



## XVIII EDICIÓN DEL SIMPOSIO DE INGENIERÍA ELÉCTRICA, SIE-2019

### Efectos de la normalización en la compresión de señales electroencefalográficas.

#### *Effects of normalization on the compression of electroencephalographic signals.*

**Ing. Rigoberto Acosta-González<sup>1</sup>, Dr. Carlos A. Bazán-Prieto<sup>2</sup>, Lester Frias-  
Dominguez<sup>3</sup>**

1- Rigoberto Acosta-González. Universidad Central "Marta Abreu" de las Villas, Cuba.

E-mail: riacosta@uclv.cu

2- Carlos A. Bazán-Prieto. Universidad Central "Marta Abreu" de las Villas, Cuba. E-

mail: cabazan@uclv.edu.cu

3- Lester Frias-Dominguez. Universidad Central "Marta Abreu" de las Villas, Cuba. E-

mail: lfdominguez@uclv.cu

#### **Resumen**

Comprimir señales electroencefalográficas (EEG) es vital, por el uso creciente de dispositivos portátiles, por la necesidad del ahorro de energía de estos para la transmisión de la información. El empleo de autocodificadores para comprimir requiere de varias etapas de preprocesamiento de la señal. Dentro de estas etapas la normalización de la señal EEG es de importancia pues reduce el tiempo de entrenamiento y alcanza un mejor aprendizaje. Pero la normalización introduce errores por aproximación. En esta investigación se analizan los efectos de la normalización en la compresión de señales EEG mediante autocodificadores. Se proponen dos sistemas de compresión, uno donde se emplea la normalización y otro donde se omite esta etapa dada las bondades de la base de datos seleccionada. Se utilizan las métricas RMSE y PRD para comparar ambos sistemas. Aunque la diferencia de RMSE y PRD entre ambos sistemas sea pequeña, el sistema que omite normalizar posee una cantidad de fragmentos inferior al 22% del total con un PRD mayor al 20% de distorsión. Mientras que el sistema que emplea normalización la cantidad de fragmentos es mayor al 25% del total. Por tanto la normalización influye en el proceso de recuperación introduciendo mayor distorsión. Se hace necesario otros métodos de normalización donde los errores por aproximación sean



menores, pues las bases de datos de señales EEG no siempre poseen sus valores en el rango de -1 hasta 1.

### **Abstract**

*Compressing electroencephalographic (EEG) signals is vital, due to the increasing use of portable devices, due to the need to save energy for the transmission of information. The use of self-encoders to compress requires several stages of preprocessing the signal. Within these stages, the normalization of the EEG signal is of importance because it reduces the training time and achieves a better learning. But normalization introduces errors by approximation. In this research, the effects of normalization on the compression of EEG signals through autocoders are analyzed. Two compression systems are proposed, one where normalization is used and another where this step is omitted given the benefits of the selected database. The RMSE and PRD metrics are used to compare both systems. Although the difference between RMSE and PRD between both systems is small, the system that fails to normalize has a number of fragments less than 22% of the total with a PRD greater than 20% distortion. While the system that uses normalization the amount of fragments is greater than 25% of the total. Therefore, normalization influences the recovery process introducing greater distortion. It is necessary other normalization methods where the approximation errors are smaller, since the databases of EEG signals do not always have their values in the range of -1 to 1.*

**Palabras Clave:** Aprendizaje profundo; Aprendizaje automático; Funciones de activación; Redes neuronales; Autocodificadores apilados

**Keywords:** Deep learning; Machine learning; Activation functions; Neural networks; Stacked autoencoder

## **1. Introducción**

La compresión de datos es una de las áreas de las ciencias de la computación más interesantes y de gran investigación. La compresión de señales electroencefalográficas (EEG) es hoy una necesidad para el desarrollo de las aplicaciones que, de manera creciente, están orientadas a los equipos portátiles y ambulatorios. Muchos de estos equipos realizan estudios de larga duración y se comunican de forma inalámbrica,



utilizando baterías, por lo que al aplicar algoritmos de compresión se puede reducir el consumo de energía [1].

Un autocodificador es una red neuronal que está entrenada para intentar copiar su entrada a su salida. La red se puede considerar que consta de dos partes: una función de codificador  $h = f(x)$  y un decodificador que produce una reconstrucción  $r = g(h)$ . Si un codificador automático tiene éxito simplemente aprendiendo a configurar  $g(f(x)) = x$  en todas partes, entonces no es especialmente útil. En cambio, los autocodificadores están diseñados para no poder aprender a copiar perfectamente. Por lo general, están restringidos en formas que les permiten copiar solo aproximadamente y copiar solo la entrada que se parece a los datos de entrenamiento. Debido a que el modelo se ve obligado a priorizar qué aspectos de la entrada deben copiarse, a menudo aprende propiedades útiles de los datos [2].

Antes de construir un modelo de aprendizaje profundo, es necesario un paso más: la normalización de los datos. La normalización de datos es una práctica común en los sistemas de aprendizaje automático, principalmente porque disminuye el tiempo de entrenamiento de la red y aumenta el rendimiento general de la red [3].

El empleo de autocodificadores es pobre en la compresión de señales EEG. En [4] se emplean autocodificadores apilados totalmente conectados y en [5] autocodificadores convolucionales apilados para comprimir señales EEG. En estas publicaciones nunca se menciona una desnormalización en el proceso de reconstrucción de la señal EEG. Por tanto el error por normalización es despreciado. En [6] se utilizan autocodificadores apilados totalmente conectados donde se emplea la normalización y desnormalización, en las etapas de compresión y descompresión respectivamente, aunque se emplee la transformada discreta del coseno (DCT siglas en inglés) en ambas etapas.

Los artículos revisados anteriormente emplean la normalización como una etapa en el preprocesamiento de la señal EEG antes de entrenar el autocodificador. El proceso de desnormalización introduce un error por aproximación en la reconstrucción de la señal EEG. Por tanto en este estudio se propone analizar los efectos de la normalización en la compresión de señales EEG usando autocodificadores apilados.



## 2. Metodología

Para el estudio de los efectos de normalización se proponen dos sistemas de compresión como se ilustra en la Figura 1. Cada sistema es dividido en dos etapas: Compresión y Descompresión.

Las señales EEG utilizadas para el estudio provienen de la base de datos CHB-MIT Scalp EEG Database. Esta base de datos [7] es una colección de grabaciones de múltiples señales fisiológicas durante el sueño. Los pacientes fueron monitoreados en "Boston's Beth Israel Hospital Sleep Laboratory" para evaluar enfermedades relacionadas con el

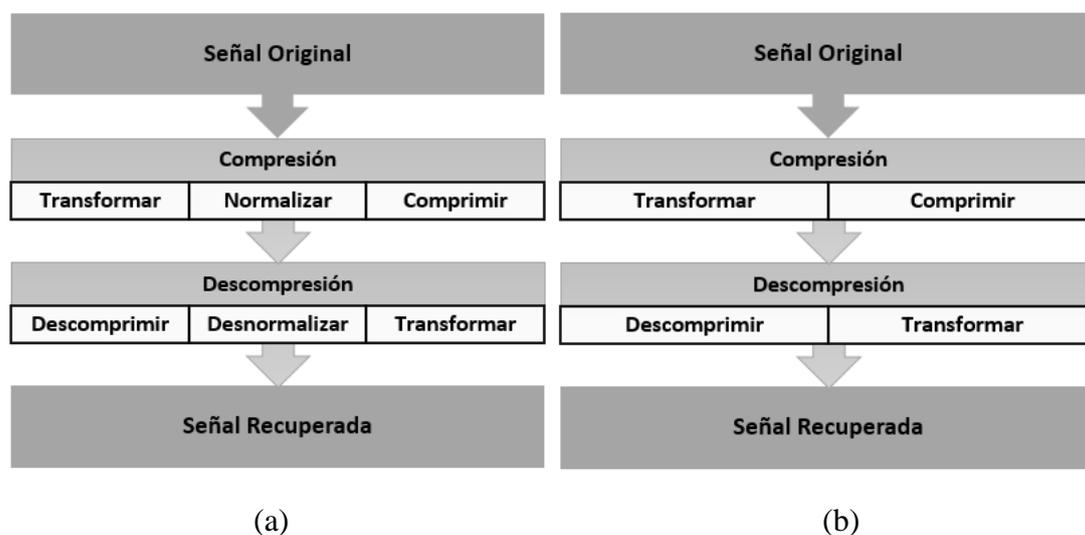


Figura 1. Modelos de compresión utilizados usando normalización (a) y sin normalización (b).

sueño. La base de datos está formada por más de 80 horas divididas en 18 casos con 4 o más señales cada uno, entre las que se encuentra el electrocardiograma, la presión arterial, el EEG y la señal de respiración. La señal EEG, fue adquirida a 250 muestras por segundo con 12 bit de resolución y está anotada con respecto a las etapas del sueño y la apnea [1].

### 2.1 Etapa de compresión

El primer proceso en la etapa de compresión es la transformación. Este proceso consiste en dividir los 72 millones de muestras, existentes en la base de datos, en varios fragmentos de tamaño 128, 256 y 512 muestras. Estos tamaños de fragmentos fueron escogidos basados en el estudio realizado en [8].

El segundo proceso en esta etapa del sistema de compresión de la Figura 1a emplea la normalización. El resultado de esta normalización son valores comprendidos entre 0 y 1. La normalización es realizada utilizando la siguiente ecuación:

$$x_{normalizada} = \frac{x_{original} - \min(x_{original})}{\max(x_{original}) - \min(x_{original})} \quad (1)$$

Donde  $\min(x_{original})$  y  $\max(x_{original})$  es el valor mínimo y máximo de la señal original respectivamente. En el sistema de compresión de la Figura 1b se omite la normalización debido a que los valores, de la señal EEG de la base de datos, están comprendidos en el rango de -0.3 hasta 0.3.

Comprimir es el tercer proceso de ambos sistemas de compresión, se construye el mismo modelo red neuronal para ambos esquemas como se muestra en la Figura 2. En esta etapa solo se emplea el codificador destacado en la Figura 2. El codificador, un autocodificador apilado, posee tres capas de neuronas totalmente conectadas. La cantidad de neuronas en la capa de entrada es dependiente de la longitud de los fragmentos construidos en la etapa de transformación. Modificando el número de neuronas en la tercera capa del codificador ("Señal Comprimida") se pueden obtener diferentes razones de compresión para las señales EEG.

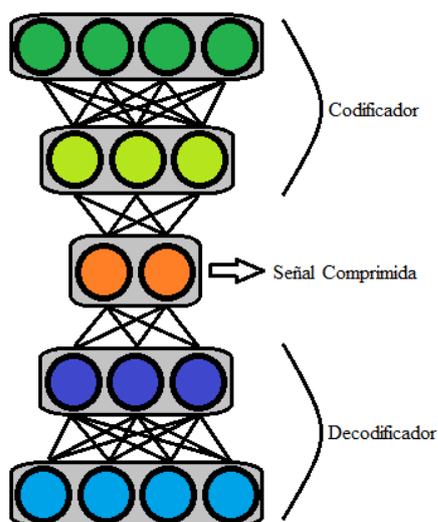


Figura 2. Autocodificadores apilados totalmente conectados empleado en los sistemas de compresión.

La cantidad de neuronas en la segunda capa del codificador cumple con la siguiente ecuación:

$$Cant_{n_2} = \left( (Cant_{n_1} - Cant_{n_3}) \div 2 \right) + Cant_{n_3} \quad (2)$$

Donde  $Cant_{n_1}$  y  $Cant_{n_3}$  es el número de neuronas en la primera y tercera capa del codificador respectivamente. La ecuación 2 es empleada para que el número de neuronas, en la capa 2, no quede a la elección de un número al azar. Este tipo de autocodificadores



apilados son conocidos como sub-completos pues el número de neuronas disminuye en cada capa del codificador.

## 2.2 Etapa de descompresión

Descomprimir es el primer proceso, en esta etapa, de ambos sistemas de compresión. Como se ilustra en la Figura 2 el decodificador es un autocodificador apilado de igual estructura que el codificador, la misma cantidad de neuronas en cada capa, lo que construido de forma inversa.

El proceso de desnormalización solo ocurre en el sistema de compresión de la Figura 1a siguiendo la siguiente ecuación:

$$x_{original} = x_{normalizada} \times (\max(x_{original}) - \min(x_{original})) + \min(x_{original}) \quad (3)$$

Por tanto los procesos de normalización y desnormalización de cada fragmento de señal EEG implica el envío de 2 valores, máximo y mínimo de cada fragmento, unido a la señal comprimida. En los estudios [4] y [5] no fueron incluidos estos valores junto a la señal comprimida demostrando que el proceso de desnormalización no es tomado en cuenta en la reconstrucción de la señal EEG. Mientras en [6] estos valores viajan junto a la señal comprimida pues forman parte del proceso de reconstrucción de la señal EEG. En el sistema de compresión de la Figura 1b no es necesario el proceso de desnormalización.

El último proceso, en ambos sistemas de compresión, es la transformación de la señal EEG descomprimida a su longitud original.

## 2.3 Funciones de activación

Cada neurona debe tener una función de activación. Esta da a la neurona la propiedad no lineal necesaria para modelar los conjuntos de datos complejos no lineales. La función toma la suma ponderada de todas las entradas y genera una señal de salida [9]. La elección de las funciones de activación en ambos sistemas de compresión esta relacionada con los valores de amplitud de la señal EEG. Para el sistema de la Figura 1a, luego de la normalización, estos valores quedan comprendidos entre 0 y 1. Para el sistema de la Figura 1b toma los valores entre -0,3 y 0,3.

Las funciones de activación para el sistema de compresión usado en la Figura 1a son las utilizadas en [10], donde se utiliza la unidad lineal rectificadora (ReLU siglas en inglés)[11] para todas las capas a excepción de la última donde se emplea una función de activación lineal.



En el sistema de compresión de la Figura 1b, fue seleccionada la función tangente hiperbólica ( $\tanh$ ) [12] para todas las capas, menos la última donde se emplea una función de activación lineal. La elección de la función tangente hiperbólica es debido a los valores que entrega a su salida, estos valores tienen un rango entre -1 y 1.

#### 2.4 Configuración de los hiperparámetros de entrenamiento

El lenguaje de programación utilizado para la construcción de los sistemas de compresión fue Python [13]. La realización de los modelos de autocodificadores fue a través de la biblioteca de alto nivel Keras [10]. Esta permite usar a Tensorflow [14] como biblioteca de funciones de aprendizaje profundo de bajo nivel.

La función de costo utilizada para optimizar fue el error medio cuadrático (MSE en inglés) como se muestra en la ecuación 4.

$$MSE = \frac{1}{N} \sum_{n=0}^{N-1} (x_{original}[n] - x_{reconstruida}[n])^2 \quad (4)$$

Donde N es la cantidad de muestras en cada fragmento generado en el proceso de transformación. El optimizador utilizado fue Adam[15] donde los hiperparámetros  $\beta_1$ ,  $\beta_2$  y  $\epsilon$  toman sus valores típicos 0.9, 0.999 y  $10^{-8}$  respectivamente. El valor de la tasa de aprendizaje (learning rate) tiene un valor de 0.0001. El número de iteraciones empleada es de 50. Para ambos sistemas los hiperparámetros se mantiene constantes.

### 3. Resultados

La base de datos fue dividida en tres partes: para entrenamiento un 70 %, para validación un 15 % y para prueba un 15 % de las señales. Esta forma de división de los datos es llamada 70-30 en la jerga del aprendizaje profundo. Cuando la base de datos posee menos de un millón de datos para el entrenamiento, validación y prueba, este tipo de división es la opción recomendable.

Existen diversas expresiones matemáticas para evaluar el desempeño cuantitativo de las diversas técnicas de compresión de señales EEG. Se prefiere la valoración subjetiva de los médicos especializados para definir si no existen pérdidas significativas en una señal comprimida con pérdidas, lo que constituye una tarea compleja [16].

Una medida para evaluar la de compresión de datos es definida en [17] usando la siguiente ecuación:

$$C = 100 \times \frac{L_{orig} - L_{comp}}{L_{orig}} \% \quad (5)$$



Donde  $L_{orig}$  y  $L_{comp}$  son las longitudes de los fragmentos originales y comprimidos respectivamente. Ambos sistemas de compresión planteados en la sección 2 tienen un valor de compresión de datos fijo e igual a 50 %. Esto significa que de 128, 256 y 512 muestras luego de la transformación son comprimidas a 64, 128, 256 respectivamente.

El parámetro más empleado en la literatura es "Percentage Root mean square Difference" (PRD) el cual queda definido por la siguiente ecuación:

$$PRD = \sqrt{\frac{\sum_{n=0}^{N-1} (x_{original}[n] - x_{reconstruida}[n])^2}{\sum_{n=0}^{N-1} (x_{original}[n])^2}} \times 100 \quad (6)$$

Este parámetro indica la distorsión global obtenida en términos de porcentaje y permite estimar en qué grado se ha conservado la forma de onda de la señal [1].

Otro parámetro es la raíz del error medio cuadrático "Root Mean Square Error" (RMSE) explica la distorsión en términos absolutos. Para el caso de la señal EEG, este se puede expresar en  $\mu V$ , por lo que está muy ligado al manejo de los especialistas encargados del análisis de estas señales bioeléctricas [1]. Este parámetro es descrito usando la siguiente ecuación:

$$RMSE = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} (x_{original}[n] - x_{reconstruida}[n])^2} \quad (7)$$

En [1] se selecciona RMSE como el más adecuado para evaluar el proceso de compresión. Como refleja la Tabla 1 los resultados cuando se calcula el PRD con normalización (PRD\_CN), es decir no se realiza el proceso de desnormalización, los resultados son excelentes, inferiores al 10% para todos los tamaños de fragmentos. Pero estos no son correctos pues se omite un paso en el sistema de compresión de la Figura 1a. Si calculamos el PRD luego de la desnormalización (PRD\_SN) observamos que estos incrementan hasta valores superiores al 16%. Mientras que en el sistema de compresión de la Figura 1b estos valores son inferiores al 14%.

El sistema de compresión que omite la normalización (Figura 1b) obtiene valores de RMSE inferiores a los  $1,8\mu V$ , aunque estén alejados de las recomendaciones es un buen resultado. Para el sistema de compresión de la Figura 1a el valor de RMSE calculado es superior a los  $2\mu V$  para todos los tamaños de fragmentos como se refleja en la Tabla 1. Como se observa en la última fila de la Tabla 1 se utiliza una métrica llamada  $PRD > 0.2$  esta significa la cantidad de fragmentos con PRD mayor al 20% de distorsión. Es



Métricas	Normalizando			Sin normalizar		
	128	256	512	128	256	512
PRD_CN	0.0905	0.0791	0.0746			
PRD_SN	0.1640	0.1655	0.1770	0.1378	0.1317	0.1262
RMSE	0.0021	0.0022	0.0025	0.0018	0.0018	0.0018
PRD > 0.2	23631 (26,17%)	11711 (25,94%)	6383 (28,28%)	19544 (21,64 %)	9076 (20,10%)	4377 (19,39%)

Tabla 1. Resultados de los sistemas de compresión.

importante destacar que los resultados de esta métrica se expresan en porcentaje pues la cantidad de fragmentos para cada una de las longitudes no son iguales. El sistema de compresión de la Figura 1a posee valores superiores, de esta métrica (PRD > 0.2), al 25% para todas las longitudes de fragmentos. Como ilustra la Tabla 1 el sistema de compresión de la Figura 1b posee valores inferiores al 22% de esta métrica.

Aunque la diferencia de PRD y RMSE de ambos sistemas de compresión sea mínima, esta pequeña diferencia se ve reflejada en la cantidad de fragmentos con PRD menor al 20%. Como muestra la Figura 3 cuando un fragmento de señal recuperada supera el 20% de PRD, la información sufre una distorsión importante.

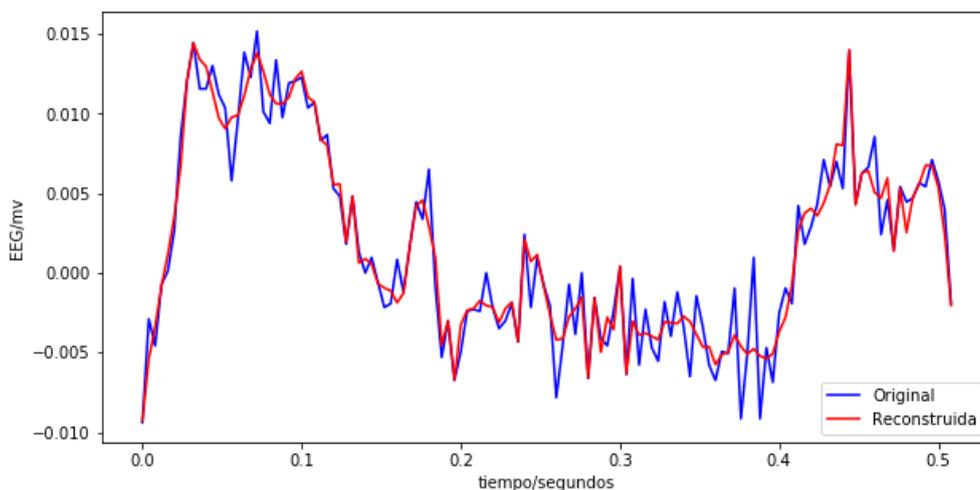


Figura 3. Señal EEG original y reconstruida.

#### 4. Conclusiones

El sistema de compresión que no emplea la normalización (Figura 1b) obtiene mejores resultados reflejados en los parámetros PRD y RMSE. El principal avance es que reduce la cantidad de fragmentos con PRD mayor al 20%. Mientras que el sistema de compresión que emplea normalización (Figura 1a) aumenta la distorsión en la señal EEG recuperada.



Esto es provocado por las propias características de los autocodificadores, pues el proceso de decodificación, realiza la reconstrucción de la señal a partir de los rasgos extraídos. Este proceso introduce distorsión debido a las aproximaciones. Además de las aproximaciones en el cálculo del proceso de desnormalización.

Aunque no todas las bases de datos brinda la posibilidad de omitir la normalización, el uso de la función de activación tangente hiperbólica ofrece una oportunidad de realizar alguna operación matemática que disminuya los errores por aproximación. Pues nos permite acotar la señal EEG en el rango de valores de -1 hasta 1. El uso de la tangente hiperbólica provoca un lento proceso de entrenamiento de la red neuronal, comparado con otras funciones de activación, pero logrando una menor distorsión en la señal recuperada.

## 5. Referencias bibliográficas

- [1] Bazán-Prieto C. Alberto, "Compresión de señales electroencefalográficas con sistemas de tasa múltiple y calidad bajo demanda," 2013.
- [2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, First E. MIT Press, 2016.
- [3] A. Galea and L. Capelo, *Applied deep learning with Python : use scikit-learn, TensorFlow, and Keras to create intelligent systems and machine learning solutions*, First E. Packt Publishing, 2018.
- [4] A. Ben Said, A. Mohamed, and T. Elfouly, "Deep learning approach for EEG compression in mHealth system," in *2017 13th International Wireless Communications and Mobile Computing Conference, IWCMC 2017*, 2017, pp. 1508–1512.
- [5] A. Z. Al-Marridi, A. Mohamed, and A. Erbad, "Convolutional Autoencoder Approach for EEG Compression and Reconstruction in m-Health Systems," in *2018 14th International Wireless Communications and Mobile Computing Conference, IWCMC 2018*, 2018, pp. 370–375.
- [6] B. Hejrati, A. Fathi, and F. Abdali-Mohammadi, "A new near-lossless EEG compression method using ANN-based reconstruction technique," *Comput. Biol. Med.*, vol. 87, pp. 87–94, 2017.



**II Convención Científica Internacional 2019**  
**Universidad Central "Marta Abreu" de Las Villas**  
**CIENCIA, TECNOLOGÍA Y SOCIEDAD. PERSPECTIVAS Y RETOS**

- [7] A. L. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet," *Circulation*, vol. 101, no. 23, 2012.
- [8] J. L. Cárdenas-Barrera, J. V Lorenzo-Ginori, and E. Rodríguez-Valdivia, "A wavelet-packets based algorithm for EEG signal compression," *Informatics Heal. Soc. Care*, vol. 29, no. 1, pp. 15–27, Mar. 2004.
- [9] A. Gulli and S. Pal, *Deep Learning with Keras*, First E. Packt Publishing, 2017.
- [10] A. Fandango, *Mastering TensorFlow 1 . x*, First E. Packt Publishing, 2018.
- [11] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 2010, pp. 807–814.
- [12] S. Raschka and V. Mirjalili, *Python Machine Learning*, Second E. Packt Publishing, 2017.
- [13] F. Romano, *Lerning Python*, First E. Packt Publishing, 2015.
- [14] B. Farnham *et al.*, *Learning TensorFlow A Guide to Building Deep Learning Systems*, First E. O'Reilly, 2017.
- [15] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," Dec. 2014.
- [16] A. Naït-Ali and C. Cavaro-Ménard, *Compression of biomedical images and signals*. ISTE, 2008.
- [17] G. Antoniol and P. Tonella, "EEG data compression techniques," *IEEE Trans. Biomed. Eng.*, vol. 44, no. 2, pp. 105–114, 1997.