos y explosiones

3. DATOS Y METODOLOGÍA

mineras, utilizados en nuestro modelo.

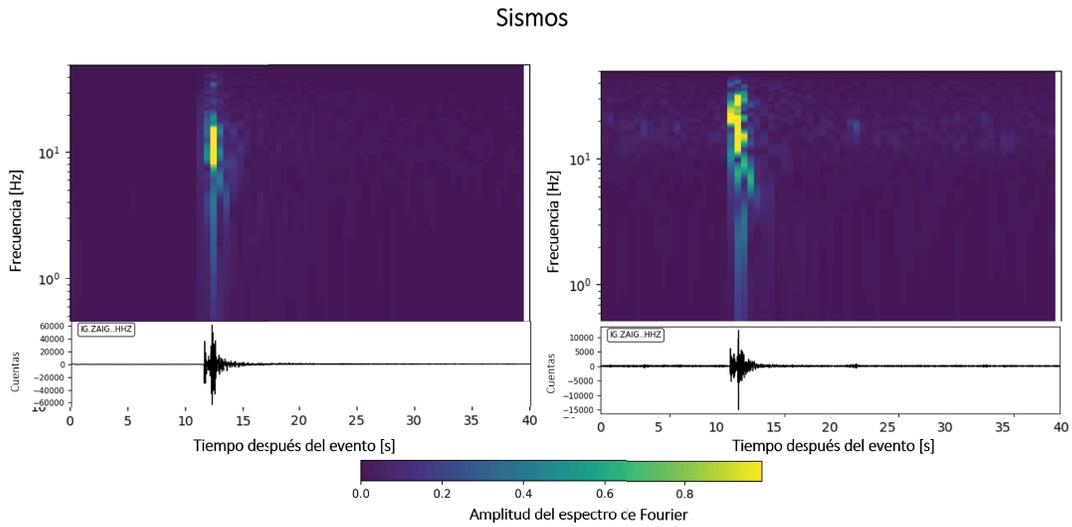


Figura 3.2. Espectrogramas de sismos.

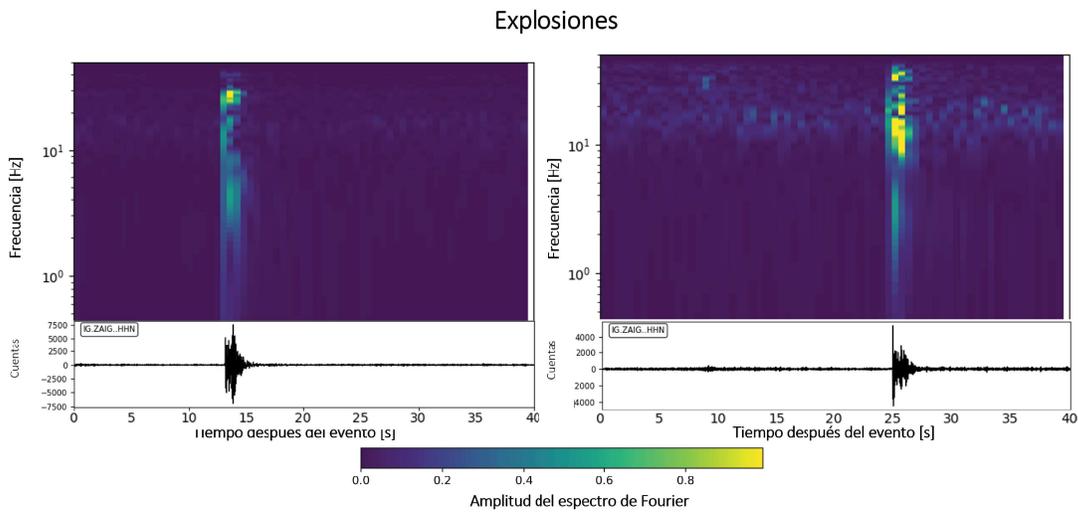


Figura 3.3. Espectrogramas de explosiones mineras.

Este conjunto de espectrogramas incluye las tres componentes de los 118 sismos y 535 explosiones mineras, dando un total de 653

eventos, es decir 1959 registros que incluye las componentes N-S, E-O y vertical.

Componente	Sismos	Explosiones	Total
Norte-Sur	118	535	653
Este-Oeste	118	535	653
Vertical	118	535	653
Total	354	1605	1959

Tabla 3.1. Distribución de datos

Para la implementación del modelo, el conjunto de eventos se dividió aleatoriamente, en tres subconjuntos: entrenamiento, validación y prueba. El subconjunto de entrenamiento incluye el 60 % de los eventos, validación otro 20 % y prueba con el 20 %, restante, considerando el mismo porcentaje de sismos y explosiones en cada subconjunto, como se muestra en la tabla 3.7.

	Entrenamiento	Prueba	Validación	Total
	60 %	20 %	20 %	100 %
Sismos	72	23	23	118
Explosiones	321	107	107	535

Tabla 3.2. División de datos para cada subconjunto de datos

3.2.1.1. Prueba 1

Para la primera prueba se propone construir los subconjuntos de entrenamiento, validación y prueba, incluyendo las tres componentes de un evento en el mismo subconjunto. Esto debido a que una o más componentes de un evento podrían tener una mala calidad de datos y afectar el aprendizaje y desarrollo del modelo. La can-

3. DATOS Y METODOLOGÍA

tividad de espectrogramas incluidas en cada uno de los subconjuntos se presenta en la tabla 3.3.

3 Componentes	Entrenamiento	Prueba	Validación	Total
	60 %	20 %	20 %	100 %
Sismos	216	69	69	354
Explosiones	953	321	321	1605

Tabla 3.3. Cantidad de espectrogramas incluidas en cada subconjunto de datos.

3.2.1.2. Prueba 2

Para la segunda prueba se utilizaron los espectrogramas de la componente N-S para construir con los subconjuntos de entrenamiento, validación y prueba, como se observa en la tabla 3.4.

Norte-Sur	Entrenamiento	Prueba	Validación	Total
	60 %	20 %	20 %	100 %
Sismos	72	23	23	118
Explosiones	321	107	107	535

Tabla 3.4. División de datos para el conjunto de datos que utiliza los espectrogramas de las componente Norte-Sur

3.2.1.3. Prueba 3

La prueba tres considera únicamente los espectrogramas de la componente E-O en la construcción de los subconjuntos de entrenamiento, validación y prueba, el desglose de los subconjuntos se presenta en la tabla 3.5.

Este-Oeste	Entrenamiento	Prueba	Validación	Total
	60 %	20 %	20 %	100 %
Sismos	72	23	23	118
Explosiones	321	107	107	535

Tabla 3.5. División de datos para el conjunto de datos que utiliza los espectrogramas de las componente Este-Oeste

3.2.1.4. Prueba 4

Finalmente, la cuarta prueba utiliza los espectrogramas de la componente vertical para la partición en los subconjuntos de entrenamiento, validación y prueba, como se observa en la tabla 3.6.

vertical	Entrenamiento	Prueba	Validación	Total
	60 %	20 %	20 %	100 %
Sismos	72	23	23	118
Explosiones	321	107	107	535

Tabla 3.6. División de datos para el conjunto de datos que utiliza los espectrogramas de las componente Vertical

3.2.2. Arquitectura del modelo

El diseño del modelo se realizó con la biblioteca PyTorch de Python. Consiste en una arquitectura simple de una capa convolucional y 16 filtros, dado que un modelo de mayor complejidad requiere de una mayor cantidad de datos. El tamaño de los filtros de la capa convolucional es de 3×3 , con un paso de 1 y un relleno igual a cero. Adicionalmente, se utilizó una ReLU que elimina características poco significantes con valores menores a cero, una capa *flatten* y una capa lineal donde se clasifica un vector de característi-

3. DATOS Y METODOLOGÍA

cas, calculando la probabilidad de pertenecer a las clases de sismos y explosiones.

El modelo toma como entrada el conjunto de espectrogramas de (28×28) píxeles.

Capa	Características
Capa convolucional	16 filtros de 3x 3
Capa ReLU	Elimina características <0
<i>Flatten</i>	$16 \times 28 \times 28 \Rightarrow 1 \times 12544$
Capa linear	Transforma de 12544 características a 2 clases

Tabla 3.7. Descripción de la arquitectura del modelo

Resultados

En este capítulo se describen los resultados obtenidos con el modelo propuesto para las cuatro pruebas descritas anteriormente. Analizando las curvas de exactitud y pérdida para los subconjuntos de entrenamiento y validación. Adicionalmente, se analiza la matriz de confusión para el conjunto de prueba.

4.1. Prueba 1

Los resultados de exactitud y pérdida obtenidos en la primera prueba, que considera las tres componentes de cada evento, se muestran en la figura 4.1.

En color azul, se indican las curvas de exactitud y pérdida para el conjunto de entrenamiento, por otro lado, las curvas de color corresponden al conjunto de validación. El modelo muestra mejores resultados durante la época 10, alcanzando una exactitud del 96.77 %, 95.19 % y 91.83 %, así como una pérdida del 8.07 %, 14.53 %

4. RESULTADOS

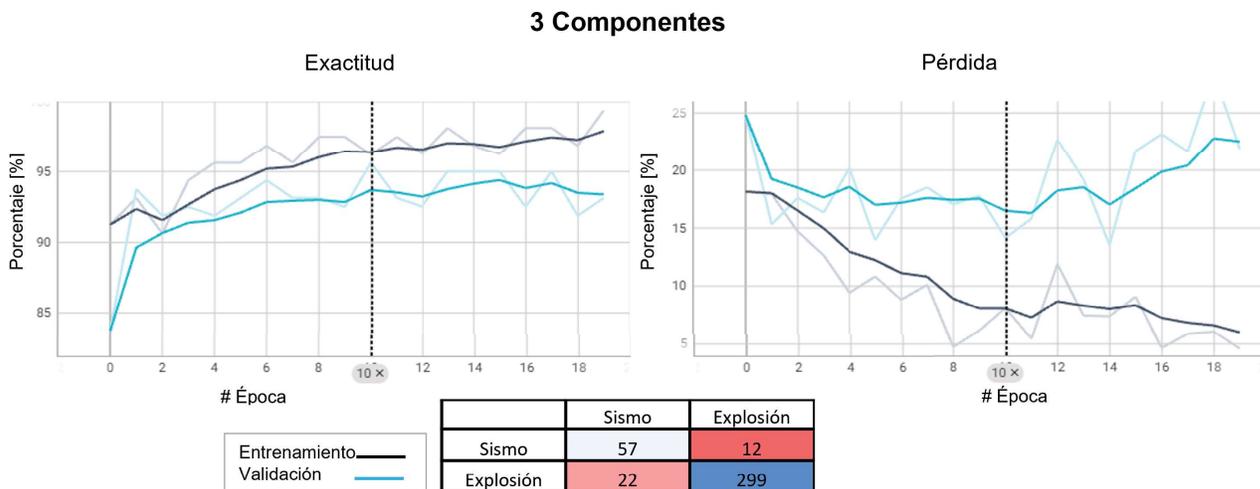


Figura 4.1. Matriz de confusión para el conjunto de prueba, y curvas de exactitud y pérdida, donde la curva color azul corresponde al conjunto de entrenamiento y la color negro al conjunto de validación. Prueba 1: Utilizando los espectrogramas de las tres componentes de cada evento

y 20.64 %, durante las etapas de entrenamiento, validación y prueba, respectivamente. El tiempo de ejecución fue de 1.27 minutos.

De acuerdo con la matriz de confusión para el conjunto de prueba, el modelo etiquetó correctamente 57 espectrogramas sísmicos y 299 espectrogramas de explosiones mineras. Por otro lado, clasificó incorrectamente 12 espectrogramas sísmicos y 22 espectrogramas de explosiones mineras.

4.2. Prueba 2

Los resultados del modelo para la prueba dos, que utiliza los espectrogramas de la componente N-S se muestran en la figura 4.2.

Las curvas de exactitud y pérdida para el conjunto de entrena-

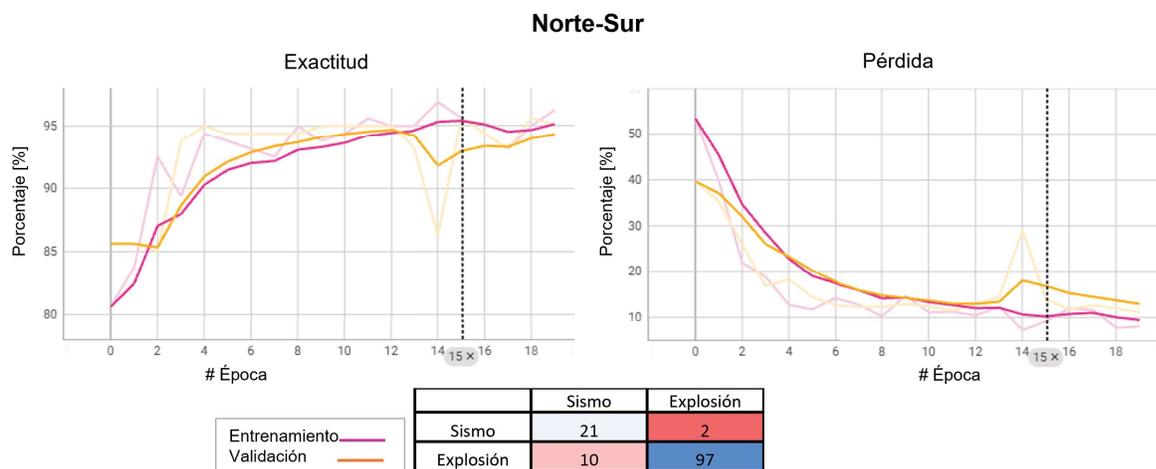


Figura 4.2. Matriz de confusión para el conjunto de prueba, y curvas de exactitud y pérdida, donde la curva color rosa corresponde al conjunto de entrenamiento y la color naranja al conjunto de validación. Prueba 2: Utilizando los espectrogramas de las componentes N-S de cada evento

miento se presentan en color rosa, y en color naranja para el conjunto de validación. Los mejores resultados se presentan en la época 15, con una exactitud del 95.43 %, 95.62 % y 83.12 %, así como una pérdida del 8.85 %, 12.25 % y 26.64 %, para los subconjuntos de entrenamiento, validación y prueba, respectivamente. El tiempo de ejecución fue de 1.41 minutos.

La matriz de confusión muestra los resultados del modelo clasificando el conjunto de prueba, etiquetando correctamente 21 espectrogramas sísmicos y 97 espectrogramas de explosiones. Por otro lado, el modelo clasificó de manera incorrecta 2 espectrogramas sísmicos y 10 espectrogramas de explosiones mineras.

Al analizar la interacción de las curvas para los subconjuntos de

entrenamiento y validación, se observa que el modelo si logró generalizar el problema. Sin embargo, el comportamiento de las curvas para el conjunto de validación son muy similares a las curvas de entrenamiento, esto podría ser dado que este conjunto tiene 130 espectrogramas.

4.2.1. Prueba 3

Los resultados del modelo utilizando como entrada la componente E-O se muestran en la figura 4.3. Las curvas de exactitud y pérdida para conjunto de entrenamiento en color morado y validación en color verde. Los mejores resultados se obtuvieron durante la época 12, con una exactitud del 95.91 %, 95.62 % y 90.62 %, así como una pérdida del 9.65 %, 11.83 % y 24.13 %, para los subconjuntos de entrenamiento, validación y prueba, respectivamente. El tiempo de ejecución fue de 1.46 minutos.

La matriz de confusión para el conjunto de prueba, muestra que el modelo clasificó correctamente 15 espectrogramas sísmicos y 100 espectrogramas de explosiones. Así como, 8 espectrogramas sísmicos y 7 espectrogramas de explosiones etiquetados de manera incorrecta.

Analizando las gráficas de exactitud y pérdida para el conjunto de entrenamiento y validación, podemos ver que inician con un buen comportamiento pero justo antes de la época 2, el modelo parece que

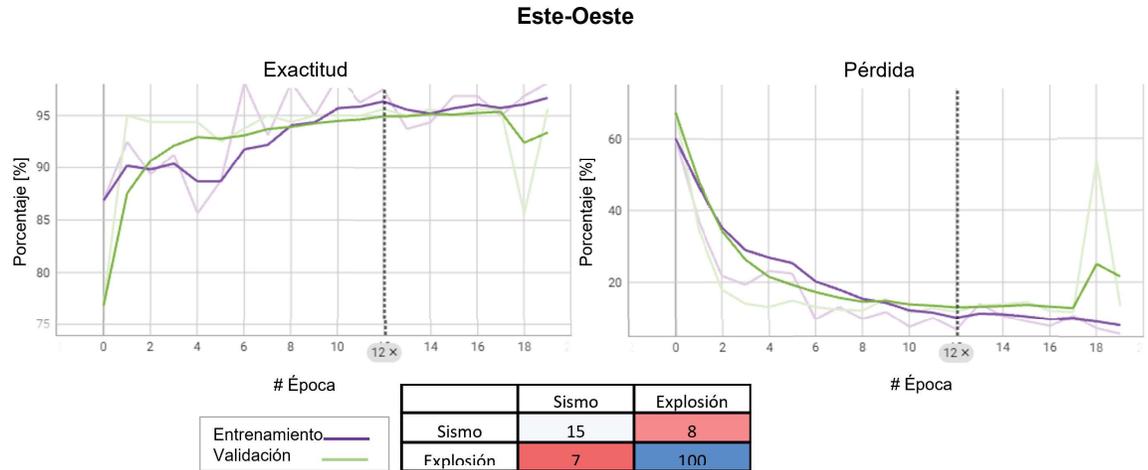


Figura 4.3. Matriz de confusión para el conjunto de prueba, y curvas de exactitud y pérdida, donde la curva color morado corresponde al conjunto de entrenamiento y la color verde al conjunto de validación. Prueba 3: Utilizando los espectrogramas de las componentes E-O de cada evento

ya no tiene mejoras con el conjunto de entrenamiento. Esto podría indicar que el conjunto de datos es tan pequeño que no necesita de muchas épocas de entrenamiento para etiquetar correctamente el problema pero al mismo tiempo, tiene tan pocos ejemplos que tiene algunas dificultades durante el aprendizaje.

4.2.2. Prueba 4

Finalmente, los resultados para la componente vertical se observan en la figura 4.4, donde el conjunto de entrenamiento corresponde a la línea en color naranja y validación en color negro. Los mejores resultados se presentan en la época 15, con una exactitud del 96.88%, 96.88% y 83.12%, así como una pérdida del 8.92%,

4. RESULTADOS

13.77 % y 44.40 %, para los subconjuntos de entrenamiento, validación y prueba, respectivamente. El tiempo de ejecución fue de 1.46 minutos.

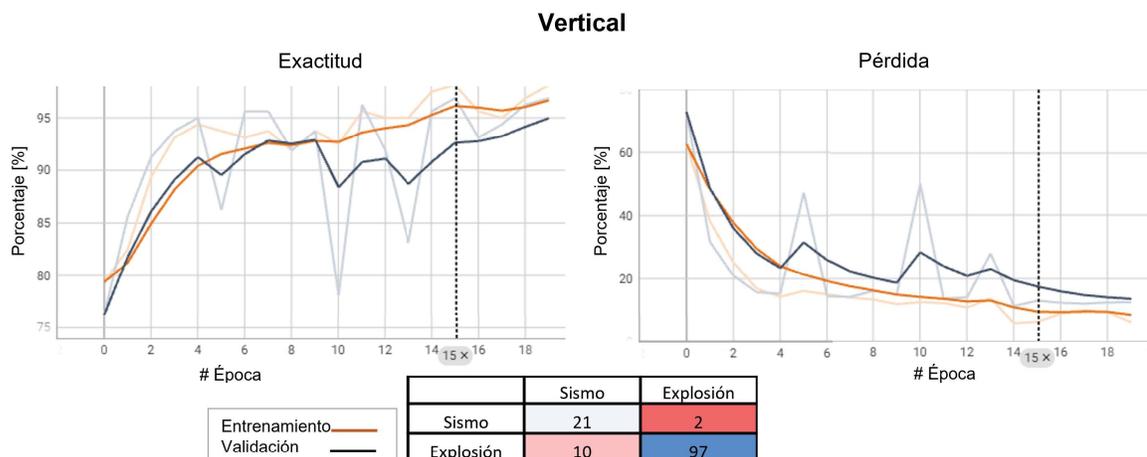


Figura 4.4. Matriz de confusión para el conjunto de prueba, y curvas de exactitud y pérdida, donde la curva color naranja corresponde al conjunto de entrenamiento y la color negro al conjunto de validación. Prueba 4: Utilizando los espectrogramas de las componentes verticales de cada evento

La matriz de confusión, que el modelo clasificó correctamente 21 espectrogramas sísmicos y 97 espectrogramas de explosiones. Adicionalmente, el modelo etiquetó incorrectamente 2 espectrogramas sísmicos y 10 espectrogramas de explosiones.

Analizando las curvas de exactitud y pérdida, podemos inferir que el modelo tiene un ligero sobre ajuste después de época 8. Aunque el modelo logró generalizar correctamente el problema de clasificación, las variaciones en las curvas denotan que el modelo tiene cierta cierta incertidumbre.

4.3. Análisis de los resultados

En la tabla 4.1 se presenta un resumen dos valores obtenidos para la exactitud y pérdida, en cada una de las pruebas descritas anteriormente.

Prueba	Parámetro	Entrenamiento (%)	Validación (%)	Prueba (%)
3 componentes	Exactitud	96.77	95.19	91.83
	Pérdida	8.07	14.53	20.64
N-S	Exactitud	95.43	95.62	83.12
	Pérdida	8.85	12.25	26.64
E-O	Exactitud	95.91	95.62	90.62
	Pérdida	9.65	11.83	24.13
vertical	Exactitud	96.88	96.88	83.12
	Pérdida	8.92	13.77	44.40

Tabla 4.1. Valores de exactitud y pérdida

Considerando los valores de exactitud y pérdida para las cuatro pruebas durante las etapas de entrenamiento, validación y prueba, podemos decir que el modelo logra buenos indicadores, logrando clasificar correctamente entre un 80 y 97% de los datos. Finalmente, analizando las curvas de exactitud y pérdida para los conjuntos de entrenamiento y validación, se observa que el modelo tiene una buena capacidad para clasificar los espectrogramas sísmicos y de explosiones mineras, especialmente cuando se utilizan las tres componentes.

Conclusiones y trabajo futuro

El modelo de una capa convolucional propuesto en este proyecto de tesis para clasificar espectrogramas de eventos sísmicos y explosiones mineras, utilizando las tres componentes logró clasificar correctamente más del 90 % de los espectrogramas en los conjuntos de entrenamiento, validación y prueba.

Analizando las curvas de exactitud y pérdida, podemos inferir que el modelo logró una generalización del problema, utilizando las tres componentes de manera individual, aunque su desempeño mejora cuando utiliza las tres componentes juntas.

Podemos concluir que la limitada cantidad de datos disponibles, así como la diferencia entre las proporciones de eventos sísmicos y explosiones mineras, podrían ser factores en el desempeño del modelo propuesto. Sin embargo, también podría estar relacionado a la calidad de los registros.

Finalmente, a pesar de la desventaja que representa la cantidad

5. CONCLUSIONES Y TRABAJO FUTURO

de datos limitada y la baja calidad de datos en algunos registros, el modelo logró clasificar correctamente más del 90 % de los espectrogramas en un tiempo menor a dos minutos, utilizando las componentes juntas o de manera individual. Este trabajo de clasificación le tomaría días a un analista experimentado.

Cabe mencionar que una ventaja de utilizar un modelo de arquitectura convolucional es que al aumentar el tamaño del conjunto de datos, el modelo podría generalizar mejor el problema de clasificación.

Como trabajo futuro sería interesante explorar otros conjuntos de datos como sismos volcánicos, ruido ambiental, etc, y revisar el comportamiento de este tipo de arquitecturas convolucionales.

REFERENCIAS

1. Caltech. <https://scedc.caltech.edu/data/stp/>. Consultada el 20 Diciembre del 2022.
2. Chollet, F. (2017). Deep Learning with Python. Manning Publications Company
3. Convolutional Neural Networks. <https://stanford.edu/~Eshervine/teaching/cs-230/cheatsheet-convolutional-neural-networks/>. Consultada el 20 Julio del 2022.
4. Curso Aprendizaje Profundo: Gibranfp. Consultada el 20 mayo del 2022.
5. Del Pezzo et al., (2003). Discrimination of Earth- quakes and Underwater Explosions Using Neural Networks. Bulletin of the Seismological Society of America, 93(1):215–223
6. Descenso por gradiente. <https://turing.iimas.unam.mx/~Eivanvladimir/posts/gradientdescent/>. Consultada el 20 mayo del 2022.
7. Kostoglodov V. y Pacheco J. F. (2000). Cien años de sismicidad en México: <http://usuarios.geofisica.unam.mx/vladimir/sismos/100aF1os.html>. Consultada el 01 enero del 2022.
8. Haykin, S. (1998). Neural Networks: A Comprehensive Foundation. Prentice Hall, subsequent edition.

9. IGI Global. <https://www.igi-global.com/dictionary/amplitude-spectrum/56397>. Consultada el 15 de mayo del 2023.
10. IRIS. <https://www.iris.edu>. Consultada el 01 enero del 2022.
11. Ketkar, N. (2017). *Deep Learning with Python: A Hands-On Introduction*. Apress.
12. Kortström et al., (2016). Automatic classification of seismic events within a regional seismograph network. *Computers Geosciences*, 87:22–30.
13. Lederer, J. (2021). *Activation Functions in Artificial Neural Networks: A Systematic Overview*.
14. Levinzon, F. A. (2017). Ultra-Low-Noise Seismic Accelerometers for Earthquake Prediction and Monitoring. *InTech*. doi: 10.5772/65925
15. Linville et al., (2019). Deep Learning Models Augment Analyst Decisions for Event Discrimination. *Geophysical Research Letters*, 46(7):3643–365.
16. Manaswi, N. K. (2018). *Deep Learning with Applications Using Python*.
17. SSN (2022): Universidad Nacional Autónoma de México, Instituto de Geofísica, Servicio Sismológico Nacional, México. Di-

rección electrónica: <http://www.ssn.unam.mx>

18. Nguyen et al., (2021). A Layer-wise. Theoretical Framework for Deep Learning of Convolutional Neural Networks
19. Redes neuronales con Python. <https://www.cienciadedatos.net/-documentos/py35-redes-neuronales-python>. Consultada el 01 enero del 2022.
20. Rojas, R. y Feldman, J. (1996). Neural Networks. Springer Publishing, New York, United States
21. SEDESOL (2020). Atlas de Riesgos de la Ciudad de Zacatecas. Reporte Técnico.
22. SGM (2020). Panorama Minero del Estado de Zacatecas. Reporte Técnico.
23. Tibi et al., (2019). Classification of Local Seismic Events in the Utah Region: A Comparison of Amplitude Ratio Methods with a Spectrogram-Based Machine Learning Approach. Bulletin of the Seismological Society of America, 109(6):2532–2544
24. Tutorial — ObsPy 1.4.0 documentation.
<https://docs.obspy.org/tutorial/>. Consultada el 01 enero del 2022.

25. USGS Earthquake Hazards Program. <https://earthquake.usgs.gov>.
Consultada el 01 enero del 2022
26. Wang et al., (2020). The Influence of the Activation Function in a Convolution Neural Network Model of Facial Expression Recognition. *Applied sciences*, 10(5):1897
27. Yildirim et al., (2011). Discrimination of quarry blasts and earthquakes in the vicinity of Istanbul using soft computing techniques. *Computers Geosciences*, 37(9):1209–1217.