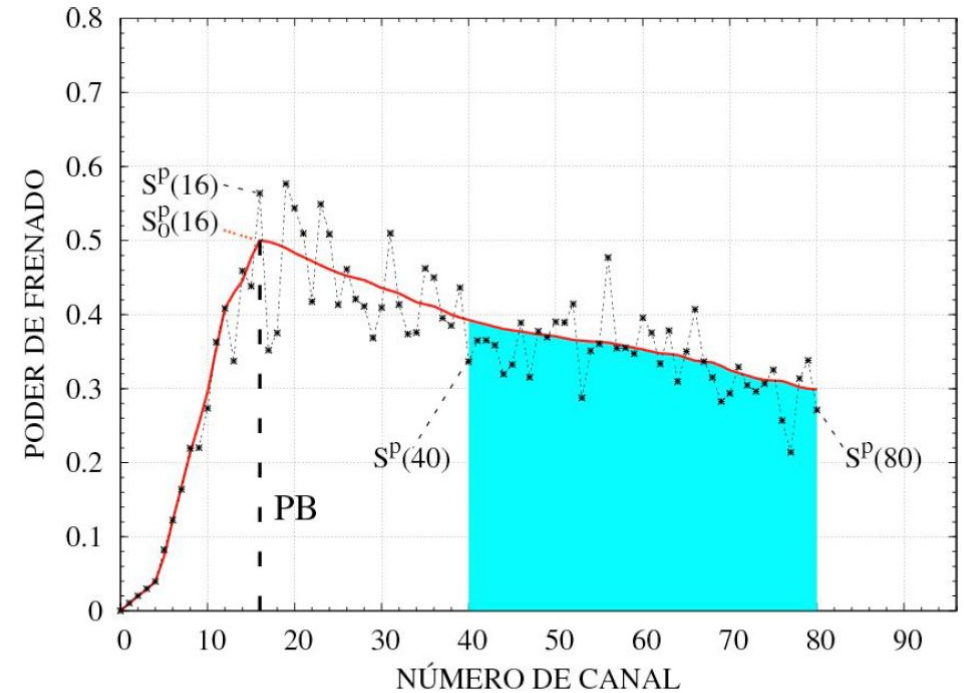


Rubro	Descripción	Puntos descontados al no respetar lo indicado (p. = puntos)
Etapas	Deben presentar las 7 etapas de su proyecto.	-15p. (por etapa que no aparezca)
Alto contraste	Deben usar colores de alto contraste.	-10p. (en general, es decir: no por diapositiva)
Palabras por diapositiva	Usar 20 palabras o menos por diapositiva. (Las citas o referencias de imágenes no son parte del conteo).	-7p. (por cada diapositiva con más de 20 palabras)
Portada	<ul style="list-style-type: none"> • Título del curso • Tema de su proyecto (o el título de su proyecto) • Nombre de los integrantes (que sí trabajaron) • Año 	-5p. (si cualquiera de los puntos indicados no se cumple)
Gráficas	Poner gráficas de buen tamaño. Es recomendable que ocupen toda la diapositiva si es posible.	-4p. (por cada diapositiva con gráficas que no se alcancen a distinguir)
Numeración	Numeren las diapositivas. La portada no hace falta numerarla.	-2p. (por cada diapositiva sin numeración)
Citas y referencias	<p>Todas las imágenes deben tener cita.</p> <p>Si ustedes las hicieron deben indicarlo (creado por #nombre del autor).</p>	<p>-1p. (por cada imagen que no sea suya y no este citada)</p> <p>*Si ustedes hicieron la imagen, pero no pusieron cita, no les bajaremos puntos.</p>
Código	<p>Por ningún motivo incluir fragmentos de código. Si desean mostrar los parámetros que usaron coloquenlos en una tabla.</p> <p>* Solo el equipo que trabajó el proyecto con el Dr. Romel, pueden incluir código. Sin embargo deben entender que de incluirlo deben explicarlo con lujo de detalle para que cualquiera lo pueda entender.</p>	-3p. (por cada diapositiva con código)
Hablar de problemas y soluciones	Pueden incluir una sección que hable de los problemas mas importantes que tuvieron. Aquellos retos que técnicos o teóricos que los mantuvieron despiertos en la noche tratando de entender o encontrar una solución.	<p>" +20p." (por incluir la sección)</p> <p>*Sí, son puntos positivos. Los profesores discutirán si sus problemas presentados merecen los puntos extra.</p>

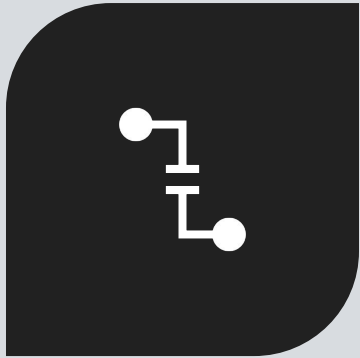
INTRODUCCIÓN

- Las Curvas de Bragg tienen dos parámetros: La Energía Total de un ion (E) o partícula y el Pico de la Curva (BP).
- El Análisis Digital de Forma de Pulsos (DPSA) con Redes Neuronales Artificiales nos permitirán estimar estos dos parámetros.



Curva CV (rojo) y datos con ruido (puntos negros). Imagen extraída de

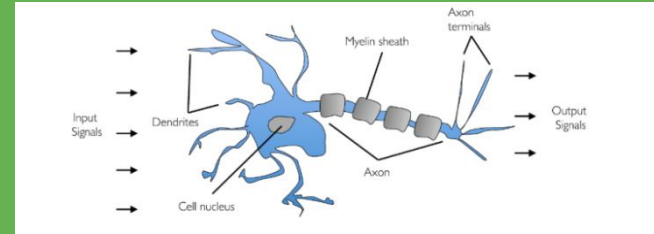
¿Qué es un Perceptrón? – Parte I



UN PERCEPTRÓN ES LA UNIDAD MÁS FUNDAMENTAL UTILIZADA PARA CONSTRUIR UNA RED NEURONAL.



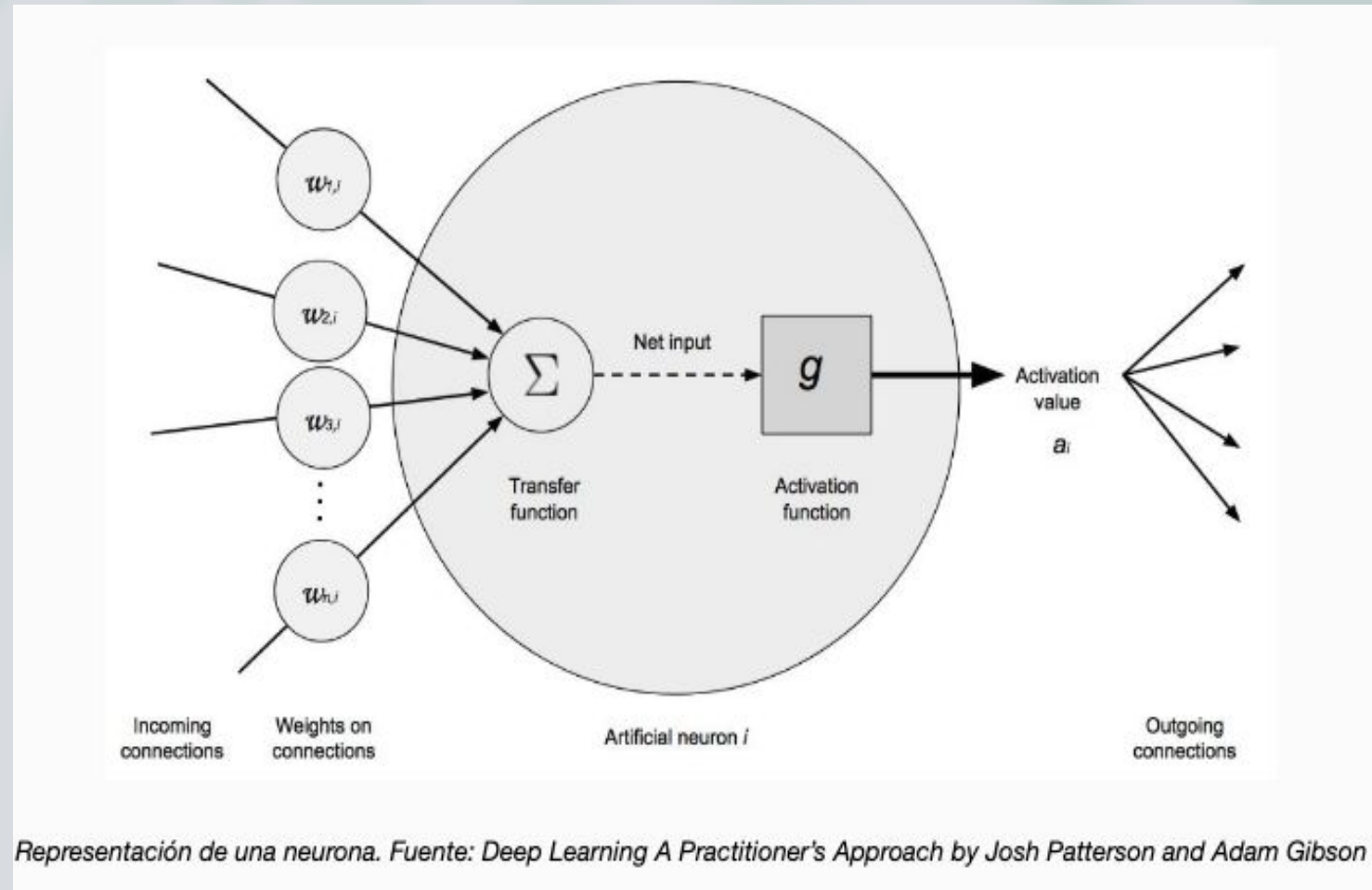
UN PERCEPTRÓN ES EL HOMÓLOGO A UNA NEURONA EN EL CEREBRO HUMANO



EN EL CASO DE UNA NEURONA BIOLÓGICA, MÚLTIPLES SEÑALES DE ENTRADA INGRESAN A TRAVÉS DE DENDRITES , Y UNA SEÑAL DE SALIDA SE DISPARA (BASÁNDOSE EN LA FUERZA DE LAS SEÑALES DE ENTRADA Y OTROS MECANISMOS.

¿Qué es un Perceptrón? – Parte II

- Un perceptrón se asemeja conceptualmente a una neurona.
- Dentro de cada neurona ocurren dos operaciones: la suma ponderada de sus entradas y la aplicación de una función de activación.
- El valor neto de entrada pasa por una **función de activación**, que lo transforma en un valor de salida.



Función de activación



Las funciones de activación controlan la información que se propaga de una capa a la siguiente.



Gracias a que se combinan funciones de activación no lineales con los múltiples capas, los modelos de redes son capaces de aprender relaciones no lineales.

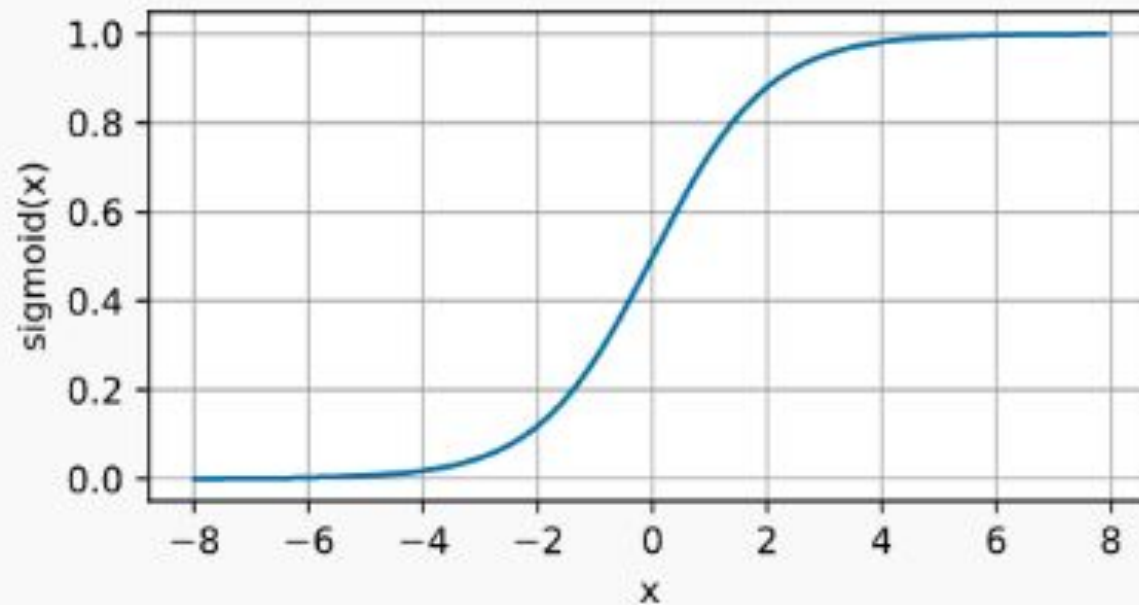


La gran mayoría de funciones de activación convierten el valor de entrada neto de la neurona en un valor dentro del rango $(0,1)$ o $(-1,1)$.



Cuando el valor de la función de activación es cero, se tiene una neurona inactiva.

$$\text{sigmoid}(x) = \frac{1}{1 + \exp(-x)}$$



Representación función activación sigmoide.

Función de activación Sigmoide

La función sigmoide transforma valores en el rango de $(-\infty, +\infty)$ a valores en el rango $(0, 1)$.

Función de coste (loss function)

- Esta función es la encargada de cuantificar la distancia entre el valor real y el valor predicho por la red.

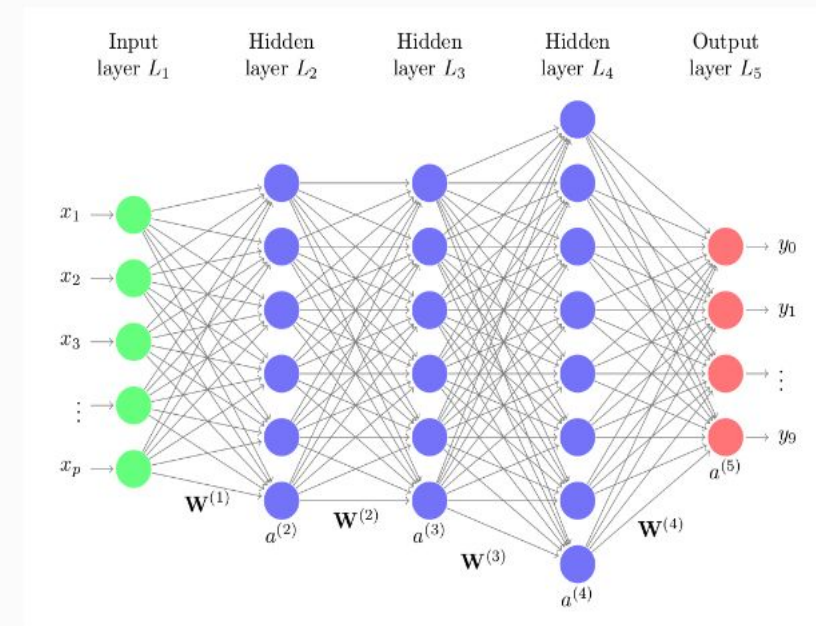
Error cuadrático medio

- Para una determinada observación i , el error cuadrático se calcula como la diferencia al cuadrado entre el valor predicho y el valor real:

$$l^{(i)}(\mathbf{w}, b) = (\hat{y}^{(i)} - y^{(i)})^2$$

Multilayer Perceptron (MLP)

- Cuando se combinan múltiples capas, la red puede aprender relaciones mucho más complejas entre los predictores y la variable respuesta.
- Nosotros utilizaremos un algoritmo de múltiples capas. De este modo podemos tener dos variables output: Energía y el Pico de Bragg.



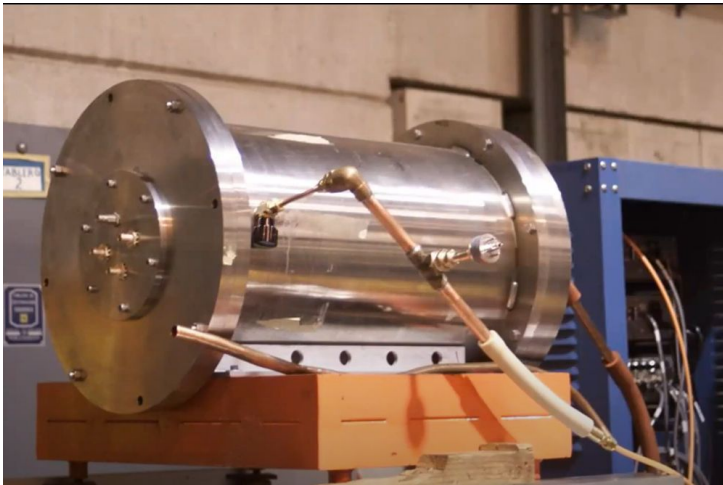
Representación de una red neuronal feed-forward (multi-layer perceptron). Fuente: Computer Age Statistical Inference 2016

CURVAS DE BRAGG : UNA APLICACIÓN DE ANÁLISIS DE SEÑALES DIGITALES BASADA EN REDES NEURONALES ARTIFICIALES (ANN)

Equipo 06
Integrantes:

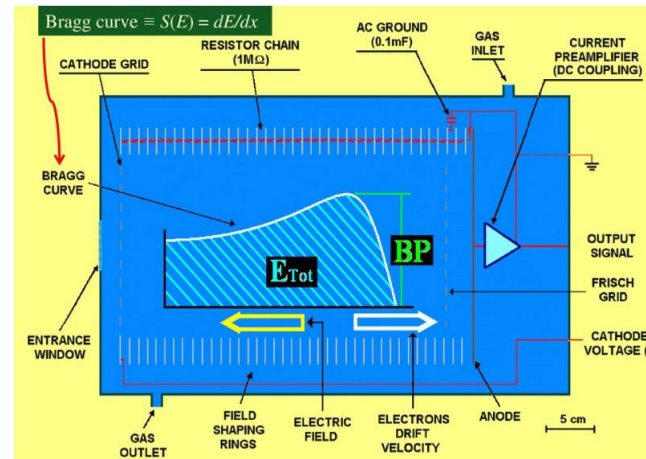
- Casas Espinosa Axel
- Espinoza Torres Víctor Hugo
- Gómez Garduño Leonardo Arturo
- Sánchez Reyes Amelia

¿Alguna vez se han preguntado cuál es la relación entre las señales digitales y las redes neuronales artificiales?

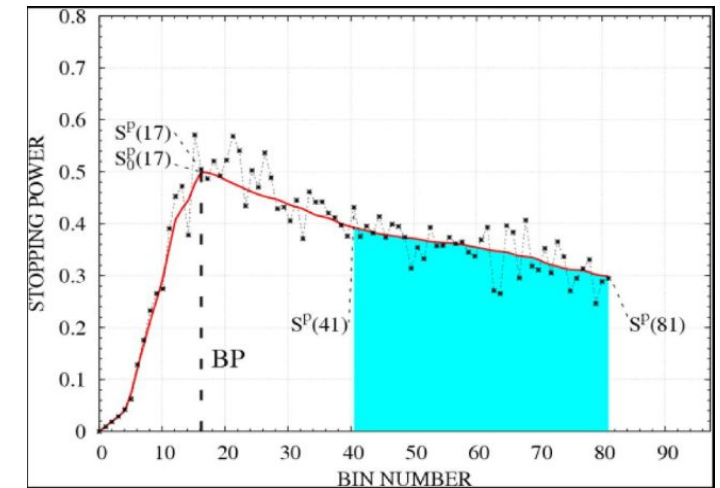


Fuente: Imagen tomada de PPT Doctor Jaime Vega Castro

Bragg curve spectrometer



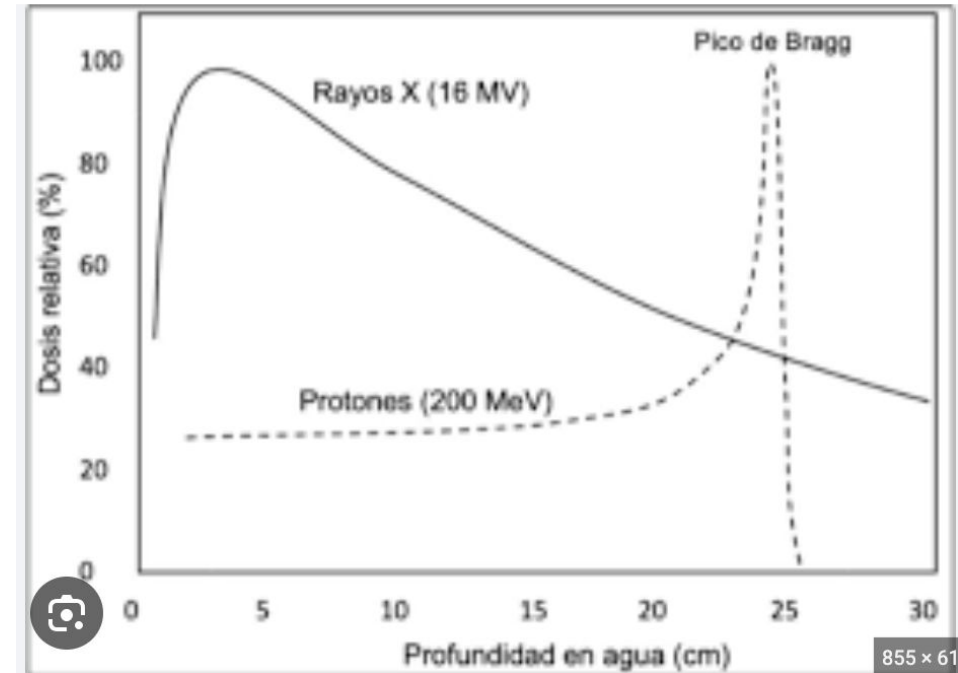
Fuente: Imagen tomada de Effect of Signal Noise on the Learning Capability of an ANN. Juan Jaime Vega Castro, Maria del Rocio Reynoso Vallecillo, Humberto Carrillo Calvet



Fuente: Imagen tomada de Effect of Signal Noise on the Learning Capability of an ANN. Juan Jaime Vega Castro, Maria del Rocio Reynoso Vallecillo, Humberto Carrillo Calvet

CURVAS DE BRAGG

- La curva de Bragg es típica para partículas con carga pesada y describe la pérdida de energía de la radiación ionizante durante el viaje a través de la materia.



Pico de Bragg obtenido en agua. Imagen extraída de la red.

DEFINICIÓN DEL PROYECTO

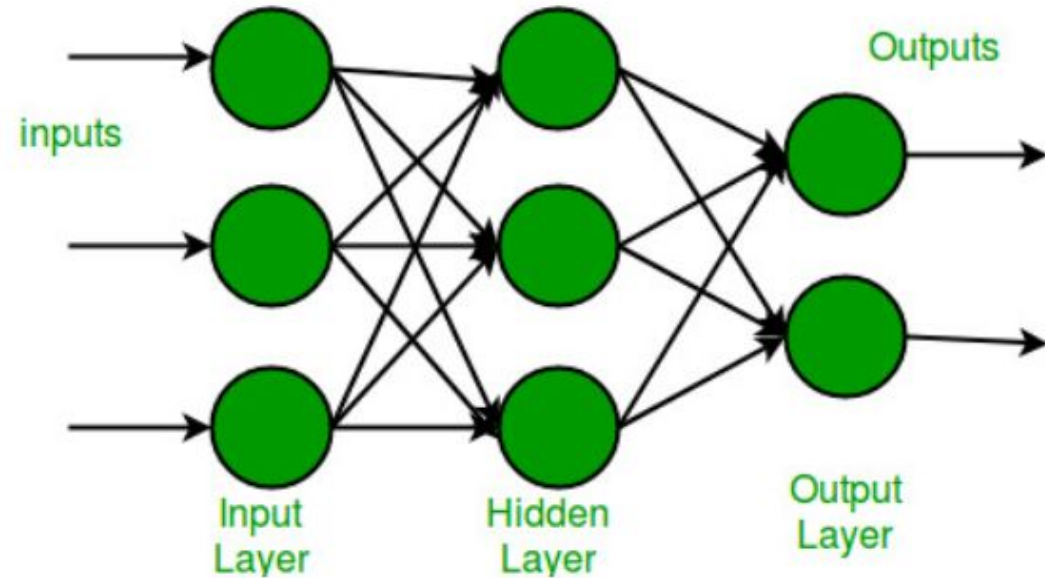
Nuestro proyecto de Curvas de Bragg pertenece a la categoría de aprendizaje supervisado.

Analizaremos curvas sintéticas de Bragg.

Cada curva se compone de 81 valores

Las variables output (respuesta) son Energía (E) y Pico de Bragg (PB)

La Red Neuronal Artificial a utilizar será la de Perceptrón Multicapa.



Fuente: Imagen tomada de <https://www.geeksforgeeks.org/multi-layer-perceptron-learning-in-tensorflow/>

ESTRUCTURA BASE DE DATOS

- Trabajamos un dataset con 10% de ruido
- Se tienen 100 muestras de curvas de bragg con ruido del 10%
- Cada muestra tiene 451 clases
 - (41 Energías asociadas a 11 Picos de Bragg)
- Así, se logra una base de datos con 45,100 registros.

Obs:

Nuestro dataset ya se encuentra estandarizado, con valores en el intervalo: $(0,1)$

OBTENCIÓN DE LOS DATOS

Las curvas de Bragg y sus variables descriptivas se obtuvieron a partir del Análisis Digital de Señales.

Para ampliar la base de datos se crearon 451 clases por curva.

Obs:

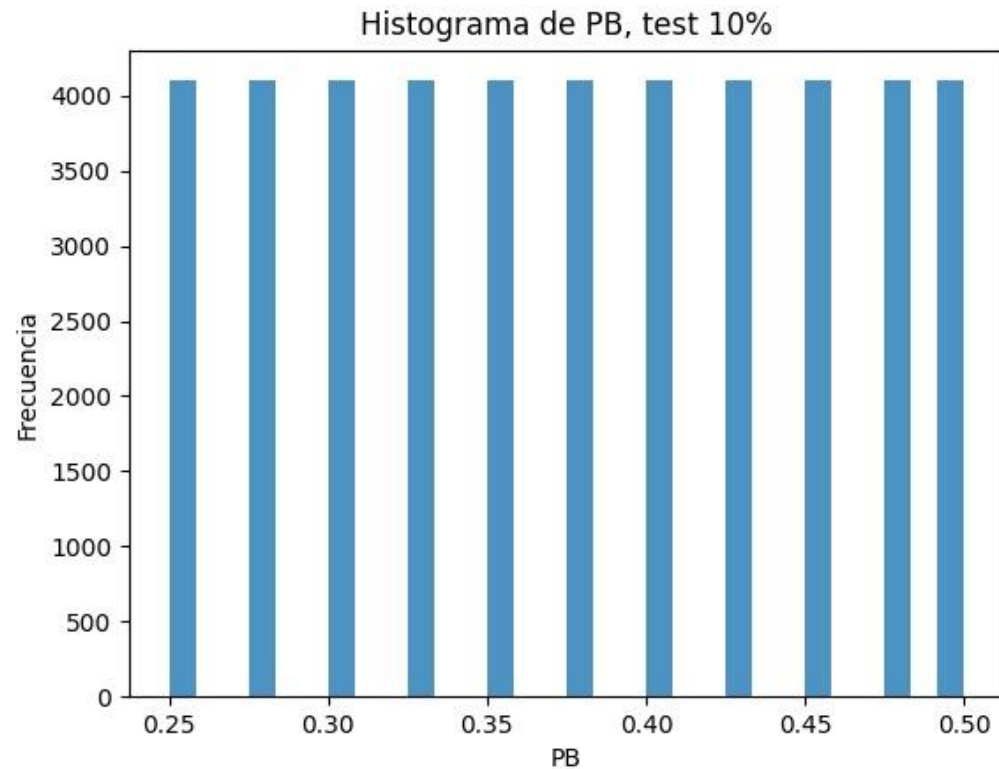
A nosotros se nos compartió la base de datos con 45,100 observaciones limpia y estandarizada.

ANÁLISIS EXPLORATORIO

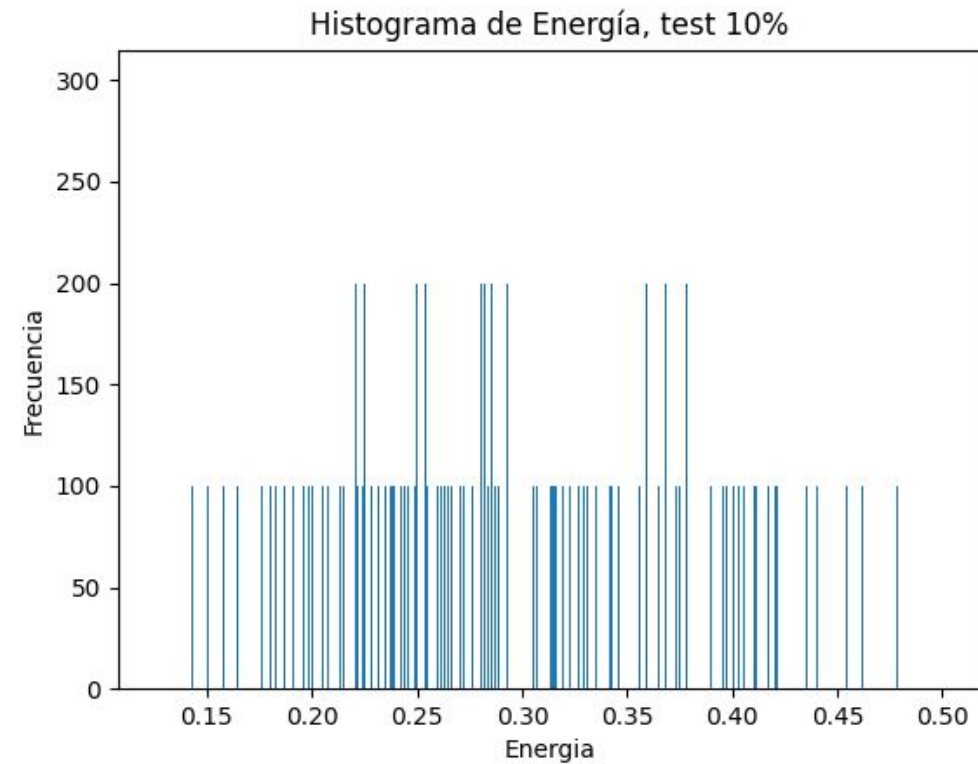
Resumen de datos:

Columna	Tipo dato	Descripción	Rango
ID	Integer	Variable que indica el número de observación	[1, 45100]
0:80	Float	Variables descriptivas (x1, x2, ..., x80)	[0,1]
PB	float	y1: Pico de Bragg	[0,1]
E	float	y2: Energía	[0,1]

Con el histograma de Pico de Bragg se puede ver que existen 4100 curvas asociadas a un mismo PB; en total tenemos 11 PB.

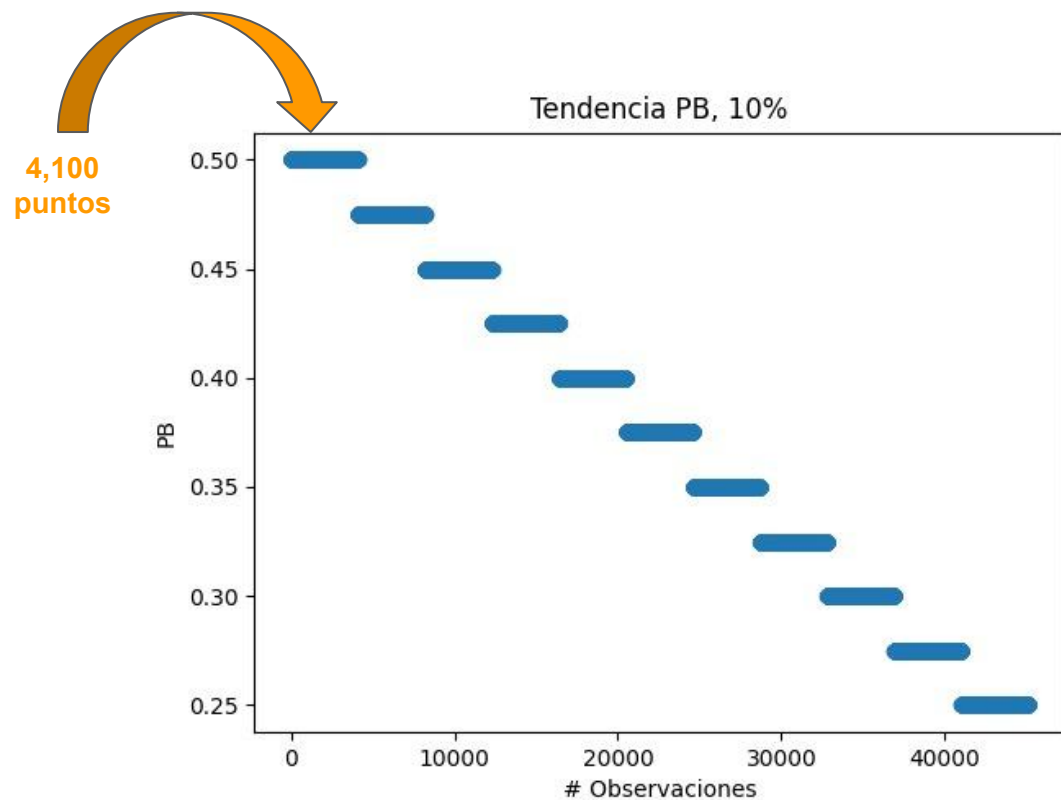


Fuente: imagen generada a partir de la base de datos con ayuda de librerías de python.

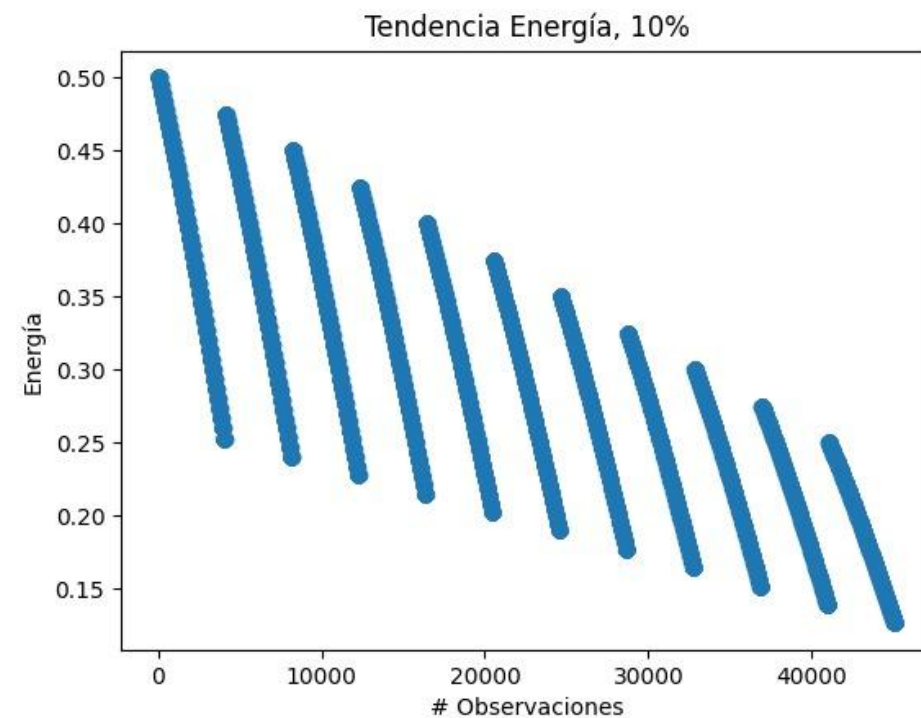


Fuente: imagen generada a partir de la base de datos con ayuda de librerías de python.

Nuevamente, tenemos 4,100 puntos por cada valor diferente de Pico de Bragg; así como, 41 energías asociadas a cada Pico de Bragg.

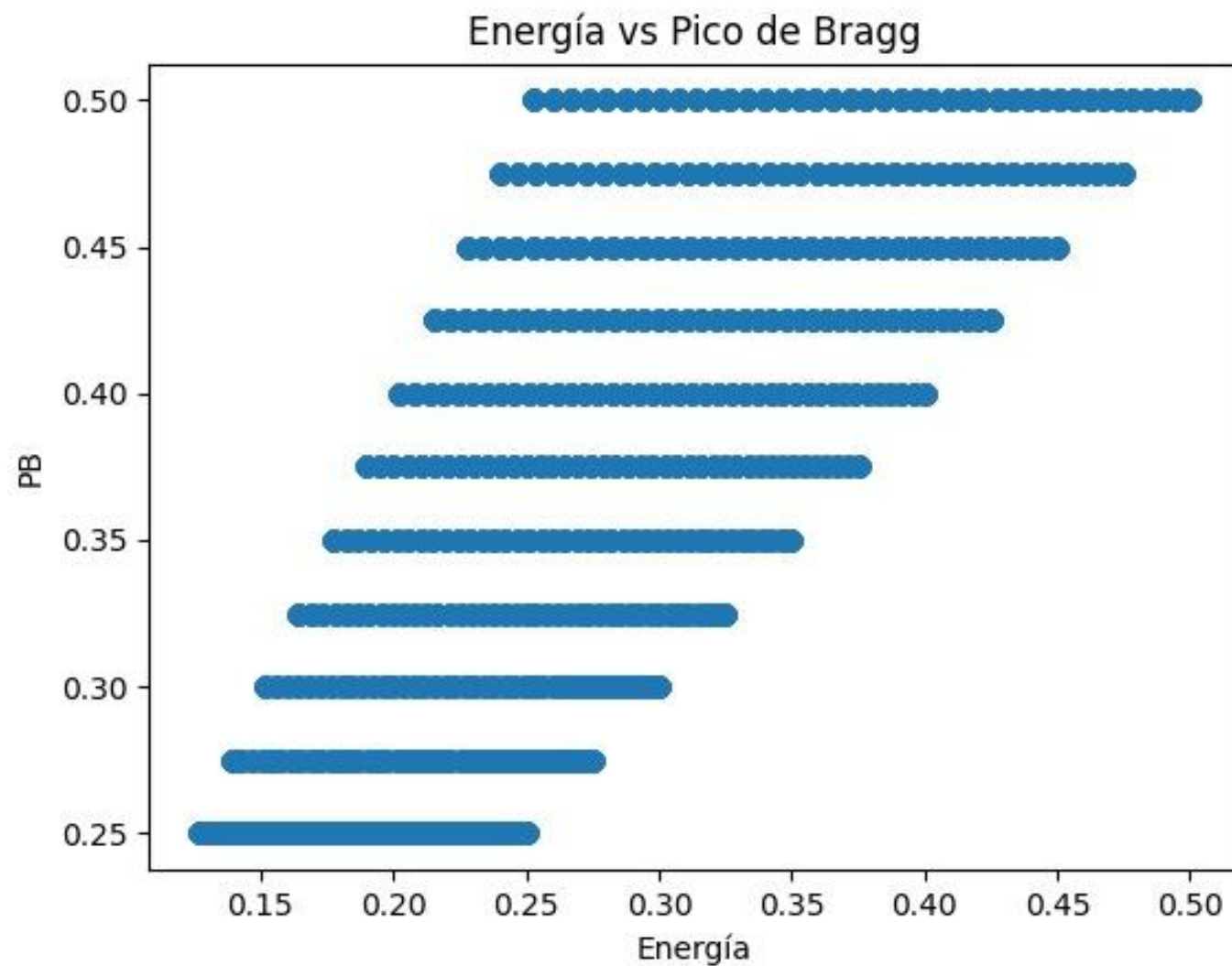


Fuente: imagen generada a partir de la base de datos con ayuda de librerías de python.



Fuente: imagen generada a partir de la base de datos con ayuda de librerías de python.

MODELO IDEAL



Fuente: imagen generada a partir de la base de datos con ayuda de librerías de python.

Etapa 5: Preparación de datos

Datos Numéricos:

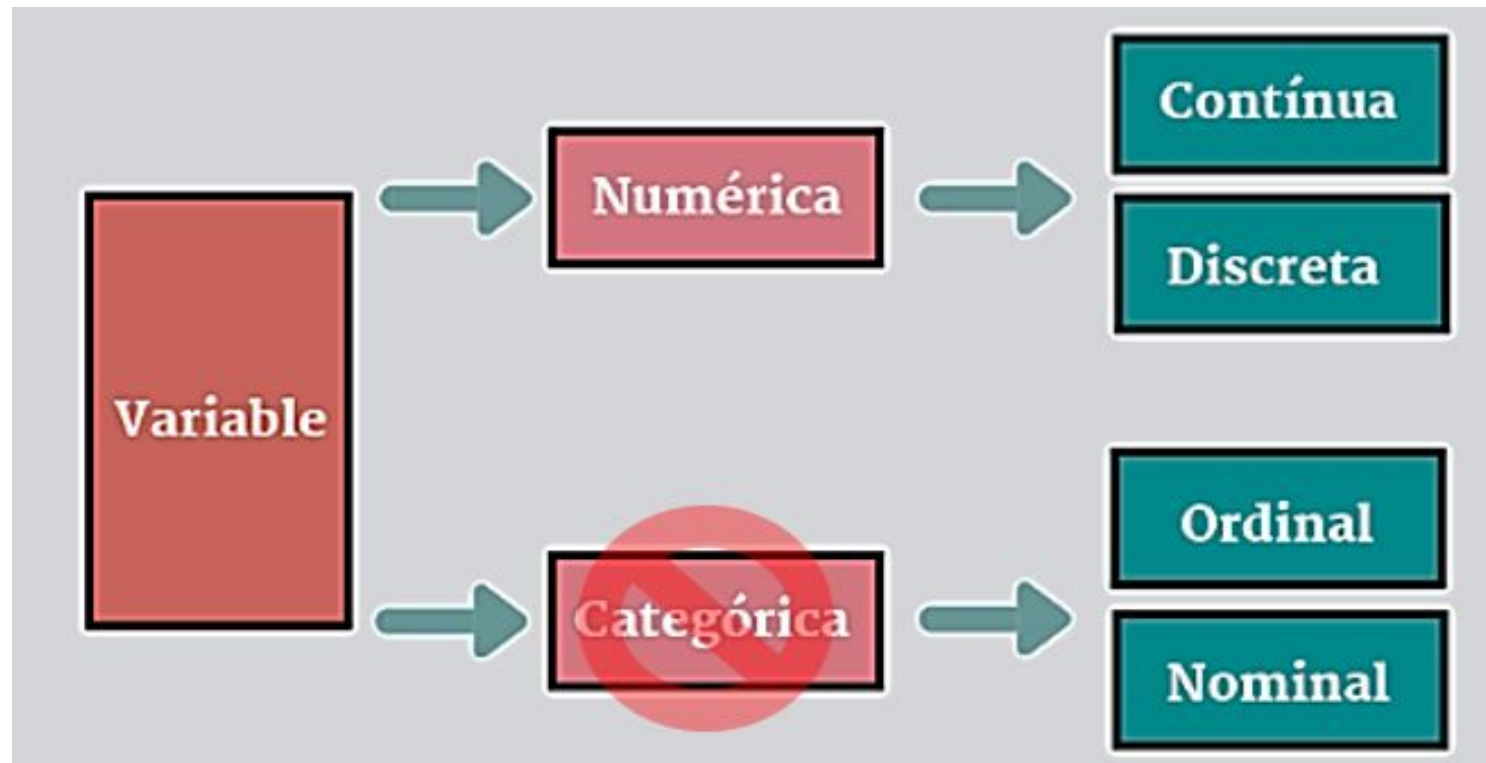
Todos los datos experimentales son esenciales; no se agregan ni eliminan atributos ni se añaden ceros.



Fuente: Imagen tomada de: <https://labeleyourdata.com/articles/machine-learning-and-training-data>

Datos Categóricos:

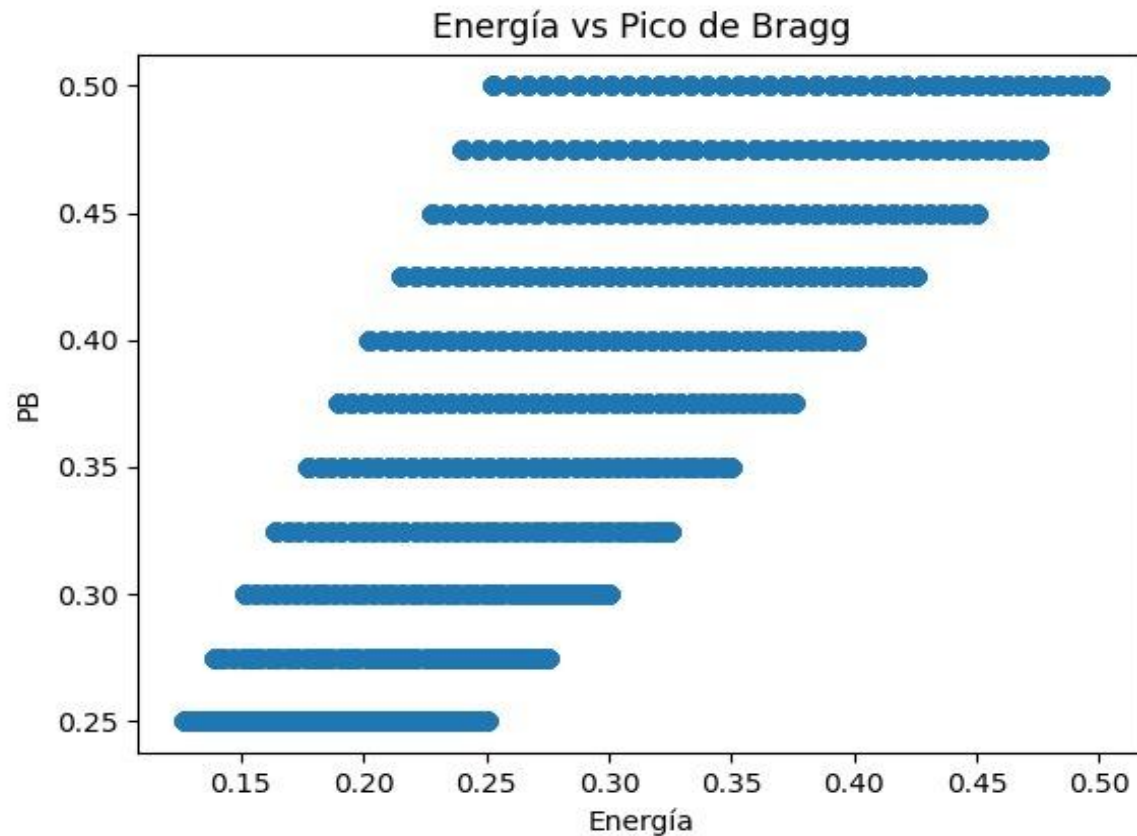
No hay valores categóricos, ya que trabajamos exclusivamente con datos numéricos.



Fuente: Imagen tomada de:
<https://www.marketing-analitico.com/analitica-web/tipos-de-variables-en-el-analisis-de-datos/>

Escalado de Atributos:

Los valores objetivo (energía, picos de Bragg) ya están en el rango 0-1, entonces no se escaló.



Fuente: imagen generada a partir de la base de datos con ayuda de librerías de python.

Selección de Atributos:

Todos son relevantes; un atributo lleno de ceros no se eliminó porque no afecta al modelo.



Fuente: Imagen generada con "Bing Image Creator"

AJUSTE DEL MODELO

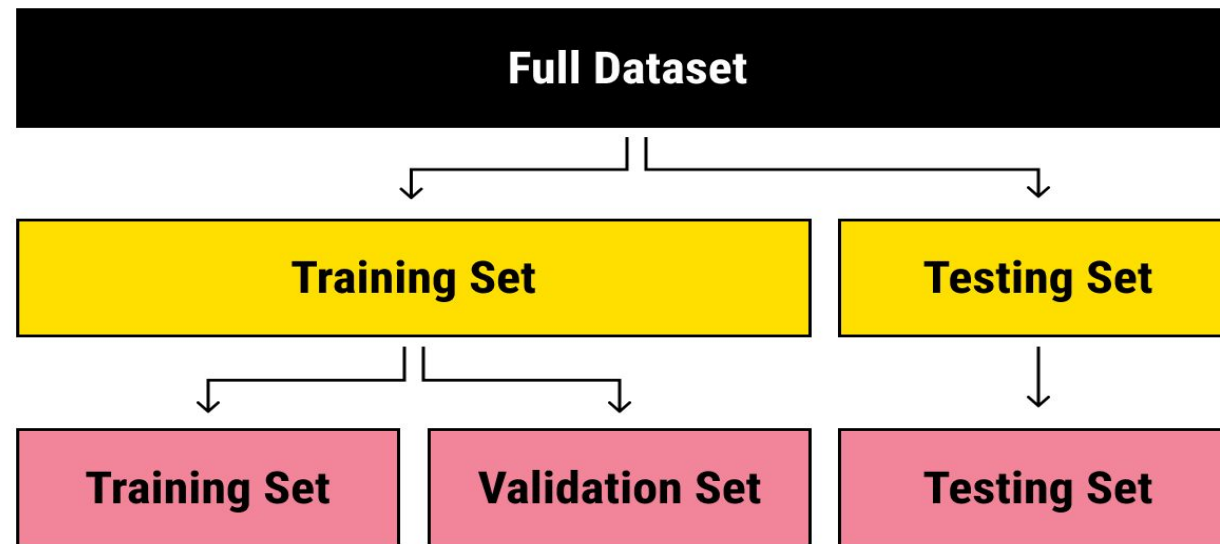
- Nuestro modelo utiliza una Red Neuronal que resolverá un problema de regresión: predecir los valores de la Energía y el Pico de Bragg a partir de un conjunto de variables predictoras.
- El algoritmo de ANN que utilizaremos será el Multi-layer Perceptron.

CONJUNTO DE ENTRENAMIENTO

Se usó la división existente entre conjuntos de prueba y entrenamiento proporcionados.

Training: 45,100 obs con 10% ruido

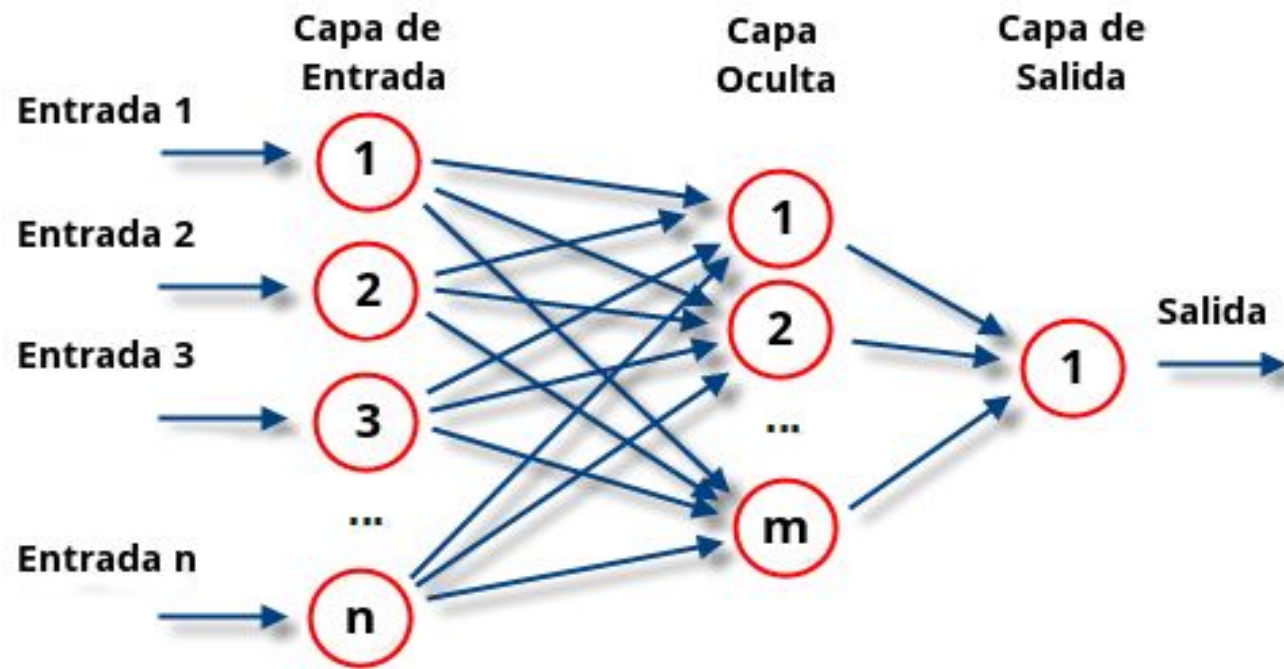
Testing: 45,100 obs con 10% ruido



Fuente: Imagen tomada de:
<https://labelyourdata.com/articles/machine-learning-and-training-data>

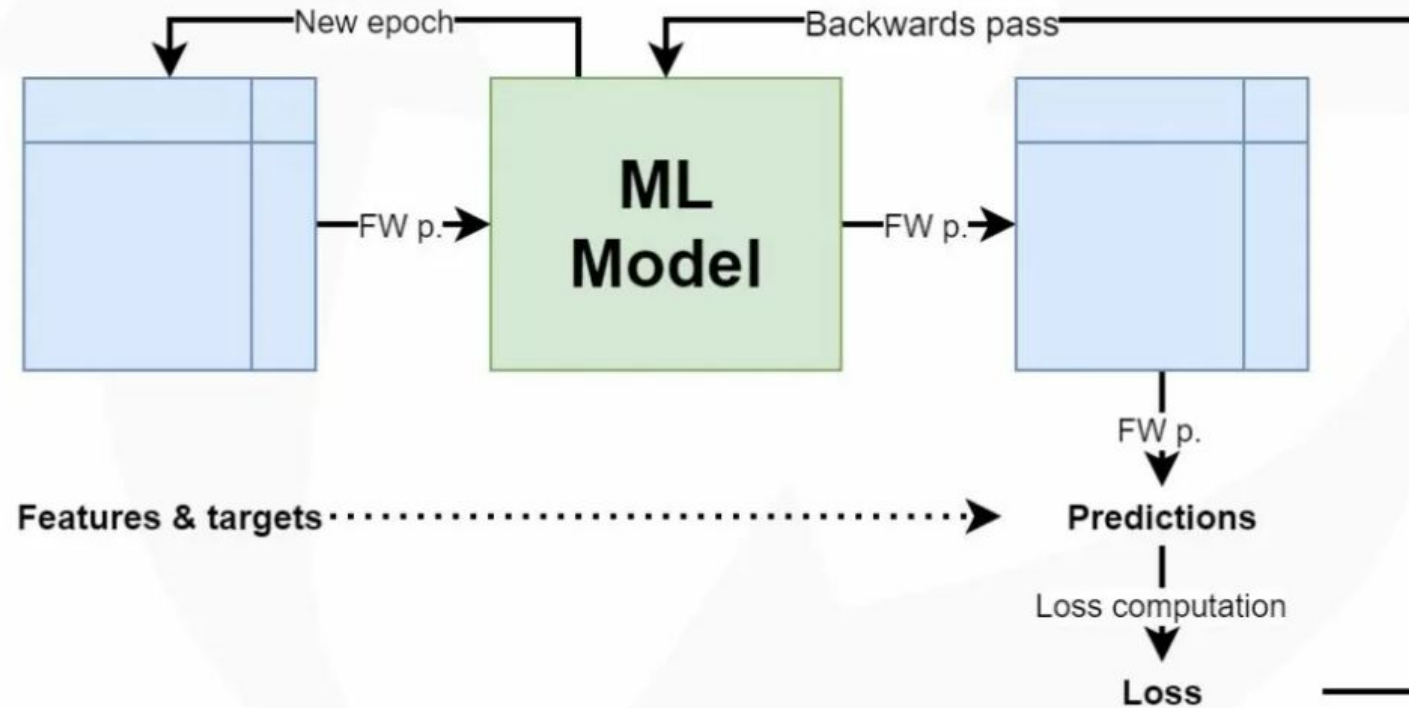
RESUMEN DEL MODELO

El modelo consta de una capa de entrada de 81 unidades, después tenemos tres capas ocultas, con 33, 39 y 9 unidades respectivamente, finalmente tenemos la capa de salida con solo 2 unidades.



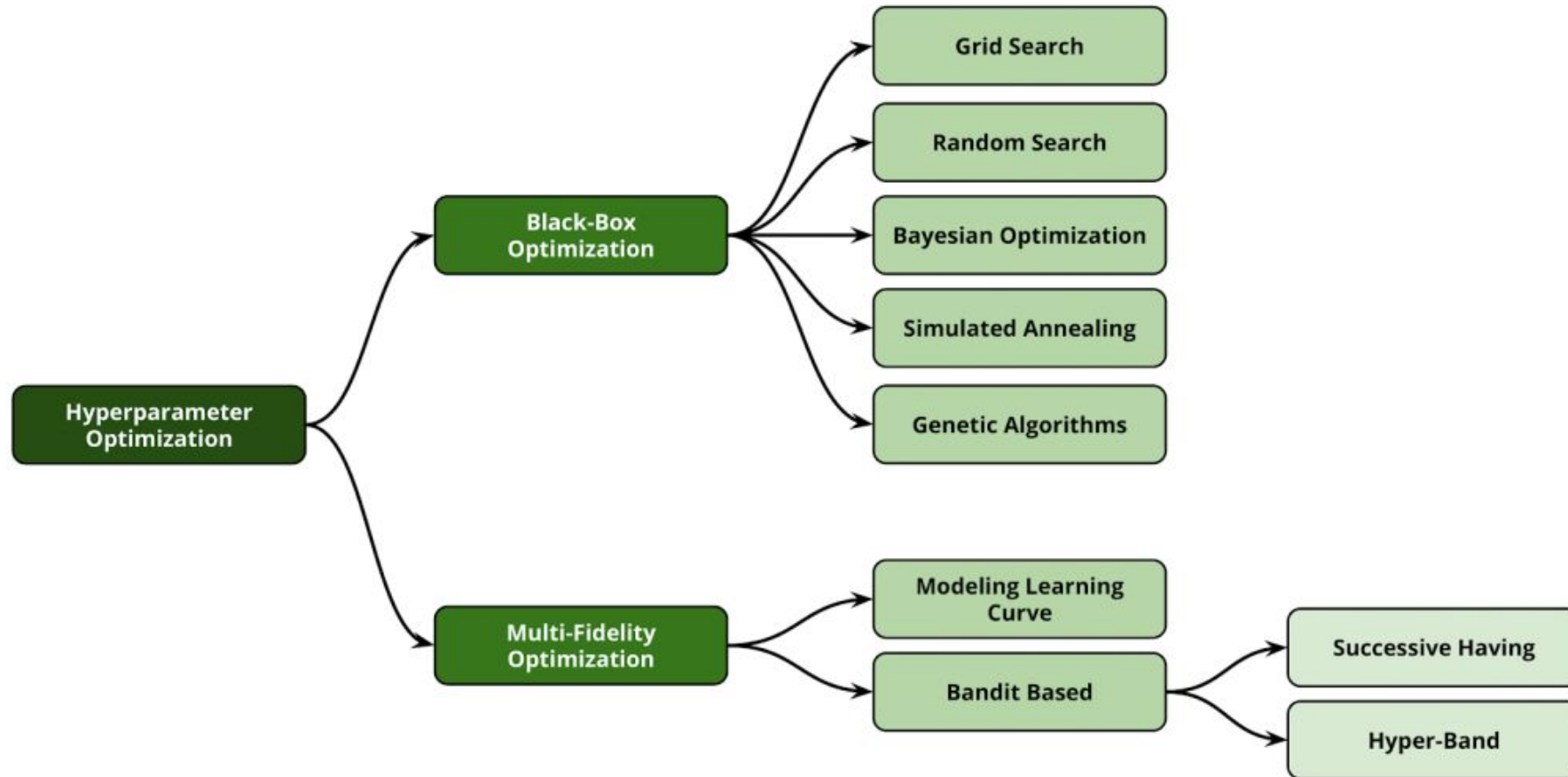
Fuente: Imagen tomada de:
https://es.wikipedia.org/wiki/Perceptr%C3%B3n_multicapa

¿CÓMO SE VA ENTRENANDO EL MODELO?



Fuente: Imagen tomada de:
<https://github.com/christianversloot/machine-learning-articles/blob/main/how-to-use-k-fold-cross-validation-with-keras.md>

Busqueda de hiper-parametros



Fuente: Imagen tomada de:
<https://peepdata.github.io/hyperparameter-optimization-hyperband/>

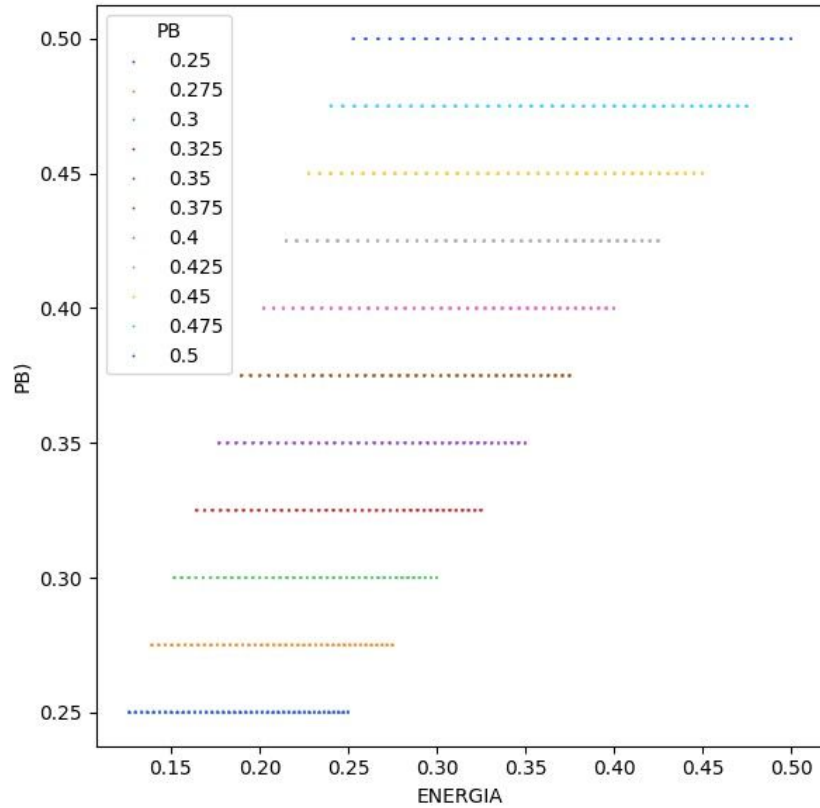
ARQUITECTURA DEL MODELO

Parametros	Valor
Metrics	Error Cuadrático Medio
Optimizer	Adam
loss	mean_absolute_error (error absoluto medio)
callbacks	earlystopping
validation_split	0.2 (20% de datos)
learning rate	.01
epocas	2000
función de activación	sigmoide

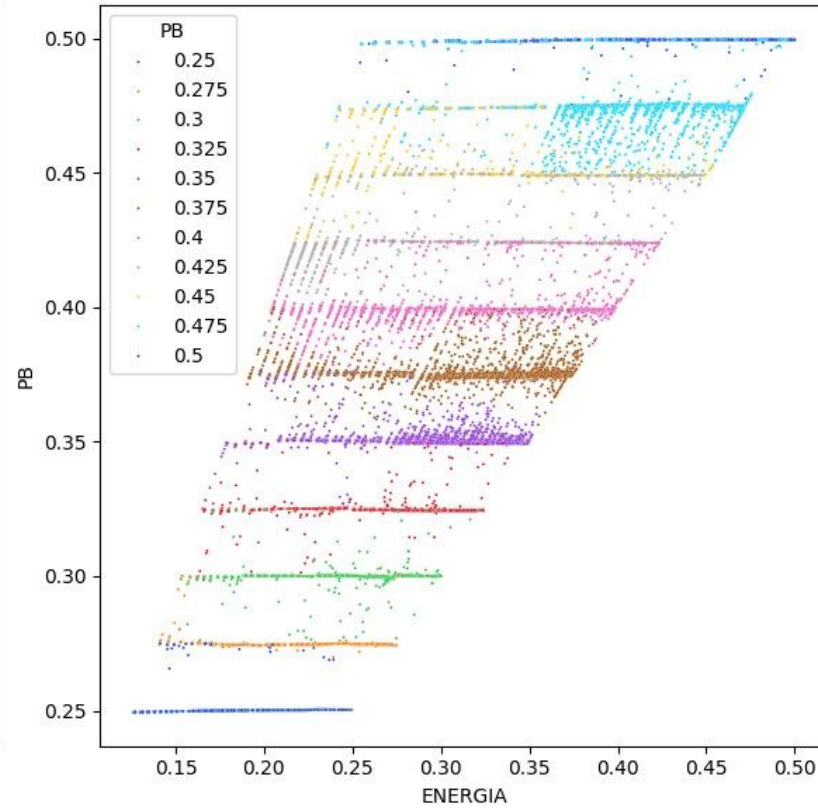
Fuente: Tabla generada a partir de parámetros obtenidos del algoritmo hyperband y algunos parámetros preestablecidos. .

ANÁLISIS Y RESULTADOS

Testing data



Training data

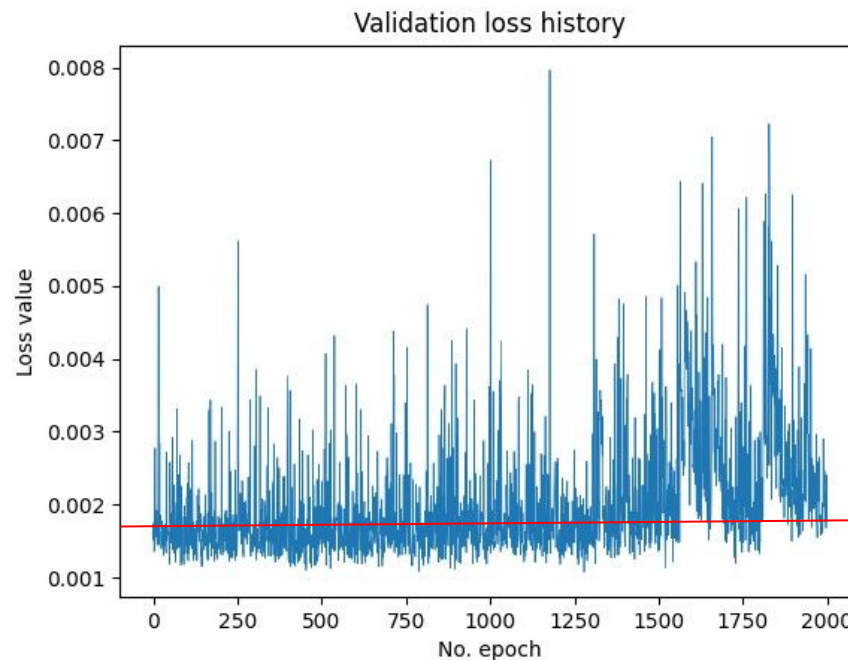


Test	Value
Test loss:	0.001836022711 3589644
Test accuracy:	1.879028604889 7542e-05

Fuente: imagen generada a partir de la base de datos con ayuda de librerías de python.

VALIDATION LOSS HISTORY

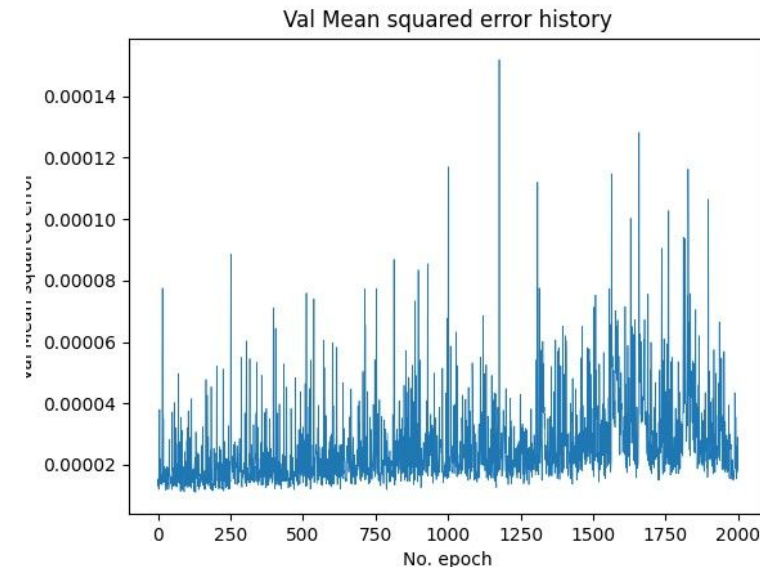
- Visualizamos la pérdida de validación a lo largo de las épocas.
- Identificamos patrones de aprendizaje y posibles problemas de **sobreajuste**.



Monitoreando la función `val_loss`, el modelo se detuvo en la época 369, a partir de ahí se dejó de minimizar el error.

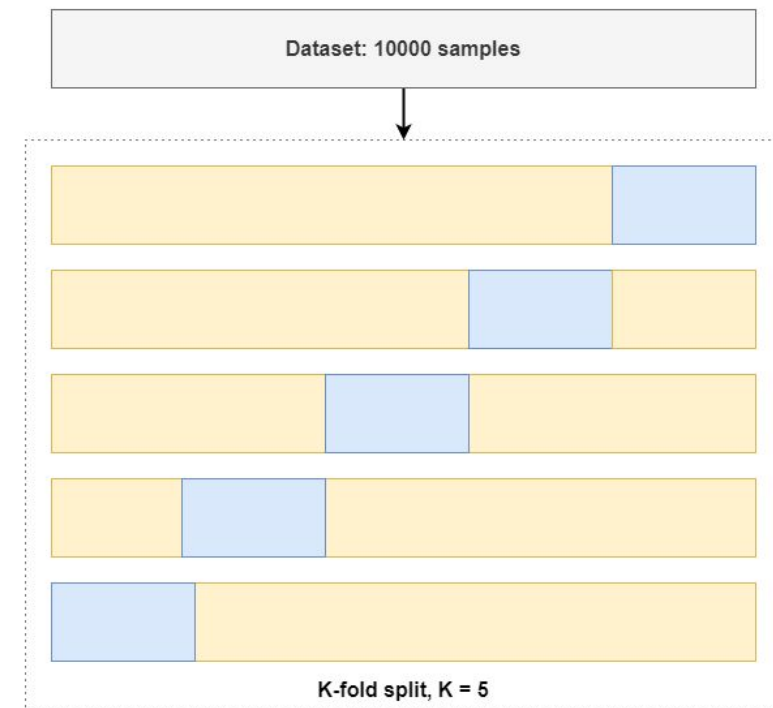
EVALUACIÓN DEL NÚMERO DE ÉPOCAS

- Monitoreando la función 'accuracy', el modelo encontró el mayor ajuste hasta la época límite (2000).
- Por tanto, ajustamos modelo con **2000** épocas para aprender relaciones complejas. (callbacks)
- Controlamos el **sobreajuste** y la **convergencia** (early stopping, K-Fold Cross Validation).



CROSS VALIDATION CON K-FOLDS

- Nuestra red neuronal aprende de una determinada base de datos (training dataset), pero, ¿cómo saber que nuestro modelo servirá para la vida real?
- Para validar nuestro modelo, unimos datos de nuestro training data set + testing data set y validamos.
- Evaluamos accuracy y función loss.



Fuente: Imagen tomada de:
<https://github.com/christianversloot/machine-learning-articles/blob/main/how-to-use-k-fold-cross-validation-with-keras.md>

PROBLEMAS Y HALLAZGOS

1. Problemas con el modelo: Consideramos como mayor problema o dificultad la búsqueda de hiperparámetros
2. Problemas con el hardware y tiempos: Las arquitecturas propuestas por el Dr. Humberto y Dr. Jaime muestran un gran número de épocas, lo que representó una limitación para nuestros recursos tecnológicos.
3. Otros problemas: Errores humanos que nos obligaban a repetir el modelo.

PROBLEMAS Y HALLAZGOS

Pareciera que con un mayor trabajo/análisis en los hiperparametros nos puede permitir tener muy buenos resultados sin la necesidad de un gran número de épocas.

Si se tiene un modelo más óptimo en cuanto a procesamiento se refiere, es posible incrementar el número de variables predictoras (X's) y tener mayor detalle.

Bibliografía

1. *Learning limits of an artificial neural network, por J.J. Vega y R. Reynosa, y H. Carrillo Calvet.*
2. *Redes Neuronales Artificiales para el Reconocimiento de Curvas de Bragg, por C. RENDÓN BARRAZA.*
3. *Regularization methods vs large training sets, por J.J. Vega, H. Carrillo Calvet y José Luis Jiménez Andrade*