

# Predicción de la cinética de movimientos imaginados a partir de EEG de época única utilizando redes neuronales convolucionales

Ramiro H. Gatti<sup>1,2</sup>[0000-0001-7367-0791], Yanina V. Atum<sup>2</sup>[0000-0002-4219-9471],  
and José A. Biurrun Manresa<sup>1,2</sup>[0000-0003-4060-9665]

<sup>1</sup> Instituto de Investigación y Desarrollo en Bioingeniería y Bioinformática (IBB), Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Oro Verde, Argentina

<sup>2</sup> Laboratorio de Ingeniería en Rehabilitación e Investigaciones Neuromusculares y Sensoriales (LIRINS), Facultad de Ingeniería, Universidad Nacional de Entre Ríos, Oro Verde, Argentina

**Abstract.** La decodificación de la cinética de movimientos imaginados a partir del electroencefalograma (EEG) permite obtener más grados de libertad en el control de dispositivos por medio de una interfaz cerebro computadora, a costa de un aumento en la complejidad del problema. Intentos recientes para discernir entre diferentes niveles de fuerza y velocidad en la imaginación de movimientos obtuvieron resultados cercanos al nivel de aleatoriedad, lo que demanda una mejora en la exactitud de la predicción. En este sentido, las redes neuronales profundas han demostrado ser capaces de resolver problemas complejos a partir de la abstracción de la señal de entrada; entre ellas, las redes neuronales convolucionales (ConvNet) además son capaces de aprender relaciones topológicas en los datos. En particular, en el presente trabajo se implementó una ConvNet para distinguir entre niveles de fuerza y velocidad durante la imaginación de una tarea de agarre, a partir de EEG de época única. Se obtuvo una exactitud mayor a 70% con un pre-procesamiento mínimo del EEG, constituyendo una mejoría sensible por sobre resultados previos a partir de los mismos datos utilizando estrategias alternativas.

**Keywords:** Movimientos cinéticos imaginados · Interfaz cerebro computadora · Redes neuronales convolucionales · Electroencefalograma.

## 1 Introducción

La decodificación de señales cerebrales relacionadas a tareas motoras permite establecer una comunicación humano-máquina por medio de una interfaz-cerebro computadora (BCI por sus siglas en inglés), sin necesidad de una salida neuromuscular [1]. De esta forma, se pueden generar comandos y establecer un canal de comunicación alternativo con el medio, conveniente en pacientes con enfermedades neuromusculares como esclerosis lateral amiotrófica, accidente cere-

brovascular o lesiones de la médula espinal; particularmente en grados avanzados donde la persona queda completamente paralizada, en un estado de enclaustramiento. Adicionalmente, aunque en principio las BCI fueron pensadas para comandar y controlar dispositivos, actualmente las áreas de investigación y aplicación se ampliaron hacia la neurociencia y la rehabilitación [2].

En una BCI, las señales cerebrales son comúnmente registradas en un electroencefalograma (EEG) y analizadas en tiempo real para generar comandos que permitan controlar un dispositivo. Si las salidas de comando están basadas en la actividad cerebral motora, en primer lugar es de interés detectar la intención de movimientos imaginados o ejecutados, contemplando que ambos tipos de movimiento muestran procesos corticales similares. Las técnicas actuales se basan principalmente en el análisis de los ritmos sensoriomotores o de los potenciales corticales lentos, ya que contienen información anticipada del comportamiento [3]. Entre los potenciales corticales lentos, los potenciales corticales relacionados a movimientos reales o imaginados fueron ampliamente estudiados para detectar la intención de movimiento [4]. En contraste, el reconocimiento de tareas cinemáticas y cinéticas a partir del EEG aún sigue siendo un reto. Gran cantidad de estudios se enfocaron en clasificar entre el movimiento de las manos (derecha-izquierda) y entre ambas manos o ambos pies [5], mientras que unos pocos se centraron en tareas relacionadas a movimientos cinemáticos y aún menos en aspectos cinéticos [6, 7], como la fuerza y velocidad durante una tarea de agarre. La posibilidad de diferenciar entre tareas motoras imaginadas con diferentes parámetros cinemáticos y cinéticos otorga al sistema mayor grado de libertad y permite un control natural, a costa de un aumento en la complejidad de la decodificación.

En este sentido, los métodos de aprendizaje por representación permiten extraer de forma automática las características óptimas para la etapa de clasificación. Entre ellos, son de particular interés los métodos basados en el aprendizaje profundo, en donde se crean múltiples niveles de representación compuestos por módulos simples y no lineales que transforman la señal de entrada en niveles cada vez más abstractos [8]. Particularmente, las redes neuronales convolucionales (ConvNets) han tenido un desarrollo notable en el campo de la visión computacional [9]. Recientemente, algunos estudios comenzaron a utilizar las ConvNets para clasificar la señal del EEG, implementando redes que permiten mejorar los resultados obtenidos hasta el momento en múltiples paradigmas y entre varios sujetos [10].

En función de lo expuesto anteriormente, el objetivo del presente estudio es evaluar si las ConvNets pueden predecir la cinética de movimientos imaginados a partir de EEG de época única. A tal fin, se analizó una base de datos previamente publicada [6] y se compararon los resultados de la clasificación obtenidos por la ConvNet contra una clasificación por chance. Para definir el nivel de exactitud por chance se permutaron las etiquetas de las clases aleatoriamente. Como índices de medición del desempeño de la red se calcularon: la exactitud, el valor predictivo positivo (VPP) y tasa de verdaderos positivos (TVP). Adicionalmente, se evaluó

el comportamiento de la red con canales intercambiados para determinar si existe un aprendizaje de la relación espacial entre canales.

## 2 Materiales y Métodos

### 2.1 Base de datos

La base de datos contiene registros de EEG de 16 voluntarios sanos. El experimento se desarrolló bajo un protocolo aprobado por el comité de ética de la region Nordjylland, respetando la declaración de Helsinki, y los sujetos firmaron un consentimiento informado antes de su partición. El EEG se registró durante la imaginación de cuatro tareas de agarre isométrico con diferentes niveles de fuerza y velocidad, donde la fuerza se expresa como porcentaje de la máxima contracción voluntaria (MCV) y para plasmar el cambio de velocidad se fijaron dos periodos de tiempo diferentes para alcanzar dicha fuerza. Las categorías o clases son las siguientes: *Slow20*, contemplando un periodo de tiempo de 3 s para alcanzar el 20% de la MCV; *Slow60*, con 3 s para alcanzar el 60% de la MCV; *Fast20*, con 0.5 s para alcanzar el 20% de la MCV y *Fast60*, con 0.5 s para alcanzar el 60% de la MCV. Se realizaron 40 repeticiones de cada clase, por lo que se dispone de 160 realizaciones por cada sujeto. Para el registro se utilizaron 20 canales del dispositivo Neuroscan NuAmp, de acuerdo con el sistema 10/10 (fig. 1) y manteniendo una impedancia menor a 5 K $\Omega$ . Todos los electrodos fueron referenciados al lóbulo derecho y puestos a tierra en el nasion. La frecuencia de muestreo fue de 500 Hz. Para más detalles referirse a [6].

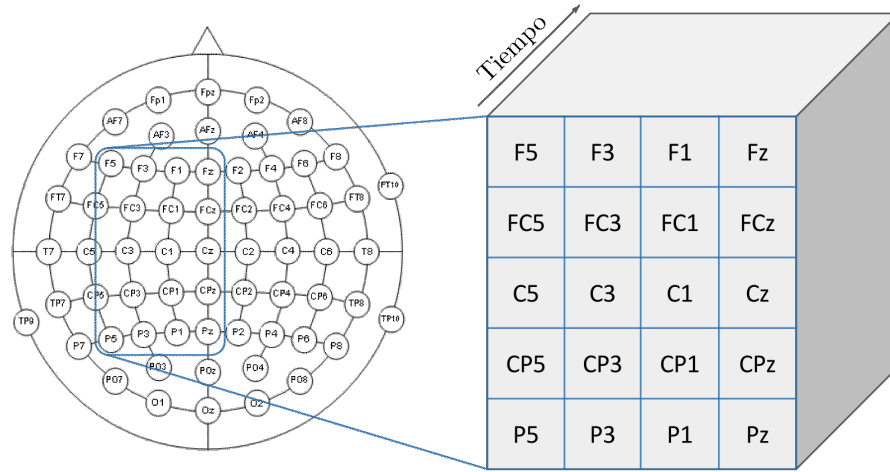
### 2.2 Pre-procesamiento

Se implementó un filtro notch para reducir el ruido de línea de 50 Hz. De cada realización se tomaron épocas de 500 ms (250 muestras), a partir de los 100 ms antes de la imaginación del movimiento (marcador), es decir de 600 a 100 ms previos al marcador. Los datos se dispusieron en arreglos de 5 x 4 x 250 (fig. 1), donde las dos primeras dimensiones corresponden a la distribución espacial de los 20 canales y la tercer dimensión a las muestras temporales. Los datos fueron divididos en 128 realizaciones (80%) para entrenamiento y validación, y 32 realizaciones (20%) para prueba. De las 128 realizaciones, 102 fueron usadas para entrenar (80%) y 18 para validación (20%).

### 2.3 Red neuronal convolucional

El modelo se basó en la red EEGnet propuesta por Lawhern et al. [10] y fue construido en Keras 2.0.5 [11] usando TensorFlow 1.2.1 [12] y entrenado en una placa de video NVIDIA Quadro K620 GPU, con CUDA 8 y cuDNN 8.

El modelo consistió en cuatro capas (Tabla 1). La entrada de la primera capa es una matriz tridimensional (3D) preprocesada para cada época, la cual se reformó (*reshape*) agregando una cuarta dimensión de valor uno (ya que cada



**Fig. 1.** Disposición de los electrodos utilizados en el sistema 10/10 y su organización espacial en la matriz ingresada a la red.

voxel de la matriz 3D contiene un solo valor y no 3 como es en las imágenes RGB). La primera capa convolucional consiste de 16 kernels de  $2 \times 2$ , que se aplican a cada muestra de tiempo, generando 16 mapas, formados por el conjunto de combinaciones lineales de cuatro canales vecinos en la dimensión espacial (filtro espacial). Los pesos del kernel fueron inicializados distribución normal y se regularizaron con una red elástica ( $L_1 + L_2$ ), con  $L_1 = .0016$  y  $L_2 = .0346$ , debido a que se esperaba que los pesos fueran pequeños y raros. No se utilizó relleno (padding) y el paso (stride) se estableció en 1 para todas las dimensiones. Se aplicó una activación lineal exponencial (ELU) después de cada capa, con  $\alpha = 1$ , seguida de una normalización (*batch normalization*) y una tasa de eliminación (*dropout*) de .34.

En la segunda capa, se tuvieron en cuenta las dimensiones espaciales y temporales, utilizando ocho núcleos (*kernels*) convolucionales de  $2 \times 2 \times 27$  para cada mapa. En este caso, el tamaño de la dimensión espacial se mantuvo constante rellenando con ceros. Luego, se aplicó un submuestreo (*max pooling 3D*) con un paso igual a 1, sin relleno, que seleccionó el valor máximo de un núcleo de  $2 \times 3 \times 4$ . Se aplicó una activación ELU, normalización y eliminación utilizando los mismos hiperparámetros que en la primera capa.

La tercera capa consiste de cuatro núcleos convolucionales ( $2 \times 2 \times 27$ ) para cada mapa, que se corresponden con las cuatro clases del problema. El submuestreo en esta capa fue  $2 \times 1 \times 4$ . Se realizó un promedio global de los 4 mapas obtenidos (*globalaverage3D*) al final de esta capa. Finalmente, las cuatro puntuaciones resultantes se transformaron en probabilidades mediante una activación softmax.

**Table 1.** Arquitectura de la red neuronal convolucional, donde  $C_x$  = canales en dirección medio-lateral,  $C_y$  = canales en dirección antero-posterior,  $T$  = muestras temporales,  $N$  = número de clases.

Capa	Tipo	Salida	Parámetros #
1	Input	$(C_y \times C_x \times T)$	0
	Reshape	$(C_y \times C_x \times T \times 1)$	0
	Conv3D (2, 2, 1) $\times$ 16	$(4 \times 3 \times T \times 16)$	80
	Batch normalization	$(4 \times 3 \times T \times 16)$	64
	ELU	$(4 \times 3 \times T \times 16)$	0
	Dropout (.25)	$(4 \times 3 \times T \times 16)$	0
2	Conv3D (2, 2, 27) $\times$ 8	$(4 \times 3 \times T \times 8)$	13832
	Batch normalization	$(4 \times 3 \times T \times 8)$	32
	ELU	$(2 \times 1 \times 62 \times 8)$	0
	Max pooling 3D (2, 3, 4)	$(2 \times 1 \times 62 \times 8)$	0
	Dropout (.25)	$(2 \times 1 \times 62 \times 8)$	0
3	Conv3D (2, 1, 3) $\times$ N	$(2 \times 1 \times 62 \times N)$	196
	Batch normalization	$(4 \times 3 \times 62 \times N)$	16
	ELU	$(1 \times 1 \times 15 \times N)$	0
	Max pooling 3D (2, 1, 4)	$(1 \times 1 \times 15 \times N)$	0
	Dropout (.25)	$(1 \times 1 \times 15 \times N)$	0
	Global Average Pooling 3D	$(N)$	0
4	Activation (Softmax)	$(N)$	0

## 2.4 Ajuste del modelo

El proceso de aprendizaje consistió en un número fijo de iteraciones durante el entrenamiento, utilizando un mini-conjunto (*mini-batch*) de 10 épocas seleccionadas aleatoriamente y se utilizó la optimización de Adam,  $lr = 0.0238$ ,  $\beta_1 = \beta_2 = .9$ ,  $decay = 0.023$ . La cantidad de iteraciones de entrenamiento se estableció en 137. Se utilizó como métrica la exactitud obtenida del conjunto de validación y se guardó el modelo a medida que la exactitud mejoraba. Para evitar un sobreajuste del modelo, solo se mantuvo el modelo con mayor exactitud de validación. Este procedimiento de regularización es similar al obtenido con una detención temprana (*early stopping*), con la desventaja de que se realiza el proceso de entrenamiento con una cantidad de iteraciones prefijada. Sin embargo, este método fue seleccionado porque la ConvNet tiene un número relativamente pequeño de ejemplos de entrenamiento, y la detención temprana no siempre dio como resultado la mejor precisión posible. Finalmente, se realizó un procedimiento de validación cruzada de 5 grupos para verificar la generalización.

El modelo se determinó por medio de una búsqueda aleatoria de 300 iteraciones, variando algunos parámetros e hiperparámetros. El número de capas varió entre 2 y 5 para verificar la capacidad del modelo, y como no es común utilizar batch normalization con dropout se probó el desempeño de la red sin dropout y con diferentes tasas de eliminación entre 0 y 1. También se probaron

los optimizadores de Adam y gradiente estocástico descendente (SGD de sus siglas en inglés), así como la inicialización de pesos normal, uniforme y Glorot uniforme. El número de iteraciones en el entrenamiento varió entre 20 y 150, la cantidad de épocas utilizadas en el mini-batch de cada iteración varió entre 8 y 150. Se varió la tasa de aprendizaje entre 0.0001 y 0.01, con un decaimiento entre 0 y 0.01. Los regularizadores L1 y L2 variaron entre 0.001 y 0.1.

## 2.5 Verificación de la red

Para asegurar que la red realmente estaba aprendiendo a partir de los datos se realizó un desordenado aleatorio de las etiquetas de cada clase, de esa forma se definió el nivel de exactitud por chance. También se cambió el orden espacial de los canales para cada ejemplo, con el objetivo de verificar la existencia de una relación espacial aprendida por la red. Por otro lado, como se disponía de pocos ejemplos de cada sujeto para el entrenamiento, se realizó una curva de aprendizaje en función de la cantidad de realizaciones para verificar si existía una mejora en la exactitud incrementando la cantidad de datos y si se alcanzaba un nivel de estabilidad.

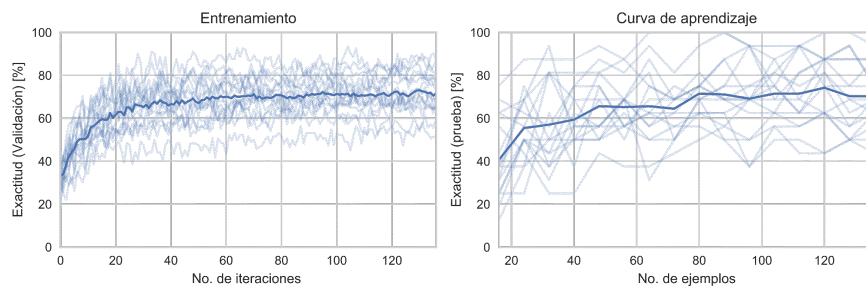
## 3 Estadísticas

La exactitud, el VPP y la TVP para cada sujeto se determinaron como la media de los valores obtenidos durante la validación cruzada de 5 grupos. Se realizó una prueba t de muestras pareadas para determinar si existe una diferencia entre la exactitud del modelo entrenado con las etiquetas correctas (EC) y desordenadas (ED). Un análisis de la varianza para medidas repetidas fue realizado para determinar la diferencia en los índices VPP y TVP, en función del modelo (EC y ED), velocidad (imaginación de movimiento rápido y lento) y fuerza (20% y 60% de MCV) como factores. Los efectos principales e interacciones de hasta dos vías fueron analizados para determinar si existe una diferencia en los índices de desempeño. La prueba de Shapiro-Wilk fue realizada para verificar la normalidad de los datos. Las interacciones fueron analizadas utilizando la prueba post hoc de Tukey cuando fue requerido. Los índices fueron reportados como *media ± desvío estándar*. Los valores de p menores a 0.05 fueron considerados como estadísticamente significativos.

## 4 Resultados

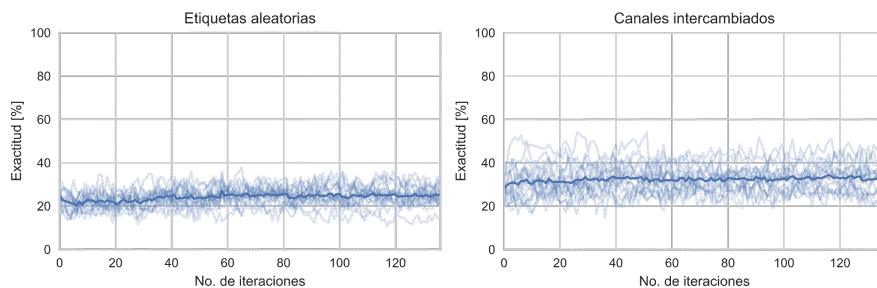
A partir de la curva de validación durante el entrenamiento (fig. 2) se observó que la red llega a un punto de estabilidad alrededor de 100 iteraciones. También se puede observar que se llegó a una meseta en la curva de aprendizaje a partir de 120 ejemplos, por lo tanto es probable que no exista una mejora importante si entrenamos con más datos.

Como se muestra en la fig. 3, para etiquetas permutadas aleatoriamente la red no pudo aprender a distinguir entre las clases, obteniendo valores de exactitud



**Fig. 2.** *Izquierda:* evolución de la exactitud durante la validación para todos los sujetos en función del número de iteraciones. *Derecha:* Exactitud en la clasificación en los datos de prueba en función de la cantidad de ejemplos para cada voluntario. En ambas gráficas la línea oscura representa la exactitud media para todos los sujetos, mientras que las líneas claras representan la media de cada sujeto, derivada de la validación cruzada de 5 grupos.

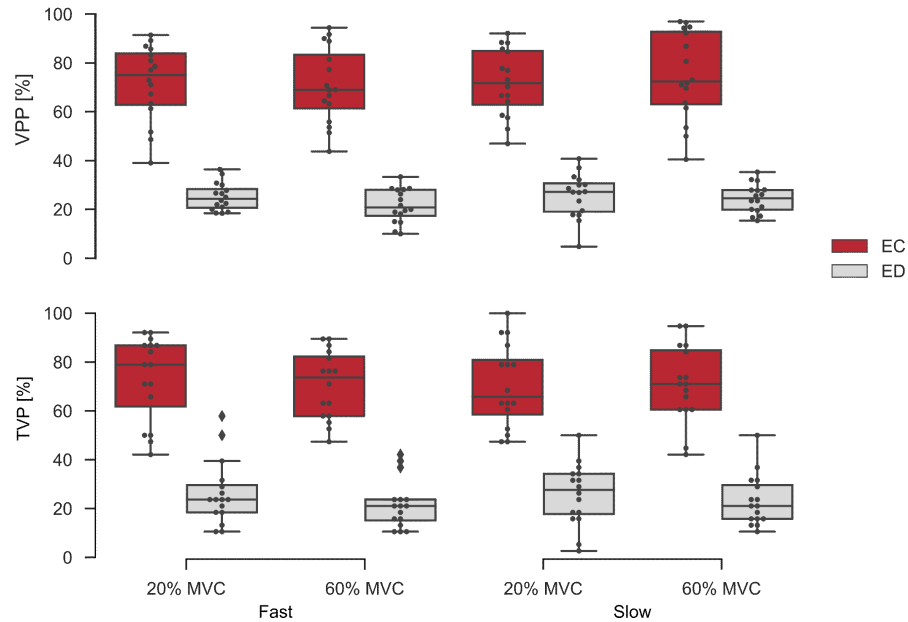
cercanos al valor teórico de chance para 4 clases (25 %). También podemos verificar que sin una relación espacial entre canales la red no puede aprender a distinguir entre las diferentes clases, con un valor cercano a la chance.



**Fig. 3.** Exactitud de la validación en los datos de prueba para todos los voluntarios con las etiquetas permutadas de forma aleatoria (*izquierda*) y removiendo la relación espacial entre los canales (*derecha*).

La exactitud en la clasificación de la ConvNet ( $71.3 \pm 12.0\%$ ) fue significativamente mayor que la de la clasificación cuando las etiquetas fueron desordenadas de forma aleatoria ( $24.3 \pm 4.2\%$ ,  $t_{15} = 15.04$ ,  $p < 0.001$ ). El VPP y la TVP para todos los voluntarios se muestra en la fig. 4, donde puede verse que el VPP de la ConvNet fue significativamente mayor con las etiquetas correctas ( $72.3 \pm 15.3\%$ ) que con las etiquetas desordenadas ( $24.2 \pm 6.9\%$ ,  $F_{1,15} = 288.7$ ,  $p < 0.001$ ), no encontrándose interacciones significativas. Para la TVP los resultados de la ConvNet con las etiquetas correctas ( $71.3 \pm 15.6\%$ ) también fueron significativamente mayores que con las etiquetas aleatorias ( $24.2 \pm 11.4\%$ ,  $F_{1,15} = 242.9$ ,  $p < 0.001$ ).

Además, se encontró interacciones significativas entre el modelo y los factores de velocidad ( $F_{1,15} = 7.0, p = 0.018$ ) y fuerza ( $F_{1,15} = 4.7, p = 0.046$ ). Luego de un análisis post-hoc se encontró que la TVP es ligeramente más alta para los movimientos lentos y para para fuerzas al 60% de la MVC para la clasificación con ED.



**Fig. 4.** Resultado de la clasificación sobre los datos de prueba. Se muestra el valor predictivo positivo (VPP) y la tasa de verdaderos positivos (TVP) para todos los voluntarios ( $n = 16$ ). Las cajas representan la mediana y los percentiles 25 y 75, los bigotes los percentiles 5 y 95, los diamantes los valores fuera del rango de los percentiles [5 - 95], mientras que los puntos representan el valor promedio de cada sujeto a partir de los 5 grupos de la validación cruzada.

## 5 Discusión

El objetivo del estudio consistió en evaluar si la ConvNet propuesta es capaz de aprender a clasificar la cinética de movimientos imaginados a partir del EEG de época única. Los resultados mostraron que, mientras que la clasificación con ED está dentro del rango de exactitud por chance para un problema de clasificación de cuatro clases (alrededor del 25%), los índices de clasificación obtenidos con la ConvNet son significativamente más altos, en el orden del 70%. A modo de



referencia, trabajos anteriores donde se clasificó la tarea de agarre con la misma base de datos, obtuvieron resultados en el rango de 37-48% [6], cercanos al nivel de chance y similares a los obtenidos cuando se intercambiaron los canales espacialmente. Aunque se encontraron interacciones en la TVP para la velocidad y la fuerza en las ED, las diferencias son mínimas y debidas a la chance.

Adicionalmente, a partir de las estrategias de entrenamiento propuestas se pudo verificar que la red aprende la relación espacial existente entre los canales, ya que al permutarlos aleatoriamente la red no puede clasificar entre las diferentes clases. En el estudio se mantuvo la distribución espacial de los electrodos en la matriz de datos de entrada a la red, a diferencia de trabajos previos publicados [13–16]. Además, en la curva de aprendizaje se observa un incremento en la exactitud del modelo al aumentar la cantidad de ejemplos que entran a la red hasta llegar a una aparente estabilización con una exactitud promedio mayor al 70% (fig. 2).

Como un próximo paso en la presente línea de investigación, se pretende evaluar el rendimiento de la red en otras bases de datos, con problemas de mayor y menor complejidad, en función de la cantidad de clases a distinguir y a partir de movimientos reales o imaginados entre diferentes partes del cuerpo o diferentes tipos de movimientos, con la salvedad de que al aumentar la complejidad del modelo, se espera un mayor número de hiperparámetros a entrenar, por lo que es necesario una mayor cantidad de datos.

## 6 Conclusión

En conclusión, la red ConvNet propuesta demostró ser capaz de predecir con una exactitud mayor a 70% la cinética de la imaginación de tareas motoras a partir del EEG de época única, sin necesidad de realizar un pre-procesamiento complejo y aprovechando la relación espacial existente entre los canales registrados.

## References

1. Wolpaw, J.R., Birbaumer, N., McFarland, D.J., Pfurtscheller, G., Vaughan, T.M.: Braincomputer interfaces for communication and control. *Clinical Neurophysiology* 113(6), 767–791 (2002)
2. Machado, S., Araújo, F., Paes, F., Velasques, B., Cunha, M., Budde, H., Basile, L.F., Anghinah, R., Arias-Carrión, O., Cagy, M., Piedade, R., de Graaf, T.A., Sack, A.T., Ribeiro, P.: EEG-based brain-computer interfaces: an overview of basic concepts and clinical applications in neurorehabilitation. *Reviews in the neurosciences* 21(6), 451–68 (2010)
3. Brunia, C.: Neural aspects of anticipatory behavior. *Acta Psychologica* 101(2-3), 213–242 (1999)
4. Shakeel, A., Navid, M.S., Anwar, M.N., Mazhar, S., Jochumsen, M., Niazi, I.K.: A review of techniques for detection of movement intention using movement-related cortical potentials. *Computational and Mathematical Methods in Medicine* 2015, 346217 (2015)

10 R. H. Gatti et al.

5. Naeem, M., Brunner, C., Leeb, R., Graimann, B., Pfurtscheller, G.: Seperability of four-class motor imagery data using independent components analysis. *Journal of Neural Engineering* 3(3), 208–216 (2006)
6. Jochumsen, M., Khan Niazi, I., Taylor, D., Farina, D., Dremstrup, K.: Detecting and classifying movement-related cortical potentials associated with hand movements in healthy subjects and stroke patients from single-electrode, single-trial EEG. *Journal of Neural Engineering* 12(5), 056013 (2015)
7. Jochumsen, M., Rovsing, C., Rovsing, H., Niazi, I.K., Dremstrup, K., Kamavuako, E.N.: Classification of Hand Grasp Kinetics and Types Using Movement-Related Cortical Potentials and EEG Rhythms. *Computational Intelligence and Neuroscience* 2017, 1–8 (2017)
8. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* 521(7553), 436–444 (5 2015)
9. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet Classification with Deep Convolutional Neural Networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) *Advances in Neural Information Processing Systems* 25, pp. 1097–1105 (2012)
10. Lawhern, V.J., Solon, A.J., Waytowich, N.R., Gordon, S.M., Hung, C.P., Lance, B.J.: Eegnet: A compact convolutional network for eeg-based brain-computer interfaces (2016), arXiv: 1611.08024 [cs.LG]
11. Chollet, F., others: Keras. Github repository <https://github.com/fchollet/keras> (2015)
12. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: A system for large-scale machine learning (2015), software available from tensorflow.org
13. Tang, Z., Li, C., Sun, S.: Single-trial EEG classification of motor imagery using deep convolutional neural networks. *Optik - International Journal for Light and Electron Optics* 130, 11–18 (2017)
14. Schirrneister, R.T., Springenberg, J.T., Fiederer, L.D.J., Glasstetter, M., Eggenesperger, K., Tangermann, M., Hutter, F., Burgard, W., Ball, T.: Deep learning with convolutional neural networks for EEG decoding and visualization. *Human Brain Mapping* 38(11), 5391–5420 (2017)
15. Soare, C.v.: Brain Computer Interface using Machine Learning. In: Iftene, A., Vanderdonckt, J. (eds.) *13th International Conference on Human Computer Interaction, RoCHI*. pp. 65–68 (2016)
16. Lu, Y., Jiang, H., Liu, W.: Classification of EEG Signal by STFT-CNN Framework: Identification of Right-/left-hand Motor Imagination in BCI Systems. In: *The 7th International Conference on Computer Engineering and Networks* (2017)