Análisis, diseño, y simulación de un sistema de clasificación basado en técnicas de aprendizaje de máquina para Human Activity Recognition

Samuel Acevedo Bustamante
Ingeniería de Sistemas
Universidad de Antioquia
Medellín, Colombia
samuel.acevedob@udea.edu.co

Danilo Antonio Tovar Arias
Ingeniería de Sistemas
Universidad de Antioquia
Medellín, Colombia
danilo.tovar@udea.edu.co

Oswald Daniel Gutiérrez Cortina
Ingeniería de Sistemas
Universidad de Antioquia
Medellín, Colombia
odaniel.gutierrez@udea.edu.co

***Resumen*—Este documento contiene la información relacionada con el proceso de investigación, análisis y evaluación de diferentes modelos de inteligencia artificial para un conjunto de datos de Reconocimiento de Actividad Humana usando teléfonos inteligentes; a través de la exploración de artículos con diferentes aproximaciones para la solución de este problema y la realización de diferentes experimentos se busca comprender el proceso de modificación y utilización de modelos de inteligencia artificial.**

***Palabras Clave*—Reconocimiento de Actividad Humana, Inteligencia Artificial, Clasificación**

I. Introducción

El reconocimiento automático de actividades humanas (En inglés, Human Activity Recognition, HAR) mediante sensores embebidos en dispositivos móviles ha emergido como un campo clave en el desarrollo de aplicaciones inteligentes para la salud, el deporte, la seguridad y la interacción humano-computador. En este contexto, el conjunto de datos *Human Activity Recognition Using Smartphones*, disponible públicamente en [1], representa una valiosa fuente de datos para la evaluación y comparación de algoritmos de aprendizaje automático en tareas de clasificación multiclase basadas en señales temporales.

Este dataset fue recopilado a partir de un experimento conducido con 30 voluntarios (19 hombres y 11 mujeres) de entre 19 y 48 años. Cada participante portó un *smartphone* "Samsung Galaxy S II" adherido a la cintura, mientras realizaba seis actividades físicas: caminar, subir escaleras, bajar escaleras, sentarse, estar de pie y acostarse. Durante la ejecución de estas actividades, se registraron datos del acelerómetro y giroscopio triaxial del dispositivo a una frecuencia de muestreo de 50 Hz.

II. Descripción del problema

A. Contexto del problema

El reconocimiento automático de actividades humanas a partir de datos de sensores móviles es un desafío fundamental para desarrollar aplicaciones inteligentes que mejoren la calidad de vida. Identificar con precisión qué actividad está realizando una persona, como caminar, sentarse o subir escaleras, permite crear sistemas personalizados para monitoreo de salud, asistencia a personas mayores, control deportivo y entornos de interacción adaptativa.

Sin embargo, distinguir entre estas actividades usando únicamente datos crudos de acelerómetros y giroscopios resulta complejo debido a la variabilidad natural de los movimientos humanos y el ruido inherente a los sensores. Por ello, se requiere una solución capaz de extraer patrones significativos y generalizables que permitan clasificar correctamente cada actividad.

El uso de técnicas de *Machine Learning* (ML) ofrece un enfoque efectivo para resolver este problema, ya que permite construir modelos que aprenden a clasificar actividades a partir de un conjunto de características previamente definidas. Estos modelos pueden adaptarse a diferentes usuarios y condiciones, ofreciendo precisión y robustez en entornos reales.

Desarrollar una solución basada en ML para el reconocimiento de actividades humanas facilita la creación de aplicaciones inteligentes con alto impacto social y comercial, tales como sistemas de rehabilitación remota, monitoreo continuo de salud, y dispositivos *wearables* que promuevan un estilo de vida saludable.

B. Composición de la base de datos

De acuerdo a la información asociada de [1], durante la recolección de datos, 30 voluntarios realizaron seis actividades mientras portaban un smartphone con sensores triaxiales (acelerómetro y giroscopio), muestreados a 50 Hz, es decir, 50 datos por segundo por eje. Las señales fueron segmentadas en ventanas de 2.56 segundos, equivalentes a 128 muestras por ventana. Para incrementar la cantidad de ejemplos y capturar mejor la variabilidad, se aplicó una superposición del 50% entre ventanas consecutivas, generando un desplazamiento de 1.28 segundos (64 muestras).

Este procedimiento generó una secuencia de ventanas etiquetadas de forma supervisada, asociadas tanto a la actividad realizada como al sujeto correspondiente. Como resultado, el conjunto de datos contiene 10,299 ventanas, cada una representando un segmento temporal en el que se llevó a cabo una de las seis actividades predefinidas.

Las características corresponden a estadísticas calculadas sobre series temporales preprocesadas, tales como medias, desviaciones estándar, energías, magnitudes de señal, coeficientes de correlación, transformadas rápidas de Fourier (FFT), entre otras.

Las características se extraen de señales en los ejes X, Y y Z provenientes de dos fuentes: aceleración total y aceleración corporal (filtrada para eliminar la componente gravitacional), así como velocidad angular (giroscopio). Todas las señales fueron previamente normalizadas y transformadas para capturar tanto información temporal como frecuencial.

El dataset incluye además dos columnas meta: *subject*, que identifica al individuo que realizó la actividad (entero de 1 a 30), y *activity*, que representa la etiqueta de clase (una de seis categorías predefinidas).

La estructura se encuentra dividida en dos archivos principales:

- *train*, que contiene */X_train.txt*, */y_train.txt*, y */subject_train.txt*
- *test*, que contiene */X_test.txt*, */y_test.txt*, y */subject_test.txt*

De estas estructuras se tiene que */X_train.txt* y */X_test.txt* corresponden a las muestras por cada actividad y sujeto con un total de 561 características cada una, luego */y_train.txt* y */y_test.txt* son las actividades asociadas a cada muestra denominadas con valores numéricos de 1 a 6, denotando las actividades en el siguiente orden *WALKING*, *WALKING_UPSTAIRS*, *WALKING_DOWNSTAIRS*, *SITTING*, *STANDING* y *LAYING*; finalmente */subject_train.txt* y */subject_test.txt* contienen los sujetos asociados a cada muestra, tal que los sujetos de *train* y *test* son diferentes para la totalidad de las muestras.

C. Paradigma a utilizar

El paradigma a utilizar corresponde a un aprendizaje multi-instancia y multiclase, donde el objetivo es agrupar diferentes muestras en ”bolsas” según el sujeto y la actividad realizada, tal que al momento de realizar el entrenamiento y predicción se obtienen un conjunto de clasificaciones sobre las muestras o ”instancias” en cada bolsa que permiten clasificar cada una de acuerdo a un proceso de agregación de los resultados.

III. Estado del arte

Se realizó una exploración sobre diferentes estudios previos relacionados a HAR para entender diferentes perspectivas y soluciones aplicadas al problema. Entre ellas se destacan las siguientes:

- En [2] se desarrolló un modelo de *Support Vector Machines* (SVM) clasificador binario generalizado para casos multiclase mediante la aproximación *One-vs-All* (OVA), y se realizó la selección de los hiperparámetros a través de una validación cruzada de 10 pliegues o *folds* y *kernels* Gaussianos. A través de la evaluación por *Precision* (inexactitud del modelo respecto a valores negativos predichos como positivos) y *Recall* (inexactitud del modelo respecto a valores positivos predichos como negativos) se obtiene un modelo con una precisión general del 96% para los datos de prueba, destacando que existen dificultades para la clasificación correcta entre *STANDING* y *SITTING* debido a la ubicación del sensor (cintura).
- En [3] se explora cómo las características físicas de los sujetos influyen en los modelos de inteligencia artificial utilizando una aproximación de ”ensemble learning” con una multitud de algoritmos como *Random Forest, eXtremeGradientBoost (XGB), AdaBoost, Artificial Neural Network, Vanilla Recurrent Neural Network, Long Short Term Memory (LSTM), K-Nearest Neighbors (KNN),* etc. Para el entrenamiento, proceden a agrupar los datos por sujetos de peso y altura similares, y posteriormente iteran clasificadores sobre el dataset construido utilizando validación cruzada de 10 *Folds* para asegurar resultados robustos, escogiendo aquel con mejores resultados de *Accuracy*. Adicionalmente, realizan comparaciones con *Recall*, Precision y *F1-Score* (media armónica entre precision y recall). Teniendo en cuenta que durante los entrenamientos .
  Al final se encuentra que entrenar considerando los atributos físicos de los individuos ayuda a entrenar modelos más consistentes en sus predicciones y más resistentes a variaciones de *dataset*. Así mismo, los algoritmos que alcanzan el mayor *accuracy* son *Random Forest* y *XGB*; mientras que la familia de algoritmos que menos se beneficia son los algoritmos de *deep learning*, que en consideración de los autores tienden a sobreajustarse con facilidad y tienen menos capacidad para reconocer patrones estadísticos.
- En [4] se emplearon algoritmos como *Support Vector Machine*, *Random Forest*, *K-Nearest Neighbors* y redes neuronales profundas, que fueron entrenados y comparados en varios conjuntos de datos estándar, evaluando su desempeño a través de la validación cruzada entre dominios, que consistió en entrenar los modelos en un conjunto de datos y probarlos en otro diferente con el objetivo de medir su capacidad para generalizar y funcionar en nuevas condiciones (diferentes unidades de medida, frecuencias de muestreo o tamaños de ventana temporal) sin necesidad de reentrenamiento.
  Adicionalmente, el rendimiento se midió principalmente con la exactitud promedio en cinco ejecuciones por modelo, y también se utilizó la métrica *Maximum Mean Discrepancy* (MMD) para evaluar la diferencia estadística entre los conjuntos de datos. Al final, los resultados mostraron que el desempeño de los modelos disminuye notablemente al evaluarlos en datos de otro conjunto, incluso cuando la diferencia estadística es baja, debido a factores como la posición del dispositivo, el tipo de smartphone y características de los participantes. Además, las actividades de baja intensidad resultaron más estables entre dominios que las dinámicas.
- En [5] se busca mejorar la precisión del reconocimiento de actividades humanas usando *Exploratory Data Analy-*

sis (EDA) junto con técnicas de reducción de dimensionalidad, intentando superar los resultados de los modelos tradicionales con altos costos computacionales y baja eficiencia. Para lograr este objetivo proponen utilizar un EDA que concentre su atención en la distribución real de los datos en conjunto con la técnica de reducción de la dimensionalidad *T-Distributed Stochastic Neighbor Embedding* (t-SNE) que permite la visualización de datos de alta dimensionalidad. .

Posteriormente, para los experimentos utilizaron los componentes de aceleración gravitacional y su ubicación en el cuerpo como variables de entrada para realizar la clasificación mediante distintos modelos, tales como Regresión Logística, *LinearSVC*, *KernelSVM*, Árboles de decisión, *RandomForest* y *Gradient Boosting Decision Trees*. Así mismo, los parámetros de cada modelo fueron optimizados a través de la técnica de validación cruzada *GridSearchCV* y la evaluación se realizó a través de las métricas *Accuracy*, *Recall*, *F1-score* y Matrices de confusión. Finalmente, se obtuvo que los mejores modelos fueron *LinearSVC* y *KernelSVM* cada uno con *accuracy* de 96,66% y 96,46% respectivamente, que corresponden a valores superiores cuando se comparan con otros estudios presentados en el artículo.

A partir de los experimentos realizados en otros artículos, podemos observar que existen factores clave en la obtención de buenos resultados usando modelos de inteligencia artificial para la clasificación de HAR, como son la posición de los sensores en el cuerpo humano, las características físicas del sujeto y la utilización de métodos de validación cruzada robustos.

IV. Entrenamiento y Evaluación de los modelos

A. Configuración Experimental

1) Metodología de Validación: En este trabajo, se evaluó el desempeño de cinco modelos de clasificación en un escenario de Aprendizaje Multi-Instancia: Regresión Logística Multiclase, *K-Nearest Neighbors*, *Random Forest*, Red Neuronal Multicapa y Máquinas de Vectores de Soporte. Todos los modelos comparten un enfoque común de entrenamiento y evaluación, adaptado a la naturaleza de las bolsas (*bags*) de instancias.

División de Datos: La partición entre los conjuntos de entrenamiento y prueba fue definida previamente por los desarrolladores del conjunto de datos *UCI HAR Dataset*. Específicamente, los datos fueron cargados desde las rutas correspondientes a las carpetas `/train/` y `/test/` dentro del archivo ZIP original, lo cual garantiza una separación clara entre ambas particiones. Esta organización asegura que los sujetos presentes en el conjunto de entrenamiento no se repiten en el conjunto de prueba, lo cual es esencial para evitar el sesgo por familiaridad y permite evaluar con mayor rigurosidad la capacidad de generalización de los modelos a individuos completamente nuevos.

Transformación a Nivel de Instancia: Dado que los modelos de aprendizaje automático utilizados en este estudio operan de forma tradicional a nivel de instancia y no directamente sobre estructuras jerárquicas, se implementó una estrategia de transformación a nivel de instancia. Esta técnica consiste en reorganizar las bolsas de datos —conjuntos de múltiples instancias etiquetados de forma colectiva— en una única estructura plana que agrupa todas las instancias de entrenamiento de manera individual.

Para lograrlo, se consolidaron todas las instancias contenidas en las bolsas del conjunto de entrenamiento en un único conjunto de datos plano, el cual puede ser interpretado por los algoritmos convencionales de clasificación. A su vez, la etiqueta asignada a cada bolsa fue replicada para cada una de sus instancias, permitiendo mantener la correspondencia semántica entre las instancias individuales y la clase global a la que pertenecen. Esta estrategia permite aprovechar modelos discriminativos clásicos sin necesidad de adaptarlos explícitamente al paradigma multi-instancia.

Estandarización de Características: Como parte del proceso de preprocesamiento, se aplicó una estandarización a las características de las instancias con el fin de mejorar la estabilidad numérica y el rendimiento de los modelos de aprendizaje automático. Para ello, se empleó el método `StandardScaler`, el cual transforma los datos de manera que cada característica tenga media cero y desviación estándar unitaria.

El escalador fue ajustado únicamente utilizando las instancias del conjunto de entrenamiento, sin considerar en ningún momento la información del conjunto de prueba. Posteriormente, el mismo transformador fue aplicado tanto a los datos de entrenamiento como a los de prueba. Esta práctica responde a una recomendación metodológica ampliamente aceptada en el campo del aprendizaje automático, ya que evita la fuga de información (*data leakage*) desde el conjunto de prueba hacia el proceso de entrenamiento, lo cual podría comprometer la validez de los resultados y generar una sobreestimación del desempeño del modelo.

Entrenamiento: El proceso de entrenamiento del modelo se estructuró bajo el paradigma de Aprendizaje Multi-Instancia, donde la unidad de análisis fundamental es la *bolsa*, compuesta por múltiples *instancias*. En el contexto de este estudio, cada bolsa corresponde al conjunto de mediciones de sensores recopiladas para un sujeto específico mientras realiza una actividad singular, siendo cada ventana temporal de datos una instancia.

Para el entrenamiento, se adoptó un enfoque basado en clasificadores a nivel de instancia. El entrenamiento del clasificador se llevó a cabo sobre este conjunto de datos a nivel de instancia estandarizado, utilizando la etiqueta de bolsa propagada a cada instancia individual. Durante la fase de experimentación, se evaluaron diversos modelos de *Machine Learning* (Regresión Logística, K-Means, Random Forest, Red Neuronal, SVM) con diferentes configuraciones de hiperparámetros. Cada modelo aprendió a predecir la etiqueta de una instancia individual en función de sus características.

Este procedimiento de entrenamiento a nivel de instancia con posterior agregación a nivel de bolsa permitió aprovechar

la granularidad de los datos del sensor (instancias) mientras se evaluaba el rendimiento en la unidad de interés a nivel superior (bolsas), alineándose con los principios del aprendizaje multi-instancia.

2) Configuración de Hiperparámetros: A continuación, se describen brevemente los hiperparámetros explorados para cada uno de los modelos empleados en el estudio y el resumen de los posibles valores para cada hiperparametro evaluado se puede observar en la tabla I:

- **Regresión Logística (MIL)**: Se evaluaron diferentes algoritmos de optimización mediante el parámetro `solver` (*lbfgs*, *newton-cg*, *newton-cholesky*). El hiperparámetro `C` controla la regularización (valores: 10, 1, 0.1, 0.01, 0.001). Se exploraron distintos valores de `max_iter` para asegurar la convergencia (500 a 5000 iteraciones).
- **K-Nearest Neighbors (KNN, MIL)**: Se ajustó el número de vecinos mediante el parámetro `n_neighbors`, considerando valores desde 5 hasta 50. Asimismo, se probó el impacto de distintas métricas de distancia con el parámetro `metric` (*euclidean*, *manhattan*, *haversine*, *cosine*).
- **Random Forest (MIL)**: Se evaluaron distintas configuraciones de `n_estimators` (número de árboles), `max_depth` (profundidad máxima), `min_samples_split` (mínimas muestras para dividir un nodo) y `min_samples_leaf` (mínimas muestras por hoja). Adicionalmente, se consideró el método de agregación de predicciones a nivel de bolsa (`aggregation`) con valores *max* y *mean*.
- **Red Neuronal Multicapa (MLP, MIL)**: Se probaron distintas configuraciones del número de capas ocultas (`num_hidden_layers`) y el tamaño de cada capa (`hidden_layer_sizes`), abarcando entre 1 y 3 capas con 4 a 256 neuronas por capa. La función de activación utilizada fue `relu`, con un número máximo de iteraciones (`max_iter`) de 350 y una penalización L2 (`alpha`) de 0.01.
- **Máquinas de Vectores de Soporte (SVM, MIL)**: Se exploraron dos tipos de kernel (`kernel`): *linear* y *rbf*. También se evaluaron distintos valores del parámetro de regularización `C` y el límite de iteraciones `max_iter`.

3) Métricas de Desempeño: En este análisis para la tarea de reconocimiento de actividad humana (*Human Activity Recognition, HAR*), se utilizaron principalmente dos métricas de evaluación para medir el rendimiento de los modelos de clasificación: **Accuracy** y **F1-score macro**.

- **Accuracy (Exactitud):** mide la proporción de bolsas correctamente clasificadas con respecto al total. Proporciona una visión general sobre qué tan bien el modelo identifica las actividades, siendo útil como primera métrica de referencia en problemas multiclase como HAR.
- **F1-Score (Macro Average):** representa la media armónica entre la *precisión* y la *exhaustividad* (*recall*). La precisión evalúa cuántas de las instancias predichas como una clase realmente pertenecen a dicha clase, mientras que la exhaustividad mide cuántas de las instancias reales de una clase fueron correctamente identificadas. El promedio *macro* indica que se calcula el F1-score de forma independiente para cada clase y luego se promedia sin ponderar. Esta métrica es particularmente útil en escenarios con clases desbalanceadas, donde el accuracy puede resultar engañoso.

B. Resultados del entrenamiento de modelos

La evaluación del rendimiento se realizó a nivel de bolsa utilizando el conjunto de prueba. Cada bolsa fue transformada mediante el mismo `StandardScaler` ajustado con los datos de entrenamiento, asegurando la consistencia en la representación de las características. Posteriormente, el modelo entrenado generó predicciones individuales (ya sea etiquetas directas o probabilidades) para cada instancia en la bolsa.

Estas predicciones a nivel de instancia se consolidaron en una única predicción de bolsa mediante una estrategia de agregación:

- Para modelos probabilísticos como Regresión Logística, Red Neuronal (MLP) y SVM, se promediaron las probabilidades de las instancias para determinar la clase final de la bolsa.
- Para modelos como KNN y Random Forest, se empleó una estrategia de votación mayoritaria entre las etiquetas predichas a nivel de instancia.

Este enfoque permitió aprovechar la granularidad de los datos del sensor (instancias), mientras se mantenía la evaluación a nivel de bolsa, en coherencia con el paradigma de aprendizaje multi-instancia (MIL).

TABLA I
TABLA DE CONFIGURACIÓN PARA LOS MODELOS

Regresión Logística			**KNN**			**Random Forest**					**MLP**			**SVM**		
Solver	*C*	*Iter*	k	*Dist*	-	*Árboles*	*Prof*	*Split*	*MinSamp*	*Aggreg*	*Capas*	*Neur*	*Act*	*Kernel*	*C*	*Iter*
lbfgs	10	500	5	euclidean	-	50	5	2	1	Max	1	4	relu	linear	10	500
newton-cg	1	1000	10	manhattan	-	100	10	10	5	Mean	2	12	-	rbf	1	1000
newton-chol	0.1	3000	15	cosine	-	200	None	20	10	-	3	32	-	-	0.1	3000
-	0.01	5000	20	haversine	-	-	-	-	-	-	-	64	-	-	0.01	5000
-	0.001	-	25	-	-	-	-	-	-	-	-	128	-	-	0.001	-
-	-	-	30	-	-	-	-	-	-	-	-	256	-	-	-	-
-	-	-	35	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	40	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	45	-	-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	50	-	-	-	-	-	-	-	-	-	-	-	-	-

La evaluación del desempeño se basó en las métricas de **Accuracy** y **F1-score macro**, calculadas exclusivamente a nivel de bolsa sobre el conjunto de prueba.

Adicionalmente, se evaluó la robustez de los modelos en escenarios ruidosos mediante una versión perturbada del conjunto de prueba original.

Los resultados experimentales, cuyos mejores resultados se observan en la tabla II, mostraron que el modelo **SVC (Support Vector Classifier)** obtuvo el mejor desempeño general, destacándose en ambas métricas incluso bajo la presencia de ruido. A este le siguieron los modelos **KNN**, **Regresión Logística**, **Red Neuronal (MLP)** y finalmente **Random Forest**, que fue el más afectado por el ruido, indicando una posible tendencia al sobreajuste en el entrenamiento a nivel de instancia.

Este procedimiento, basado en entrenamiento a nivel de instancia con agregación posterior y evaluación bajo métricas robustas, permitió una comparación rigurosa de los modelos en el contexto de HAR bajo el enfoque MIL.

TABLA II
MEJORES RESULTADOS PARA CADA MODELO

Modelos	**Sin ruido**		**Con ruido**	
	Accuracy	**F1-Score**	**Accuracy**	**F1-Score**
SVC	1.0000	1.0000	0.9444	0.9440
KNN	0.9815	0.9814	0.9444	0.9430
Red Neuronal	1.0000	1.0000	0.8333	0.8292
Regresión Logística	1.0000	1.0000	0.9074	0.9072
Random Forest	0.9815	0.9814	0.5556	0.4709

V. REDUCCIÓN DE DIMENSIÓN

Para reducir la complejidad y los tiempos de procesamiento para los dos mejores modelos, aplicamos diferentes técnicas de selección y extracción de características, evaluando los resultados obtenidos de los mismos modelos al utilizar las muestras posteriores a la reducción de dimensión.

A. Selección de características

Para esta sección utilizamos dos métodos para revisar la viabilidad de la reducción de dimensión y, posteriormente, utilizamos el método *Sequential Forward Selection* para realizar la selección de las mejores características que describen la variable objetivo.

- **Información Mútua (IM):** Es una medida basada en la teoría de la información que cuantifica cuánta información comparte una variable con otra. Si dos variables son independientes, su IM es cero; mientras que cuanto mayor sea la IM, mayor es la dependencia entre ellas. A partir de esta medida se obtuvieron las relaciones de cada característica respecto a la variable objetivo y se marcaron aquellas variables que tenían un valor de información mutua por debajo del percentil 20, es decir, el 20% inferior de las características. En la figura 1 se puede observar la distribución de las diferentes características respecto a su valor de IM.

Si se revisa con detalle la figura, existe una característica que tiene un valor IM de 1; esto indica que es una variable decisiva a la hora de predecir la variables objetivo; Sin embargo, después de realizar unas pruebas preliminares sencillas respecto a esta característica, se descubrió que no es suficiente para determinar con gran exactitud la variable objetivo, obteniendo un 50 a 60 por-ciento de *Accuracy* y *F1-Score* con diferentes valores de prueba en un modelo KNN.

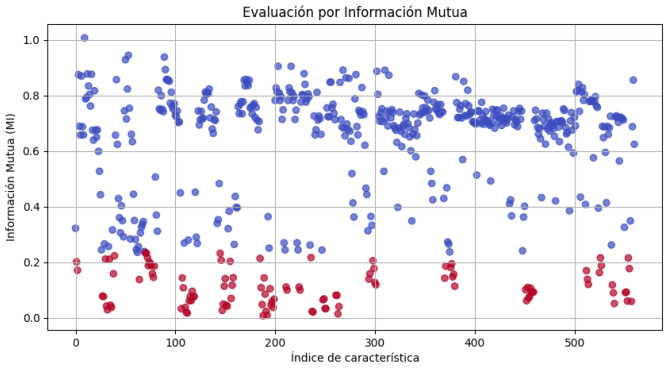

Fig. 1. Información Mutua respecto a la variable objetivo

A partir de lo descrito por el análisis de IM se encontraron diferentes características con indices de relación extremadamente bajos, candidatas a ser eliminadas para la posterior evaluación. Así mismo, si se eliminaran las características marcadas en rojo se obtendría una reducción de 112 variables y se realizaría una evaluación con 449 características.

- **Análisis de varianza:** Se examinó la varianza por característica de las muestras y se seleccionaron aquellas que poseen una varianza inferior a 0.01. De esta manera, se obtuvieron 37 variables que podrían ser eliminadas, permitiendo una reducción del 6.6%.
- ***Sequential Forward Selection (SFS):*** A través del recorrido secuencial de características y su evaluación respecto a un modelo, en nuestro caso respecto al KNN, se determina un conjunto de variables que mejor describe la variable objetivo dentro de un limite de selección. Dentro de las pruebas realizadas se limitó el conjunto de características a 30 como máximo debido a las limitaciones de los recursos del programa y la alta complejidad algorítmica producida por la alta dimensión del conjunto de datos. Una vez obtenido el conjunto de 30 características, se realizaron las modificaciones a los datos de entrenamiento y prueba para la evaluación respecto a los dos mejores modelos, correspondientes a SVM y KNN. Similarmente al entrenamiento y evaluación realizada previamente, se obtuvieron los resultados de aplicar el modelo utilizando diferentes hiper-parámetros y modificando el conjunto de datos de prueba con ruido; como se puede ver en la tabla III los valores de las métricas para ambos modelos se ven afectadas por utilizar el conjunto de características reducido y se vuelven

TABLA III
MEJORES RESULTADOS DE SFS PARA LOS 2 MEJORES MODELOS

Modelos	Sin ruido		Con ruido	
	Accuracy	F1-Score	Accuracy	F1-Score
SVC	0.8518	0.8286	0.7407	0.6763
KNN	0.8333	0.8302	0.7777	0.7359

menos robustos al ruido en comparación con los modelos cuando utilizan el conjunto de variables completo. Entre las razones por la cuáles ocurre esto se debe mencionar que un conjunto de 30 características no es suficiente para describir apropiadamente la variable objetivo o, en su defecto, las características seleccionadas por SFS no soportan correctamente las decisiones tomadas por el modelo y se requeriría aplicar algoritmos adicionales como *Sequential Backward Selection (SBS)*, *More-L Less-R Selection (LRS)* o *Sequential Floating Forward/Backward Selection (SFFS/SFBS)* para encontrar un conjunto de 30 características más robusto.

B. Extracción de características

Se aplicó el proceso *Principal Component Analysis (PCA)* para encontrar un conjunto de valores derivados de las características originales que mantuviese la mayor cantidad de varianza posible mientras se reduce la dimensión. Al momento de ajustar las características del conjunto de datos originales, se especificó que mantuviera el 95% de la varianza, generando una reducción del 81.82% y creando un conjunto de características de únicamente 102 variables; posteriormente se realizó la evaluación de este conjunto respecto a los dos mejores modelos y se obtuvieron los resultados descritos en la tabla IV, donde al realizar la comparación con SFS y los resultados originales se observa que el modelo de SVC mantiene el desempeño cuando no hay ruido y, cuando existe ruido, sorprendentemente es más robusto que el modelo generado por el entrenamiento con todas las características originales; mientras que el modelo KNN sigue viendose afectado por la reducción de dimensión, presentando peor desempeño que el modelo original, pero presenta mayor robustez ante el ruido respecto al modelo KNN generado por el conjunto de datos al aplicar SFS.

TABLA IV
MEJORES RESULTADOS DE PCA PARA LOS 2 MEJORES MODELOS

Modelos	Sin ruido		Con ruido	
	Accuracy	F1-Score	Accuracy	F1-Score
SVC	1.0000	1.0000	0.9814	0.9814
KNN	0.8518	0.8355	0.8518	0.8506

VI. DISCUSIÓN SOBRE LOS RESULTADOS

La obtención de un rendimiento perfecto, manifestado en un *accuracy* y un *F1-score macro* de 1.0 en el conjunto de prueba original, si bien matemáticamente posible, requiere un análisis cuidadoso en el contexto del aprendizaje automático. En el caso particular de conjuntos de datos como el UCI HAR, es común observar rendimientos muy altos cuando los datos de entrenamiento y prueba provienen de la misma fuente y se han procesado de manera similar. Esto se debe a que las características extraídas pueden ser extremadamente discriminatorias para las condiciones específicas bajo las cuales se recolectó el dataset.

No obstante, este alto rendimiento en el conjunto de prueba original no siempre se traduce en una generalización efectiva a escenarios más diversos. Según lo revisado en otros artículos y experimentos, cuando se prueba con datos provenientes de otros sensores, dispositivos o recolectados bajo condiciones significativamente diferentes (distintas posiciones del sensor, variaciones en la ejecución de actividades, etc.), el rendimiento de los modelos tiende a disminuir. Esto subraya que el rendimiento perfecto en el conjunto de prueba original puede ser un reflejo de la especificidad del dataset más que de una robustez inherente del modelo a variaciones amplias en los datos de entrada.

Por lo tanto, si bien un *accuracy* y *F1-score* de 1.0 en el conjunto de prueba original son resultados notables, es crucial evaluar la robustez del modelo ante la variabilidad inherente a los datos de sensores en el mundo real. La experimentación con datos ruidosos, como la realizada en este estudio mediante la adición de ruido gaussiano, proporciona una evaluación más realista de la capacidad del modelo para generalizar y mantener su rendimiento en condiciones menos ideales. Una caída significativa en el rendimiento al introducir ruido es un fuerte indicativo de sobreajuste a los datos originales, mientras que un rendimiento relativamente estable sugiere una mayor robustez y potencial de aplicación práctica.

Así mismo, la aplicación de un aprendizaje multi-instancia permite reducir la inestabilidad de las predicciones debido a que se evalúan los diferentes modelos a nivel de bolsas y no de instancias, permitiendo que errores sobre instancias puntuales sean corregidos por el resto de instancias correctas al interior de la bolsa, generando que la bolsa se clasificada correctamente. Esto permite que los resultados obtenidos para los experimentos tengan, en la mayoría de los casos, un rendimiento mejor o igual comparado con los modelos revisados en la sección III.

VII. CONCLUSIONES

La aplicación del paradigma multi-instancia permite obtener resultados más robustos ante las inexactitudes y errores cometidos durante las predicciones del modelo, sin embargo hay que tener en cuenta que la cantidad de instancias en las bolsas influye en la capacidad de corrección sobre la predicción final de la bolsa; y así mismo, es necesario tener en consideración el error que se comete a nivel de instancias para mantener un alto nivel predictivo ante las variaciones sobre los datos. Similarmente, tener en cuenta las variaciones implícitas de las muestras debido al sujeto evaluado favorece a la generalización de las predicciones sobre actividades especificas, tal que para cada actividad se tienen diferentes puntos de referencia durante el entrenamiento que favorecen la predicción para nuevos sujetos.

Los resultados obtenidos respaldan los hallazgos del estado del arte en cuanto a la influencia determinante de factores como la posición del sensor, las condiciones del entorno y las características individuales del sujeto en el rendimiento del modelo. En particular, se evidenció que la generalización del sistema se ve severamente limitada cuando los modelos son entrenados con datos capturados en un único contexto controlado. Esta dependencia sugiere que, aunque los modelos puedan alcanzar métricas sobresalientes en un entorno de prueba específico, su rendimiento puede deteriorarse significativamente cuando se despliegan en condiciones reales, donde las fuentes de variabilidad son mucho más amplias y menos predecibles.

En este sentido, para lograr una verdadera capacidad de generalización, es imperativo avanzar hacia estrategias de entrenamiento más diversificadas, en las cuales los modelos sean expuestos desde su etapa de aprendizaje a una gama más amplia de condiciones. Esto incluye no solo la incorporación de datos de distintos sujetos, sino también el uso de diferentes ubicaciones de sensores (e.g., muñeca, cintura, tobillo), dispositivos con diferentes características técnicas, y contextos de recolección variados (iluminación, nivel de ruido, superficie de movimiento, etc.).

Tal enfoque permitiría no solo capturar la variabilidad intersujeto e interdispositivo, sino también robustecer la representación interna de los modelos ante eventos atípicos. Esto es especialmente relevante en sistemas HAR orientados a la vida cotidiana, donde los usuarios pueden portar los sensores de forma no estandarizada y las condiciones de operación pueden cambiar drásticamente a lo largo del tiempo.

REFERENCIAS

[1] J. Reyes-Ortiz, D. Anguita, A. Ghio, L. Oneto, and X. Parra. ”Human activity recognition using smartphones”, UCI Machine Learning Repository, 2013. [En línea]. Disponible: https://doi.org/10.24432/C54S4K.

[2] D. Anguita, A. Ghio, L. Oneto, X. Parra, y J. L. Reyes-Ortiz, ”A Public Domain Dataset for Human Activity Recognition using Smartphones”, The European Symposium on Artificial Neural Networks, 2013.

[3] N. A. Choudhury, S. Moulik, y D. S. Roy, ”Physique-Based Human Activity Recognition Using Ensemble Learning and Smartphone Sensors”, IEEE Sensors Journal, vol. 21, no. 15, pp. 16852-16860, 2021. [En línea]. Disponible: https://doi.org/10.1109/JSEN.2021.3077563

[4] O. Napoli, D. Duarte, P. Alves, et al., ”A benchmark for domain adaptation and generalization in smartphone-based human activity recognition”, Sci Data, vol. 11, p. 1192, 2024. [En línea] Disponible: https://doi.org/10.1038/s41597-024-03951-4

[5] W. Kong, L. He, y H. Wang, ”Exploratory Data Analysis of Human Activity Recognition Based on Smart Phone”, IEEE Access, vol. 9, pp. 73355-73364, 2021. [En línea]. Disponible: https://doi.org/10.1109/ACCESS.2021.3079434