



Clasificación de actividades humanas aplicando Inteligencia Computacional Human activity classification applying Computational Intelligence

López-Reynaga Brandon Alexis

Instituto Politécnico Nacional

Escuela Superior de Ingeniería Mecánica y Eléctrica, Zacatenco, México

Correo: blopezr1507@alumno.ipn.mx

<https://orcid.org/0009-0007-2091-6496>

Acevedo-Mosqueda María Elena

Instituto Politécnico Nacional

Escuela Superior de Ingeniería Mecánica y Eléctrica, Zacatenco, México

Correo: eacevedo@ipn.mx

<https://orcid.org/0000-0001-8179-0398>

Acevedo-Mosqueda Marco Antonio

Instituto Politécnico Nacional

Escuela Superior de Ingeniería Mecánica y Eléctrica, Zacatenco, México

Correo: macevedo@ipn.mx

<https://orcid.org/0000-0002-3535-1164>

Gómez-Coronel Sandra Luz

Instituto Politécnico Nacional

Unidad Profesional Interdisciplinaria

en Ingeniería y Tecnología Avanzadas, México

Correo: sgomez@ipn.mx

<https://orcid.org/0000-0003-4383-9239>

Resumen

La actividad diaria es de gran importancia para llevar una vida saludable. El monitoreo de esa actividad permite a una persona saber si es necesario incrementar sus movimientos diarios para, al menos, cubrir la actividad mínima diaria para vivir con salud. El monitoreo de la actividad humana tiene varias aplicaciones, en este trabajo el objetivo es la salud. El primer paso para lograr dicho objetivo es identificar los movimientos que realiza una persona. El conjunto de datos utilizado se obtuvo del repositorio de datos de UCI. La propuesta consiste en aplicar tres algoritmos de clasificación: *K-NN*, Regresión Logística y Redes Neuronales Convolucionales para identificar seis movimientos que son: Caminar, subir escaleras, bajar escaleras, tomar asiento, ponerse de pie y acostarse. El método de validación empleado fue Hold-Out (80-20). Los mejores resultados se obtuvieron con las Redes Neuronales Convolucionales y se usaron las métricas: Exactitud: 99.03 %, Precisión: 99.08 % y F1-score: 99.12 %. La aplicación de estos métodos logró mejorar el porcentaje de exactitud en la clasificación de actividades físicas.

Descriptores: Inteligencia computacional, redes neuronales, aprendizaje profundo, clasificación, actividad humana.

Abstract

Daily activity is of great importance to lead a healthy life. The monitoring of this activity allows a person to know if it is necessary to increase their daily movements to, at least, cover the minimum daily activity to live in health. The monitoring of human activity has several applications but in this work the objective is health. The first step to achieve the objective is to be able to identify the movements that a person makes. The data set used was obtained from the UCI data repository. Our proposal is to apply three classification algorithms: *K-NN*, Linear Regression and Convolutional Neural Networks to identify six movements that are: walking, going up stairs, going down stairs, sitting down, standing up and lying down. The validation method used was Hold-Out (80-20). The best results were obtained with Convolutional Neural Networks and were: Accuracy: 99.03 %, Precision: 99.08 % and F1-score: 99.12 %. The application of these methods managed to improve the percentage of accuracy in the classification of physical activities.

Keywords: Computational intelligence, neural networks, deep learning, classification, human activity.

INTRODUCCIÓN

El reconocimiento de la actividad humana es una nueva tecnología que permite reconocer actividades o gestos realizados por personas de manera automática, esto ha despertado un gran interés en el campo de las ciencias computacionales, debido a la amplia gama de aplicaciones que pueden presentarse, tales como: Análisis de comportamientos, monitoreo de salud, entrenamiento, juegos interactivos, reconocimiento de gestos, videovigilancia.

Esta investigación se realizó durante el período de la pandemia debida al SARS-COV2 y estuvo enfocada a la detección de las actividades que realizaban las personas dentro de su casa, de manera que se pudiera llevar un registro de ellas y emitir una recomendación de los movimientos que se debían de incrementar o decrementar para poder llevar, en la manera de lo posible, una mejor calidad de vida dentro del aislamiento. Aun cuando se ha regresado al desarrollo de las actividades cotidianas, sigue siendo importante llevar un registro de los movimientos que se realizan durante el día, con el mismo propósito del monitoreo de la salud, como ya se indicó. El nivel suficiente de actividad es la práctica físico-deportiva de al menos tres días por semana, acumulando a la semana un tiempo mínimo de 75 minutos y con una intensidad vigorosa, o un acumulado de 150 minutos con intensidad moderada, recomendado por la Organización Mundial de la Salud.

En la Figura 1, se muestra la actividad física registrada antes y durante la pandemia (Flores *et al.*, 2021).

	Antes del confinamiento	Durante del confinamiento
	Mediana (IIC)	Mediana (IIC)
Actividad física vigorosa		
Días de actividad	5 (3-5)	3 (2-5)
Tiempo de actividad (min)	60 (40-120)	60 (30-90)
Actividad física moderada		
Días de actividad	5 (3-6)	4 (2-5)
Tiempo de actividad (min)	90 (60-120)	60 (40-90)
Caminar		
Días de actividad	5 (3-7)	4 (2-6)
Tiempo de actividad (min)	40 (20-60)	30 (15-50)
Actividades sedentarias		
Tiempo de actividad (min)	6 (3-6)	5 (3-8)

Figura 1. Cifras de la actividad física que realizaban las personas en México antes y durante el confinamiento por COVID-19

De la Figura 1, IIC es el Intervalo Intercuartílico. En esta misma figura se puede observar que aun cuando las

personas disminuyeron los días de actividad, respecto a la actividad física vigorosa, en promedio realizaban el mismo tiempo de ejercicio. En el caso de la actividad física moderada, el tiempo dedicado a esta actividad disminuyó 30 minutos durante el confinamiento. También disminuyó el tiempo dedicado a caminar, el decremento promedio fue de 10 minutos.

Ahora se presentan las estadísticas de la actividad física en México (INEGI, 2021), tomando como referencia el año 2021. En ese año, un 39.6 % de la población mayor de 18 años de edad afirmó que era físicamente activa. El 73.9 % de la población establece que el motivo de hacer ejercicio físico o deporte es la salud. En cuanto a la población activa físicamente que alcanzó el nivel de suficiencia para obtener beneficios a la salud según las recomendaciones de la OMS, más de la mitad de esa población (53.5 %) realizó la actividad cumpliendo los criterios antes mencionados.

Durante el 2020, año de la pandemia, esa cifra descendió 3.3 puntos porcentuales. En este mismo año, las personas dejaron de hacer deporte en lugares o instalaciones privados.

Se puede observar que la actividad física de las personas sí tuvo un decremento durante la pandemia, y que durante el año 2021 regresó a las cifras promedio. También se puede concluir que la salud es la mayor razón para ejercitarse. Debido a esto, en este trabajo se propone un clasificador de actividades físicas utilizando herramientas de Inteligencia Artificial, como primer paso, para lograr un sistema completo que permita sugerirle al usuario el incremento del ejercicio físico para mantener saludable su cuerpo.

TRABAJOS RELACIONADOS

En la Tabla 1, se muestran los trabajos relacionados con esta propuesta. Se presentan el autor y el año, así como el conjunto de datos utilizados y los algoritmos aplicados, finalmente, se muestran los resultados de las métricas empleadas.

Murad & Pyun (2017) aplican sus algoritmos a cinco conjuntos de datos: UCI-HAD, USC-87 HAD, Opportunity, Daphnet FOG y Skoda. La métrica fue la exactitud, el método de 88 validación fue Hold-Out (80-20) y el resultado para el conjunto UCI-HAD fue de 96.7 % de exactitud. El conjunto de datos UCI-HAD es el mismo conjunto que se utiliza en este 90 trabajo. En el caso de Sansano *et al.* (2020) utilizaron diez conjuntos de 91 datos de actividades humanas, entre ellos está UCI-HAR. Las métricas aplicadas fueron 92 exactitud y F1-score, el método de validación fue Hold-Out (80-20). Los mejores resultados 93 se obtuvieron con una Red Neuronal Convolutiva y fueron: 98.74 % y 98.59 % de

94 exactitud y F1-score, respectivamente. Hossain *et al.* (2020) 95 dividen el conjunto de datos en actividades estáticas y dinámicas. Para la clasificación de 96 estos dos tipos de actividades aplican el algoritmo de Árboles de Decisión. Después, para 97 identificar las actividades estáticas: Sentarse, pararse y descansar, usan el método de 98 Máquina de Soporte Vectorial (SVM) y, finalmente, utilizan una Red Neuronal 99 Convolutiva 1D para reconocer actividades dinámicas: Caminar, subir escaleras, bajar 100 escaleras. Obtuvieron 97.66 % de exactitud con el método de validación Hold-Out (80-20), el conjunto de datos fue UCI-HAR. Un trabajo más reciente es el de Satya *et al.* (2023), donde se reporta una exactitud de 96.7 % aplicando un clasificador lineal de Soporte Vectorial.

Los trabajos que no utilizaron el mismo conjunto de datos se describen a continuación. 103 Babiker *et al.*

(2017) obtienen las actividades mediante una 104 cámara de vigilancia. Las actividades detectadas son: Caminar, descansar, saludar con la 105 mano agitándola, boxear, sentarse. Se aplica un Perceptrón Multicapa y la exactitud mayor 106 obtenida fue de 100 % para boxear y sentarse. La menor exactitud fue de 81.3 % y fue para 107 la actividad de descansar. Sun *et al.* (2018) también detectan actividades 108 físicas como: Abrir lavavajillas, abrir cajón 1, abrir cajón 2, abrir cajón 3, abrir puerta 1, 109 abrir puerta 2, tomar de una taza, apretar el interruptor, actividad nula. La clasificación 110 la realizan mediante una combinación de Redes Neuronales Convolutivas, unidades 111 recurrentes de memoria a corto plazo y máquinas de aprendizaje extremas. La actividad 112 nula fue eliminada porque ocupaba 70 % del conjunto de datos y no aportaba 113 información. El promedio obtenido de F1-score fue de 90.6 %.

Tabla 1. Trabajos relacionados con esta investigación

Autor-Año	Características	Resultados (%)
Murad & Pyun (2017)	En este trabajo se aplican Redes neuronales recurrentes profundas con memorias de corto plazo, se utilizan arquitecturas unidireccionales, bidireccionales y en cascada. Las pruebas se realizaron con 5 conjuntos de datos diferentes: UCI-HAD, USC-HAD, Opportunity, Daphnet FOG y Skoda. Cabe mencionar que el primer conjunto de datos es el mismo que el utilizado en nuestro trabajo. El método de validación fue Hold-Out 80-20	El porcentaje de exactitud para los cinco conjuntos de datos fueron: UCI-HAD = 96.7, USC-HAD = 97.8, Opportunity = 92.5, Daphnet FOG = 94.1 y Skoda = 92.6
Babiker <i>et al.</i> (2017)	Utilizan un Perceptrón multicapa para clasificar las actividades humanas aplicado a video vigilancia. Trabajan con imágenes extraídas de una cámara de vigilancia. Las actividades detectadas son: Caminar, descansar, saludar con la mano agitándola, boxear, sentarse. Se obtuvieron 5000 muestras y se utilizó 70 % para entrenar, 15 % para probar y 15 % para validar	La métrica utilizada fue la exactitud. En la etapa de Validación se obtuvieron los siguientes porcentajes: Caminar: 99.3, Descansar: 81.3, Saludar: 97.1, Boxear: 100, Sentarse: 100. El promedio de la exactitud de todas las actividades fue de 94.7
Sun <i>et al.</i> (2018)	Aplican una combinación de Redes Neuronales Convolutivas, unidades recurrentes de memoria a corto plazo y máquinas de aprendizaje extremas. Utilizan el conjunto de datos de reconocimiento de actividad OPPORTUNITY. Este conjunto contiene datos numéricos. La métrica utilizada es F_1 . Las actividades a reconocer son: Abrir lavavajillas, abrir cajón 1, abrir cajón 2, abrir cajón 3, abrir puerta 1, abrir puerta 2, tomar de una taza, apretar el interruptor, actividad nula	Los autores decidieron ignorar la actividad nula, porque desbalanceaba el conjunto de datos, ya que esta actividad ocupaba 70 % de conjunto completo. El promedio de la métrica F_1 fue de 90.6
Sansano <i>et al.</i> (2020)	Se realiza una comparación de resultados entre varios algoritmos para el reconocimiento de actividades humanas. Los algoritmos son: Redes neuronales convolutivas (CNN), redes de memorias de largo plazo (LSTM), redes de memorias de largo plazo bidireccionales (biLSTM), redes de unidades recurrentes cerradas (GRU) y redes de creencias profundas DBN. Utilizan 10 conjuntos de datos de actividades humanas, entre ellos está Opportunity y UCI-HAR. Las métricas que aplican son F_1 -score y exactitud. Método de Validación: Hold.Out (80-20)	Para UCI-HAR Exactitud CNN= 98.74, LSTM=96.41, biLSTM=97.66, GRU=98.21, DBN=93.49. F_1 -score CNN= 98.59, LSTM=96.04, biLSTM=97.56, GRU=97.98, DBN=93.58

continúa ...

Tabla 1. Trabajos relacionados con esta investigación ... continuación

Autor-Año	Características	Resultados (%)
Randhawa <i>et al.</i> (2020)	Se detectan actividades violentas de ataque y se comparan con las actividades normales. Para la detección se utiliza una chamarra con sensores. El algoritmo de clasificación es Support Vector Machine. Se detectan seis clases de actividades: Estático/estacionario, caminar, caminar a paso ligero, salto-voltar-girar, bailar tango, ataque violento. La métrica utilizada es exactitud. Se aplica validación Hold-Out 70-30	Exactitud SVM = 98.8
Hossain <i>et al.</i> (2020)	Se utiliza una Máquina de Soporte Vectorial (SVM) para identificar actividades estáticas: Sentarse, pararse y descansar, así como una Red Neuronal Convolutiva 1D para reconocer actividades dinámicas: Caminar, subir escaleras, bajar escaleras. Antes de la aplicación de ambos algoritmos, se aplican Árboles de Decisión para identificar si la actividad es estática o dinámica. Algoritmo de validación: Hold-Out 80-20. La métrica utilizada es la exactitud y el conjunto de datos es UCI-HAR.	Exactitud 97.66
Ibrahim <i>et al.</i> (2022)	Utilizan pre-procesamiento, extracción de características, selección de características, fusión de características y técnicas de Machine Learning para la clasificación. Los conjuntos de datos son Weizmann y KTH, los cuales contienen imágenes de actividades humanas	Exactitud 99.5 % para el conjunto de datos de Weizmann 99.9 % para el conjunto de datos KTH
Satya <i>et al.</i> (2023)	Para el análisis de los datos se utiliza t-distributed Stochastic Neighborhood Embedding y para la clasificación se aplica Regresión Logística, SVC Lineal, Kernel de SVM, Árboles de decisión. Las métricas utilizadas son exactitud, matriz de confusión, precisión y recuperación. El conjunto de datos es el mismo que se utiliza en este trabajo.	Exactitud Clasificador Lineal de Soporte Vectorial 96.7
Sri <i>et al.</i> (2023)	Se obtienen las características de los movimientos de los cambios de la distancia articular aplicando el Euclidiano a los fotogramas en cada segmento de la actividad. Estos datos son la entrada a una Red Neuronal Convolutiva que se utiliza como clasificador. El conjunto de datos se obtuvo de la Universidad de Florencia y consiste en videos capturados con una cámara Kinect, en donde se pueden distinguir nueve actividades como son saludar, beber de una botella, contestar el teléfono, aplaudir, amarrar las agujetas, sentarse, levantarse, leer, mirar y reverencia	Exactitud 94.08
Hassan <i>et al.</i> (2023)	Los datos de los movimientos se simulan con un captador de movimientos. Después, estos datos son analizados y se obtienen los espectrogramas interferométricos y variables Doppler en el tiempo, obtenidos a partir del análisis tiempo-frecuencia de un único receptor Doppler y los datos de salida interferométricos, respectivamente, son alimentados a una red neuronal convolutiva profunda (DCNN) para la extracción de características y el proceso de entrenamiento/prueba. Las actividades son caminar, correr, saltar, dar puñetazos, agacharse, preparar y sentarse/pararse	Exactitud 98 para movimientos simulados 90 para movimientos reales
Praneeth <i>et al.</i> (2023)	Se recolectaron datos propios utilizando un acelerómetro y un giroscopio integrados para recopilar la aceleración lineal de 3 ejes y la velocidad angular de 3 ejes a una velocidad constante de 50 Hz. Los algoritmos de clasificación utilizados son K-NN, SVM, Árboles de decisión y Bosques aleatorios. Las actividades son caminar, trotar, subir escaleras, bajar escaleras, correr, dormir	Exactitud K-NN: 96.78 SVM: 94.19 Árboles de decisión: 93.78 Bosques aleatorios: 97.68

DESARROLLO

ALGORITMOS DE CLASIFICACIÓN

En esta sección se describen los algoritmos que se utilizaron para la clasificación de las actividades humanas. Se aplicaron tres algoritmos que son: *K* vecinos más cercanos (*KNN*, *K Nearest Neighbor*), Regresión Logística y Redes Neuronales Convolucionales.

K VECINOS MÁS CERCANOS

Este algoritmo pertenece al grupo de los algoritmos de clasificación de aprendizaje automático. Para llevar a cabo su implementación es necesario aplicar los siguientes pasos:

1. Seleccionar el número de los *K* vecinos, es decir, se elige el número de vecinos que permitan una mejor clasificación.
2. Seleccionar a los *K* vecinos más cercanos con base en la distancia euclidiana respecto al nuevo elemento. La distancia Euclidiana entre los puntos $X = (x_1, x_2, \dots, x_n)$ y $Y = (y_1, y_2, \dots, y_n)$ se muestra en la ecuación (1) (Wazery *et al.*, 2021).

$$d(X, Y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2} \quad (1)$$

Contar el número de elementos que pertenece a cada categoría, esto se ilustra en la Figura 2.

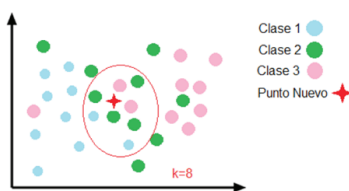


Figura 2. Conteo de los vecinos más cercanos al nuevo punto

3. Asignar el nuevo elemento a la categoría donde se contaron más vecinos (Figura 3).

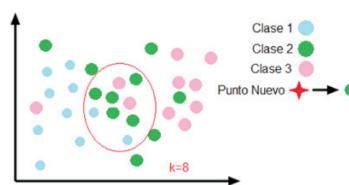


Figura 3. Asignación del nuevo elemento a la clase con más vecinos

REGRESIÓN LOGÍSTICA

Este modelo probabilístico se enfoca más en los modelos de clasificación binaria (Raschka & Mirjalili, 2019). Para la explicación de este modelo, primeramente se presenta la razón de probabilidades (*RP*), es decir, la probabilidad de que ocurra un evento, lo cual se expresa en la ecuación (2).

$$RP = \frac{p}{1-p} \quad (2)$$

Donde *p* es la probabilidad de un evento positivo, esto solo implica que el evento pertenece a una clase, por ejemplo $y = 1$, que puede equivaler a la clase de tener una enfermedad. Ahora, se define la función *logit* que es el logaritmo de la razón de probabilidades, como se muestra en la ecuación (3).

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right) \quad (3)$$

Esta función toma como entrada valores en el rango de 0 a 1 y los transforma en valores de todo el rango de números reales. Esto permite relacionar características de un conjunto de datos con logaritmos de la razón de probabilidades, definido en la ecuación (4).

$$\begin{aligned} \text{logit}(p(y = 1 | x)) &= w_0x_0 + w_1x_1 + \dots + w_nx_n \\ &= \sum_{i=1}^n w_i x_i = \mathbf{w}^T \mathbf{x} \end{aligned} \quad (4)$$

Donde $p(y = 1 | x)$ es la probabilidad condicional de que una instancia del conjunto de datos con las características \mathbf{x} , pertenezca a la clase 1; \mathbf{w} son los pesos de las características.

Ahora, para realmente predecir la clase concreta de una determinada instancia, se aplica la inversa de la función *logit* denominada función sigmoide logística o simplemente función sigmoide, la cual se muestra en la ecuación (5).

$$\phi(z) = \frac{1}{1 + e^{-z}} \quad (5)$$

Donde $z = \mathbf{w}^T \mathbf{x}$.

La función sigmoide se muestra en la Figura 4.

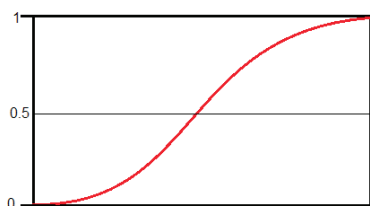


Figura 4. Función Sigmoide para clasificación de valores binarios, arriba de 0.5 se convierte en 1 y por debajo de 0.5 se convierte en 0

REDES NEURONALES CONVOLUCIONALES

Este tipo de redes pertenecen a la clase de redes neuronales profundas. La arquitectura general de una Red Neuronal Convolutiva (*Convolutional Neural Network*, CNN), se muestra en la Figura 5 (Moo Carme *et al.*, 2020).

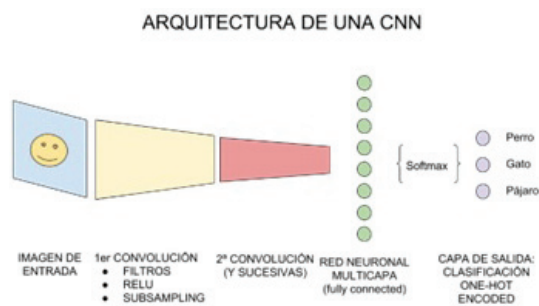


Figura 5. Arquitectura general de una red neuronal convolucional

Como se puede observar en la Figura 5, los principales componentes de una red neuronal convolucional son: La imagen de entrada, la capa de convolución, capa de submuestreo o *pooling* y el cambio de tamaño o *flattening*. La imagen de entrada puede ser de cualquier tipo y se convierte en una imagen binaria con valores ceros y unos. En la capa de convolución, se realiza la convolución entre la imagen de entrada y una máscara que funge como un selector de características. Después de la convolución se aplica la función ReLU (*Rectifier Linear Unit*), que es una de las más utilizadas. La capa de submuestreo permite ignorar los datos menos importantes y conservar las características de la imagen, en este paso, la imagen decreta su tamaño. La convolución y el submuestreo se aplican de manera sucesiva las veces que establezca el diseñador. Una vez que termina este proceso, la imagen resultante se convierte en un vector que es la entrada a un Perceptrón Multicapa que realiza la tarea de clasificación, al final se aplica la función de activación Softmax para obtener una clase.

CONJUNTO DE DATOS

Para el desarrollo de este trabajo se hizo uso del conjunto de datos titulados: "Reconocimiento de la actividad humana utilizando el conjunto de datos de teléfonos inteligentes". Este conjunto de datos está disponible en el repositorio de aprendizaje automático de la Universidad de California Irvine (UCI dataset, 2012).

Las características del conjunto de datos se describen a continuación. Este conjunto de datos fue construido a partir de las grabaciones de 30 personas voluntarias, quienes se encontraban en el rango de edades de 19 a 48 años. Cada persona de este grupo realizó seis actividades, las cuales fueron *caminar*, *subir escaleras*, *bajar escaleras*, *tomar asiento*, *ponerse de pie y acostarse*. Estas actividades fueron realizadas mientras llevaban consigo un teléfono inteligente (Samsung Galaxy S II) colocado en la cintura, haciendo uso del acelerómetro y giroscopio integrados. Se realizó la captura de la aceleración lineal en 3 ejes y la velocidad angular en 3 ejes a una velocidad constante de 50 Hz.

El conjunto de datos obtenido de las actividades se dividió de forma aleatoria en dos subconjuntos, donde se seleccionó un 70 % de los voluntarios para generar los datos de entrenamiento y 30 % de los voluntarios para generar los datos de prueba. Cabe destacar que las características están normalizadas y limitadas dentro de del rango [-1,1]. En la Tabla 2, se muestran las características cuantitativas.

Es importante señalar que en este trabajo se unieron los dos archivos, el de entrenamiento y el de prueba en uno solo para poder dividir el conjunto de datos en diferentes porcentajes de prueba y entrenamiento. Lo anterior se realizó porque los trabajos del estado del arte aplican el método de validación de Hold-Out con 80-20.

En la Tabla 2, se puede observar que las clases están desbalanceadas, es decir, las clases no tienen el mismo número de registros, por lo que se procede a balancear estos conjuntos.

Los métodos de balanceo de datos también suelen ser conocidos como métodos de preprocesado para conjuntos de datos, y estos se dividen en dos tipos:

Submuestreo: Consiste en eliminar objetos de la clase mayoritaria, es decir, la clase que contenga mayor cantidad de datos dentro del conjunto, con el fin de crear un conjunto que se encuentre equilibrado y con una menor cantidad de información; pero la principal desventaja del submuestreo es que, al realizar la exclusión de objetos, se pueden eliminar datos que sean importantes dentro del conjunto, afectando de esta manera el modelo construido para realizar la clasificación.

Sobremuestreo: Consiste en la creación de nuevos objetos de la clase minoritaria, es decir, la clase que cuenta con la menor cantidad de datos, para producir unos nuevos datos con una distribución equilibrada de clase. Sin embargo, el principal inconveniente del enfoque de sobre muestreo es que puede incluir también muchos objetos artificiales que pueden producir sobreajuste.

Tabla 2. Características del conjunto de datos

Información de atributos	
Número total de instancias	10299
Número de atributos	561
Conjunto balanceado	No
Cantidad de registros por clase	
Clase 1: Caminar	1722
Clase 2: Subir escaleras	1544
Clase 3: Bajar escaleras	1406
Clase 4: Tomar asiento	1777
Clase 5: Ponerse de pie	1906
Clase 6: Acostarse	1944

En este trabajo, se utilizó la función SMOTE de Python que realiza un sobremuestreo.

En la Tabla 3 se muestran los datos balanceados.

Tabla 3. Balanceo del conjunto de datos

Información de atributos balanceados	
Número de instancias	11664
Número de atributos	561
Conjunto balanceado	Si
Cantidad de registros por clase	
Clase 1: Caminar	1944
Clase 2: Subir escaleras	1944
Clase 3: Bajar escaleras	1944
Clase 4: Tomar asiento	1944
Clase 5: Ponerse de pie	1944
Clase 6: Acostarse	1944

Posteriormente se aplica una estandarización de manera que la distribución tenga un valor medio igual a 0 y una desviación estándar que sea igual a 1.

DISCUSIÓN Y ANÁLISIS DE RESULTADOS

Para la implementación de los tres algoritmos de clasificación se utilizó el lenguaje de programación Python

3.7. El software se ejecutó en una laptop OMEN Core-i7 con 64 GB en RAM.

A continuación se presentan, en tres subsecciones, los resultados obtenidos después de haber aplicado los tres algoritmos de clasificación al conjunto de datos UCI-HAR.

Se debe tomar en cuenta, como ya se indicó anteriormente, que a los datos se les aplicaron dos pre-procesamientos: Balanceo y estandarización.

El método de validación utilizado fue Hold-Out. Las métricas que se aplicaron fueron precisión, exactitud y F1-score.

Cada una de las tablas de resultados que se mostrarán a continuación se enfocarán en el porcentaje de entrenamiento de 80 %, ya que los trabajos que utilizan el mismo conjunto de datos aplican este porcentaje.

RESULTADOS DEL ALGORITMO K-NN

Para este algoritmo se propuso utilizar una $K = 1$ y la distancia Euclidiana. En la Figura 6 se muestra la gráfica que sustenta el valor de $K = 1$. Se puede observar que con este valor de K se alcanza el valor máximo de exactitud que es mayor de 0.97.

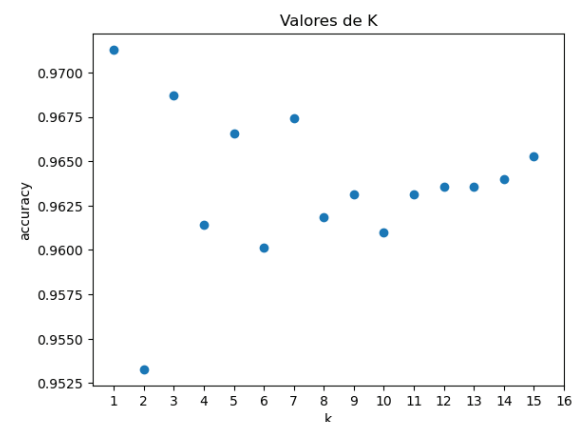


Figura 6. Gráfica que muestra el valor de K que arroja una mejor efectividad

En la Tabla 4 se muestran los resultados al aplicar los dos pre-procesamientos junto con el método de validación Hold-Out (H-O) para diferentes porcentajes de aprendizaje y prueba.

Tabla 4. Resultado de las métricas con balanceo y estandarización aplicando el algoritmo K-NN con H-O

% Entrenamiento	Precisión (%)	Exactitud (%)	F1-score (%)
10	91.13	91.13	91.13
20	93.07	93.07	93.07
30	94.13	94.13	94.13
40	95.04	95.04	95.04
50	95.83	95.83	95.83
60	96.46	96.46	96.46
70	96.85	96.85	96.85
80	97.25	97.25	97.25
90	97.17	97.17	97.17

De la Tabla 4 se puede observar que el resultado de las tres métricas es de 97.25 %.

RESULTADOS DEL ALGORITMO DE REGRESIÓN LOGÍSTICA

En esta sección se presentan los resultados de las pruebas realizadas aplicando el algoritmo de clasificación de Regresión Logística. De igual manera que en la sección anterior, para la validación de los resultados se aplicó el método de Hold-Out con porcentajes de entrenamiento desde el 10 % hasta 90 %.

La Tabla 5 presenta los resultados de las métricas de precisión, exactitud y F1-score.

Tabla 5. Resultados de la aplicación de Regresión Logística, utilizando el algoritmo de validación H-O

% Entrenamiento	Precisión (%)	Exactitud (%)	F1-score (%)
10	96.29	96.29	96.29
20	96.84	96.84	96.84
30	97.62	97.62	97.62
40	97.97	97.97	97.97
50	97.95	97.95	97.95
60	98.30	98.30	98.30
70	98.17	98.17	98.17
80	98.24	98.24	98.24
90	98.54	98.54	98.54

La Tabla 5 indica que el porcentaje en cada una de las métricas empleadas es de 98.24 %. El algoritmo de Regresión Lineal muestra una mejora del 0.99 % respecto al método del 1-NN.

RESULTADOS DEL ALGORITMO DE REDES NEURONALES CONVOLUCIONALES

La arquitectura de la Red Neuronal Convolutiva (RNC) utilizada en este trabajo es la siguiente:

- Capa de entrada: 1.
- Capas de convolución: 4.
- Capas de agrupación: 4.
- Capa totalmente conectada: 1.
- Cantidad de filtros por convolución: 16, 32, 64 y 128, respectivamente.
- Tamaño de filtro: 3 filtros de (7x7) y un filtro de (5x5).
- Función de activación: ReLu.
- Algoritmo de optimización: Adam.

En el caso del Perceptrón Multicapa, se propusieron los siguientes parámetros:

- Dos capas ocultas con 100 y 80 neuronas, respectivamente.
- Función de activación: ReLu.
- Algoritmo de optimización: Adam.
- Núm. de épocas: 50.
- Batch-size: 300.

Cabe hacer mención que en este caso, solo se aplicó el balanceo de datos, pero no el pre-procesamiento de los datos, es decir, no se aplicó la estandarización debido a que los resultados obtenidos de cada una de las métricas empleadas eran menores, en comparación con los que se obtuvieron usando los datos sin procesamiento.

En la Tabla 6 se muestran los valores de exactitud obtenidos al aplicar el método de validación de Hold-Out.

Tabla 6. Resultados de las métricas con el algoritmo de Redes Neuronales Convolucionales y validando con Hold-Out

% Entrenamiento	Precisión (%)	Exactitud (%)	F1-score (%)
10	66.28	54.45	59.79
20	94.59	94.27	94.43
30	96.85	96.82	96.83
40	96.02	96.01	96.02
50	97.82	97.80	97.81
60	98.21	98.14	98.12
70	98.67	98.62	98.64
80	99.03	99.08	99.12
90	99.04	99.06	99.13

La red neuronal convolutiva muestra los mejores resultados de las métricas de entre los tres métodos aplicados en este trabajo, como se puede apreciar en la Tabla 6. En el caso de la precisión, se tiene una mejora del 1.78 % y del 0.79 % respecto al 1-NN y Regresión Logística, respectivamente. Para la exactitud, se muestra una mejora de 1.82 % y 0.84 % en comparación de

los otros dos métodos. Finalmente, para la métrica del F1-score, la RNC supera al método del 1-NN y Regresión Logística en 1.87 % y 0.88 %, respectivamente.

Ahora, en la Tabla 7 se muestra la comparación de esta propuesta con los resultados obtenidos por los trabajos relacionados.

En la Tabla 7 se han resaltado los trabajos de Sansano *et al.* (2020), Satya *et al.* (2023) y nuestra propuesta, debido a que se utiliza el mismo conjunto de datos.

Tabla 7. Comparación de los resultados de exactitud entre los trabajos del estado del arte y nuestra propuesta

Autor-Año	Características	Resultados (%)
Murad <i>et al.</i> , 2017	Redes neuronales recurrentes profundas con memorias de corto plazo, se utilizan arquitecturas unidireccionales, bidireccionales y en cascada. Método de validación: Hold-Out 80-20	Exactitud: 96.7
Sansano <i>et al.</i> , 2020	Redes Neuronales Convolucionales. Validación: Hold-Out (80-20)	Precisión: 98.74
Hossain <i>et al.</i> , 2020	Máquina de Soporte Vectorial (SVM) para identificar actividades estáticas y una Red Neuronal Convolutiva 1D para reconocer actividades dinámicas antes de la aplicación de ambos algoritmos, aplican Árboles de Decisión para identificar si la actividad es estática o dinámica. Algoritmo de validación: Hold-Out 80-20	Exactitud: 97.66
Ibrahim <i>et al.</i> , 2022	Pre-procesamiento, extracción de características, selección de características, fusión de características y técnicas de Machine Learning para la clasificación	Exactitud: 99.5 para el conjunto de datos de Weizmann 99.9 para el conjunto de datos KTH
Satya <i>et al.</i> , 2023	Regresión Logística, SVC Lineal, Kernel de SVM, Árboles de decisión	Exactitud: Clasificador Lineal de Soporte Vectorial 96.7
Sri <i>et al.</i> , 2023	Red Neuronal Convolutiva	Exactitud: 94.08
Hassan <i>et al.</i> , 2023	Red Neuronal Convolutiva profunda	Exactitud: 98 para movimientos simulados 90 para movimientos reales
Praneeth <i>et al.</i> , 2023	Algoritmos: K-NN, SVM, Árboles de decisión y Bosques aleatorios	Exactitud: K-NN: 96.78 SVM: 94.19 Árboles de decisión: 93.78 Bosques aleatorios: 97.68
Este trabajo	Algoritmos 1-NN, Regresión Logística (RL) y Redes Neuronales Convolucionales (RNC). Validación con Hold-Out (80-20)	1-NN Exactitud: 97.25, Precisión: 97.25 y F1-score: 97.25
		RL Exactitud: 98.24, Precisión: 98.24 y F1-score: 98.24
		RNC Exactitud: 99.03, Precisión: 99.08 y F1-score: 99.12

De la Tabla 7 se derivan varias observaciones. Primero, la precisión de Sansano *et al.* (2020) la cual es de 98.74 % aplicando RNC, mientras que la precisión presentada en este trabajo, y utilizando la misma herramienta, fue de 99.08 %, mostrando un incremento de 0.34 %. Hay que resaltar que Sansano *et al.* utiliza 3 capas de convolución mientras que en esta propuesta se aplican 4 capas de convolución. Se realizaron pruebas adicionando más capas de convolución, sin embargo, los resultados ya no mejoraron. El trabajo de Hossain *et al.* (2020) reporta una precisión de 97.66 % con una combinación de tres algoritmos: SVM, RNC y Árboles de decisión, mientras que, en este trabajo, la Regresión Logística muestra una precisión de 98.24 %.

Para hacer la comparación del trabajo elaborado por Satya *et al.* (2023) y nuestra propuesta, se presenta la Tabla 8 donde se muestran los resultados de exactitud para cada una de las actividades.

De la Tabla 8 podemos observar que la Red Neuronal Convolutiva propuesta en este trabajo presenta

los mejores resultados que los algoritmos de Machine Learning. También se puede observar que el método del 1-NN también supera los resultados presentados en Satya *et al.* (2023).

En general, las RNC aplicadas en esta propuesta presentan un mejor resultado que los trabajos del estado del arte. En el caso de la Regresión Logística los resultados son similares.

En las pruebas, también se obtuvo el reporte de clasificación que indica las tres métricas para cada una de las actividades. Se observó que las actividades de sentarse y pararse son las que obtienen resultados en promedio de 95 % para exactitud, precisión y F1-score, y son las que decrecientan los valores finales, ya que las demás actividades muestran valores muy cercanos a 100 %. Por ejemplo, en la Figura 7 se muestra una captura de pantalla de los resultados de la ejecución del algoritmo 1-NN con Hold-Out. Se puede observar que las métricas tienen un valor menor en las actividades de

Tabla 8. Resultados de exactitud para cada una de las actividades, obtenidos por Satya *et al.* (2023) y nuestra propuesta (1-NN, Regresión Logística y RNC)

Algoritmo	Acostarse	Sentarse	Pararse	Caminar	Subir	Bajar
Regresión logística	100 %	88 %	97 %	99 %	96 %	95 %
SVM Lineal	100 %	88 %	98 %	100 %	98 %	96 %
SVM con kernel RBF	100 %	90 %	98 %	99 %	95 %	96 %
Árboles de decisión	100 %	75 %	89 %	95 %	84 %	77 %
Regresión logística	100 %	95 %	92 %	99 %	95 %	97 %
1-NN	100 %	92 %	95 %	100 %	99 %	100 %
RNC	100 %	97 %	97 %	100 %	100 %	100 %

```
Presicion del clasificador K-NN con los datos de entrenamiento: 1.0000
Presicion del clasificador K-NN con los datos de prueba: 0.9734
```

```
-----
Exactitud: 0.9734247749678525
F1 score: 0.9734247749678525
Precision: 0.9734247749678525
```

Matriz de confusion

```
[[380  1  0  0  0  0]
 [ 0 372  0  0  0  0]
 [ 0  3 380  0  0  0]
 [ 0  0  0 366 20  1]
 [ 0  0  0  37 349  0]
 [ 0  0  0  0  0 424]]
```

```
Reporte de clasificación
precision  recall  f1-score  support
Walking           1.00    1.00    1.00    381
Walking Upstairs  0.99    1.00    0.99    372
Walking DownStairs 1.00    0.99    1.00    383
Sitting           0.91    0.95    0.93    387
Standing          0.95    0.90    0.92    386
Laying            1.00    1.00    1.00    424

accuracy          0.97
macro avg         0.97
weighted avg      0.97
```

Figura 7. Captura de pantalla de la ejecución de la clasificación utilizando el algoritmo K-NN con el método de validación Hold-Out

sentarse y pararse, pero es de 100 % o 99 % para las actividades restantes.

CONCLUSIONES

La detección de actividades humanas tiene varias aplicaciones y este trabajo se enfocó en detectar seis actividades que son: Caminar, subir escaleras, bajar escaleras, tomar asiento, ponerse de pie y acostarse, esto con el objetivo de poder recomendarle a una persona incrementar o decrementar sus movimientos para tener una mejor salud. El objetivo principal fue el detectar esas seis actividades y se logró.

Se aplicaron tres algoritmos de clasificación: 1-Nearest Neighbors (1-NN), Regresión Logística (RG) y Redes Neuronales Convolucionales (RNC). Las clases se balancearon, como primer paso de pre-procesamiento y en seguida, se aplicó una estandarización a los datos. El método de validación fue Hold-Out (80-20). Los mejores resultados se obtuvieron con las RNC con una Exactitud de 99.03 %, Precisión: 99.08 % y F1-score: 99.12 %. Los resultados de exactitud y precisión superan a los resultados mostrados por los trabajos del estado del arte que utilizan el mismo conjunto de datos, por lo que se puede asegurar que esta propuesta mejora los resultados en la detección de actividades humanas. Esta mejora se logró adicionando una capa de convolución a la RNC. Es importante resaltar que esto no implica que agregando más capas de convolución se pueden mejorar los resultados, ya que se hicieron las pruebas correspondientes y a pesar de usar más capas, los resultados obtenidos no mejoraron.

Otro punto importante por resaltar es que las actividades de sentarse y pararse afectan de manera negativa a los resultados, las demás actividades, de manera individual, muestran valores de 99 % y 100 % en las métricas de exactitud, precisión y F1-score, mientras que sentarse y pararse muestran porcentajes alrededor de 95 %.

Como trabajo futuro se propone aplicar un pre-procesamiento a los datos para tratar de que los porcentajes de exactitud aumenten en las actividades de sentarse y pararse. Aplicar una función no lineal a los datos puede subsanar este problema.

AGRADECIMIENTOS

Los autores agradecen al Instituto Politécnico Nacional (COFAA y SIP) y al Sistema Nacional de Investigadores por el apoyo económico para el desarrollo de este trabajo.

REFERENCIAS

- Babiker, M., Khalifa, O., Htike, K., Hassan, A., & Zaharadeen, M. (Noviembre de 2017). Automated daily human activity recognition for video surveillance using neural network. 2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA), 1-5. <https://doi.org/10.1109/ICSIMA.2017.8312024>.
- Flores, L. A., Cervantes, N., Quintana, E., & Enriquez-del Castillo, L. A. (2021). Actividad física y estilo de vida sedentario en adultos, cambios durante el confinamiento por la pandemia de Covid-19. *Salud Pública de México*, 63(6), 825-826. <https://doi.org/10.21149/13199>
- Hassan, S., Wang, X., Ishtiaq, S., Ullah, N., Mohammad, A., & Noorwali, A. (2023). Human activity classification based on dual micro-motion signatures using interferometric radar. *Remote Sensing*, 15(7), 1752. <https://doi.org/10.3390/rs15071752>
- Hossain, M., Ahmed, N., Nouduri, K., & Palaniappan, K. (2020). A hybrid approach for human activity recognition with support vector machine and 1D convolutional neural network. 2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR), 1-5, <https://doi.org/10.1109/AIPR50011.2020.9425332>
- Ibrahim, M. Kainat, AlSalman, H., Ullah, S., Al-Hadhrami, S., & Hussain, S. (2022). An effective approach for human activity classification using feature fusion and machine learning methods. *Applied Bionics and Biomechanics*, 22, 1-14, <https://doi.org/10.1155/2022/7931729>
- INEGI (2021). Recuperado de <https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2022/moprade/moprade2021.pdf>
- Moocarme, M., Abdolahnejad, M., & Bhagwat, R. (2020). *The deep learning with keras workshop*. Packt Publishing, 2a ed., p. 227.
- Murad, A., & Pyun, J.-Y. (2017). Deep recurrent neural networks for human activity recognition. *Sensors*, 17(11), 2556. <https://doi.org/10.3390/s17112556>
- Praneeth, Ch., Chowdary, P., Sairam, M., Janardhan, K., & Sankar, S. (2022). Human activity classification based on smartphone data using ML techniques. *International Journal of Innovative Engineering and Management Research*, 12(2), 100-104, <https://doi.org/dx.doi.org/10.2139/ssrn.4382590>
- Randhawa, P., Shanthagiri, V., Kumar, A., & Yadav, V. (2020). Human activity detection using machine learning methods from wearable sensors. *Sensor Review*, 40(5), 591-603. <https://doi.org/10.1108/SR-02-2020-0027>
- Raschka, S., & Mirjalili, V. (2019). *Python machine learning: Aprendizaje automático y aprendizaje profundo con Python, sci-kit learn y Tensorflow*. 2a ed., Marcombo, p. 140.
- Sansano, E., Montoliu, R., & Belmonte-Fernández, S. (2020). A study of deep neural networks for human activity recognition. *Computational Intelligence*, 36(3), 1113-1139. <https://doi.org/10.1111/coin.12318>
- Satya, U., Thanuja, P., Jaya, L., & Chivukula, R. (2023). Classifying human activities using machine learning and deep learning

techniques. En: Bhateja, V., Carroll, F., Tavares, J.M.R.S., Sengar, S.S., & Peer, P. (eds). *Intelligent Data Engineering and Analytics. FICTA 2023. Smart Innovation, Systems and Technologies*, Vol. 371. Springer, Singapore. https://doi.org/10.1007/978-981-99-6706-3_2

Sri, R., Mulyanto, Y., Eddy, P., & Hery, P. (2023). Human activity classification using deep learning based on 3D motion feature. *Machine Learning with Applications*, 12(15), 1-10. <https://doi.org/doi.org/10.1016/j.mlwa.2023.100461>

Sun, J., Fu, Y., Li, S., He, J., Xu, C., & Tan, L. (2018). Sequential human activity recognition based on deep convolutional network and extreme learning machine using wearable sensors. *Journal of Sensors*, 1-10. <https://doi.org/10.1155/2018/8580959>

UCI. Dataset. Recuperado de <https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones>

Wazery, Y., Saber, E., Houssein, E., Ali, A., & Amer, E. (2021). An efficient slime mould algorithm combined with k-nearest neighbor for medical classification tasks. *IEEE Access*, 9, 113666-113682. <http://dx.doi.org/10.1109/ACCESS.2021.3105485>

Cómo citar: López-Reynaga, B. A., Acevedo-Mosqueda, M. E., Acevedo-Mosqueda, M. A., & Gómez-Coronel, S. L. (2024). Clasificación de actividades humanas aplicando Inteligencia Computacional. *Ingeniería Investigación y Tecnología*, 25(02), 1-12. <https://doi.org/10.22201/fi.25940732e.2024.25.2.009>