



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Universidad Politécnica de Valencia
Escuela Técnica Superior de Ingeniería del Diseño

Detección no supervisada de sonidos anómalos para el reconocimiento del estado de cajas de cambios

Trabajo fin de Máster:

Máster Universitario en Ingeniería del Mantenimiento

Autor: Wang, Yicheng

Tutor: Climent Puchades, Héctor

Curso: 2021-2022

Data: Valencia, Enero, 2022

RESUMEN

Detección no supervisada de sonidos anómalos para el reconocimiento del estado de cajas de cambios

Aprendizaje automático no supervisado es una tecnología esencial en la cuarta revolución industrial incluida la automatización industrial basada en inteligencia artificial, que se lleva a cabo una investigación para la predicción y diagnóstico de fallas de la maquinaria.

En este artículo, prueba unos métodos de diagnóstico de anomalías de la caja de cambios sobre base de aprendizaje automático mediante la detección no supervisada de los sonidos, y aplica el algoritmo propuesto para esa. Este experimento verifica la influencia del tiempo de los datos en experimentos, las rondas de entrenamiento (Epoch) y la estructura del modelo de la red neuronal. Al mismo tiempo, el artículo muestra algunas características extraídas de las muestras de sonidos.

Al final, analiza la eficiencia del modelo y propone la perspectiva de investigación futura mediante la comparación entre los resultados actuales.

Palabras clave: detección de anomalías, aprendizaje automático, características, la red neuronal, autoencoder, CNN, FNN



AGRADECIMIENTOS

En primer lugar, agradezco a mi profesor, Héctor Climent Puchades, que me guió a estudiar en este fascinante campo, quiero decir que el diagnóstico mediante análisis del ruido es un curso muy interesante en el que habrá más ramas que vale la pena investigar.

La integración de aprendizaje automático y análisis de ruido está orientada al futuro en la ciencia de la detección. A pesar de que no todo salió bien durante el experimento, pero superé muchas dificultades bajo la paciencia del profesor para conseguir los resultados actuales. Gracias de nuevo por su rigor académico, su esmero y humor en la docencia, lo que me ayudó a resolver muchas dudas.

Asimismo, quiero agradecer a todos los profesores, Vicente Macian Martinez y Bernardo Vicente Tormos Martínez, les agradezco por enseñarme y ayudarme en mis estudios de posgrado.

Finalmente, Gracias a mi familia y su apoyo detrás de mí. Creo que este será el comienzo del próximo camino.

ÍNDIX

1	MEMORIA.....	1
1.1	Objetivo y Motivación del trabajo	1
1.2	Metodología del trabajo	2
1.3	Estructura del trabajo	4
2	TEORÍA BÁSICA DEL PREPROCESADO PARA SONIDOS.....	6
2.1	Preprocesado para las señales acústicas.....	6
2.2	Extracción de características	8
2.2.1	Coeficientes Cepstrales en las Frecuencias de Mel (MFCCs).....	9
2.2.2	Cromaticidad-STFT(Chroma-STFT).....	12
2.2.3	Centroide Espectral	14
2.2.4	RollOff Espectral.....	16
2.2.5	Energía a Corto Plazo	18
2.2.6	Tasa de Cruce por Cero(Zero Crossing Rate)	21
2.2.7	Curtosis	23
2.3	Lista de características acústicas.....	25
3	FUNDAMENTOS DEL APRENDIZAJE AUTOMÁTICO	26
3.1	Descripción del aprendizaje profundo.....	26
3.2	Diagrama del aprendizaje automático.....	28
3.3	La red neuronal profunda (Deep Neural Network)	29
3.3.1	Algoritmo hacia adelante	30
3.3.2	Algoritmo de retropropagación de errores.....	31
4	MÉTODO AE DE LA DETECCIÓN DE ANOMALÍAS PARA SONIDOS DE LA CAJA DE CAMBIOS	32
4.1	Aprendizaje automático supervisado y no supervisado	32
4.2	Detección de anomalías por el aprendizaje automático no supervisado	33
4.3	Indicadores de evaluación para el modelo.....	34
4.4	Sistema basado en BP-FNN	35
4.4.1	Diagrama del sistema sobre BP-FNN	37

TRABAJO FIN DE MÁSTER

4.4.2	Regularización	38
4.4.3	Optimizador	40
4.4.4	Función de pérdida	42
4.4.5	Resultados y Análisis	43
4.5	Sistema basado en CNN-Conv1D	51
4.5.1	La red neuronal convolucional	51
4.5.2	La capa convolucional	51
4.5.3	La capa de agrupación	53
4.5.4	Diagrama del CNN-Conv1D	57
4.5.5	Resultados y Análisis	58
5	CONCLUSIÓN Y PERSPECTIVA	60
5.1	Conclusión	60
5.2	Perspectiva	61
6	ANEXOS	63
6.1	Biblioteca de sonidos	63
6.2	Sistema de software	64
6.3	Visualización de código	65
7	ÍNDICE DE FIGURAS Y TABLAS	67
7.1	Índice de figuras	67
7.2	Índice de tablas	68
8	BIBLIOGRAFÍA	69



1 MEMORIA

1.1 Objetivo y Motivación del trabajo

La tecnología de detección de fallas ha desarrollado a la aplicación de detección de sonidos complejos, el objeto de investigación se extiende desde las señales de vibración hasta el ruido de maquinaria en funcionamiento.

La detección de sonido anómalo es una de las investigaciones populares en este campo, que se utiliza para detectar si se ocurre un evento sonoro que preocupa al usuario en un entorno, por ejemplo, el sonido anómalo de rodaje de engranajes o el de rodamiento, lo que hace es para confirmar si hay algún peligro potencial en el ambiente de trabajo. En realidad, aunque hay muchas maneras para la seguridad como la videovigilancia, pero la mayoría de ellas son costosas. Dado que la gran cantidad de informaciones de imágenes están en procesamiento, generalmente, se ignora los detalles contenida en la señal de sonidos.

La gente comenzó a considerar la manera más cómoda y eficaz para la seguridad del medioambiente, por tanto, el sonido es un tipo de señal muy común en el entorno, si la gente puede fortalecer la seguridad mediante la detección de sonido, es cierto que se disminuirá los problemas insignificantes. Debido al desarrollo de la tecnología, la detección de sonidos anómalos se ha aplicado en muchos campos, por ejemplo, en la producción industrial, el sonido como el indicador es para probar la calidad de los productos, o en la medicina moderna, los médicos pueden juzgar la afección mediante escucho de la frecuencia cardíaca del paciente. En un vehículo, una ocurrencia de falla en la caja de cambios o de daño a alguna pieza causará los sonidos anómalos cuando la caja de cambios está funcionando. Por eso, el diagnóstico por

análisis de sonido anormal puede predecir el estado de funcionamiento de la máquina con antelación. La realización de mantenimiento preventivo y predictivo es para prevenir los accidentes innecesarios.

En este caso, utilizamos un método de aprendizaje no supervisado para que el sistema reconozca si la muestra de sonido de la caja de cambios es normal o anormal a través del aprendizaje automático.

1.2 Metodología del trabajo

La filosofía del experimento en este caso es que si entrenamos el modelo con los datos de muestras normales o los no etiquetados con pocos sonidos anormales, aprenderá una función de reconstrucción efectiva para los datos de muestras normales que tienen el error de reconstrucción bajo al final, lo contrario es que es eficaz para los datos de anomalías que tiene el error más alto tras de reconstrucción. Entonces, el error de reconstrucción es como la señal para la detección de fallas, luego visualiza el mapa de errores de reconstrucción, se puede observar que la distribución de errores de la muestra normal es totalmente pequeña y está separada de la de la muestra anormal.

Sin embargo, los resultados de investigación muestran que es cierto que se puede interferir la exactitud de la detección si mezcla las muestras de sonidos anómalos con diferentes proporciones en equipo de entrenamiento. En realidad, debido a la rareza de ocurrencia de anomalías en trabajo, se supone que los datos recopilados en este experimento para el entrenamiento vienen de las cajas de cambios sanas y buenas.

En este caso, se completa el experimento en Google Colab Notebook a través del lenguaje Python. Mediante el uso de métodos de aprendizaje profundo, va a mejorar sus marcos de las dos redes neuronales, que son las de la capa completamente conectada basada

TRABAJO FIN DE MÁSTER

en un algoritmo de retropropagación de errores (BP-FNN) y de la convolucional (CNN) utilizado en el autoencoder, crea unas estructuras de la red para identificar sonidos anómalos y mejorar la tasa de detección. Por eso, el trabajo como los siguientes son:

- a. Establece un paquete de sonido para la detección de sonidos anómalos de la caja de cambios y analiza su efectividad. Dado que falta de una base de datos de dichos sonidos sobre estándar abierta, por lo tanto, va a recopilar y clasificar unos paquetes de sonidos de la caja de cambios a partir de DCASE Challenge, eso es para hacer el trabajo previo de recopilación de base de datos.
- b. La tarea principal es extraer las características adecuadas a partir de sonidos, cuyos formatos son todos WAV., luego se componen de vectores de características y hacen la normalización. A pesar de que el sonido de la caja de cambios es distinto al de la voz humana, pero también contiene unos abundantes parámetros físicos, es concebible que los diferentes parámetros de características puedan mostrar las informaciones muy distintas. Dado que el estudio de las características acústicas sobre la caja de cambios es aún un campo nuevo, por ello, se seleccionan y extraen algunas características importantes desde la señal acústica, luego va a analizar sus significados físicos y estudiar sus teorías básicas y métodos de extracción.
- c. La parte principal en este caso es la viabilidad y exactitud de las dos redes neuronales (BP-FNN y CNN aplicadas en autoencoder) en la detección de sonidos anómalos para la caja de cambios. El sistema está basado en estas dichas redes neuronales, construye los modelos en el marco de Tensorflow y prográmelo en Python. La construcción del marco de modelo utiliza principalmente la biblioteca Keras. Al final, se ha construido ocho capas en el modelo de BP-FNN y diez capas en el modelo de CNN. Este experimento muestra que se puede obtener los resultados mejores y precisos después de ejecutar el programa.



1.3 Estructura del trabajo

La estructura del trabajo en este artículo incluyen las cinco partes:

- a. En capítulo I, describe este tema y su significado trascendental, expone la situación actual de la investigación del aprendizaje profundo en el campo del diagnóstico de fallas mecánicas.
- b. En capítulo II, generalmente, el diagnóstico de fallas de maquinaria se basa en las señales de vibración en la mayoría de estudios pero menos en las acústicas. A pesar de esto, en este artículo se presenta la extracción de características del preprocesamiento de señales de audio. Conforme a las características de la caja de cambios teniendo en cuenta la generalización del modelo, selecciona algunas características especiales y las extrae desde muestras.
- c. En capítulo III y capítulo IV, la red neuronal profunda basada en el autoencoder se aplica al diagnóstico de fallas de la caja de cambios. Se compone de dos partes, la primera parte es utilizada por el codificador para analizar la secuencia de entrada y el decodificador usa en la segunda parte para generar la secuencia de salida. Por tanto, en este capítulo, crea una variedad del autoencoder de aprendizaje automático en la que contiene varias capas de redes neuronales. Hay las dos redes neuronales de uso común que son la de capa completamente conectada basada en un algoritmo de retropropagación de errores (BP-FNN) y la convolucional (CNN) utilizado en el autoencoder. También analiza los diferentes algoritmos de las dos redes neuronales. El autoencoder puede reconstruir los datos desde muestras que son las informaciones de la extracción de características.

La idea es que los datos reconstruidos por el autoencoder no pueden restaurar como los datos originales, pasará un error de reconstrucción, que es la pérdida después del entrenamiento. Para que la máquina identifique las muestras anómalas, es necesario verificar y clasificar dichos errores, ya que el error de reconstrucción a partir de datos de muestras anormales es mayor y más claro que el de datos de muestras normales.

TRABAJO FIN DE MÁSTER

El método es que los datos preprocesados de características que vienen de muestras normales se ingresan al autocodificador a través de la capa de entrada, luego se alcanza a la capa de embotellamiento mediante el codificador, que es para encontrar las reglas de los datos enormes y eliminar los restantes. El segundo paso se decodifica en la capa de salida, se restaura a unos datos digitales que la máquina puede aprender. Al mismo tiempo, se utiliza la técnica 'Dropout' para el ajuste de la red, entrenamiento y la realización del diagnóstico de fallas.

- d. En capítulo V, esta parte está compuesto de conclusión y expectativa. Los sonidos anómalos no es solo una ocurrencia rara, sino también muy diversa. Por tanto, no es fácil a recopilar patrones detallados de sonidos anómalos, es decir que debe detectar los desconocidos que no se observan en los datos de entrenamiento (datos de muestras normales). En esta parte, describe las deficiencias del método tradicional de diagnóstico de fallas, que se usa las informaciones aisladas de un solo canal. Esperamos integrar las informaciones multicanales y aplicarlas a las redes de aprendizaje profundo en los trabajos futuros.

2 TEORÍA BÁSICA DEL PREPROCESADO PARA SONIDOS

El sistema de la detección de sonidos anómalos contiene las tres fases, la primera es el preprocesado para todas las muestras de sonidos, la segunda es la etapa de entrenamiento y la tercera es la etapa de reconocimiento de sonidos anómalos.

En la fase de preprocesado, debe regularizar las muestras originales de sonido. Primero, se elimina las partes inválidas, luego se divide en el conjunto de entrenamiento y el conjunto de validación, el conjunto de entrenamiento se compone de la mayoría de las muestras de sonido de puntos normales, sin embargo, el conjunto de validación mezcla una gran parte de los sonidos anómalos más distinguibles. Al final, el proceso es la extracción de características en la que logra los parámetros informáticos que pueden representar los sonidos. Por ello, en este capítulo, presenta el preprocesado de señales de sonido y varios métodos para la extracción de características comunes.

2.1 Preprocesado para las señales acústicas

Lo primero que hay que hacer antes de entrenar es el preprocesado para las señales originales. Las fases de preprocesado son el pre-énfasis, encuadre, la agregación de ventanas (ventana de Hamming, etc.) y la detección y respuesta de Endpoints para eliminar el ruido inválido.

a. Pre-énfasis

Dado que la complejidad de recopilación de muestras de sonido, generalmente la señal se caerá en la parte de frecuencia alta, por ello, el motivo del pre-énfasis es para mejorar el punto de caída, la señal será más plana y se puede aumentar la resolución de aspecto en la parte de frecuencia alta. La función del filtro digital de primer orden para realizar el pre-énfasis es:

$$H(z) = 1 - az^{-1} \quad (0.9 < a < 1.0)$$

a : el coeficiente de pre - énfasis

Figura. 1 Función de Pre-énfasis



b. Encuadre

La señal de sonido es una señal variable en el tiempo, pero será estable a corto plazo en tiempo, es decir que tiene la característica que se llama la estacionariedad a corto plazo. Entonces, cortar un pedazo de sonido estable a corto plazo como un cuadro a partir del sonido continuo, lo que es la técnica de encuadre. La duración de cada cuadro es generalmente 10 ~ 30ms, lo que se analiza es la serie temporal de características de sonido, así que este es un método sobre pronósticos con análisis de series de tiempo.

Además, nuestro objeto en este caso de detección de fallas es la caja de cambios, no es el reconocimiento de la voz humana, por lo tanto, debe monitorear la señal de sonido completa para la maquinaria. Las señales de la máquina industrial se ven afectada por los factores externos, es decir que las muestras quizás provengan de la misma máquina pero con distintos parámetros de funcionamiento, diferentes condiciones ambientales, como la temperatura y la humedad, por ello, el error de reconstrucción por el entrenamiento aumentaría inesperadamente si solo se considera unos bloques elegidos a corto plazo de la señal completa.

c. La agregación de ventanas

El propósito de la agregación de ventanas es para enfatizar las ondas de cada cuadro y debilita el resto, es decir que se usa para evitar las discontinuidades de los bloques analizados al principio y la final. Hay muchas funciones para la agregación de ventanas como la ventana de Hamming, la ventana de Hann, la ventana rectangular y la ventana triangular, etc.

En este caso, la ventana de Hann se utilizará para extraer la característica MFCC, que tiene una buena resolución de frecuencia y menos leakage en espectro. La función de Hann es:

$$w(n) = \begin{cases} 0.5[1 - \cos(\frac{2\pi n}{N-1})] & (0 \leq n \leq N-1) \\ 0, & \end{cases}$$

Figura. 2 Función de ventana Hann

d. La detección y respuesta de Endpoints

Se refiere al uso de tecnología digital para encontrar el punto de inicio y la posición final de la señal de sonido, además de aumentar la tasa de reconocimiento de sonido, sino que también reduzca el cálculo. Los métodos de detección de Endpoints incluyen energía a corto plazo, la amplitud promedio, la tasa de cruce por cero (ZCR), etc. Entre ellos, la energía y la tasa de cruce por cero es el más común, por lo tanto, los extrae como las características y los envía al modelo de aprendizaje automático.

2.2 Extracción de características

Una de las dificultades en la detección de anomalías es la extracción de características, cuyas características extraídas a partir de la señal acústica pueden contener las máximas informaciones disponibles para que el aprendizaje automático encuentre una regla específica entre ellos. Por lo tanto, es esencial que seleccionará las características adecuadas tanto como sea posible para el entrenamiento de modelos.

En este caso, ha seleccionado 13 características como los objetos de aprendizaje, que son 'MFCCs', 'SpectralCentroid', 'BandWidth', 'SpectralRolloff', 'ZeroCrossingRate', 'PolyFeature', 'ChromaSTFT', 'Curtosis', 'MaxAmplitude', 'Std', 'Mean', 'Short-term Energy' y 'RMS', luego analizará algunas de ellas más populares.



2.2.1 Coeficientes Cepstrales en las Frecuencias de Mel (MFCCs)

MFCCs son los coeficientes para la representación del habla basados en la percepción auditiva humana, aunque se usa generalmente en los casos de reconocimiento de la voz, pero también se puede usar en el caso de la detección de fallas como una característica eficaz. MFCCs es una característica desarrollada en base al sistema auditivo humano cuya característica es que cuanto menor es la frecuencia, mayor la resolución será, en contrario, cuanto mayor es la frecuencia, menor la resolución será. Debido a que Cepstrum tiene la ventaja de que puede separar las frecuencias de espectro en altas y bajas, por eso, es realizable que utilizar MFCCs como el objeto específico para aprendizaje automático.

Generalmente, MFCCs pasa por varios pasos: pre-énfasis, encuadre, agregación de ventana, transformada de Fourier (FFT), banco de filtro Mel, y transformada de coseno discreta (DCT). Por lo tanto, MFCCs es el coeficiente de Cepstrum, que debe realizar la transformada de DCT en los coeficientes del espectro Mel después de 'log'. De hecho, cuando se usa realmente la señal en el dominio de la frecuencia como una señal en el dominio del tiempo, se realiza la transformada del dominio de la frecuencia para obtener las magnitudes de frecuencia en el dominio de la frecuencia mapeada.

Las siguientes 4 muestras de sonido normal y sonido anormal se extraen de la misma parte de una caja de cambio (section 1) desde el conjunto de prueba, y tras la extracción de MFCCs a través de Python, observa los cambios en el dominio del tiempo y el dominio de la frecuencia sobre esta característica. El número de MFCCs se establece en 20 en este caso y cada uno es el paso banda.

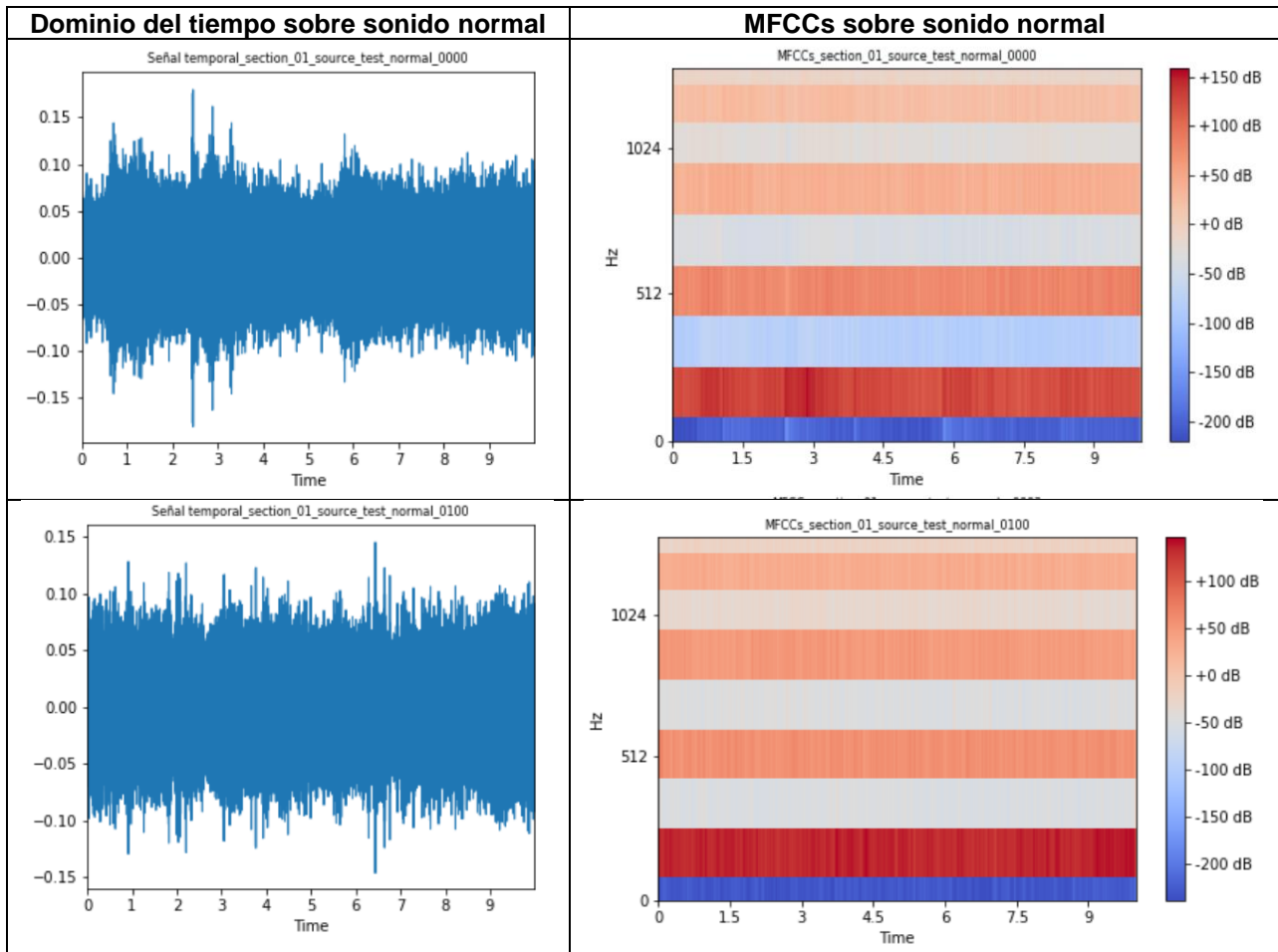
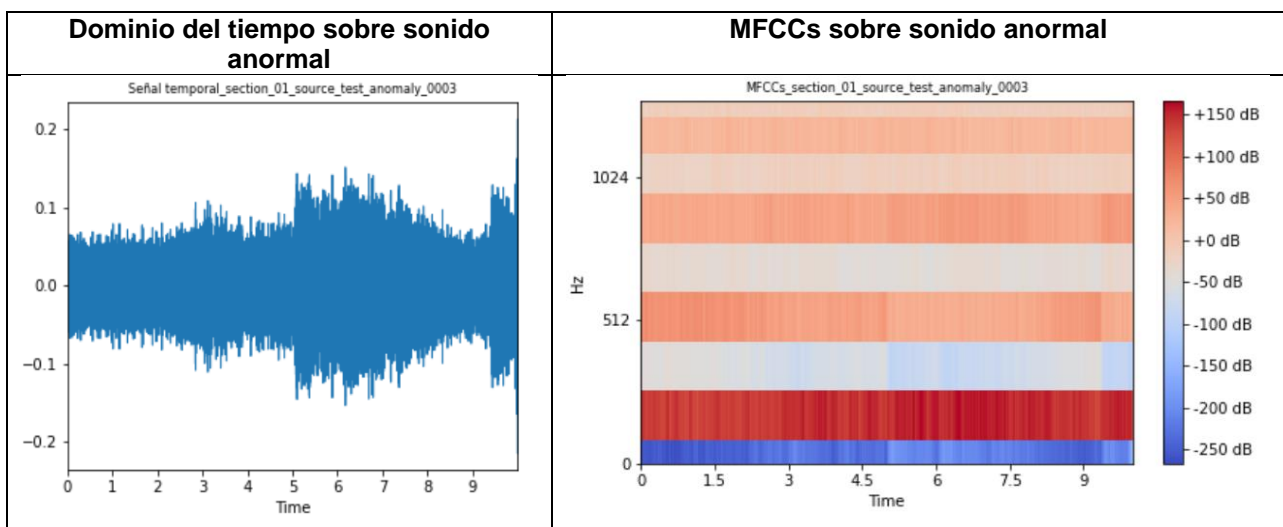


Figura. 3 Señal temporal y MFCCs
(Section_01_source_test_normal_0000,
Section_01_source_test_normal_0100)



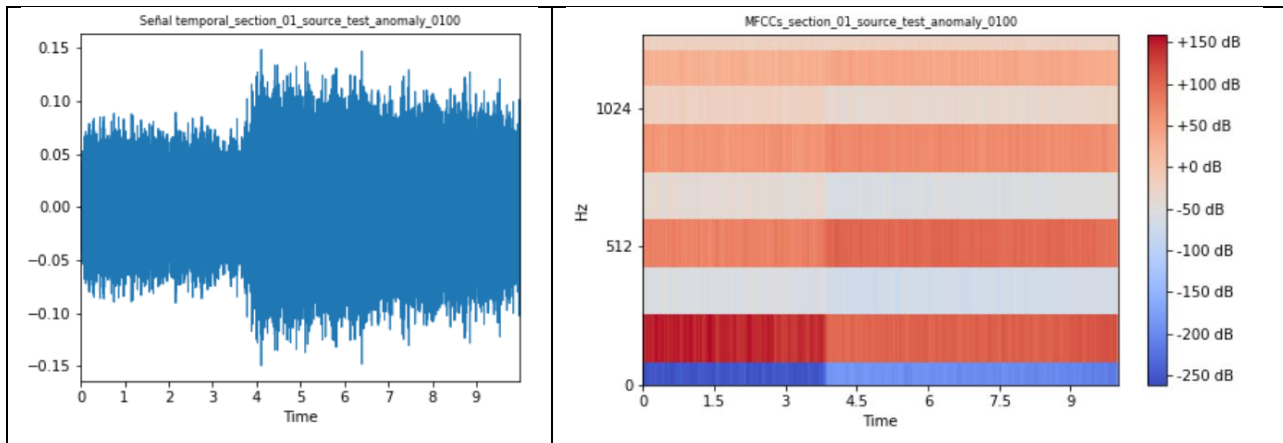


Figura. 4 Señal temporal y MFCCs
(Section_01_source_test_anomaly_0003,
Section_01_source_test_anomaly_0100)

Sobre todo, se puede observar que el cambio de la señal de anomalías en espectrograma está debajo de 2000Hz, en este caso, el primer paso de la extracción es encuadre para filtrar algunos ruidos pequeños en la señal y luego agrega la ventana de Hann, dado que está en la parte de frecuencia baja y lo que nos concentra es el cambio de frecuencia de anormal sonido, por eso, no es necesario a usar la técnica de pre-énfasis.

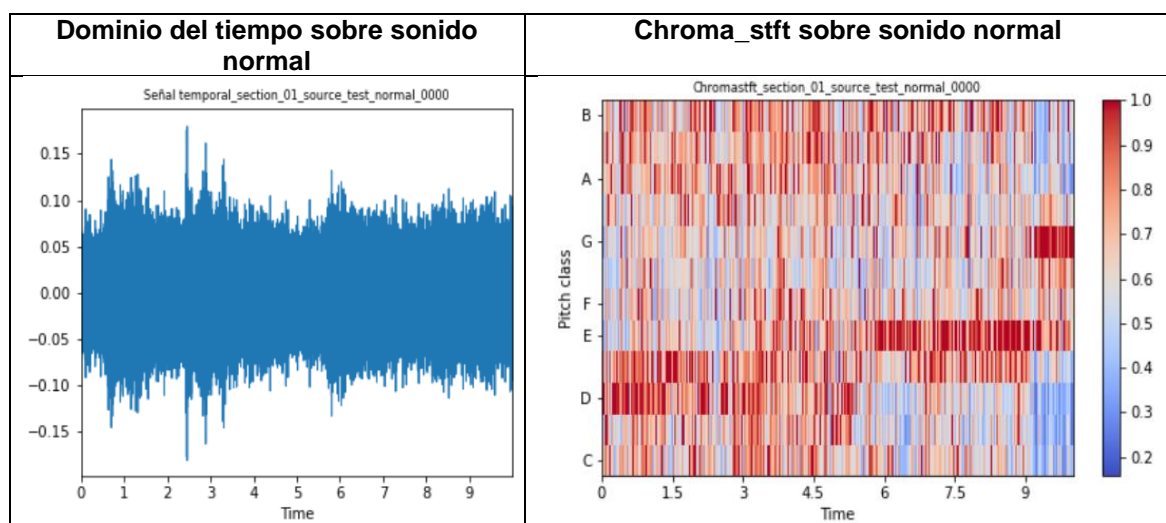
La frecuencia del sonido normal es relativamente estable, pero la frecuencia del sonido anormal puede ver los cambios claros desde el bloque de color. Además, puede ver un cambio más claro entre los decibeles altos y bajos en bloques de color en cuanto la máquina pasaría una anomalía, figura 3 y 4, cuando la onda en el dominio del tiempo sube mucho, se puede ver que la intensidad de la señal también cambia mucho, hay decibelios más altos en la parte de frecuencia alta.

2.2.2 Cromaticidad-STFT(Chroma-STFT)

La frecuencia cromática es una característica específica e interesante para los sonidos, ya que todo el espectro se proyecta en 12 intervalos que representan 12 semitonos distintos de octavas, es decir que cada semitono o cromaticidad representa la energía de 12 niveles sonoros en un período de tiempo (un cuadro), la energía del mismo tono de diferentes octavas se acumula, por tanto, la cromatograma es una secuencia de vectores de cromaticidad en la que el eje horizontal es la serie de tiempo y el eje vertical el tono.

El proceso de extracción es que el primer paso es la transformada de Fourier para audios, de forma que los transforme del dominio del tiempo al dominio de la frecuencia, el segundo paso es la reducción de ruido y la afinación como afinar un instrumento musical a una frecuencia estándar, el tercer paso es que convierta el tiempo absoluto en cuadros según la longitud elegida y anota la energía de cada tono en cada cuadro para hacer un gráfico de tono. Al final, va a acumular la energía de las notas musicales del mismo tiempo y mismo tono pero diferentes octavas para formar una cromatograma.

Observa el cambio de tono en el cromatograma cuando utiliza las mismas muestras de sonido.



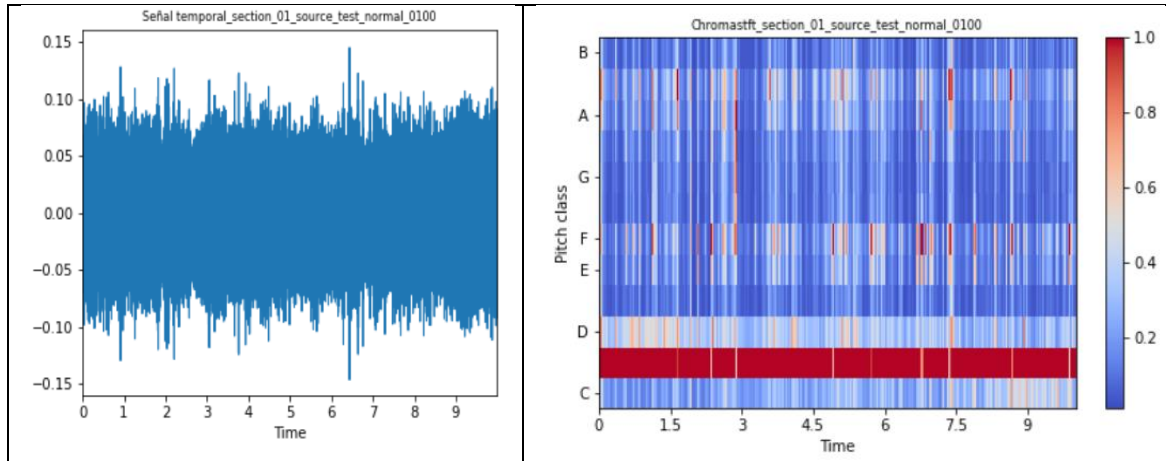


Figura. 5 Señal temporal y Cromaticidad-STFT
(Section_01_source_test_normal_0000,
Section_01_source_test_normal_0100)

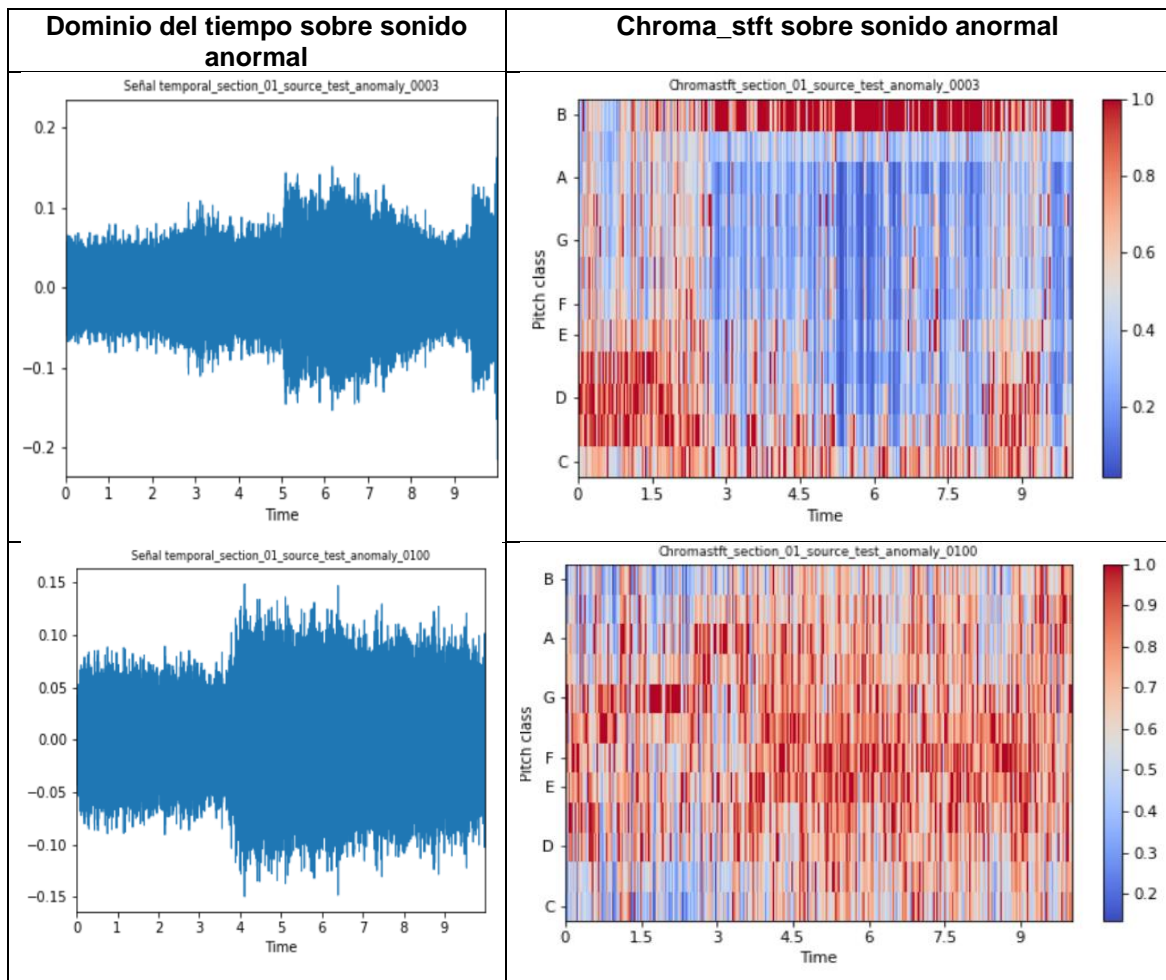


Figura. 6 Señal temporal y Cromaticidad-STFT
(Section_01_source_test_anomaly_0003,
Section_01_source_test_anomaly_0100)

Sobre todo, cuando la caja de cambios funciona en el estado sano, el tono se distribuye principalmente entre C y G, pero si se produce un ruido anormal, el tono subirá a A y B, por ello, puede considerarse que si se ha pasado una anomalía. Normalmente, el método de la detección de falla por tono del sonido es bastante común para las máquinas industriales, ya sea que se pase la falla abrasiva o por fatiga, se producirá un sonido anormal más ruidoso.

2.2.3 Centroides Espectral

El centroide del espectro es uno de las características físicas muy importantes que describe la propiedad de timbre, también es el centro de gravedad de las frecuencias. Su unidad es Hertzio que significa que es la frecuencia promedio ponderada por energía dentro de un cierto rango de frecuencia.

Es la información importante de la señal acústica sobre la distribución de frecuencia y de energía, porque describe el brillo del sonido, es decir que los sonidos más oscuros y profundos tienen más frecuencias bajas y el centroide espectral es más bajo. Por el contrario, la mayoría de sonidos más alegres y brillantes cuyas frecuencias serán más altas y el centroide espectral será más alto.

En este caso, se utiliza la misma muestra que antes para visualizar el centroide espectral. Después de calcularlo en cada cuadro, se lleva a cabo la normalización para mejorar la visualización.

$$C_t = \frac{\sum_{n=1}^N n M_t[n]}{\sum_{n=1}^N M_t[n]}$$

$M_t[n]$: La magnitud de la transformada de Fourier en el cuadro t y el punto de frecuencia n .

Figura. 7 Función de Centroides

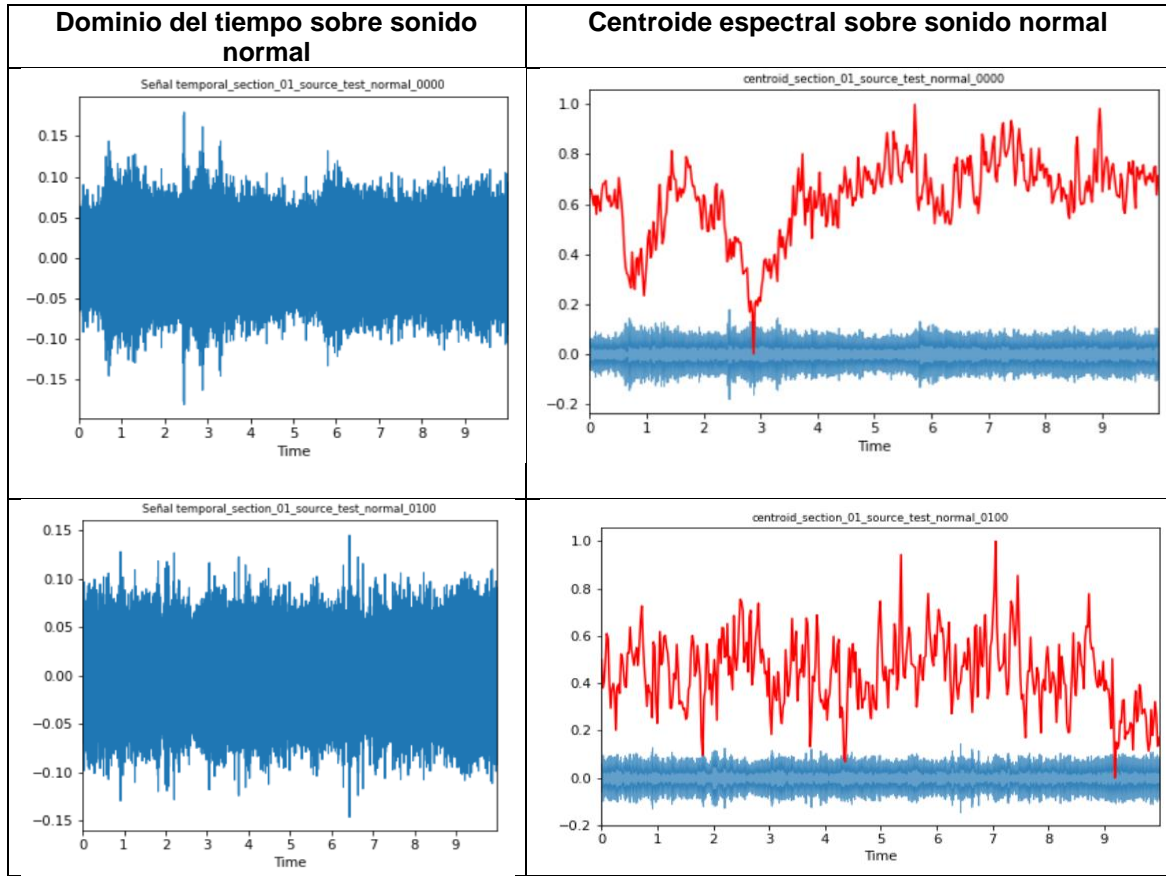
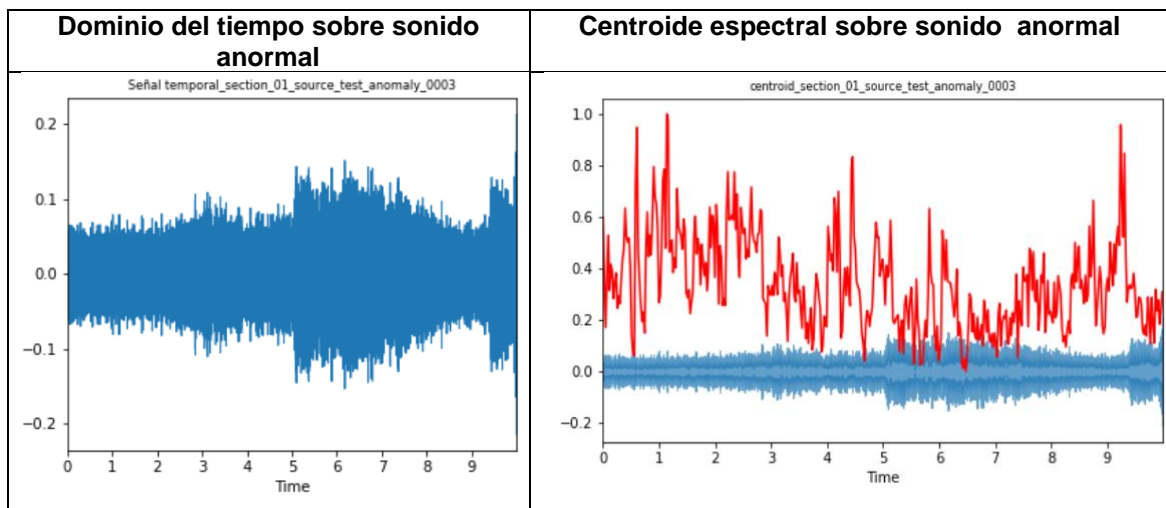
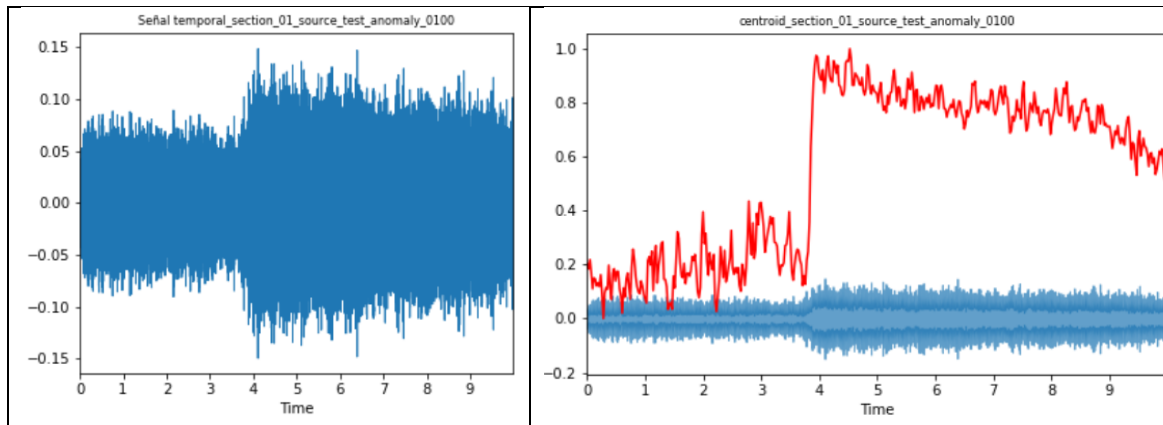


Figura. 8 Señal temporal y Centroide Espectral (Section_01_source_test_normal_0000, Section_01_source_test_normal_0100)





**Figura. 9 Señal temporal y Centroide Espectral
(Section_01_source_test_anomaly_0003,
Section_01_source_test_anomaly_0100)**

Dado que las muestras de sonido no están etiquetadas, por eso, no sabe con qué velocidad funcionó esta caja de cambios. La única forma es solo ampliar la cantidad de muestras para que la computadora aprenda la regla de distribución de frecuencia tanto como sea posible.

La verdad es que la distribución de timbre depende de qué falla pasaría en la máquina. Por ejemplo, en muestra anormal de la Figura 4, el centroide espectral va a subir abruptamente en el cuarto segundo, el sonido es más brillante y nítido, más energía se concentra en la parte de frecuencia alta, lo que podría pasar una anomalía.

2.2.4 RollOff Espectral

Rolloff espectral (Spectral Rolloff) es una medida de la forma de la señal, se calcula el coeficiente de caída (Rolloff) para cada cuadro de la señal, por eso, representa la frecuencia en un porcentaje específico de toda la energía espectral, normalmente es 85%. De hecho, si el umbral de energía se establece en un valor del 50% en el cálculo de Rolloff, es posible que el centroide espectral sea igual al Rolloff espectral. La frecuencia de Rolloff se puede utilizar para distinguir entre las señales armónicas por debajo de Rolloff y los sonidos ruidosos por encima de Rolloff.

$$\sum_{n=1}^{R_t} M_t[n] = 0.85 \sum_{n=1}^N M_t[n]$$

La media y la variación de Rolloff a lo largo de los cuadros de tiempo en la ventana de textura se utilizan como características

Figura. 10 Función de RollOff

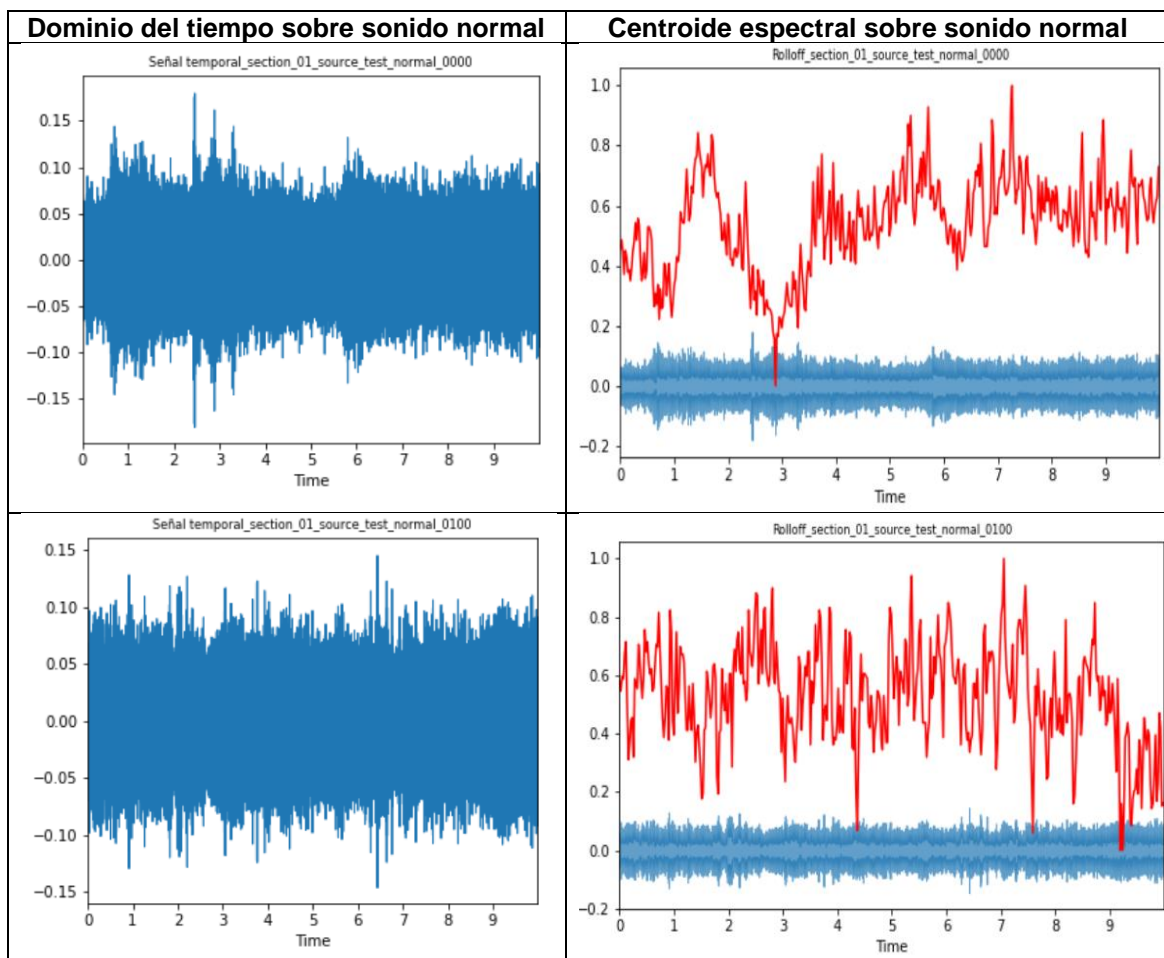


Figura. 11 Señal temporal y RollOff
(Section_01_source_test_normal_0000,
Section_01_source_test_normal_0100)

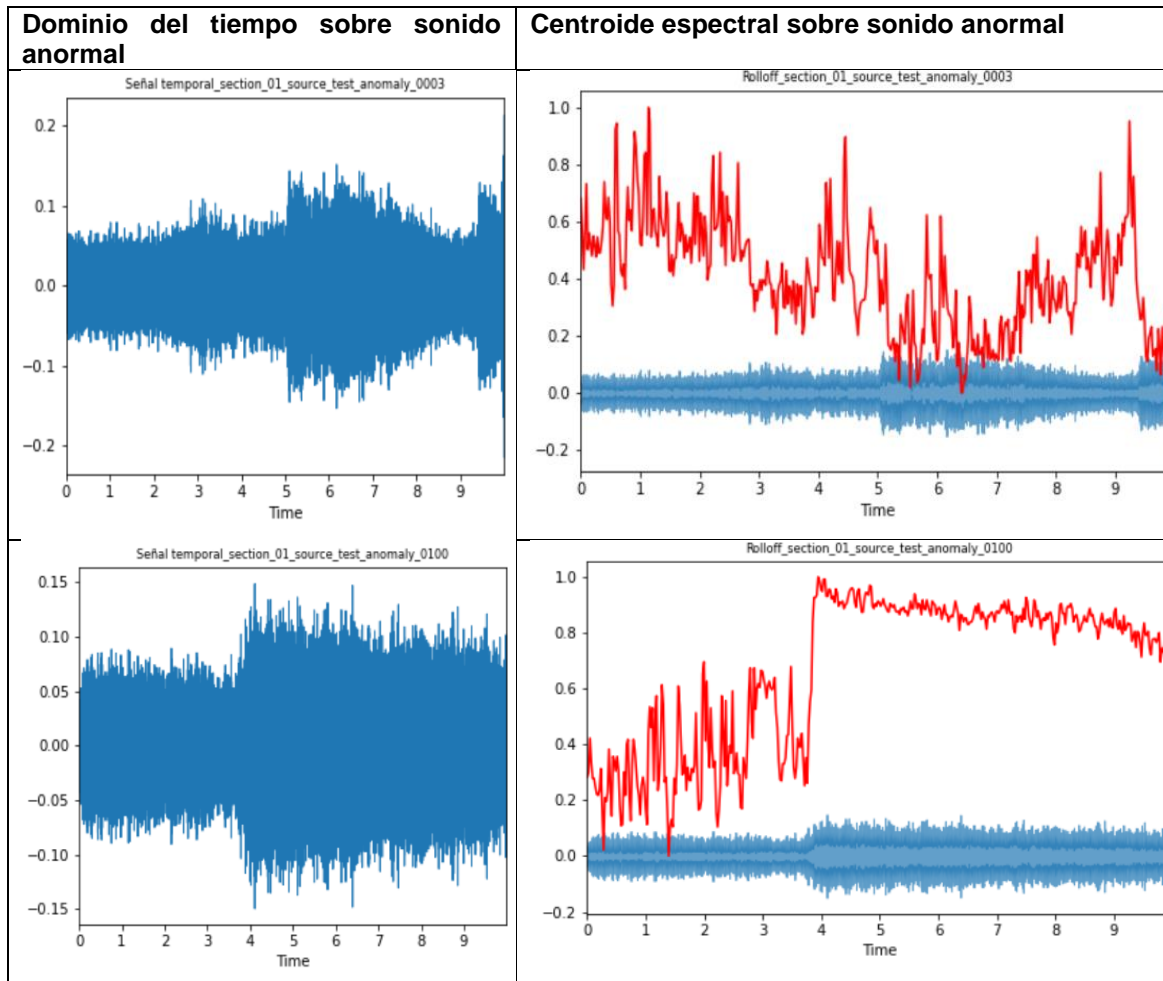


Figura. 12 Señal temporal y RollOff
(Section_01_source_test_anomaly_0003,
Section_01_source_test_anomaly_0100)

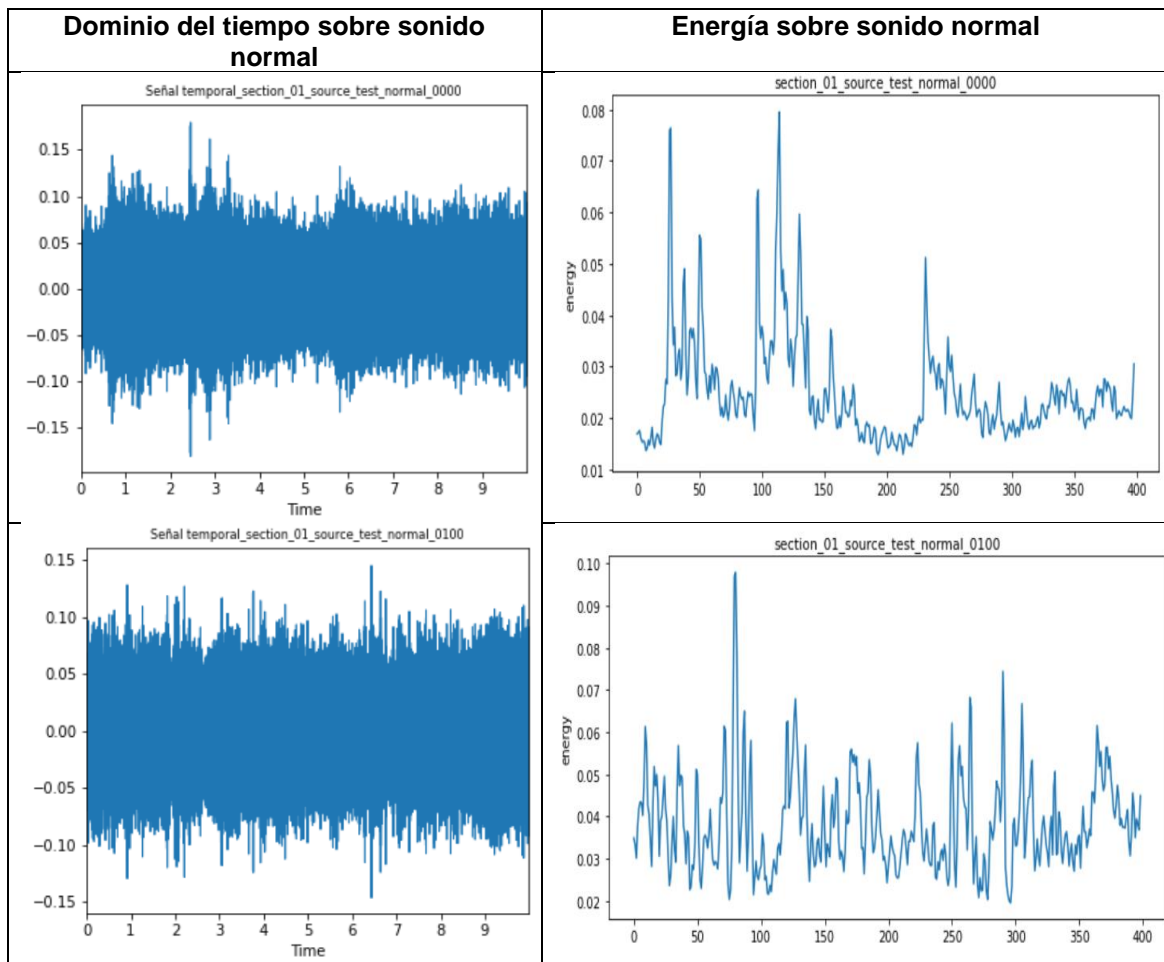
2.2.5 Energía a Corto Plazo

La energía a corto plazo es la suma de los cuadrados de la señal en cada cuadro, que muestra la intensidad de la energía de la señal. Además, la entropía de energía describe la distribución de la señal en el dominio del tiempo, que refleja la continuidad.

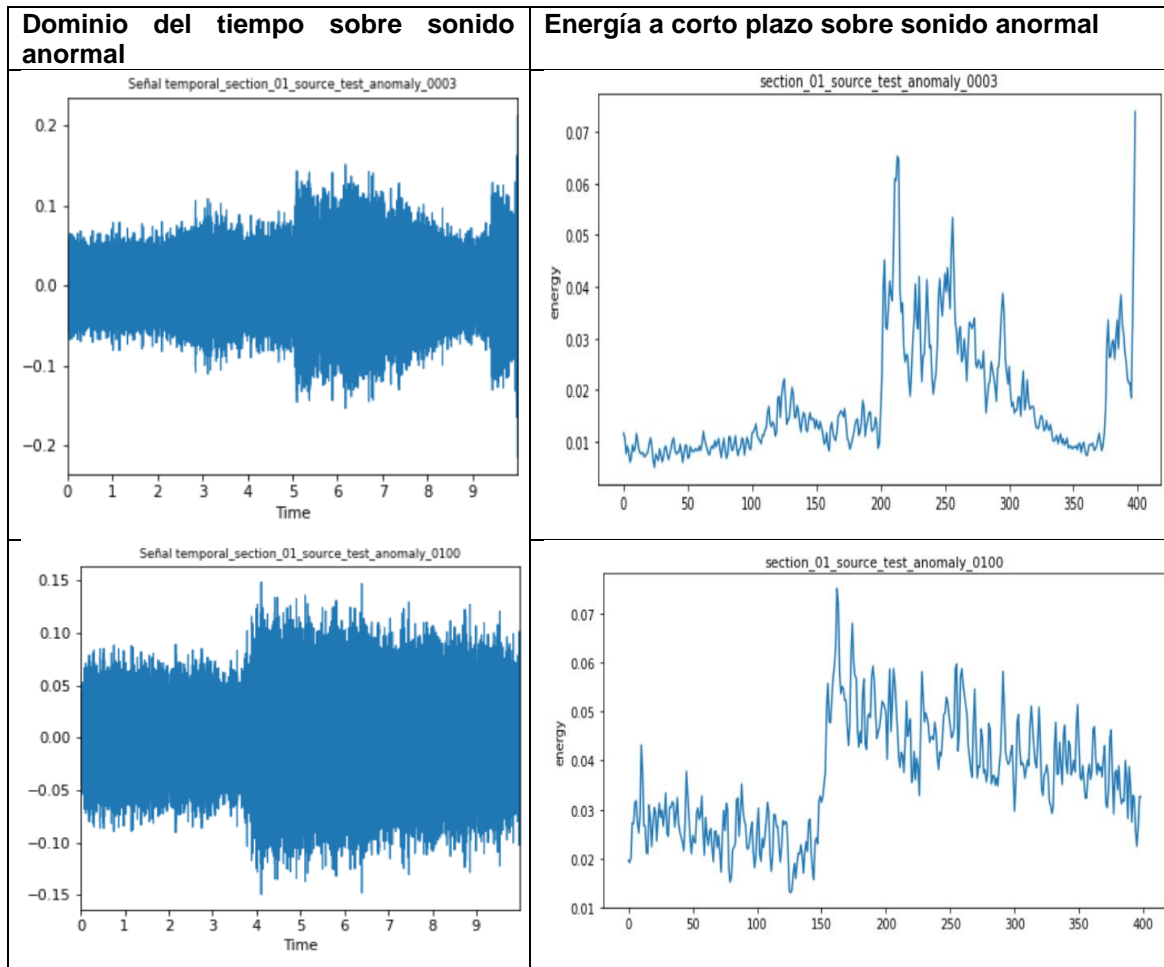
A pesar de que MFCCs y el centroide espectral puede representar las propiedades de la señal en el dominio de frecuencia, pero la energía a corto plazo puede mostrar las de la señal en el dominio del tiempo, también el espectro de la energía es un método de descripción para reflejar los cambios de amplitud.

Se puede observar la diferencia entre el sonido válido y el ruido de fondo a partir de la energía. Generalmente, la energía del sonido válido es mayor que la del ruido blanco, por lo tanto, se usa la energía a corto plazo para distinguir la señal válida del ruido de fondo.

Además de la observación de intensidad, hay una función más conocida que es la verificación del punto inicial y el final de la señal válida integrado con el uso de la tasa de cruce por cero. Cuando analiza los sonidos de la máquina industrial, sería necesario observar las partes contrarias de los dos en el gráfico en el que contiene más información.



**Figura. 13 Señal temporal y Energía a corto plazo
(Section_01_source_test_normal_0000,
Section_01_source_test_normal_0100)**



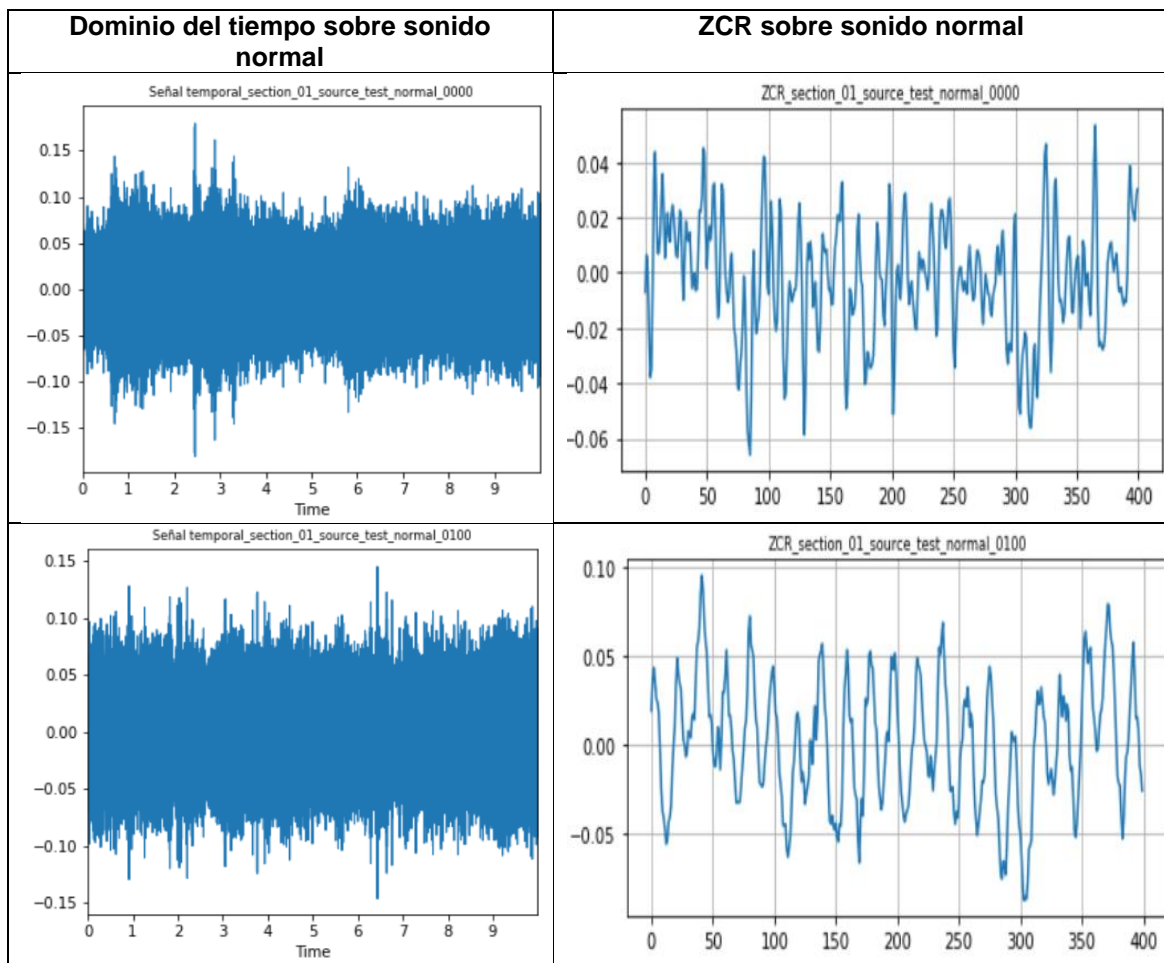
**Figura. 14 Señal temporal y Energía a corto plazo
(Section_01_source_test_anomaly_0003,
Section_01_source_test_anomaly_0100)**

Sobre todo, la energía a corto plazo es la característica en dominio del tiempo cuya amplitud cambiará claramente con el tiempo, actualmente, se permite observar directamente el cambio de energía en cada cuadro de la señal. Por lo tanto, la energía media a corto plazo se puede ser una referencia importante para la selección del umbral.

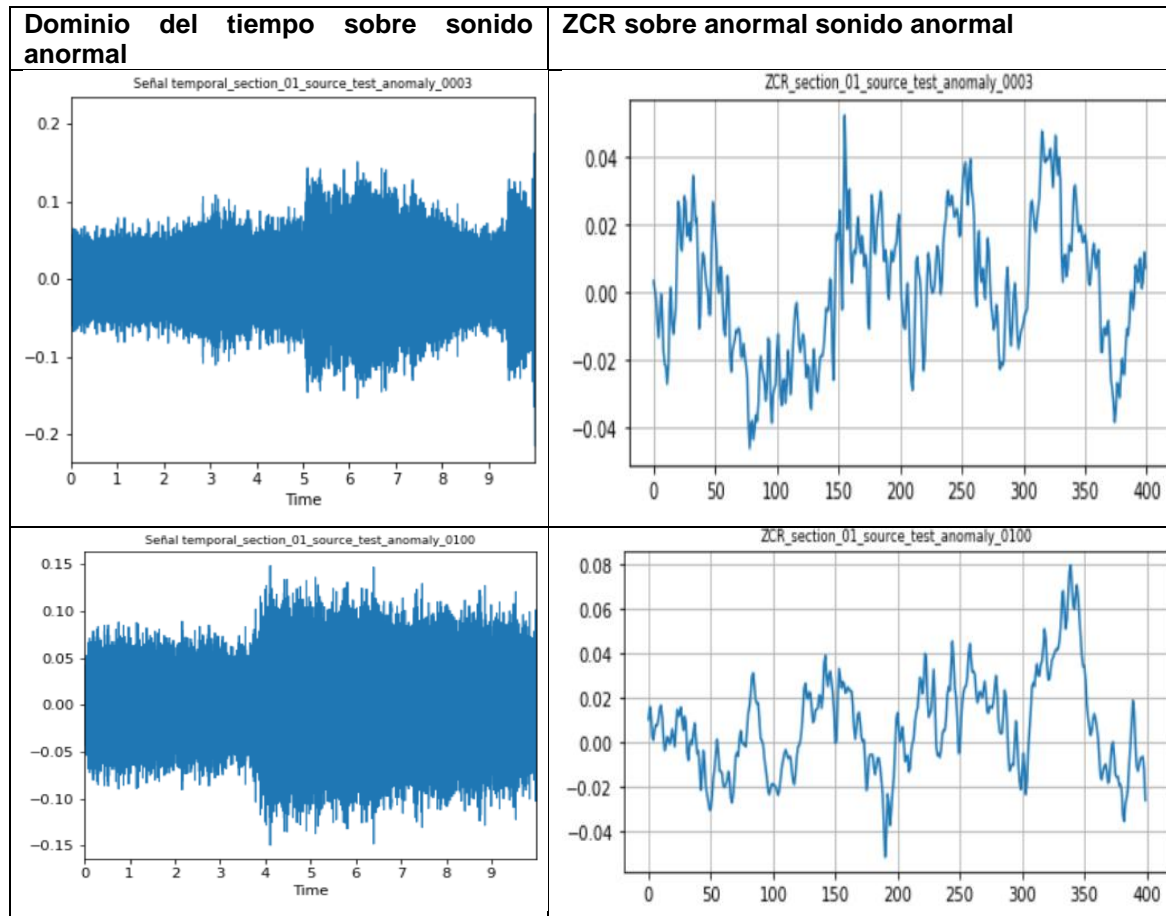
2.2.6 Tasa de Cruce por Cero(Zero Crossing Rate)

La tasa de cruce por cero se refiere a las veces que la señal de sonido pasa por el punto cero de positivo a negativo o de negativo a positivo en cada cuadro, es decir que se puede observar que la onda de una señal continua en dominio del tiempo pasa por el eje horizontal.

El siguiente es el diagrama de la tasa de cruce por cero en 400 segmentos de cuadro.



**Figura. 15 Señal temporal y ZCR
(Section_01_source_test_normal_0000,
Section_01_source_test_normal_0100)**



**Figura. 16 Señal temporal y ZCR
(Section_01_source_test_anomaly_0003,
Section_01_source_test_anomaly_0100)**

La siguiente tabla muestra la energía promedio a corto plazo y la tasa de cruce por cero en 400 cuadros, en la que se extraen un total de 9 muestras normales y 9 muestras anómalas a partir de la misma caja de cambios.

Nombre de muestra	la energía promedio a corto plazo	la tasa de cruce por cero
Section_01_source_test_normal_0000	0.024882	71
Section_01_source_test_normal_0100	0.038344	45
Section_01_source_test_normal_0050	0.031239	57
Section_01_source_test_normal_0001	0.044987	52
Section_01_source_test_normal_0090	0.030768	29
Section_01_source_test_normal_0030	0.049767	50
Section_01_source_test_normal_0005	0.040469	53
Section_01_source_test_normal_0070	0.051392	46

Section_01_source_test_normal_0064	0.048093	44
Section_01_source_test_anomaly_0003	0.017731	46
Section_01_source_test_anomaly_0100	0.036614	37
Section_01_source_test_anomaly_0050	0.027484	101
Section_01_source_test_anomaly_0001	0.045671	30
Section_01_source_test_anomaly_0090	0.055186	70
Section_01_source_test_anomaly_0030	0.045327	25
Section_01_source_test_anomaly_0005	0.049258	62
Section_01_source_test_anomaly_0070	0.046708	46
Section_01_source_test_anomaly_0064	0.058134	49

Tabla. 1 ZCR para muestras de Section_01_source_test

Las partes con mayor energía promedio a corto plazo se distribuyen principalmente en muestras anómalas, mientras que la tasa de cruce por cero cambia poco y el valor es relativamente estable en las normales.

2.2.7 Curtosis

Curtosis es una característica de forma de la distribución de frecuencias o probabilidad, también es un parámetro estadístico de la inclinación de todas las distribuciones de valores. Se muestra un pico que debe compararse con la distribución normal.

Cuando el valor de curtosis es igual al 0, significa que la distribución general de datos es la misma como la normal, que es la distribución mesocúrtica. Si la curtosis es mayor que 0, la curva será más apuntada y con colas más gruesas que la normal, indica que la distribución de datos es más abrupta que la normal, llamada la distribución leptocúrtica. Cuando es menor que 0, la curva será menos apuntada y con colas menos gruesas que la normal, significa que la distribución es plana comparado con la normal, llamada la distribución platicúrtica. Además, cuanto mayor sea el valor absoluto de la curtosis, mayor será la diferencia de la inclinación entre su distribución y la normal.

TRABAJO FIN DE MÁSTER

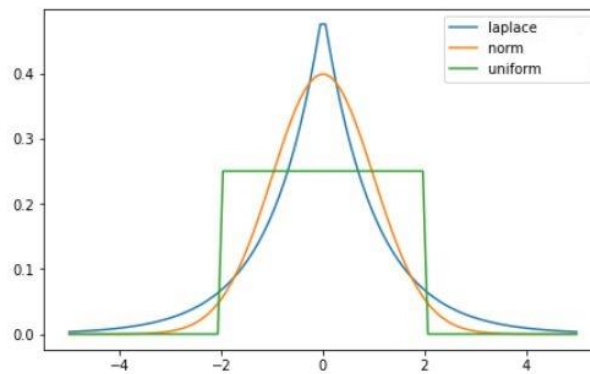


Figura. 17 La curva leptocúrtica, mesocúrtica y platicúrtica, fuente: `scipy.stats.kurtosis`

Nombre de muestra	Curtosis
Section_01_source_test_normal_0000	0.3535
Section_01_source_test_normal_0100	0.0129
Section_01_source_test_normal_0050	0.0236
Section_01_source_test_normal_0001	0.0188
Section_01_source_test_normal_0090	0.4263
Section_01_source_test_normal_0030	0.0098
Section_01_source_test_normal_0005	0.0755
Section_01_source_test_normal_0070	-0.1088
Section_01_source_test_normal_0064	-0.0163
Section_01_source_test_anomaly_0003	1.1075
Section_01_source_test_anomaly_0100	0.2431
Section_01_source_test_anomaly_0050	-1.2002
Section_01_source_test_anomaly_0001	-0.0283
Section_01_source_test_anomaly_0090	-0.0586
Section_01_source_test_anomaly_0030	0.0119
Section_01_source_test_anomaly_0005	-0.0468
Section_01_source_test_anomaly_0070	-0.0492
Section_01_source_test_anomaly_0064	-0.1213

Tabla. 2 Curtosis para muestras de Section_01_source_test

En el análisis de ruido, el valor alto de curtosis representa el ruido bajo y la frecuencia baja, pero se puede observar que la frecuencia de trabajo de caja de cambios es alta y el sonido es muy ruidoso. La curtosis de puntos anormales tiene valores más negativos, así no es accidental. Por otro lado, a pesar de que los datos de entrenamiento fueran completamente normales, habría unas muestras de interferencia. A menos que la condición de muestreo fueran consistentes en el mismo entorno, el modelo predecirá con mayor precisión.

2.3 Lista de características acústicas

Se extrae las características en este caso por varias Python bibliotecas sobre procesamiento de audio. A continuación, se muestra una tabla de principales características acústicas.

Clase	Característica	Definición
Energía	Energía de raíz de la media cuadrática	El valor medio de la energía de la señal durante un período de tiempo.
Dominio de tiempo	Autocorrelación	La similitud entre la señal original y su cambio por tiempo.
	Tasa de cruce por cero (ZCR)	Las veces que la señal pasa el punto de cero en el período de tiempo.
Dominio de frecuencia	Centroide espectral	El centro de gravedad de las frecuencias, que describe la propiedad de timbre.
	RollOff espectral	La medida de la forma de la señal, representa la frecuencia en un porcentaje específico de toda la energía espectral, normalmente es 85%.
	MFCC	Coefficientes Cepstrales en las Frecuencias de Mel, que es una característica desarrollada en base al sistema auditivo humano cuya característica es que cuanto menor es la frecuencia, mayor la resolución será.
	Ancho de banda espectral	Cuando mayor sea el rango de frecuencia de la señal, mayor será el ancho de banda espectral.
Teoría musical	Frecuencia fundamental F0	Frecuencia de tono
	Sonido de la discordia	La desviación de la frecuencia de sobretonos y la de fundamental.
Característica perceptiva	Cromaticidad	La energía de 12 niveles sonoros en un período de tiempo
	Curtosis	Un parámetro estadístico de la inclinación de todas las distribuciones de valores. Se muestra un pico que debe compararse con la distribución normal.

Tabla. 3 Lista de características acústicas



3 FUNDAMENTOS DEL APRENDIZAJE AUTOMÁTICO

3.1 Descripción del aprendizaje profundo

El aprendizaje automático es el proceso de estudio de la experiencia a partir de una gran cantidad de muestras, significa que puede analizar los datos mediante algoritmos, luego toma la decisión y la predicción para los eventos del mundo real, por tanto, contiene varias ramas de algoritmos distintos de aprendizaje como el aprendizaje supervisado (la clasificación), el aprendizaje no supervisado (el análisis de grupos), el aprendizaje semi-supervisado, el aprendizaje profundo y el aprendizaje por refuerzo, etc.

Por ejemplo, el aprendizaje profundo es un concepto nuevo después del aprendizaje automático, que es una rama del aprendizaje automático.

En 2016, AlphaGo, el sistema de inteligencia artificial sobre Go desde Google, que derrotó al campeón mundial de Go, fue la primera máquina que ganó Go en una tabla 19x19. Por eso, la victoria de AlphaGo fue un hito en la historia del desarrollo de inteligencia artificial. Al mismo tiempo, el aprendizaje profundo se ha convertido en una palabra de búsqueda más conocida por Google, cuya aplicación se ha expandido desde el reconocimiento de imágenes hasta varios campos del aprendizaje automático.

De hecho, el aprendizaje profundo no es un método independiente, sino que también integre con el uso de aprendizaje supervisado o del no supervisado para el entrenamiento de redes neuronales. Sin embargo, la principal diferencia con el aprendizaje automático tradicional es que puede ingresar directamente las señales originales en la red neuronal profunda sin preprocesamiento ni la ingeniería de característica, luego puede generar automáticamente las características adecuadas a través de estudio por las múltiples capas, al final, la predicción también será bastante precisa.

El problema de aprendizaje automático tradicional está en la extracción de características, que es necesario a encontrar las

TRABAJO FIN DE MÁSTER

características manualmente antes del proceso de entrenamiento. Debido a que requiere una gran cantidad de conocimiento relacionado con las propiedades de maquinaria, se convierte en un principal obstáculo para las tareas de aprendizaje automático. Dicho de otro modo, si no tiene los suficientes bases de dato o una computadora potente, no es tan fácil ejecutar un modelo complejo de alta potencia, así se pide una extracción de características que lleva mucho tiempo y esfuerzo.

El aprendizaje profundo aún tiene sus competencias:

- a. Se pide una gran cantidad de datos para el entrenamiento pero no es ideal para la tarea con un poco de muestras.
- b. No es igual como un cerebro humano. Debido a 'domain shift', aunque una misma máquina funciona a diferentes velocidades, diferentes cargas y ruidos de fondo, es probable que cause una gran interferencia en el resultado de la predicción.

La definición de 'Domain shift' es un cambio de distribución del espacio de entrada causado por la descripción variable de la observación o el cambio en el sistema de observación. Aunque el factor causal es igual entre los conjuntos de datos, pero hay distintos factores confusos, lo que causa la diferencia entre la distribución en conjuntos de datos.

Sobre todo, el aprendizaje automático tradicional y el aprendizaje profundo tienen sus propias ventajas y desventajas cuando se enfrentan a las tareas diferentes. En la realidad, es muy importante elegir el modo apropiado de aprendizaje según la condición específica.



3.2 Diagrama del aprendizaje automático

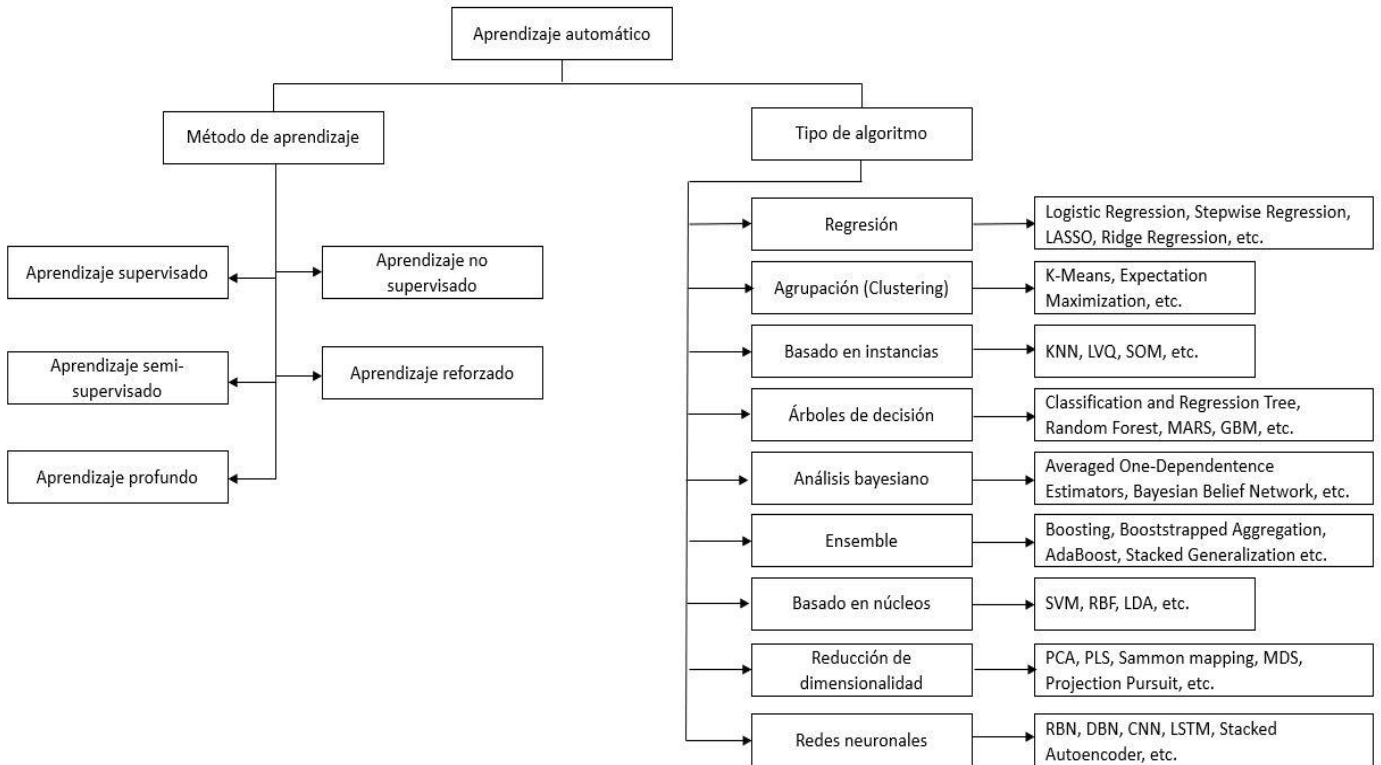


Figura. 18 Diagrama del aprendizaje automático



3.3 La red neuronal profunda (Deep Neural Network)

La red neuronal del aprendizaje automático es la simulación de la conexión de las neuronas de cerebro humano, está basado en el perceptrón pero es mejor. La red neuronal profunda consiste en muchas capas ocultas, también se llama Perceptrón multicapa.

Las capas de red neuronal interna están formadas por tres tipos, que son capa de entrada, capa oculta y capa de salida. Generalmente, la primera capa es la de entrada, la última capa es la de salida y la capa intermedia es una capa oculta. Generalmente, se usa la conexión completamente conectada entre los nodos de dos capas, pero no están conectados entre los nodos en la misma capa.

La forma de la capa de entrada está determinada por la dimensión del vector de entrada, la forma de la capa de salida está relacionado con las necesidades de la tarea. El número de capas ocultas se determina en base a datos, es decir que más complejos datos pide más números de capas.

3.3.1 Algoritmo hacia adelante

El cálculo hacia adelante comienza de la capa de entrada de la red neuronal y se propaga hacia la capa de salida. Como las capas están conectadas entre sí, el valor ponderado por el valor de salida desde la capa anterior y agrega una parcialidad, luego, le da una función no lineal como la función de activación (ReLU, Sigmoid, etc.) el resultado obtenido es el valor de salida de la siguiente capa.

El algoritmo hacia adelante será varío determinado por la diferente estructura de red neuronal. Por ejemplo, la función básica de la capa completamente conectada (BP-FNN) es el siguiente:

$$z^{(l)} = W^{(l)} * a^{(l-1)} + b^{(l)}$$

$$a^{(l)} = f_l(z^{(l)})$$

l: Número de capa
f_l(.): La función de activación de capa *l*
N^(l) : Número de neuronas de capa *l*
*W^(l) ∈ R^{N^(l) * N^(l-1)}* : Matriz de pesos posicionales de capa *l* – 1 a capa *l*
b^(l) ∈ R^{N^(l)} : bias entre capa *l* – 1 a capa *l*
z^(l) ∈ R^{N^(l)} : Valor de entrada de capa *l*
a^(l) ∈ R^{N^(l)} : Valor de salida de capa *l*

Figura. 19 Algoritmo hacia adelante

3.3.2 Algoritmo de retropropagación de errores

El algoritmo de retropropagación de errores es algoritmo de BP. Hay dos procesos de aprendizaje, uno es el de propagación hacia adelante y otro es el de propagación hacia atrás del error. El motivo es que el error de reconstrucción de alguna manera vuelve desde de las capas ocultas hasta la capa de entrada, dicho proceso es la retropropagación, de modo que pueda logra error (δ) entre la salida y el objetivo real, luego calcula el gradiente para cada neurona y actualiza constantemente los pesos (W , b) según al descenso del gradiente.

$$\delta^{(l)} = \frac{\partial \mathcal{L}(y, \hat{y})}{\partial z^{(l)}} = f'_i(z^{(l)}) \odot (W^{(l+1)})^T \delta^{(l+1)}$$

$\delta^{(l)}$: *error de cada capa*
 $f'_i(z^{(l)})$: *Derivada parcial de valor de salida*
 \odot *el producto tensorial*
 $\mathcal{L}(y, \hat{y})$: *Función de pérdida*

$$\frac{\partial \mathcal{L}(y, \hat{y})}{\partial W^{(l)}} = \delta^{(l)} (a^{(l-1)})^T \rightarrow W_{t+1}^{(l)} = W_t^{(l)} - \varepsilon (\delta^{(l)} (a^{(l-1)})^T) + \lambda W_t^{(l)}$$

$$\frac{\partial \mathcal{L}(y, \hat{y})}{\partial b^{(l)}} = \delta^{(l)} \rightarrow b_{t+1}^{(l)} = b_t^{(l)} - \varepsilon \delta^{(l)}$$

ε : *Tasa de aprendizaje*
 λ : *Coficiente de regularización*

Figura. 20 Algoritmo de retropropagación de errores

Sobre todo, el error de una neurona en la capa actual es igual al gradiente de la función de activación multiplicado por la suma ponderada de los errores de todas las neuronas en la capa anterior.



4 MÉTODO AE DE LA DETECCIÓN DE ANOMALÍAS PARA SONIDOS DE LA CAJA DE CAMBIOS

4.1 Aprendizaje automático supervisado y no supervisado

El aprendizaje automático supervisado es obtener un modelo óptimo a través del entrenamiento y la prueba sobre las muestras etiquetadas que ya sabemos, luego convierte los datos originales (Vector de número decimal o imágenes) en los datos de salida binarios. En otras palabras, los datos en el aprendizaje supervisado se han clasificado con antelación, por eso, sus muestras de entrenamiento y la prueba contienen información sobre características y etiquetas, lo que necesita hacer la computadora es la clasificación o la predicción.

El problema más común utilizado el aprendizaje supervisado es la regresión y la clasificación.

Por el contrario, el aprendizaje automático no supervisado cuyos datos de entrada no están etiquetados, por eso, no sale un resultado definido. Cuando la computadora carece de los conocimientos previos para etiquetar los datos, se utiliza un método de aprendizaje automático como el análisis de grupos o la reducción de dimensionalidad. En otras palabras, el conjunto de entrenamiento del algoritmo supervisado debe tener una gran cantidad de muestras anormales, así como las muestras normales. Después del entrenamiento, la computadora puede encontrar unos datos similares entre las nuevas muestras de prueba y las de entrenamiento.

Generalmente, el análisis de grupos sirve para descubrir la similitud entre las muestras desconocidas. La detección de anomalías es encontrar las diferencias entre las muestras.



Las diferencias	
Aprendizaje automático supervisado	Aprendizaje automático no supervisado
1. Encontrar unas reglas en el conjunto de entrenamiento y va a usarlas muy bien para las muestras de prueba	1. Solo hay un conjunto de datos en el que busca una cierta regla. A lo mejor esta regla no es adecuada para muestras de prueba.
2. Para reconocer objetos y agregar etiquetas a los datos.	2. Solo va a analizar los datos sin etiquetas, el reconocimiento o identificación no es el motivo final.
3. Reconocimiento de voz, Clasificación de imágenes y Traducción de idiomas	3. Análisis de grupos, Detección de anomalías

Tabla. 4 Las diferencias entre aprendizaje automático supervisado y no supervisado

4.2 Detección de anomalías por el aprendizaje automático no supervisado

Si hay un problema que se cumple una de las siguientes tres condiciones, se puede utilizar el algoritmo de detección de anomalías en lugar del algoritmo de aprendizaje supervisado.

- a. Hay pocas muestras positivas y muchas muestras negativas.
- b. Hay demasiados tipos anormales, pero es difícil que la computadora aprende las características de anomalías desde las muestras negativas.
- c. Los datos de las muestras positivas en el conjunto de prueba son completamente distintos con el los de entrenamiento, es decir que aparecen unas muestras anormales desconocidas.

Dado que hay muy pocas muestras anómalas de caja de cambios y muchos tipos de anomalías complejas, por tanto, el algoritmo de detección de anomalías es el más adecuado en este caso. Además, hay muchos métodos distintos como el análisis de grupos, el análisis de estadísticas, el bosque aislado y la red neuronal, etc. En este caso, se utilizará el método de autoencoder en la red neuronal.

El método de autoencoder es parecido a PCA, pero supera su limitación lineal cuando se utiliza una función no lineal de activación en las capas. Supone que los puntos anormales obedecen a la distribución diferente. El autoencoder entrenado por los datos negativos puede reconstruir y restaurar bien las muestras normales, sin embargo, es imposible que restaure los datos diferentes de la distribución normal, al

final sale un gran error de reconstrucción. Cuando dicho error es mayor que un cierto umbral, se anota como un valor anormal.

4.3 Indicadores de evaluación para el modelo

Según la complejidad de la detección de anomalías, se requiere múltiples indicadores para la evaluación. La razón es que los conjuntos de datos son los desequilibrados, ya que tiene más datos normales que los anómalos, por eso, además de 'Accuracy', es necesario introducir 'Precision', 'Recall' y 'F1 Score' basado en la confusión matriz.

Confusión Matriz			
	True	False	
Positives	TP	FP	TP+FP (Anomalías por la predicción)
Negatives	FN	TN	FN+TN (Normalidades por la predicción)
	TP+FN (Anomalías reales)	FP+TN (Normalidades reales)	

Tabla. 5 Confusión Matriz

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1\ Score = \frac{(a^2 + 1)Precision * Recall}{a^2(Precision + Recall)} = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (a = 1)$$

TP: True Positives
TN: True Negatives
FP: False Positives
FN: False Negatives

Figura. 21 Función de Indicadores



Sobre todo, 'Accuracy', que indica qué porcentaje de muestras son correctas en la predicción.

'Precision', que indica cuántas muestras positivas son verdaderas en toda las positivas de la predicción, por eso, una alta tasa de 'Precision' significa una baja tasa de FP.

'Recall', que indica cuántas muestras positivas son verdaderas en toda las positivas reales, por eso, una alta tasa de 'Recall' significa una baja tasa de FN.

'F1 Score', que es la media armónica de 'Precision' y 'Recall'. Parece que estos dos indicadores son igualmente importantes para la evaluación del modelo.

4.4 Sistema basado en BP-FNN

Autoencoder es una tecnología de aprendizaje automático no supervisada o semi-supervisada que utiliza redes neuronales para el aprendizaje de características. El método es que agrega un cuello de botella como un reloj de arena en el marco de la red para comprimir el contenido de aprendizaje, el propósito es eliminar la información inútil.

Si hay una cierta correlación entre varias características, por ejemplo, las características del dominio de la frecuencia se desarrollan a partir del espectro, dicha correlación no es necesario extraer en manual, se filtrarán y aprenderán automáticamente a través de la estructura del cuello de botella.

El modelo de BP-FNN es Autoencoder (AE), que está compuesto por el codificador y decodificador, La función del codificador es para encontrar la información clave desde los datos comprimidos, el decodificador se utiliza para reconstruir los datos comprimidos. La salida del codificador es la entrada del decodificador, por eso, la forma de la salida del decodificador es la misma como la entrada del codificador. En resumen, AE tiene tres características:

- a. AE solo puede comprimir los datos parecidos a los de entrenamiento, pero no puede hacer nada para los desconocidos.
- b. Se pasa la pérdida cuando los datos están descomprimidos.

TRABAJO FIN DE MÁSTER

- c. Dado que la entrada y la salida son parecidas, por tanto, tiene la capacidad de aprender automáticamente la codificación sin etiquetas de características.

La red neuronal de capa completamente conectada de retropropagación (BP-FNN) tiene un algoritmo poderoso y una estructura básica. Su capa de entrada, capa oculta y capa de salida están compuestas por las capas densas, incluido el algoritmo de propagación hacia adelante y el algoritmo de retropropagación, para ello, es necesario definir un optimizador para realizar el descenso de gradiente, una función de pérdida y unos indicadores de evaluación. Sobre todo, BP-FNN posee tres propiedades:

- a. Las neuronas de las capas superiores e inferiores están completamente conectadas.
- b. No hay conexiones entre neuronas en la misma capa.
- c. No hay conexiones de capa cruzada.

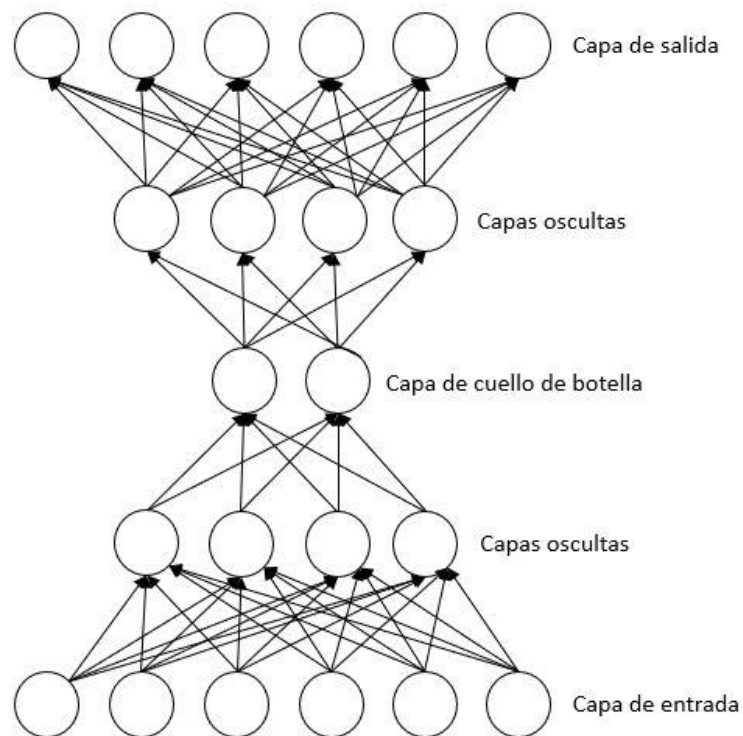


Figura. 22 Autoencoder



4.4.1 Diagrama del sistema sobre BP-FNN

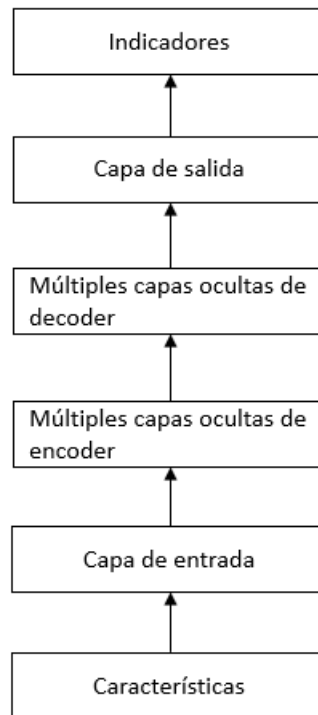


Figura. 23 Diagrama del sistema sobre BP-FNN

Capa de entrada	
Dense layer	Carácterísticas de 32 dimensiones
Múltiples capas ocultas de encoder	
Dense layers	Parámetro de dimensión de grande a pequeño, L2 regularización (0.001), Activación Relu
BatchNormalization layer	-
Dropout layer	0.5
Múltiples capas ocultas de decoder	
Dense layers	Parámetro de dimensión de pequeño a grande, L2 regularización (0.001), Activación RELU
BatchNormalization layer	-
Dropout layer	0.5

Capa de salida	
Dense layer	Datos reconstruidos de 32 dimensiones, Activación Sigmoid

Tabla. 6 Estructura de BP-FNN

4.4.2 Regularización

La regularización es un procesamiento de la reducción el error de prueba (mejorar los indicadores de evaluación de la predicción), el motivo final es que el modelo de aprendizaje automático puede funcionar bien frente a nuevos datos, debido a que las muestras negativas y positivas se mezclan en el conjunto de prueba, la computadora debe aprender a reconstruir los datos con mucha precisión para evitar la confusión.

Sin embargo, cuando utiliza un modelo más complejo como una red neuronal profunda, es posible que ocurra el riesgo del sobreaprendizaje (overfitting) en la etapa de prueba, es decir que el modelo funcionará bien para el conjunto de entrenamiento, pero tendrá un desempeño deficiente por más interferencias en la prueba, Para evitar que disminuya la generalización del modelo, la regularización será un buen método para reducir la complejidad de modelo.

‘Weight decay’, L2 regularización es una manera común para reducir el sobreaprendizaje, lo que hacer es agregar el costo asociado con un valor de peso mayor a la función de pérdida, de modo que fuerce el modelo a tomar un valor de peso pequeño, de esta manera limitará la complejidad.

$$L = E_{in} + \lambda \sum_j w_j^2 \quad \left(\sum_j w_j^2 \leq C \right)$$

E_{in}: Error de entrenamiento no regularizado
λ: Coeficiente de regularización
w: Weight, peso

Figura. 24 Función de Regularización L2



Como la formula anterior, L2 regularización es la suma de los cuadrados de cada valor de peso y agrega el error de pérdida. Cuando la parte derecha es menos que un parámetro 'C' (Se aproxima a 0), el valor de peso será pequeño. De esta forma, se minimizar el error de entrenamiento ' E_{in} ' bajo de la condición de que la suma de los cuadrados de peso sea menor que el parámetro 'C'. Cuando el parámetro sea menor, el modelo será más sencillo, por lo tanto, esto puede aliviar el riesgo causado por el sobreaprendizaje.

Además de demasiados parámetros, el modelo ocurrirá el sobreaprendizaje si tiene muy pocas muestras de entrenamiento, por ello, hay una otra técnica disponible, 'Dropout', se utiliza como un truco para entrenar la red neuronal profunda. La función es ignorar la mitad de los detectores de características en cada ronda de entrenamiento, es decir que deja que la mitad de los nodos de la capa oculta tengan un valor de 0. La manera es que algunas neuronas dejan de trabajar con una cierta probabilidad 'p' durante la propagación hacia adelante, lo que puede hacer que el modelo sea más generalizado en lugar de depender demasiado de ciertas características.

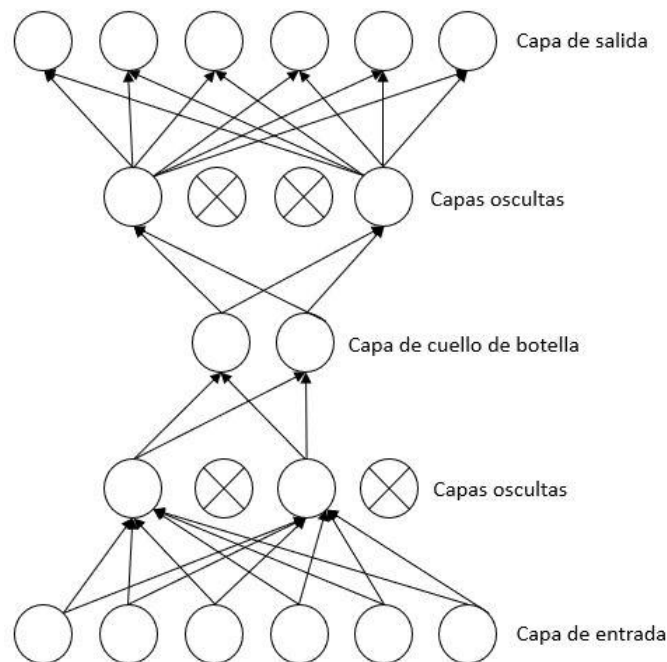


Figura. 25 Dropout

TRABAJO FIN DE MÁSTER

Conforme a la figura anterior, su primer paso es eliminar aleatoriamente y temporalmente la mitad de las neuronas ocultas en la red, pero no hay cambio para las neuronas de entrada y salida. El segundo paso es actualizar los valores de pesos w y b basado en las neuronas no eliminadas mediante el descenso de gradiente. El tercer paso es repetir este proceso en cada ronda de entrenamiento.

La tasa de 'Dropout' se establece en $(1-p)$ en Keras, por lo general, es ideal a coger 0.5 para la capa oculta de la red media o grande, pero no es recomendable para la capa de entrada y de salida. Tiene 3 ventajas para ayudar a aliviar el sobreaprendizaje.

- a. Para lograr un resultado promedio. Debido a que diferentes redes pueden causar el sobreaprendizaje distinto, el motivo del promedio es para cancelar algunos resultados opuestos entre sí.
- b. Mejora de la red neuronal mediante la prevención de la coadaptación de detectores de características. Dado que no es fácil aparecer dos mismas neuronas en la red de 'Dropout', de esta forma, la actualización de pesos no depende de la interacción entre algunos nodos que tienen la relación fija, así evita que ciertas características sean efectivas solo bajo las condiciones específicas.
- c. Mejora del modelo más preciso. Se puede eliminar los resultados opuestos con grande desviación.

4.4.3 Optimizador

La función de pérdida y la función de optimización son los dos conceptos importantes en el aprendizaje automático, el primero es el índice de evaluación y el segundo es la estrategia de optimización de la red. Hay muchos tipos de funciones de optimización, en este caso de la detección de anomalías, la función de Adam será más adecuada, que es un algoritmo de descenso de gradiente con la tasa de aprendizaje adaptativa. La tasa de aprendizaje es muy importante para la red neuronal, cuando sea pequeña, la velocidad de convergencia será muy lenta, al contrario, si la tasa de aprendizaje es enorme, ignorará algún

mínimo local, ambas situaciones no son ideales. Por lo tanto, el método de adaptación de la tasa de aprendizaje se refiere a la modificación adecuada sobre esa durante el entrenamiento para mejorar la convergencia.

$$m_t = \text{beta}_1 * m_{t-1} + (1 - \text{beta}_1) * g$$

$$v_t = \text{beta}_2 * v_{t-1} + (1 - \text{beta}_2) * g^2$$

$$\text{variable} = \text{variable} - lr_t * \frac{m_t}{\sqrt{v_t} + \epsilon} \quad (\eta = lr_t * m_t)$$

m_t: El valor de la suavización exponencial simple del gradiente histórico

v_t: El valor de la suavización exponencial simple del gradiente histórico cuadrado

variable: La actualización de variables

ε: La tasa de aprendizaje inicial

η: El parámetro de la tasa de aprendizaje

Figura. 26 Función de optimizador Adam

Conforme a las fórmulas anteriores, ' m_t ' es el promedio ponderado del gradiente histórico, cuando mayor sea la distancia del momento actual, menor será el peso. El motivo que se usa es para eliminar la oscilación cuando el modelo está actualizando la variable y lograr un valor de actualización de gradiente estable.

' v_t ' es el promedio ponderado del gradiente histórico cuadrado, que sirve para ajustar la tasa de aprendizaje. En cada ronda de entrenamiento, si el valor de la actualización de gradiente ' v_t ' es pequeño y estable, la tasa de aprendizaje adaptativo que está en la parte derecha de la fórmula 3 cambiará en más grande. De lo contrario, se volverá más pequeña.

Sobre todo, el optimizador de Adam es adecuado a actualizar la variable según la fluctuación en el gradiente histórico y la situación verdadera después de la oscilación.

4.4.4 Función de pérdida

La función objetivo es la clave para el aprendizaje de optimización en el modelo, que también se denomina la función de pérdida o la función de costo cuando el motivo es minimizarla en proceso de aprendizaje. Hay una variedad de funciones de pérdida, es esencial elegir una adecuada según el requisito de la tarea diferente de aprendizaje automático.

La tarea de la detección de anomalías es evaluar la diferencia entre datos reconstruidos y datos originales, por lo tanto, el error cuadrático medio (MSE) o el error absoluto medio (MAE) podrán ser las dos funciones de pérdida más fiables en este caso. Si hay un punto anormal que representa un importante riesgo potencial de fallo durante el funcionamiento de la máquina y es necesario detectarlo, MSE es más ideal. Por otro lado, si dicho punto anormal es solo causado por datos dañados, MAE es mejor. Dado que los datos dañados del conjunto de entrenamiento se han eliminado en el preprocesado, no es necesario ampliar el error de reconstrucción o la diferencia en la muestra normal, por lo tanto, se utiliza MSE como la función de pérdida.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - y_i^p)^2$$

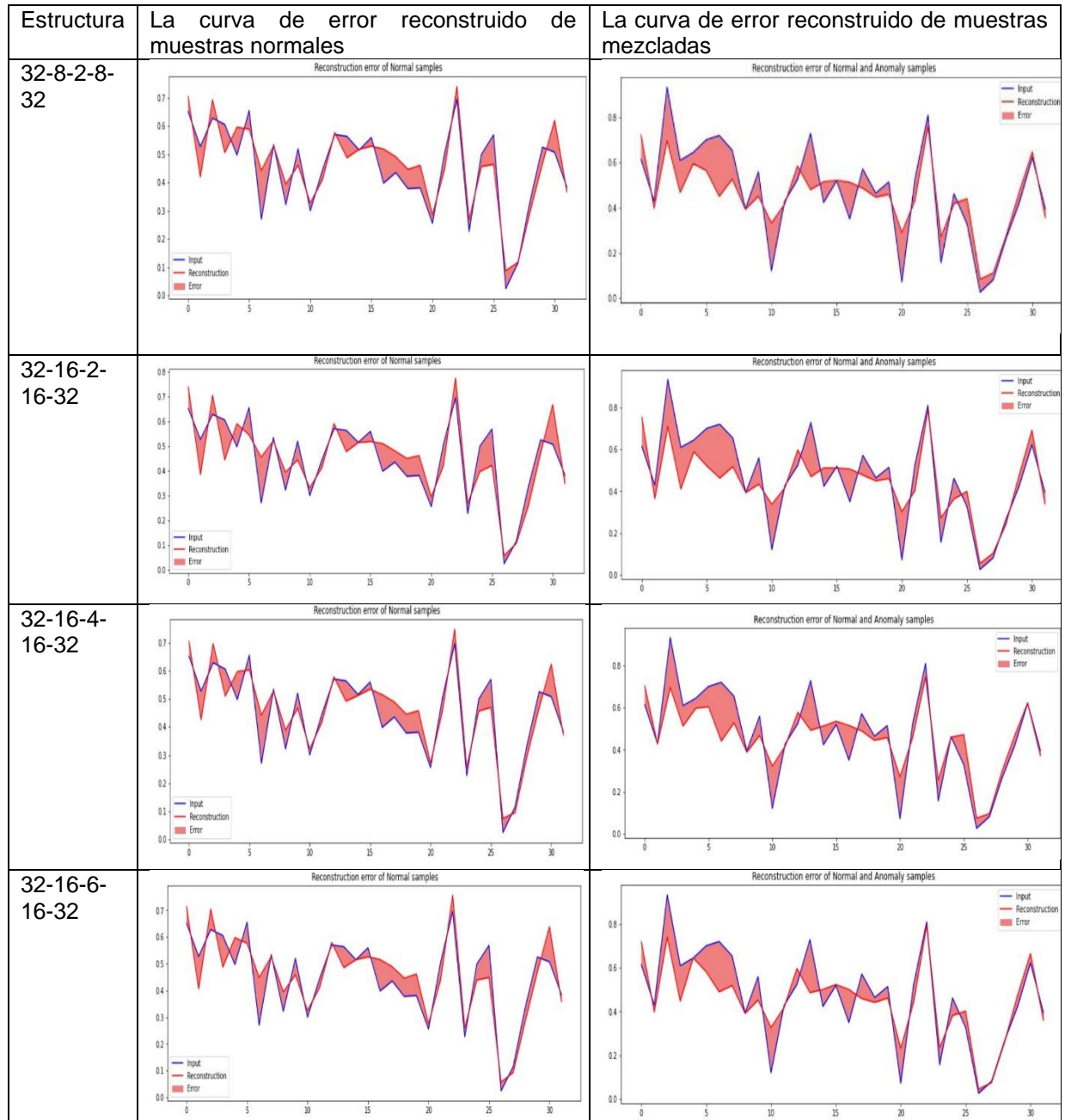
n: Número de muestras
y_i: Valor observado
y_i^p: Valor predicho

Figura. 27 Función de MSE

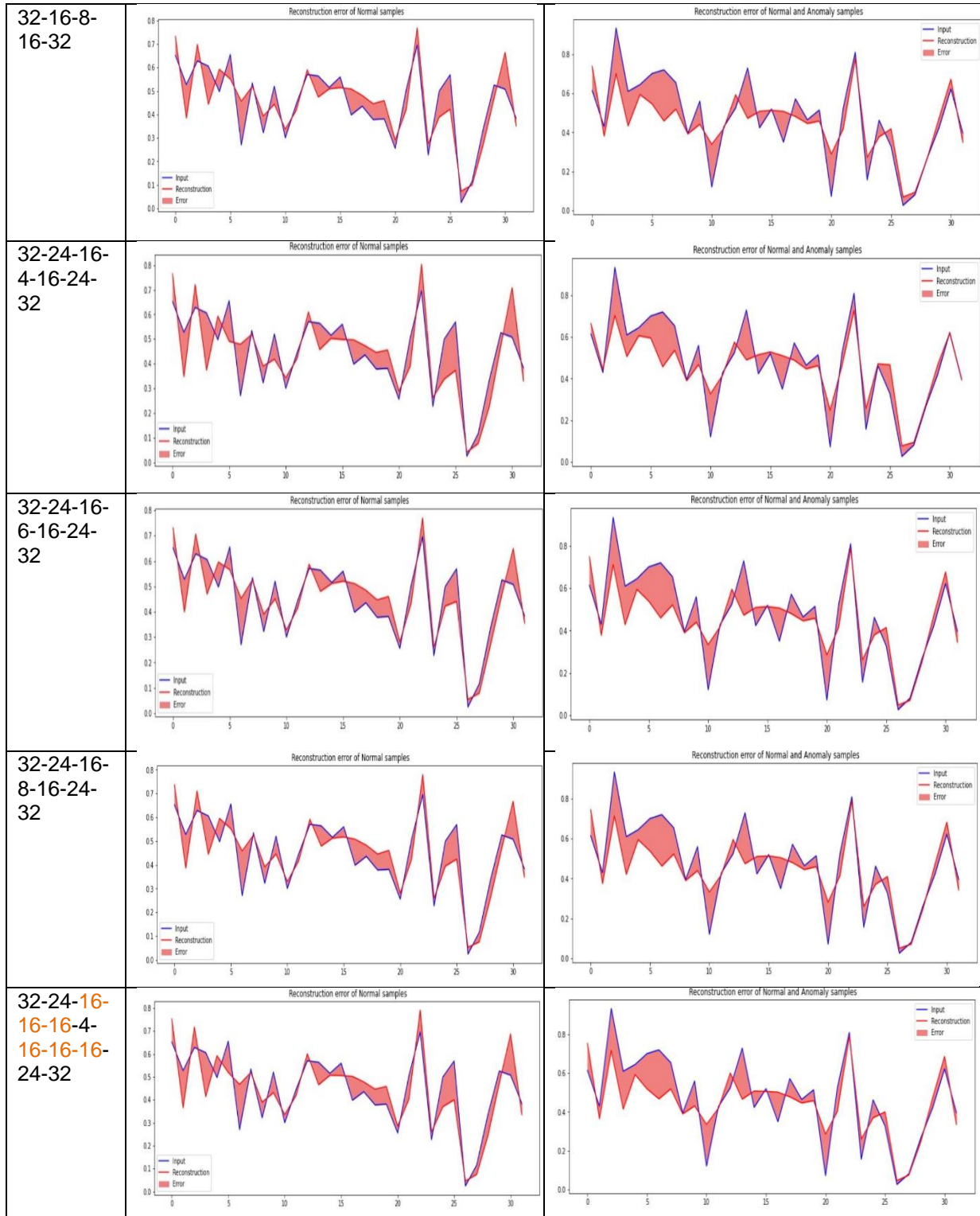
Conforme a la fórmula, se puede evaluar el cambio de datos, cuando el valor de MSE menor, será más eficaz el modelo para reconstruir los datos. MSE cuya ventaja es que la actualización del gradiente aumentará a medida que aumenta la pérdida, en contrario, disminuirá en cuanto tiende a cero la pérdida.

4.4.5 Resultados y Análisis

Las siguientes figuras muestran el error de reconstrucción, que provienen del autoencoder de BP-FNN con mejores resultados.



TRABAJO FIN DE MÁSTER



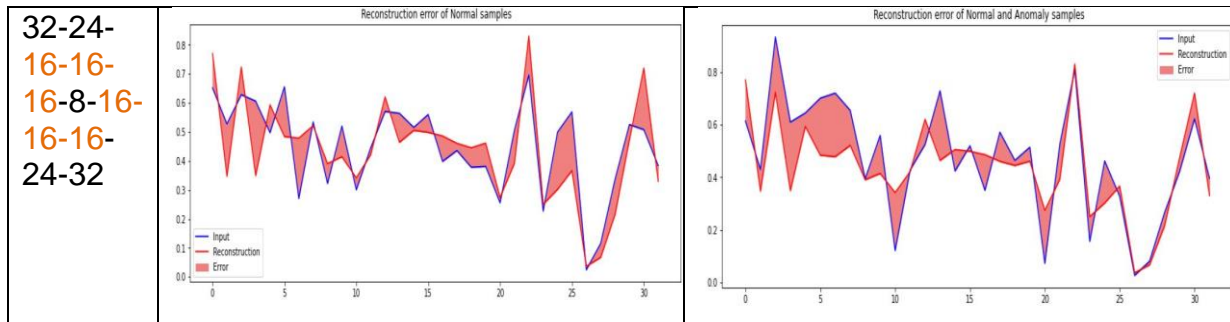
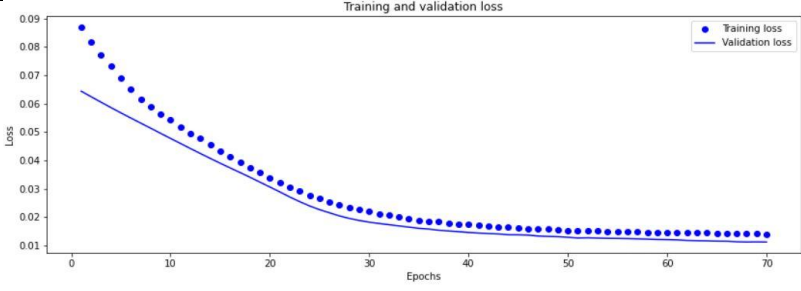
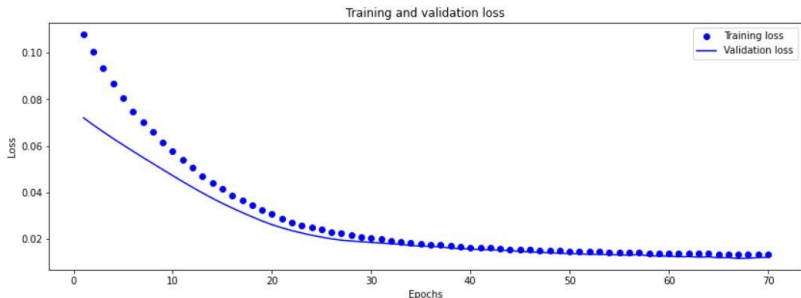
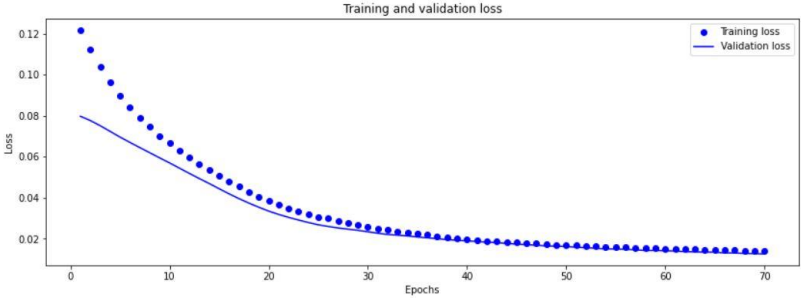
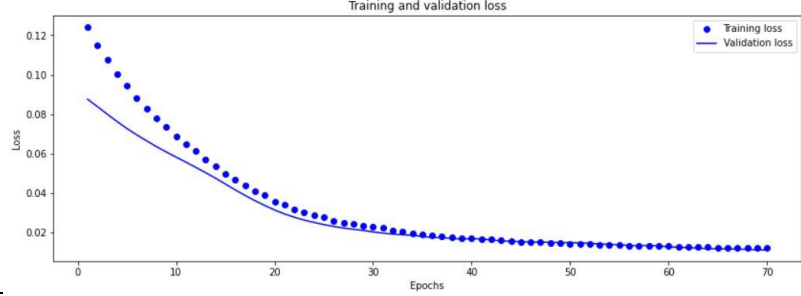
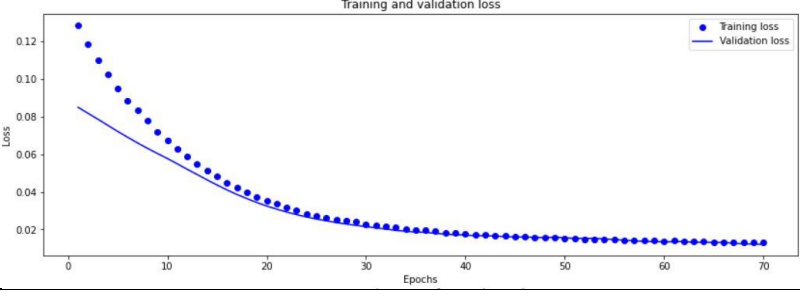


Figura. 28 La curva de error reconstruido

La segunda columna muestra la diferencia entre la señal reconstruida y la señal original de todas las muestras normales a través del aprendizaje automático. La tercera columna es la curva de error de reconstrucción de una mezcla de muestras normales y anormales en el conjunto de prueba. Obviamente, la magnitud de error de figuras en la tercera columna es mayor que la segunda, el área bajo la curva también es mayor, indica que el modelo tiene capacidad de detectar la mayoría de las muestras anormales.

Al mismo tiempo, en la quinta escala de la tercera figura, se observa que el área bajo la curva aumenta a medida que la profundidad de la red neuronal, significa que la eficiencia del sistema pone en correlación con la estructura de la red.

TRABAJO FIN DE MÁSTER

Estructura	Rondas de cruce	la curva de error reconstruido entre el conjunto de entrenamiento y el de prueba
32-8-2-8-32	-	
32-16-2-16-32	30	
32-16-4-16-32	30	
32-16-6-16-32	30	
32-16-8-16-32	30	

TRABAJO FIN DE MÁSTER

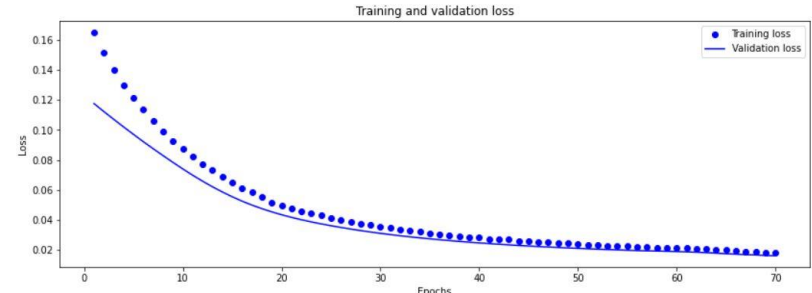
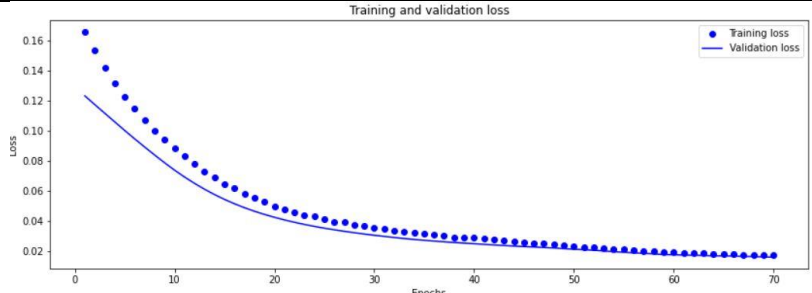
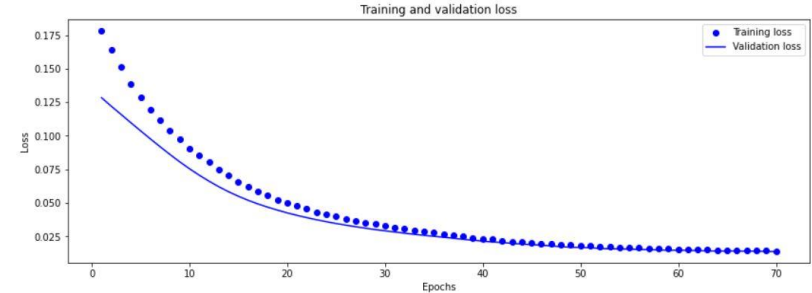
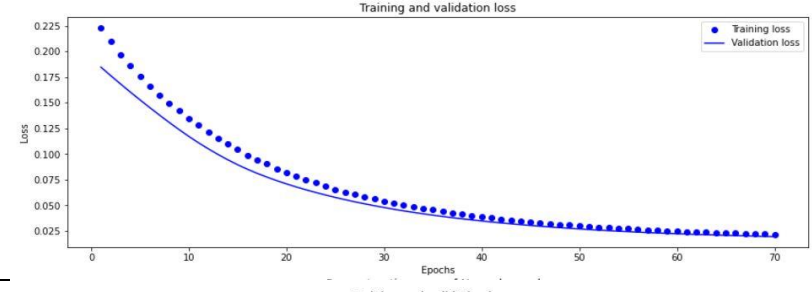
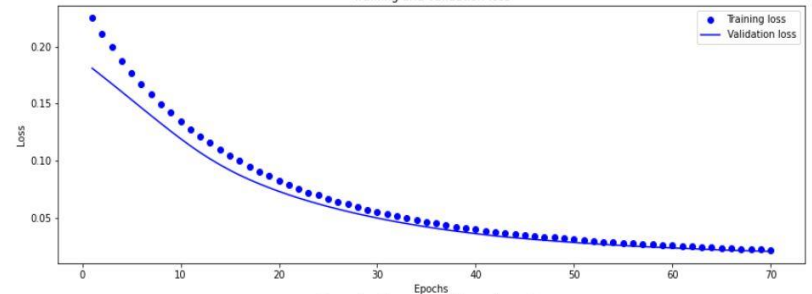
Estructura	Rondas de cruce	la curva de error reconstruido entre el conjunto de entrenamiento y el de prueba
32-24-16-4-16-24-32	35	
32-24-16-6-16-24-32	35	
32-24-16-8-16-24-32	35	
32-24-16-16-16-4-16-16-16-24-32	40	
32-24-16-16-16-8-16-16-16-24-32	40	

Figura. 29 La curva de error reconstruido y Rondas de cruce

TRABAJO FIN DE MÁSTER

Según la figura arriba, se observa que el ajuste del valor de pérdida de BP-FNN es estable y suave en 70 ronda, especialmente, alcanza la solución óptima desde 30 rondas hasta 40. Cuando la profundidad de la red neuronal es demasiado pequeña y la cantidad de capas ocultas es insuficiente, será difícil a convergerse las dos curvas de pérdida y la eficiencia del modelo no será óptima. Por el contrario, a medida que aumenta la profundidad, la eficiencia mejorará y el número de rondas de entrenamiento también aumentará.

Los resultados del experimento de BP-FNN son los siguientes.

Tipo de DNN	Estructura	Rondas de cruce	Threshold	Accurac y	Precisio n	Recall	F1 score
Dense layers	32-8-2-8-32	-	0.019189	46.85%	46.14%	92.02%	61.46%
	32-16-2-16-32	30	0.018929	46.85%	46.24%	94.58%	62.11%
	32-16-4-16-32	30	0.017951	47.90%	46.71%	93.16%	62.22%
	32-16-6-16-32	30	0.017100	47.11%	46.29%	92.59%	61.72%
	32-16-8-16-32	30	0.018669	46.45%	46.04%	94.58%	61.94%
	32-24-16-4-16-24-32	35	0.028901	48.03%	46.32%	80.91%	58.92%
	32-24-16-6-16-24-32	35	0.020760	47.24%	46.40%	93.73%	62.07%
	32-24-16-8-16-24-32	35	0.021179	46.85%	46.20%	93.73%	61.90%
	32-24-16-16-16-4-16-16-16-24-32	40	0.025724	48.16%	46.88%	94.30%	62.63%
	32-24-16-16-16-8-16-16-16-24-32	40	0.025692	48.03%	46.81%	94.30%	62.57%
32-24-16-16-16-16-16-4-16-16-16-16-24-32	40	0.025701	48.16%	46.88%	94.30%	62.63%	

El rojo marcado es uno de los valores más altos en la columna actual

Tabla. 7 Resultados finales de BP-FNN

True positive	329
True negative	39
False positive	372
False negative	22

Tabla. 8 Resultados de Confusión Matriz

Hay un total de 6140 muestras normales en conjunto de entrenamiento, y el conjunto de prueba tiene 441 muestras normales y 351 muestras anormales.

Conforme a la tabla de resultado, vemos las comparaciones interesantes allí. El primer lugar que tenemos en cuenta son la estructura, el threshold y los indicadores como 'accuracy', 'presicion',

TRABAJO FIN DE MÁSTER

´recall´ y ´f1 score´. En este caso, el experimento inició a partir de la estructura sencilla hasta la más compleja. Observamos que el threshold y los valores de indicadores se convierten en más mayores e ideales, ya que hemos agregado más capas ocultas para facilitar el aprendizaje automático, por eso, las unidades neuronales tienen capacidad de seleccionar y mejorar la información oculta en características acústicas, entonces, eso es un aprendizaje efectivo para la computadora. Sin embargo, el modelo no mejora la eficiencia, cuando más de una cierta cantidad de capas ocultas trabaja allí, por ejemplo, los resultados son parecidos a partir de 10 capas de 16 dimensiones y los de 6 capas de 16 dimensiones.

Además, a pesar de que los resultados mejores se centran en las estructuras más complejas con más capas ocultas, pero la capa de cuello de botella también es el factor clave para que la computadora pueda eliminar el resto de información no válida. Según la tabla, la mejor capa de cuello de botella será 4 en BP FNN.

El valor de ´Recall´ alcanzó aproximadamente el 94%, que es muy alto. Cuanto más efectivas sean las características que extraen, más probabilidad hay de que el detector encuentre las anomalías, significa que el detector puede ser más eficaz para muestras anómalas de conjunto de prueba, aunque el resultado de la predicción puede ser falso. En contrario, los indicadores restantes como ´Accuracy´ y ´Presicion´ no son ideales, solamente son el 48% y el 46% menos de la mitad. Lo que paso es porque se ve afectado por el riesgo de ´domain shift´, debido a que el muestreo de sonido está bajo la condición distinta, por ejemplo, las cajas de cambios trabajan a las velocidades diferentes en el entorno más seco o más húmedo, su velocidad de rotación pasará un cambio infinito por el ambiente, luego el sonido también cambiará de manera impredecible, por lo tanto, hay unas desviaciones sutiles en los datos extraídos después de la extracción de características. Posteriormente, dichas desviaciones van a cambiar y expandir a través de varias capas ocultas en el proceso de aprendizaje automático. Al

TRABAJO FIN DE MÁSTER

final, es posible que afecte la eficiencia de agrupación en este caso, la computadora puede clasificar incorrectamente los sonidos normales como anomalías.

La solución a este problema requiere el uso de ideas de aprendizaje por transferencia (transfer learning). El aprendizaje por transferencia puede transferir el modelo de gran dato a dato personalizado para descubrir el común entre sí, es decir que se aplica la similitud desde 'source domain' en el aprendizaje del 'target domain', así es la transferencia personalizada.

4.5 Sistema basado en CNN-Conv1D

4.5.1 La red neuronal convolucional

La red neuronal convolucional (CNN) es una red con estructura no completamente conectada. CNN generalmente está compuesta por una capa de entrada, unas capas convolucionales, unas capas de agrupación, una capa completamente conectada y una capa de salida. Hay una diferencia de FNN que es la conexión no completada entre la capa convolucional y la capa de agrupación, lo que puede reducir efectivamente la complejidad de la red neuronal.

CNN puede reconocer las redes neuronales con las formas multi-dimensionales porque se usa el algoritmo de convolución para mejorar el sistema de aprendizaje automático incluido el concepto de la interacción dispersa (sparse interactions) y del compartimiento de parámetros (parameter sharing). La interacción dispersa es una percepción parcial inspirado por la estimulación parcial de las neuronas de la corteza visual en el sistema visual biológico. De esta forma, se estimula parcialmente la conexión entre la capa convolucional y la capa de agrupación, lo que puede reducir los parámetros, mejorar la velocidad de cálculo y aliviar el riesgo de sobreaprendizaje. Por otro lado, el compartimiento de parámetros se refiere al uso de los mismos pesos en varias funciones, lo que puede mejorar la eficacia de aprendizaje y mejorar la generalización del modelo.

Dado que la extracción de características se trata de los datos numéricos en series temporales, por eso se utiliza la convolución unidimensional (Conv1D) para analizar los datos de señales con un período fijo como las señales de audio.

4.5.2 La capa convolucional

La capa convolucional en Conv1D tiene solo una dimensión vectorial, significa que la dirección de movimiento del núcleo de convolución en la matriz de entrada posee solo una dimensión, que se mueve de arriba hacia abajo. Si hay un plano de matriz sobre características con M filas y N columnas, M es la dimensión de dicha



característica y N representa el cuadro de audio, el núcleo de convolución debe moverse de pequeño a grande hacia cuadro.

El núcleo de convolución también es una matriz con J filas y K columnas y se puede ver que él es bidimensional con largo y ancho. Debido a que el núcleo de convolución solo se mueve en una dirección, es necesario a cubrir todos los valores de característica, significa que la fila del núcleo de convolución debe ser igual a la de matriz (M=J), por lo tanto, el tamaño de 'kernel' y número de 'kernel' es el parámetro que debe establecer en la capa convolucional. 'kernel' es una matriz de ponderaciones o un núcleo de convolución, que sirve para aprender las características relevantes por multiplicarse con datos de entrada.

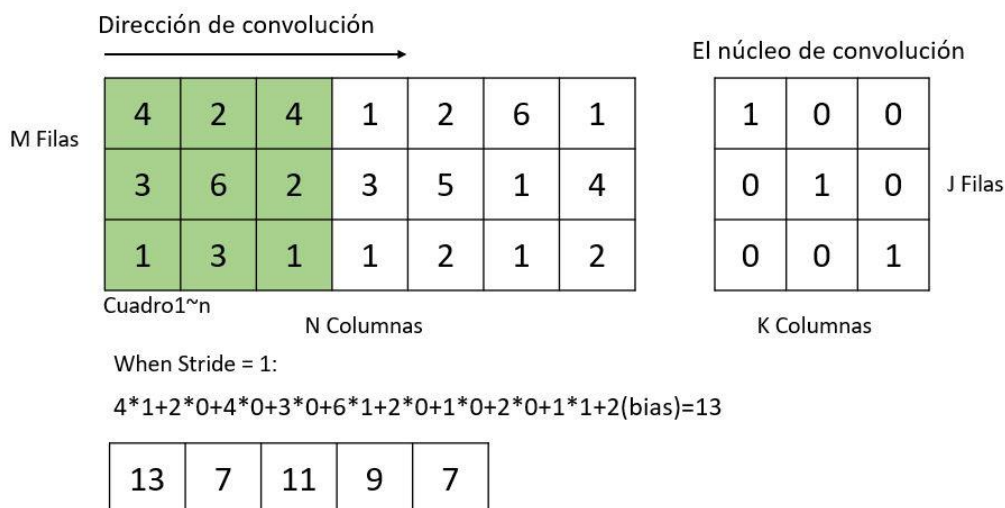


Figura. 30 Matriz de entrada y Kernel convolucional

Además de 'kernel', hay otros claves parámetros como 'filter', 'strides' y 'padding'. 'Filter' está compuesto por varios 'kernel' en series, que es la dimensión del espacio de salida o la cantidad de núcleo de convolución. 'strides' es el paso de movimiento para la convolución y en general es 1. Hay dos opciones para 'padding' que es 'same' o 'valid'. 'valid padding' significa que no llena los blancos por cero para descartar los elementos redundantes, la otra es al revés, 'same padding' hace que la entrada y la salida sea del mismo tamaño.

$$output = \left(\frac{input + 2p - k}{s} \right) + 1$$

output: El tamaño de salida
input: El tamaño de entrada
k: El tamaño de kernel (núcleo de convolución)
s: 'stride'
p: 'padding'

Figura. 31 Función de MSE

4.5.3 La capa de agrupación

Generalmente, la capa de agrupación y la capa convolucional aparecen alternativamente en pares. La capa de agrupación es una red neuronal conectada detrás de la capa convolucional, se puede dividir en la capa de muestreo descendente o muestreo ascendente.

La capa convolucional profundiza la matriz, luego la capa de muestreo descendente mantiene dicha profundidad sin cambios, pero va a reducir el tamaño de la matriz. La función es extraer el valor medio parcial o el valor máximo, que son 'AveragePooling' y 'MaxPooling'. La realización de la capa de agrupación unidimensional es parecida a la capa convolucional, cuyo filtro se mueve en una sola dirección mediante la propagación hacia adelante.

'MaxPooling' toma el valor más grande y 'AveragePooling' toma el medio en la matriz.

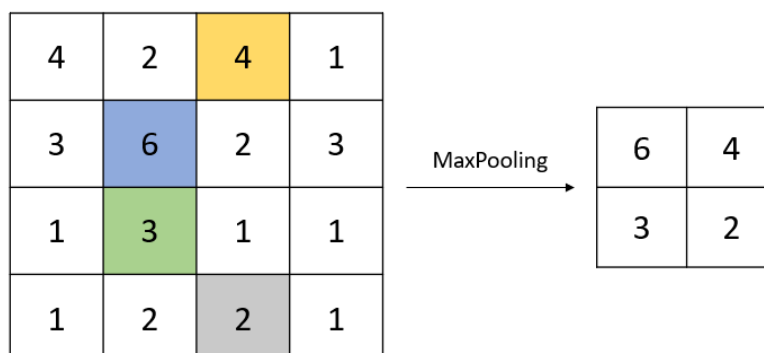


Figura. 32 MaxPooling (Stride = 2)

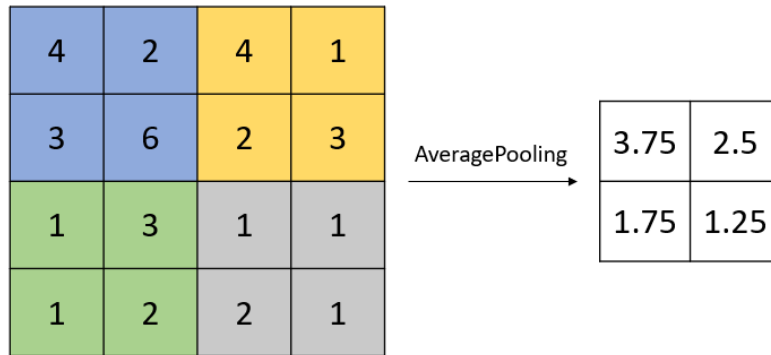


Figura. 33 AveragePooling (Stride = 2)

La capa de muestreo ascendente es una técnica de deconvolución utilizada en el decodificador en autoencoder, se refiere a la operación inversa a la convolución empleada para restaurar las señales y recuperar los datos. Keras proporciona tres métodos como 'UnPooling', 'UpSampling' y 'Transpose Convolution'.

'UnPooling' se rellena con cero en las posiciones cercanas en la deconvolución y 'UpSampling' se rellena con los mismos valores allí.

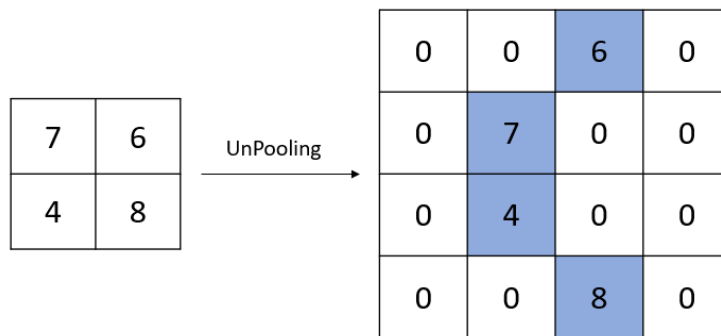


Figura. 34 UnPooling

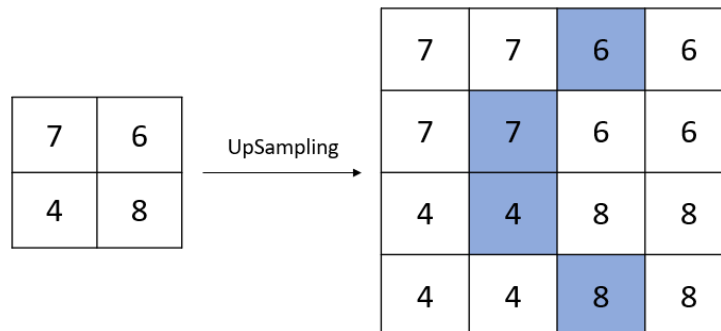


Figura. 35 UpSampling

‘Transpose Convolution’ es opuesta a la convolución en la propagación hacia adelante y la retropropagación de la red neuronal. O sea, también es una convolución especial hacia adelante, lo que hace es ampliar el tamaño de imagen o la longitud de datos numéricos a través del relleno de ceros, y luego gira el núcleo de convolución para hacer la propagación hacia adelante. La diferencia es que ‘UnPooling’ y ‘UpSampling’ tiene sus funciones fijadas y no se puede aprender, pero se permiten los parámetros de ‘Transpose Convolution’.

$$\vec{x} * \vec{a} = X\vec{a}$$

$$\begin{bmatrix} x & y & z & 0 & 0 & 0 \\ 0 & x & y & z & 0 & 0 \\ 0 & 0 & x & y & z & 0 \\ 0 & 0 & 0 & x & y & z \end{bmatrix} \begin{bmatrix} 0 \\ a \\ b \\ c \\ d \\ 0 \end{bmatrix} = \begin{bmatrix} ay + bz \\ ax + by + cz \\ bx + cy + dz \\ cx + dy \end{bmatrix}$$

\vec{x} : La entrada
 \vec{a} : El núcleo de convolución
 kernel = 3, **stride** = 1, padding = 1

Figura. 36 Algoritmo de la propagación hacia adelante de Conv1D (Stride = 1)

$$\vec{x}^T * \vec{a} = X^T \vec{a}$$

$$\begin{bmatrix} x & 0 & 0 & 0 \\ y & x & 0 & 0 \\ z & y & x & 0 \\ 0 & z & y & x \\ 0 & 0 & z & y \\ 0 & 0 & 0 & z \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} ax \\ ay + bx \\ az + by + cx \\ bz + cy + dx \\ cz + dy \\ dz \end{bmatrix}$$

\vec{x} : La entrada
 \vec{a} : El núcleo de convolución
 kernel = 3, **stride** = 1, padding = 1

Figura. 37 Algoritmo de ‘Transpose Convolution’ (Stride = 1)

$$\vec{x} * \vec{a} = X\vec{a}$$

$$\begin{bmatrix} x & y & z & 0 & 0 & 0 \\ 0 & 0 & x & y & z & 0 \end{bmatrix} \begin{bmatrix} 0 \\ a \\ b \\ c \\ d \\ 0 \end{bmatrix} = \begin{bmatrix} ay + bz \\ bx + cy + dz \end{bmatrix}$$

\vec{x} : La entrada
 \vec{a} : El núcleo de convolución
 $kernel = 3, stride = 2, padding = 1$

Figura. 38 Algoritmo de la propagación hacia adelante de Conv1D (Stride ≥ 2)

$$\vec{x}^T * \vec{a} = X^T \vec{a}$$

$$\begin{bmatrix} x & 0 \\ y & 0 \\ z & x \\ 0 & y \\ 0 & z \\ 0 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} ax \\ ay \\ az + bx \\ by \\ bz \\ 0 \end{bmatrix}$$

\vec{x} : La entrada
 \vec{a} : El núcleo de convolución
 $kernel = 3, stride = 2, padding = 1$

Figura. 39 Algoritmo de 'Transpose Convolution' (Stride ≥ 2)

Sobre todo, es importante que elige un método de deconvolución más cómodo según las tareas distintas. La capa de agrupación tiene dos ventajas en CNN.

- Se reduce los parámetros en cada ronda de entrenamiento para acelerar la velocidad de cálculo.
- Se reduce el riesgo de sobreaprendizaje.

4.5.4 Diagrama del CNN-Conv1D

Capa de entrada	
Conv1D layer	Características de 32 dimensiones
Múltiples capas ocultas de encoder	
Conv1D layers	Parámetro de filtro de grande a pequeño, Stride (4), L2 regularización (0.001), Activación Relu, Padding (same)
MaxPooling1D layers	Tamaño de ventana (2), Padding (same)
Múltiples capas ocultas de decoder	
Conv1D layers	Parámetro de filtro de pequeño a grande, Stride (4), L2 regularización (0.001), Activación Relu, Padding (same)
UpSampling1D layers	Tamaño de ventana (2)
Flatten layer or GlobalMaxPooling1D layer	-
Capa de salida	
Dense layer	Datos reconstruidos de 32 dimensiones, Activación Sigmoid

Tabla. 9 Estructura de CNN-Conv1D

4.5.5 Resultados y Análisis

Los resultados del experimento de CNN y BP-FNN son los siguientes.

Tipo de DNN	Estructura	Rondas de cruce	Threshold	Accuracy	Precisión	Recall	F1 score
Dense layers	32-8-2-8-32	-	0.019189	46.85%	46.14%	92.02%	61.46%
	32-16-2-16-32	30	0.018929	46.85%	46.24%	94.58%	62.11%
	32-16-4-16-32	30	0.017951	47.90%	46.71%	93.16%	62.22%
	32-16-6-16-32	30	0.017100	47.11%	46.29%	92.59%	61.72%
	32-16-8-16-32	30	0.018669	46.45%	46.04%	94.58%	61.94%
	32-24-16-4-16-24-32	35	0.028901	48.03%	46.32%	80.91%	58.92%
	32-24-16-6-16-24-32	35	0.020760	47.24%	46.40%	93.73%	62.07%
	32-24-16-8-16-24-32	35	0.021179	46.85%	46.20%	93.73%	61.90%
	32-24-16-16-16-4-16-16-16-24-32	40	0.025724	48.16%	46.88%	94.30%	62.63%
	32-24-16-16-16-8-16-16-16-24-32	40	0.025692	48.03%	46.81%	94.30%	62.57%
32-24-16-16-16-16-16-4-16-16-16-16-16-24-32	40	0.025701	48.16%	46.88%	94.30%	62.63%	
Conv1D	32-24-16-8-2-2-8-16-24-32 (globalmaxpooling)	15	0.025704	48.16%	46.88%	94.30%	62.63%
	32-24-16-8-4-4-8-16-24-32 (flatten)	15	0.025729	48.16%	46.87%	94.01%	62.55%
	32-24-16-8-4-4-8-16-24-32 (globalmaxpooling)	15	0.025755	48.16%	46.88%	94.30%	62.63%

El rojo marcado es uno de los valores más altos en la columna actual

Tabla. 10 Resultados finales de BP-FNN y CNN-Conv1D

BP-FNN (32-24-16-16-16-4-16-16-16-24-32)	
True positive	329
True negative	39
False positive	372
False negative	22
CNN-Conv1D (32-24-16-8-4-4-8-16-24-32 (globalmaxpooling)	
True positive	330
True negative	34
False positive	377
False negative	21

Tabla. 11 Resultados de Confusión Matriz

TRABAJO FIN DE MÁSTER

Los indicadores mejores aparecen en las dos redes neuronales, BP FNN y CNN, significa que los resultados de ambas partes se verifican mutuamente.

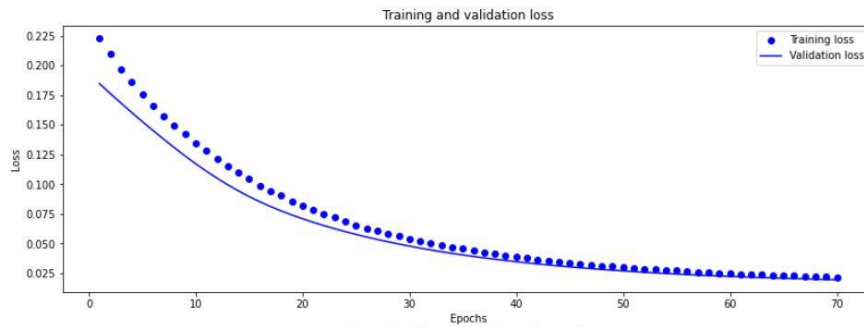


Figura. 40 La curva de error reconstruido y Rondas de cruce
BP-FNN (32-24-16-16-16-4-16-16-16-24-32)

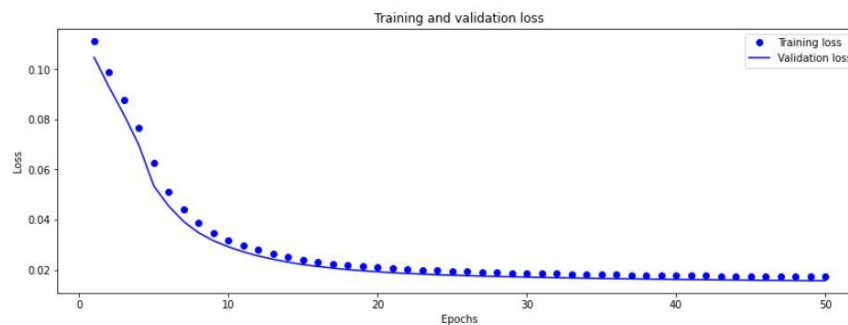


Figura. 41 La curva de error reconstruido y Rondas de cruce
CNN-Conv1D (32-24-16-8-4-4-8-16-24-32 (globalmaxpooling))

Conforme a las dos figuras arribas, observamos que la convergencia de las curvas de entrenamiento y prueba en CNN es más alta que dense layers de FNN. La red de dense layers requiere 35~40 rondas para alcanzar el óptimo, mientras que CNN solo necesita 15 rondas. La velocidad de adaptación de curvas depende de la complejidad de los datos y la eficiencia del modelo. Por lo tanto, Si hay muy pocas rondas de entrenamiento, las dos curvas no pueden intersecarse y es posible que ocurra el problema de mínimo local. Sin embargo, demasiadas rondas causarán más cargas de cálculo innecesarias, eso no será mejor. Según las necesidades reales, los

distintos tipos de modelos pueden elegir el número correcto de rondas de entrenamiento.

5 CONCLUSIÓN Y PERSPECTIVA

5.1 Conclusión

El trabajo principal es estudiar el sistema de la detección de anomalías a través de ruidos para monitorizar el estado de la caja de cambios, que se centra en el establecimiento de la biblioteca de sonidos, la extracción de características y el proceso de entrenamiento del modelo de sonido. En este caso, se estudia el modelo de red neuronal de aprendizaje automático, se analiza la estructura de la red neuronal y se construye un sistema de detección de sonido anormal basado en autoencoder.

Posteriormente, se prueba y verifica las muestras de sonido a través de BP-FNN y CNN. De acuerdo con los resultados experimentales, va a estudiar los factores de influencia en el sistema de la detección de anomalías. Los pasos de trabajo que he completado son los siguientes:

- a. Este artículo presenta brevemente la extracción de características y la etapa de preprocesado para señales. Según la señal muestreada por la caja de cambios, se extrae y analiza las características útiles como MFCCs, Centroide espectral y Curtosis, etc. La extracción de características se realiza a través de los paquetes de Python como Librosa y Pyaudioanalysis, etc.
- b. Construye el sistema de detección de anomalías para los sonidos de la caja de cambios a través de autoencoder con BP-FNN y CNN. Aplica diferentes capas ocultas y capa de cuello de botella para verificar y mejorar los indicadores del modelo, que son 'accuracy', 'precision', 'recall' y 'f1 score'.
- c. Explora los factores que afectan el rendimiento del sistema, las capas ocultas con diferentes números, la capa de cuello de botella



con diferentes tamaños, diferentes rondas de entrenamiento y distintos métodos de normalización llevarán directamente los varios resultados. Por ejemplo, el experimento muestra que pide rondas de entrenamiento en CNN menos que BP-FNN para obtener la máxima rendimiento.

- d. Resuma las ventajas y las deficiencias de este experimento a partir de los resultados actuales. El sistema puede encontrar muy bien los sonidos anormales, pero hay una gran cantidad de falso positivo en la predicción.

5.2 Perspectiva

El trabajo fin de máster investiga las características acústicas de la caja de cambios industriales y la tarea de detección de anomalías. Los sonidos muestreados se clasifican en dos categorías a través del algoritmo de la red neuronal y dos estructuras distintas de red (FNN y CNN).

Al mismo tiempo, este experimento también reveló las deficiencias de detección automática. Dado que se tomaron muestras de las tres partes de una caja de cambios al mismo tiempo, la velocidad de trabajo a corto plazo de cada parte y el entorno no fueron iguales, por lo tanto, la definición para muestras anómalas también cambió por los factores de error en la propagación hacia adelante del aprendizaje automático, lo que llevó a errores de detección en los resultados.

- a. Debido a limitaciones de datos disponibles, los datos experimentales no contenían un sonido de la caja de cambios muy completo. Se recomienda que el trabajo posterior recopile más información sobre características acústicas y establezca una biblioteca de sonidos más completa para la maquinaria.
- b. La generalización del modelo es la base más importante para el aprendizaje por transferencia (Transfer learning). Suponemos que se puede entrenar de antemano un modelo adecuado de detección de anomalías para sonidos de una sección de la caja de cambios. Luego,

TRABAJO FIN DE MÁSTER

transfiera los parámetros del modelo entrenado al segundo modelo para ayudar al entrenamiento del nuevo, porque los datos de la misma máquina o del mismo tipo de máquina están coherente. Finalmente, el modelo para cada sección de máquina se optimiza mediante el ajuste de parámetros personalizado, y luego integrará todos los modelos entrenados.

- c. La recopilación de sonido de antemano debe convertirse en la forma en tiempo real. El trabajo posterior es estudiar los métodos de recopilación de sonidos en tiempo real y el aprendizaje online. De acuerdo con la naturaleza en tiempo real de los eventos de sonido anormal, se usa los métodos en línea para el monitoreo en tiempo real.

6 ANEXOS

6.1 Biblioteca de sonidos

Los sonidos provienen del conjunto de datos de desarrollo de la tarea 2 del DCASE Challenge 2021. Los datos consisten en sonidos de funcionamiento normales o anormales de la caja de cambios. La grabación es un audio de un solo canal de 10 segundos, incluidos los sonidos de funcionamiento de la máquina y el ruido ambiental.

Hay un total de 6140 muestras normales en conjunto de entrenamiento, y el conjunto de prueba tiene 441 muestras normales y 351 muestras anormales. Se muestrea 6 secciones diferentes de la misma máquina y la cantidad de muestras en cada sección es aproximadamente el mismo.



Figura. 42 Los sonidos normales del conjunto de entrenamiento

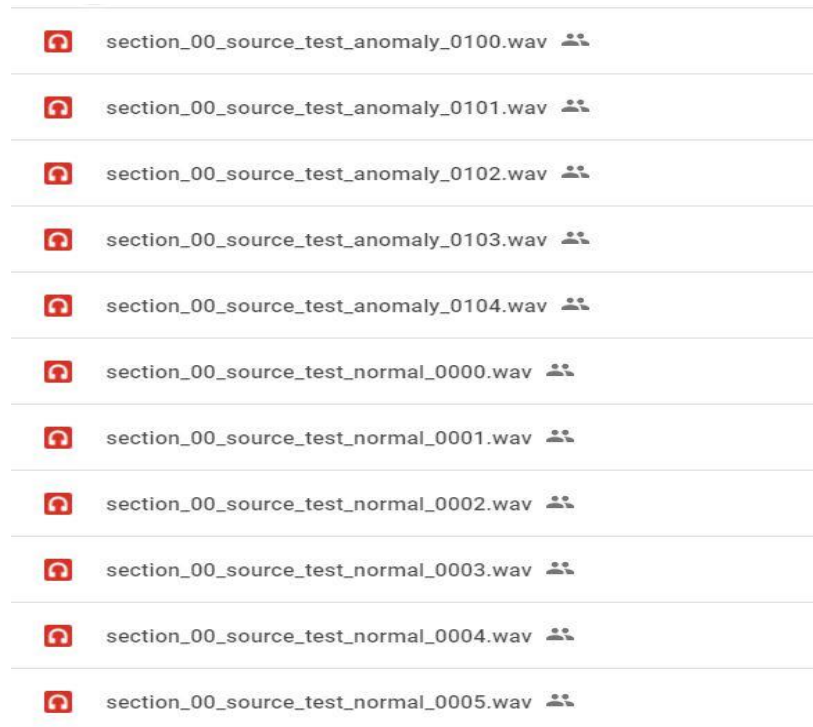


Figura. 43 Los sonidos normales/anormales del conjunto de prueba

6.2 Sistema de Software

COLABORATORY	
Descripción	Ubuntu 18.04.5 LTS
Codename	Bionic
5.4.144+	
Sistema de operación	Microsoft Windows 10
Python Versión	3.7.12
Archivo	yichengTFM_Train.ipynb

Tabla. 12 Colaboratory

6.3 Visualización de código

```

from IPython.display import IFrame
from IPython.display import Image
from IPython.display import Audio
from IPython.display import YouTubeVideo
from IPython.core.display import display, HTML
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline
import matplotlib.mlab as mlab
from matplotlib import cm
from ipywidgets import interactive
from scipy import signal # sirve para filtros
import pywt # sirve para la transformada de wavelet: pip install PyWavelets
import pandas as pd
import sklearn
from sklearn import metrics
import urllib.request
import os
import datetime
import glob
import librosa
import librosa.display
import soundfile as sf
import scipy.stats as stats
import keras
import tensorflow as tf
from keras.models
from keras import models, layers, optimizers, losses, metrics, regularizers
from keras.layers import Input, Dense, Conv1D, MaxPooling1D, UpSampling1D, LSTM, TimeDistributed, ConvLSTM1D
from keras.models import Model
from keras import backend as K

```

Figura. 44 La biblioteca de Python

```

class AnomalyDetector(model):
    def __init__(self):
        super(AnomalyDetector, self).__init__()
        self.encoder = tf.keras.Sequential([
            layers.Dense(24, kernel_regularizer=regularizers.l2(0.001), activation="relu"),
            layers.BatchNormalization(),
            layers.Dropout(0.5),
            layers.Dense(16, kernel_regularizer=regularizers.l2(0.001), activation="relu"),
            layers.BatchNormalization(),
            layers.Dropout(0.5),
            layers.Dense(16, kernel_regularizer=regularizers.l2(0.001), activation="relu"),
            layers.BatchNormalization(),
            layers.Dropout(0.5),
            layers.Dense(16, kernel_regularizer=regularizers.l2(0.001), activation="relu"),
            layers.BatchNormalization(),
            layers.Dropout(0.5),
            layers.Dense(4, kernel_regularizer=regularizers.l2(0.001), activation="relu"),
            layers.BatchNormalization()])

        self.decoder = tf.keras.Sequential([
            layers.Dense(16, kernel_regularizer=regularizers.l2(0.001), activation="relu"),
            layers.BatchNormalization(),
            layers.Dropout(0.5),
            layers.Dense(16, kernel_regularizer=regularizers.l2(0.001), activation="relu"),
            layers.BatchNormalization(),
            layers.Dropout(0.5),
            layers.Dense(16, kernel_regularizer=regularizers.l2(0.001), activation="relu"),
            layers.BatchNormalization(),
            layers.Dropout(0.5),
            layers.Dense(24, kernel_regularizer=regularizers.l2(0.001), activation="relu"),
            layers.BatchNormalization(),
            layers.Dropout(0.5),
            layers.Dense(32, activation="sigmoid")])

    def call(self, x):
        encoded = self.encoder(x)
        decoded = self.decoder(encoded)
        return decoded

autoencoder = AnomalyDetector()

#optimizer ,loss function and index
autoencoder.compile(optimizer=tf.optimizers.Adam(lr=0.001), loss='MSE')

```

Figura. 45 BP-FNN (32-24-16-16-16-4-16-16-16-24-32)



TRABAJO FIN DE MÁSTER

```
input_dim = Input(shape=(train_data.shape[1],1))

x = Conv1D(24, (4),kernel_regularizer=regularizers.l2(0.001), activation='relu', padding='same')(input_dim)
x = MaxPooling1D(2, padding='same')(x)
x = Conv1D(16, (4),kernel_regularizer=regularizers.l2(0.001), activation='relu', padding='same')(x)
x = MaxPooling1D(2, padding='same')(x)
x = Conv1D(8, (4),kernel_regularizer=regularizers.l2(0.001), activation='relu', padding='same')(x)
x = MaxPooling1D(2, padding='same')(x)
encoded = Conv1D(4, (4),kernel_regularizer=regularizers.l2(0.001), activation='relu', padding='same')(x)

x = Conv1D(4, (4),kernel_regularizer=regularizers.l2(0.001), activation='relu', padding='same')(encoded)
x = UpSampling1D(2)(x)
x = Conv1D(8, (4),kernel_regularizer=regularizers.l2(0.001), activation='relu', padding='same')(x)
x = UpSampling1D(2)(x)
x = Conv1D(16, (4),kernel_regularizer=regularizers.l2(0.001), activation='relu', padding='same')(x)
x = UpSampling1D(2)(x)
x = Conv1D(24, (4),kernel_regularizer=regularizers.l2(0.001), activation='relu', padding='same')(x)
x = layers.GlobalMaxPooling1D()(x)
decoded = Dense(32, activation='sigmoid')(x)

autoencoder = Model(input_dim, decoded)
autoencoder.compile(optimizer=tf.optimizers.Adam(lr=0.001), loss='MSE')
```

Figura. 46 CNN-Conv1D (32-24-16-8-4-4-8-16-24-32 (globalmaxpooling))



7 ÍNDICE DE FIGURAS Y TABLAS

7.1 Índice de figuras

Figura 1 Función de Pre-énfasis	6
Figura 2 Función de ventana Hann	8
Figura 3 Señal temporal y MFCCs	10
Figura 4 Señal temporal y MFCCs	11
Figura 5 Señal temporal y Cromaticidad-STFT	13
Figura 6 Señal temporal y Cromaticidad-STFT	13
Figura 7 Función de Centroide	14
Figura 8 Señal temporal y Centroide Espectral	15
Figura 9 Señal temporal y Centroide Espectral	16
Figura 10 Función de RollOff.....	17
Figura 11 Señal temporal y RollOff	17
Figura 12 Señal temporal y RollOff	18
Figura 13 Señal temporal y Energía a corto plazo	19
Figura 14 Señal temporal y Energía a corto plazo	20
Figura 15 Señal temporal y ZCR	21
Figura 16 Señal temporal y ZCR	22
Figura 17 La curva leptocúrtica, mesocúrtica y platicúrtica, fuente: scipy.stats.kurtosis	24
Figura 18 Diagrama del aprendizaje automático	28
Figura 19 Algoritmo hacia adelante	30
Figura 20 Algoritmo de retropropagación de errores	31
Figura 21 Función de Indicadores	34
Figura 22 Autoencoder	36
Figura 23 Diagrama del sistema sobre BP-FNN	37
Figura 24 Función de Regularización L2	38
Figura 25 Dropout	39
Figura 26 Función de optimizador Adam.....	41
Figura 27 Función de MSE.....	42



Figura 28 La curva de error reconstruido	45
Figura 29 La curva de error reconstruido y Rondas de cruce	47
Figura 30 Matriz de entrada y Kernel convolucional.....	52
Figura 31 Función de MSE.....	53
Figura 32 MaxPooling (Stride = 2).....	53
Figura 33 AveragePooling (Stride = 2)	54
Figura 34 UnPooling.....	54
Figura 35 UpSampling.....	54
Figura 36 Algoritmo de la propagación hacia adelante de Conv1D (Stride = 1)	55
Figura 37 Algoritmo de ´Transpose Convolution´ (Stride = 1)	55
Figura 38 Algoritmo de la propagación hacia adelante de Conv1D (Stride ≥ 2)	56
Figura 39 Algoritmo de ´Transpose Convolution´ (Stride ≥ 2)	56
Figura 40 La curva de error reconstruido y Rondas de cruce BP-FNN (32-24-16-16-16-4-16-16-16-24-32)	59
Figura 41 41 La curva de error reconstruido y Rondas de cruce CNN-Conv1D (32-24-16-8-4-4-8-16-24-32 (globalmaxpooling))	59
Figura 42 Los sonidos normales del conjunto de entrenamiento	63
Figura 43 Los sonidos normales/anormales del conjunto de prueba	64
Figura 44 La biblioteca de Python	65
Figura 45 BP-FNN (32-24-16-16-16-4-16-16-16-24-32).....	65
Figura 46 CNN-Conv1D (32-24-16-8-4-4-8-16-24-32 (globalmaxpooling))	66

7.2 Índice de tablas

Tabla 1 ZCR para muestras de Section_01_source_test.....	23
Tabla 2 Curtosis para muestras de Section_01_source_test	24
Tabla 3 Lista de características acústicas.....	25
Tabla 4 Las diferencias entre aprendizaje automático supervisado y no supervisado	33
Tabla 5 Confusión Matriz	34
Tabla 6 Estructura de BP-FNN.....	38
Tabla 7 Resultados finales de BP-FNN.....	48



Tabla 8 Resultados de Confusión Matriz.....	48
Tabla 9 Estructura de CNN-Conv1D	57
Tabla 10 Resultados finales de BP-FNN y CNN-Conv1D	58
Tabla 11 Resultados de Confusión Matriz.....	58
Tabla 12 Colaboratory.....	64

8 BIBLIOGRAFÍA

- [1] Duan Ruiqi, heart sound identification based on mfcc and short-time energy[D], Yanshan University, 2015, 05
- [2] Chen Zhouliang, Research on fault diagnosis method of machinery based on Deep Learning[D], Nanchang Hangkong University, 2018, 05
- [3] Shiraishi Hiroshi, Signal analysis on damaged rail tread detection based on Wavelet Packet and MFCC[D], South China University of Technology, 2016, 04
- [4] Xin Liu, Research on Unsupervised Anomaly Detection Algorithm and Application [D], Electronic Science and Technology University , 2018, 05
- [5] Shi Kangkai, Improvement of audio feature extraction and research on visualization method [D], Beijing Industry University , 2017, 05
- [6] Yang Haoge, Research on the abnormal sound recognition method of aircraft engine [D], Nanchang Hangkong University, 2018, 06
- [7] John Grezma; Wang Peng; Sun Chuang; Robert X.Gao, Explainable Convolutional Neural Network for Gearbox Fault Diagnosis[J], Xian Jiaotong University, 80(2019)476-481
- [8] ECE 366 Honors Section Fall 2009 [J], Project Description, 2019
- [9] Mo Yimin; Qi Fei; Sun Chuang; Zhang Jian, Transmission fault diagnosis based on dynamic vector and BP neural network [J], Wuhan University of Technology, 1001—3997(2017)11-0049—0
- [10] Abbaspour S; Gholamhosseini H; Linden M; Sun Chuang; Zhang Jian, Evaluation of wavelet based methods in removing motion artifact from ecg



- signal[J], 16th Nordic-Baltic conference on biomedical engineering. springer international publishing, 2015, 1-4
- [11] Gupta H; Gupta D, LPC and LPCC method of feature extraction in Speech Recognition System. 2016 6th International Conference - Cloud System and Big Data Engineering (Confluence), 2016, 498-502
- [12] Plchot O; Burget L; Aronowitz H; Matějka P, Audio enhancing with DNN autoencoder for speaker recognition, 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2016, 5090-5094.
- [13] Lu Xiaoyun; Wang Hongxia, Abnormal audio recognition algorithm based on MFCC and short-term energy[J], Journal of Computer Applications, 2010(03), 796-8
- [14] Wang Y; Lawlor B, Speaker recognition based on MFCC and BP neural networks, 2017 28th Irish Signals and Systems Conference (ISSC), 2017 20-21 June 2017
- [15] Kawaguchi; K.Imoto; Y.Koizumi; N.Harada; D.Niizumi; K.Dohi; R.Tanabe; H.Purohit; T.Endo, Description and Discussion on DCASE 2021 Challenge Task 2: Unsupervised Anomalous Sound Detection for Machine Condition Monitoring under Domain Shifted Conditions, in arXiv e-prints: 2106.04492, 2021
- [16] François Chollet, Deep Learning with Python[M], Manning publications, 1st edición (10 enero 2018)
- [17] Juan Ciscar, Alfons, Optimización de prestaciones en técnicas de aprendizaje no supervisado y su aplicación al reconocimiento de formas[D], Universitat Politècnica de València, 2000, 01
- [18] García García, Germán, Applying deep learning techniques to terminology extraction in specific domains, Universidad Politécnica de Madrid[D], 2021, 07
- [19] Boureau Y; Bach F; Lecun Y; et al, Learning mid-level features for recognition: Computer Vision and Pattern Recognition[C], 2010
- [20] Zeiler M D, Fergus R, Visualizing and understanding convolutional networks[J], 2013, 8689:818-833
- [21] Hinton G E, Reducing the dimensionality of data with neural networks[J], Science, 2006, 313(5786):504-507



[22] Wang Liang, Research on abnormal sound recognition technology based on MFCC[D], Harbin Engineering University, 2018, 03

[23] DCASE2021

<http://dcase.community/challenge2021/task-unsupervised-detection-of-anomalous-sounds>

[24] DCASE2020

<http://dcase.community/challenge2020/task-unsupervised-detection-of-anomalous-sounds>

[25] Deep Learning Fundamentals with Keras

<https://learning.edx.org/course/course-v1:IBM+DL0101EN+2T2021/home>

[26] Valerio Velardo, The Sound of AI

<https://www.youtube.com/channel/UCZPFjMe1uRSirmSpznqvJfQ>

[27] Beginner's Guide to Audio Data

<https://www.kaggle.com/fizzbuzz/beginner-s-guide-to-audio-data/notebook>

[28] Intro to Autoencoders

https://www.tensorflow.org/tutorials/generative/autoencoder#third_example_anomaly_detection

[29] Python processing audio signal combat: teach you to implement music genre classification and feature extraction

<https://flashgene.com/archives/17964.html>

[30] aialgorithm/Blog

<https://github.com/aialgorithm/Blog>

[31] Timeseries anomaly detection using an Autoencoder

https://keras.io/examples/timeseries/timeseries_anomaly_detection/#prepare-training-data

[32] Machine learning anomaly detection

<https://blog.csdn.net/yzhou86/article/details/78932289>

[33] Summary of anomaly detection methods

https://blog.csdn.net/qq_40195360/article/details/105380444

[34] Speech signal preprocessing and feature parameter extraction

<https://blog.csdn.net/cwfjimgudan/article/details/71112171>



[35] Detecting Heart Arrhythmias with Deep Learning in Keras with Dense, CNN, and LSTM

<https://towardsdatascience.com/detecting-heart-arrhythmias-with-deep-learning-in-keras-with-dense-cnn-and-lstm-add337d9e41f>

[36] Audio feature extraction-common audio features

<https://www.cnblogs.com/xingshansi/p/6815217.html>

[37] Keras

<https://keras.io/about/>

[38] Librosa

<https://librosa.org/doc/latest/index.html>

[39] Pyaudioanalysis

<https://awesomeopensource.com/project/tyiannak/pyAudioAnalysis>

[40] Deep learning feed-forward neural network(forward propagation and error back propagation)

<https://www.cnblogs.com/Luv-GEM/p/10694471.html>

[41] scipy.stats.kurtosis

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kurtosis.html>

[42] Transpose Convolution

https://zhuanlan.zhihu.com/p/115070523?from_voters_page=true

[43] Transpose Convolution

https://github.com/vdumoulin/conv_arithmetic

[44] The calculation process of one-dimensional convolution in neural networks

<https://www.cnblogs.com/talkaudiodev/p/14287562.html>

[45] CS231n convolution neural networks for visual recognition

<https://cs231n.github.io/convolutional-networks/>

[46] Python keras.layers.Conv1D()

<https://www.programcreek.com/python/example/89676/keras.layers.Conv1D>

[47] Data analysis and feature extraction with Python

<https://www.kaggle.com/pmarcelino/data-analysis-and-feature-extraction-with-python>