Análisis y predicción de la actividad de personas mayores

Salvador Mendoza[1 [A01067783]], Karla Gonzalez[1 [A01541526]], Alfonso Pineda[1 [A01660394]], and Mariana Rincón[1 [A01654973]]

Instituo Tecnológico y de Estudios Superiores de Monterrey, Campus Guadalajara.

Resumen En este trabajo de investigación, se emplearon técnicas de clasificación avanzada y análisis exploratorio de datos para predecir actividades físicas en personas mayores, abordando el desbalance en los datos mediante el uso de submuestreo. Se evaluaron varios modelos de clasificación en Python, seleccionando los más precisos, se llevaron a cabo procesos de validación cruzada para respaldar su eficacia, entre otros métodos de validación. Este trabajo contribuye al campo de la salud en la tercera edad y proporciona una base para futuras investigaciones y aplicaciones en la mejora de la calidad de vida de este grupo demográfico.

Keywords: Clasificación · Validación cruzada · Submuestreo.

1. Introducción

El reconocimiento de actividades humanas desempeña un papel fundamental en múltiples aplicaciones, desde la monitorización de la salud hasta la mejora de la calidad de vida de las personas mayores. El conjunto de datos *"Human Activity Recognition with sensors (HAR70+)"* se presenta como una fuente de información para abordar esta problemática. Este conjunto de datos se centra en la detección de actividades de adultos mayores, específicamente en 18 personas de 70 a 95 años de edad, que usaron dos acelerómetros (sensores) de tres ejes duarante aproximadamente 40 minutos mientras los participantes llevaban a cabo sus actividades diarias.

El objetivo principal de este conjunto de datos es entrenar modelos de aprendizaje automático para reconocer las actividades realizadas por los adultos mayores, aprovechando la información recopilada por los acelerómetros y las anotaciones asociadas.

Cada participante cuenta con un archivo .csv separado que contiene información detallada sobre los registros de aceleración en las direcciones x, y, y z, tanto para el sensor ubicado en la espalda, como para el sensor ubicado en el muslo. Además, se proporciona una columna de etiquetas que indica la actividad anotada correspondiente a cada registro.

Es importante destacar que las actividades (nuestras clases) en el conjunto de datos incluyen caminar, deslizarse, subir las escaleras, bajar las escaleras, estar de pie, sentarse y acostarse. Estas actividades se han asignado a códigos específicos que nos permiten identificarlas de manera única.

2. Metodología

2.1. Análisis Exploratorio de Datos

Comenzando con el análisis de nuestra base de datos, iniciando con el examen de un único archivo correspondiente a un paciente. El EDA por sus siglas en inglés, es la etapa inicial y fundamental en cualquier proyecto de análisis de datos. Esta fase inicial tiene como objetivo principal comprender la naturaleza y las características de los datos con los que se está trabajando. A través del EDA, se busca descubrir patrones, tendencias, relaciones y posibles anomalías en los datos antes de aplicar modelos más avanzados o tomar decisiones basadas en ellos.

Como un primer acercamiento a los datos, se utilizaron las librerías de numpy y pandas para: cargar el csv al programa, obtener el tamaño de la base de datos, la cantidad de datos nulos y el tipo de datos que contiene cada atributo. Una vez contando con estas características, se pudo realizar una estadística descriptiva, en la cual para los datos de tipo no objeto, se pueden obtener valores estadísticos como la media, la desviación estándar, mínimos y máximos, cuartiles del 25 %, 50 % y 75 %, varianzas y asimetrías.

Una manera de comprender mejor la distribución de los datos es analizándolos gráficamente, para esto se llevó a cabo la visualización de los mismos utilizando la librería de matplotlib. Mediante un diagrama de barras, pudimos observar la cantidad de datos de cada clase, este fue un indicio del desbalanceo en las clases, puesto que es muy notoria la desproporción de datos para cada clase. Por otro lado, se realizó un histograma para cada atributo del data set, y así conocer la distribución de los datos; observamos formas de campana en varios grafos, dando indicios de comportarse como distribuciones normales, no obstante, hubo ciertos atributos o variables que no pudimos describir como una distribución posible a simple vista. Finalmente, se realizaron los boxplots que grafican la variabilidad estadística de los valores por atributo.

Cómo ultimo método para la etapa de EDA se realizó una matriz de correlación, aquí no se demostró ningún tipo de correlación (ni alta, ni baja) entre los datos, por lo que no fue necesario realizar una estandarización de los mismos, al menos, no por esta razón.

2.2. Técnicas de Submuestreo

Durante este proceso de exploración, se identificó un desequilibrio significativo en la distribución de clases en la base de datos. Dado el desbalance de la base de datos, se optó por implementar técnicas de submuestreo como medida para abordar este problema. Se eligió el submuestreo debido a la preferencia de preservar datos existentes en lugar de crear datos artificiales, los cuales podrían introducir sesgos en el análisis.

El submuestreo, dentro del contexto de la clasificación de datos desbalanceados, es una técnica que se utiliza para reducir la proporción de muestras en la clase mayoritaria con el fin de equilibrar el número de datos de las clases [1].

2.2.1. ENN (Edited Nearest Neighbors) Antes de realizar este método de submuestreo se realizó una búsqueda sistemática de hiperparámetros utilizando la librería GridSearchCV de Python. Esto permitió determinar el número óptimo de vecinos para el método de submuestreo, que se encontró ser igual a 11. Además, se obtuvo una puntuación de validación cruzada de 0.95, lo que garantiza la confianza de este parámetro.

Para llevar a cabo el submuestreo, se aplicó el método de Edited Nearest Neighbors (ENN). Este método implica la eliminación de instancias que se consideran ruidosas o que no contribuyen significativamente a la representación de las clases. El proceso de implementación de ENN se basó en una búsqueda exhaustiva de hiperparámetros previa [2].

Posteriormente, todas las variables fueron estandarizadas utilizando la librería StandardScaler de Python. Esta estandarización es esencial para garantizar que todas las características tengan la misma escala y no introduzcan sesgos en los modelos de clasificación posteriores.

Con la base de datos procesada de esta manera, se estableció para la evaluación y comparación de diversos modelos de clasificación. Estos modelos serán utilizados para identificar y seleccionar el más adecuado en función de métricas de desempeño específicas, con el fin de lograr los objetivos del estudio.

2.3. Bosque Aleatorio

En el marco de este proyecto de análisis de actividad en adultos mayores, se llevaron a cabo pruebas de diversos modelos de clasificación con el objetivo de determinar el más adecuado para la tarea de reconocimiento de actividades. Entre los modelos evaluados, se destacaron el Bosque Aleatorio y el Árbol de Decisión como los más óptimos. El modelo de Bosque Aleatorio se configuró con 100 estimadores y una semilla aleatoria fija de 42. A continuación, se presentan los resultados obtenidos:

Clase	**Precisión**	**Recall**	**F1-Score**	**Support**
Caminar	0.99	1.00	0.99	34690
Arrastrar	0.97	0.42	0.59	521
Subir escaleras	0.00	0.00	0.00	18
Bajar escaleras	1.00	0.17	0.30	23
Parado	0.99	0.99	0.99	10193
Sentado	1.00	1.00	1.00	11242
Acostado	1.00	1.00	1.00	7480
Accuracy			0.99	64167
Macro Avg	0.85	0.65	0.69	64167
Weighted Avg	0.99	0.99	0.99	64167

Cuadro 1. Informe de Clasificación - Bosque Aleatorio con Validación cruzada

El informe de clasificación presenta los resultados del modelo de Bosque Aleatorio con validación cruzada. El modelo exhibe un rendimiento general sólido,

con altos valores de precisión y recall en varias clases, como 'Caminar,' 'Parado,' 'Sentado,' y 'Acostado,' superando el 99 % en la mayoría de los casos. Sin embargo, se observa un rendimiento más limitado en actividades como 'Arrastrar' y 'Bajar escaleras,' con valores de precisión y recall más bajos. Esto puede atribuirse a la falta de registros suficientes de personas que hayan realizado estas actividades, lo que dificulta que el modelo aprenda patrones distintivos para estas clases menos representadas.

A pesar de esto, el modelo logra un F1-Score promedio del 69 %, indicando un equilibrio razonable entre precisión y recall. La precisión global alcanza el 99 %, lo que resalta la capacidad general del modelo para clasificar con precisión diversas actividades.

2.4. Árbol de decisión

Como segundo modelo de clásificación más efectivo en el reconocimiento de actividades de los adultos mayores, se tiene el árbol de decisión.

Clase	**Precisión**	**Recall**	**F1-Score**	**Support**
Caminar	0.99	0.99	0.99	34690
Arrastrar	0.48	0.50	0.49	521
Subir escaleras	0.07	0.11	0.08	18
Bajar escaleras	0.44	0.30	0.36	23
Parado	0.97	0.97	0.97	10193
Sentado	1.00	1.00	1.00	11242
Acostado	1.00	1.00	1.00	7480
Accuracy			0.98	64167
Macro Avg	0.71	0.70	0.70	64167
Weighted Avg	0.98	0.98	0.98	64167

Cuadro 2. Informe de Clasificación - Árbol de Decisión con Validación cruzada

El informe de clasificación presenta los resultados del modelo de Árbol de Decisión con validación cruzada. Similar al modelo anterior, demuestra un buen rendimiento en la mayoría de las clases, con una alta precisión y recall para actividades como 'Caminar', 'Parado', 'Sentado', y 'Acostado', superando el 97 % en la mayoría de los casos. Sin embargo, muestra un rendimiento más limitado en actividades como 'Arrastrar', 'Subir escaleras', y 'Bajar escaleras', igualmente puede atribuirse a la falta de registros suficientes de personas que hayan realizado esas actividades, lo que limita la capacidad del modelo para aprender patrones distintivos.

El rendimiento general se refleja en un F1-Score promedio del 70 %, indicando un buen equilibrio entre precisión y recall. También se destaca una precisión global del 98 %, lo que puede ser muy bueno para clasificar con precisión diversas actividades.

2.5. Validación de los modelos

2.5.1. Validación cruzada En la elaboración de este proyecto se ve esencial el uso de validación cruzada para evaluar y validar el rendimiento de los modelos predictivos. Surge de la necesidad de obtener estimaciones más robustas y confiables. Esto se logra al evaluar el modelo en múltiples divisiones de datos, permitiendo así la obtención de métricas de rendimiento más consistentes y representativas.

Al interpretar los resultados de la validación cruzada, se busca evaluar la precisión del modelo, identificando posibles problemas sesgo o varianza. Se observa la consistencia de las métricas en las divisiones de los datos, se analizan las métricas de clase por clase y se seleccionan hiperpáramentros óptimos.

2.5.2. Matriz de confusión Radican en su capacidad para proporcionar una visión detallada de los aciertos y errores del modelo, incluyendo falsos positivos y falsos negativos.

2.5.3. Curvas ROC y AUC Estas curvas son herramientas funcionan para evaluar el rendimiento de modelos de clasificación binarios o multiclase. Estas métricas proporcionan una comprensión detallada del rendimiento de un clasificador en términos de su capacidad para distinguir entre clases [3].

Se entiende a una curva ROC como una representación gráfica de la relación entre la Tasa de Verdaderos Positivos (TPR) y la Tasa de Falsos Positivos (FPR) en función de un umbral de decisión variable. La TPR se refiere a la fracción de ejemplos positivos correctamente clasificados, mientras que la FPR es la fracción de ejemplos negativos incorrectamente clasificados como positivos. Esta gráfica mostrará cómo varía el rendimiento del modelo a medida que se ajusta el umbral de decisión. Un clasificador ideal tendría una TPR de 1 y una FPR de 0 en todos los umbrales de decisión, lo que se traduciría en una curva ROC que se acerca al vértice superior izquierdo del gráfico [4].

Por otro lado, el AUC (área bajo la curva ROC) es el área que se encuentra bajo la curva ROC del modelo. Es un número que cuantifica la capacidad de discriminación global del modelo; es decir, cuanto mayor sea el valor del AUC, mejor será el rendimiento del modelo en la clasificación [4].

3. Experimentos

3.1. Submuestreo

En esta sección, se detallan los experimentos realizados para abordar el desbalance en la base de datos y se explican las razones detrás de la elección del método de submuestreo final, Edited Nearest Neighbors (ENN). Comenzando con la evaluación de técnicas de submuestreo, se aborda el desequilibrio en la base de datos, evaluando tres técnicas de submuestreo: ENN, All-KNN y Tomek

Links, sin embargo, después de una evaluación exhaustiva de las técnicas de submuestreo, se seleccionó el método ENN como la solución óptima para abordar el desbalance en la base de datos. Las ventajas clave de ENN incluyen; preservación de información relevante, ya que este método elimina únicamente los ejemplos que se consideran ruidosos o ambiguos, lo que permite mantener la información más relevante para el proceso. Además que cuenta con la flexibilidad de escoger un número de vecinos óptimos, dicho valor fue calculado especialmente [2].

3.2. Modelos de clasificación

3.3. Hardware

Para ejecutar el programa implementado en este proyecto, se requiere una computadora o plataforma que cumpla con los siguientes requisitos mínimos de hardware:

- Memoria RAM: Se recomienda disponer de al menos 4 GB de memoria RAM para ejecutar el programa de manera eficiente.
- CPU: El programa puede ser ejecutado en una amplia variedad de procesadores. No se requiere ninguna característica de procesador específica para ejecutar el programa.
- Espacio en disco: El proyecto y los datos asociados requerirán espacio en disco para su almacenamiento. Se recomienda disponer de suficiente espacio en disco para almacenar los archivos de código y los conjuntos de datos utilizados.

3.4. Software

El software necesario para ejecutar el programa incluye:

- Python: Se requiere tener instalado Python en el sistema. Se recomienda utilizar Python 3.x, aunque el programa también puede ser compatible con versiones anteriores.
- Entorno de Jupyter Notebook: Se debe tener acceso a un entorno de Jupyter Notebook instalado o basado en la web.
- Bibliotecas de Python: El programa utiliza varias bibliotecas de Python, incluyendo pandas, numpy, seaborn, matplotlib, scikit-learn e imbalanced-learn.
- Conjunto de Datos: Se debe tener acceso al conjunto de datos necesario para la ejecución del programa.

4. Resultados

4.1. Metodología 80/20

En esta sección se presentan los resultados obtenidos mediante la aplicación de la metodología 80/20, que consiste en dividir el conjunto de datos en un 80 % destinado al conjunto de entrenamiento y un 20 % para el conjunto de prueba.

4.1.1. Bosque Aleatorio A continuación, se presenta el informe de clasificación obtenido para el clasificador de Bosque Aleatorio:

Clase	Precisión	Recall	F1-Score	Support
Caminar	0.93	0.99	0.96	37626
Arrastrar	0.85	0.03	0.06	2499
Subir escaleras	0.00	0.00	0.00	23
Bajar escaleras	0.00	0.00	0.00	156
Parado	0.90	0.94	0.92	11265
Sentado	1.00	1.00	1.00	11302
Acostado	1.00	1.00	1.00	7457
Accuracy			0.95	70328
Macro Avg	0.67	0.56	0.56	70328
Weighted Avg	0.94	0.95	0.93	70328

Cuadro 3. Informe de Clasificación - Bosque Aleatorio con la partición 80/20

La Figura 1 muestra la matriz de confusión resultante para el método de Bosque Aleatorio.

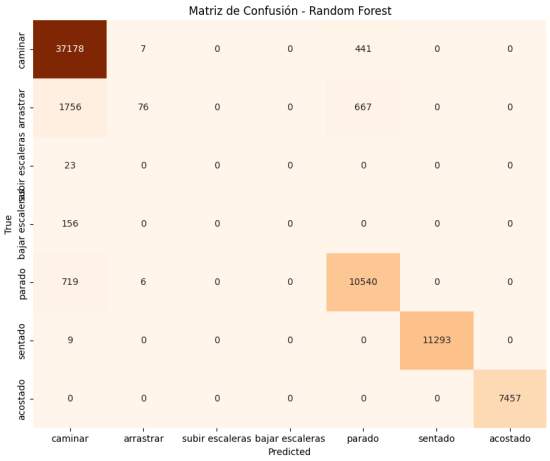

Figura 1. Matriz de Confusión - Bosque Aleatorio (Metodología 80/20)

Además, se presentan las curvas ROC y los valores de AUC para cada clase:

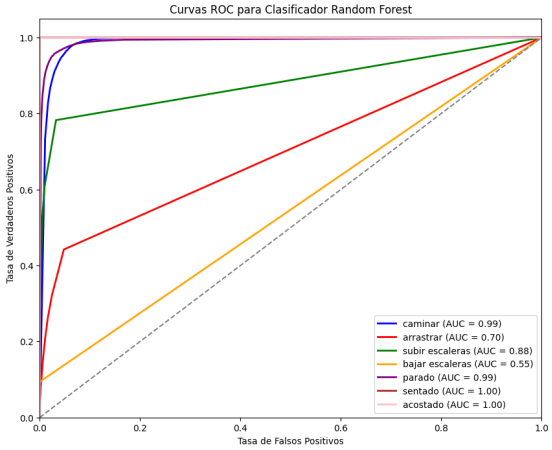

Figura 2. Curvas ROC - Bosque Aleatorio (Metodología 80/20)

Caminar (AUC = 0.99): El AUC de 0.99 indica un rendimiento excepcionalmente alto en la clasificación de la clase caminar. La curva ROC se acerca significativamente al rincón superior izquierdo, lo que sugiere que el modelo tiene una alta tasa de verdaderos positivos (TPR) para una tasa de falsos positivos (FPR) baja.

Arrastrar (AUC = 0.70): Rendimiento aceptable. La curva ROC sugiere que el modelo es capaz de distinguir esta clase, pero con una tasa moderada de FPR en comparación con otras clases.

Subir escaleras (AUC = 0.88): El AUC de 0.86 indica un rendimiento decente en la clasificación de la clase subir escaleras. La curva ROC muestra que el modelo puede distinguir esta clase con éxito, aunque podría haber margen de mejora en términos de reducción de FPR.

Bajar escaleras (AUC = 0.55): Rendimiento bajo, casi aleatorio.

Parado (AUC = 0.99): Rendimiento excepcionalmente alto

Sentado (AUC = 1.00): El AUC de 1.00 también indica un rendimiento perfecto en la clasificación de la clase sentado. Al igual que en "parado", el modelo prácticamente no comete errores en la clasificación de esta clase.

Acostado (AUC = 1.00): El AUC de 1.00 indica un rendimiento perfecto en la clasificación de la clase acostado. Al igual que en las clases anteriores, el modelo prácticamente no comete errores en la clasificación de esta clase.

4.1.2. Árbol de Decisión Continuando con nuestro proceso, se realizó para esta misma metodología el clasificador de Árbol de Decisión.

El informe de clasificación para este método resultó con los siguientes resultados.

Clase	Precisión	Recall	F1-Score	Support
Caminar	0.87	0.94	0.90	37626
Arrastrar	0.18	0.32	0.23	2499
Subir escaleras	0.05	0.04	0.04	23
Bajar escaleras	0.11	0.12	0.11	156
Parado	0.82	0.49	0.62	11265
Sentado	1.00	1.00	1.00	11302
Acostado	1.00	1.00	1.00	7457
Accuracy			0.86	70328
Macro Avg	0.58	0.56	0.56	70328
Weighted Avg	0.87	0.86	0.86	70328

Cuadro 4. Informe de Clasificación - Árbol de Decisión con partición 80/20

En la Figura 3 se muestra el resultado de la matriz de confusión.

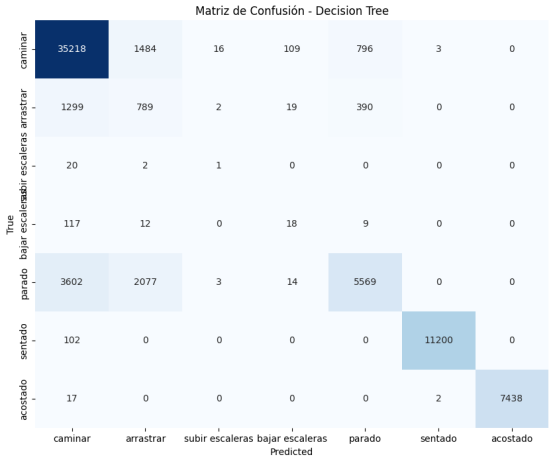

Figura 3. Matriz de confusión - Árbol de decisión con partición 80/20

Además, se presentan las curvas ROC y los valores de AUC para cada clase:

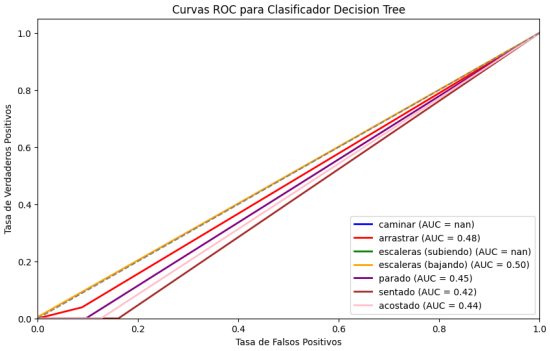

Figura 4. Curvas ROC - Árbol de decisión (Metodología 80/20)

Caminar (AUC = nan): Puede haber un problema en los datos o en la forma en que se calculó el AUC para esta clase.

Arrastrar (AUC = 0.48): La curva ROC muestra que el modelo tiene dificultades para distinguir esta clase y, en su lugar, realiza clasificaciones cercanas al azar.

Subir escaleras (AUC = nan): Al igual que en el primer caso, un valor 'nan' indica que podría haber un problema en la configuración o los datos relacionados con la clase subir escaleras.

Bajar escaleras (AUC = 0.50): Rendimiento similar al azar. El modelo no es efectivo para distinguir esta clase.

Parado (AUC = 0.45): El modelo tiene dificultades para separar esta clase de otras.

Sentado (AUC = 0.42): Rendimiento pobre. El modelo no es efectivo para distinguir esta clase de manera adecuada.

Acostado (AUC = 0.44): El modelo tiene dificultades para separar esta clase de otras.

En general, los valores de AUC cercanos a 1 indican un buen rendimiento en la clasificación de la clase correspondiente, mientras que los valores cercanos a 0.5 o por debajo sugieren un rendimiento deficiente o similar al azar.

4.2. Metodología 50-50

Similar a la metodología anterior, se presentan los resultados obtenidos al dividir el conjunto de datos de 6 pacientes, 3 para el conjunto de entrenamiento, y 3 para el conjunto de prueba.

4.2.1. Bosque Aleatorio A continuación, se presenta el informe de clasificación obtenido para el clasificador de Bosque Aleatorio:

Clase	Precisión	Recall	F1-Score	Support
Caminar	0.70	0.98	0.82	155032
Arrastrar	0.38	0.00	0.00	9536
Subir escaleras	0.00	0.00	0.00	346
Bajar escaleras	0.00	0.00	0.00	314
Parado	0.88	0.31	0.46	79850
Sentado	0.99	0.87	0.93	85760
Acostado	0.74	0.99	0.85	30640
Accuracy			0.78	360478
Macro Avg	0.53	0.45	0.44	360478
Weighted Avg	0.80	0.78	0.75	360478

Cuadro 5. Informe de Clasificación - Bosque Aleatorio con la partición 50-50

La Figura 1 muestra la matriz de confusión resultante para el método de Bosque Aleatorio.

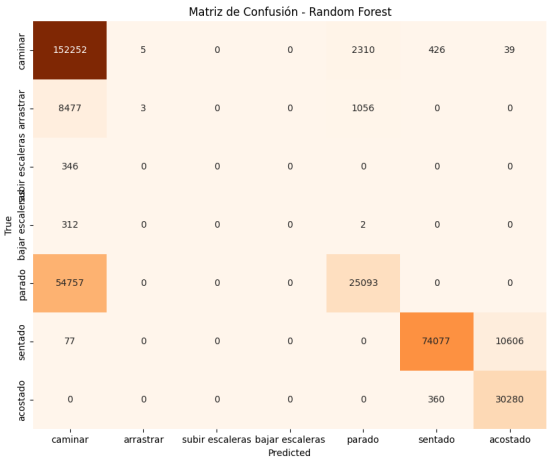

Figura 5. Matriz de Confusión - Bosque Aleatorio (Metodología 50-50)

Curvas ROC y valores AUC para cada clase:

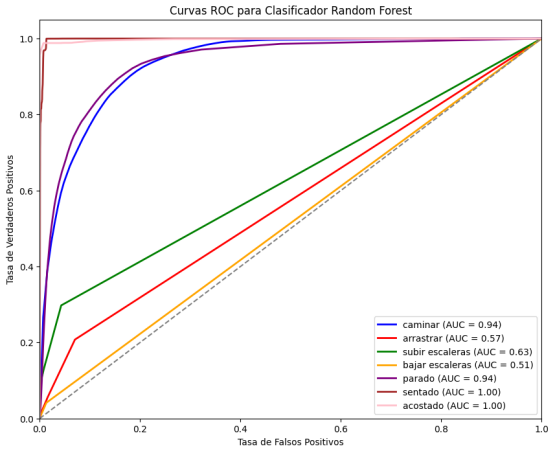

Figura 6. Curvas ROC - Bosque Aleatorio (Metodología 50-50)

Caminar (AUC = 0.94): El AUC de 0.94 indica un rendimiento bastante alto en la clasificación de esta clase. La curva ROC sugiere que el modelo tiene una tasa baastante acertada de verdaderos positivos (TPR) para una tasa de falsos positivos (FPR) baja.

Arrastrar (AUC = 0.57): Rendimiento bajo. La curva ROC sugiere que el modelo es igual de capaz de distinguir o equivocar esta clase.

Subir escaleras (AUC = 0.63): Se tiene un rendimiento medio, pero no confiable.

Bajar escaleras (AUC = 0.51): Rendimiento bajo, casi aleatorio.

Parado (AUC = 0.94): Rendimiento excepcionalmente alto

Sentado (AUC = 1.00): El AUC de 1.00 también indica un rendimiento perfecto en la clasificación de la clase sentado. El modelo no comete errores en la clasificación de esta clase.

Acostado (AUC = 1.00): El AUC de 1.00 indica un rendimiento perfecto en la clasificación de esta clase.

4.2.2. Árbol de Decisión Continuando con el proceso, se realizó para esta misma metodología de 50-50 el clasificador de Árbol de Decisión.

El informe de clasificación arrojó los siguientes resultados.

Clase	Precisión	Recall	F1-Score	Support
Caminar	0.67	0.87	0.76	155032
Arrastrar	0.09	0.13	0.11	9536
Subir escaleras	0.03	0.01	0.01	346
Bajar escaleras	0.00	0.04	0.01	314
Parado	0.60	0.18	0.28	79850
Sentado	0.92	0.98	0.95	84760
Acostado	1.00	0.88	0.93	30640
Accuracy			0.86	360478
Macro Avg	0.58	0.56	0.56	360478
Weighted Avg	0.87	0.86	0.86	360478

Cuadro 6. Informe de Clasificación - Árbol de Decisión con partición 50-50

En la Figura 7 se muestra el resultado de la matriz de confusión.

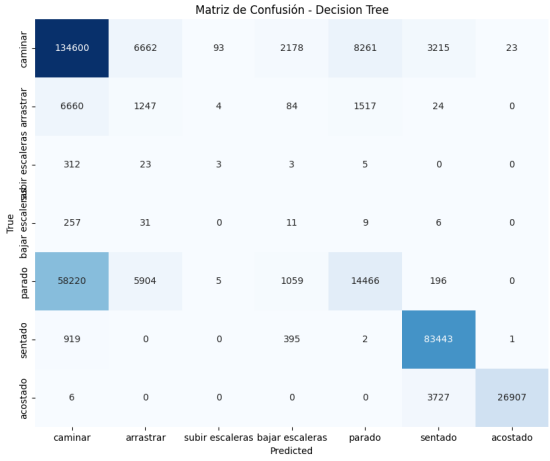

Figura 7. Matriz de confusión - Árbol de decisión con partición 50-50

Además, se presentan las curvas ROC y los valores de AUC para cada clase:

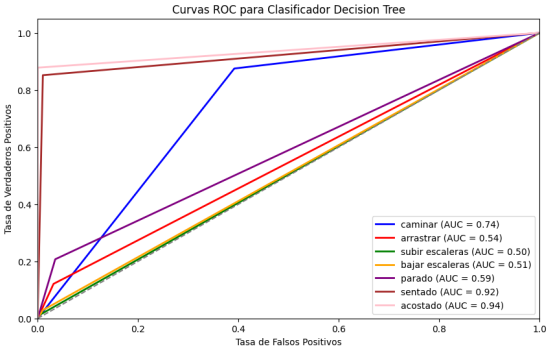

Figura 8. Curvas ROC - Árbol de decisión (Metodología 50-50)

Caminar (AUC = 0.74): El rendimiento es promedio, no es alto, pero tampoco es bajo, indica que el modelo identifica bien la clase, sin embargo, no es muy bueno.

Arrastrar (AUC = 0.54): La curva ROC muestra que el modelo tiene dificultades para distinguir esta clase.

Subir escaleras (AUC = 50): Rendimiento bajo. La curva ROC sugiere que el modelo es igual de capaz de distinguir o equivocar esta clase.

Bajar escaleras (AUC = 0.51): Rendimiento bajo. La curva ROC sugiere que el modelo es igual de capaz de distinguir o equivocar esta clase.

Parado (AUC = 0.59): El modelo tiene dificultades para separar esta clase de las demás, tiene un rendiemiento bastante bajo.

Sentado (AUC = 0.92): El rendimiento del modelo es bastante bueno, muy cercano al 1, lo que indica que el modelo no está teniendo tantas dificultades para distinguir esta clase de otras.

Acostado (AUC = 0.94): El modelo distingue mejor a esta clase que al resto de ellas, tiene el AUC más alto y es la curva ROC que está por encima de las demás.

4.3. Validación cruzada

La última metodología aplicada fue la validación cruzada, una técnica que nos permite evaluar el rendimiento del modelo de manera más robusta al dividir el conjunto de datos en múltiples subconjuntos de entrenamiento y prueba.

Posteriormente, se evaluaron los modelos de clasificación Bosque Aleatorio, K Vecinos más Cercanos (KNN) y Árbol de Decisión. Sin embargo, se observó que el modelo KNN tuvo el rendimiento menos efectivo en la clasificación de las actividades. Por lo tanto, en este informe se presentarán en detalle los resultados obtenidos para los modelos Bosque Aleatorio y Árbol de Decisión, que mostraron un desempeño sobresaliente en la tarea de clasificación.

4.3.1. Bosque Aleatorio A continuación, se muestra el informe de clasificación para Bosque Aleatorio con validación cruzada.

Clase	**Precisión**	**Recall**	**F1-Score**	**Support**
Caminar	1.00	1.00	1.00	25212
Arrastrar	1.00	0.72	0.84	47
Subir escaleras	0.00	0.00	0.00	13
Bajar escaleras	0.00	0.00	0.00	1
Parado	1.00	1.00	1.00	7267
Sentado	1.00	1.00	1.00	8980
Acostado	1.00	1.00	1.00	5989
Accuracy			1.00	47509
Macro Avg	0.71	0.67	0.69	47509
Weighted Avg	1.00	1.00	1.00	47509

Cuadro 7. Informe de Clasificación - Bosque Aleatorio con validación cruzada

El modelo tiene una precisión promedio del 100 %. Esto significa que, en general, clasifica correctamente todas las actividades. Sin embargo, tiene dificultades para detectar las actividades de 'subir escaleras' y 'bajar escaleras', con una precisión y recall del 0 %. El rendimiento es excelente en otras actividades como 'caminar', 'arrastrar', 'parado', 'sentado' y 'acostado'.

La Figura 9 muestra la matriz de confusión resultante de la aplicación del modelo de Bosque Aleatorio con validación cruzada.

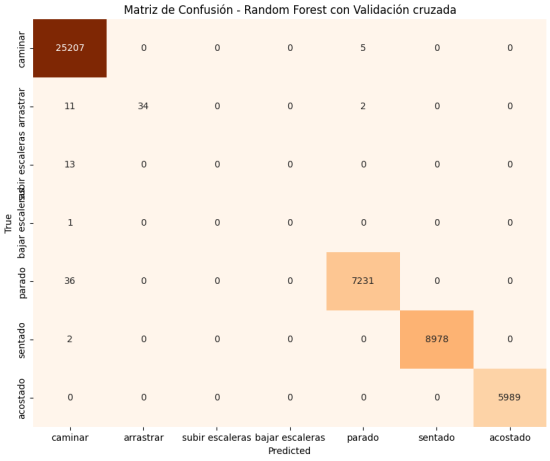

Figura 9. Matriz de Confusión - Bosque Aleatorio con Validación Cruzada

Observamos que la clase 'caminar' ha sido correctamente clasificada en 25,207 instancias, mientras que la clase 'arrastrar' ha sido correctamente clasificada en 34 instancias. Por otro lado, las clases 'subir escaleras' y 'bajar escaleras' no han sido correctamente clasificadas por el modelo en ninguna instancia. Similarmente, la clase 'parado' ha sido correctamente clasificada en 7,231 instancias y la clase 'sentado' en 8,978 instancias. Finalmente, la clase 'acostado' ha sido correctamente clasificada en 5,989 instancias.

A continuación, se presentan las curvas ROC y los valores AUC correspondientes para el modelo de Bosque Aleatorio con validación cruzada.

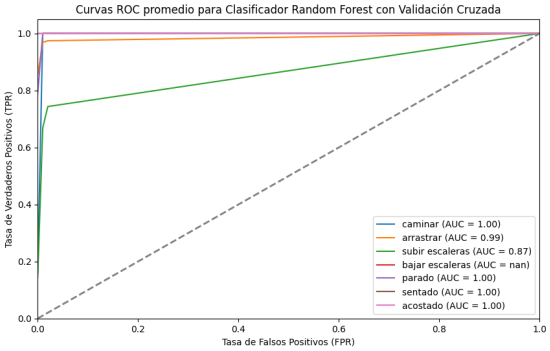

Figura 10. Curvas ROC - Bosque Aleatorio con Validación Cruzada

Caminar (AUC = 1.00): Tasa de verdaderos positivos perfecta y una tasa de falsos positivos nula. Rendimiento excelente en la detección de esta actividad.

Arrastrar (AUC = 0.99): Alto rendimiento con un área bajo la curva de 0.99. Casi todas las instancias de esta actividad se clasifican correctamente.

Subir escaleras (AUC = 0.87): Buen rendimiento, pero no tan alto como en las actividades anteriores. Puede haber algunas instancias mal clasificadas.

Bajar escaleras (AUC = nan): Puede haber un problema en los datos o en la forma en que se calculó el AUC para esta clase.

Parado (AUC = 1.00): Tasa de verdaderos positivos perfecta y una tasa de falsos positivos nula, al igual que en la clase Caminar. Indica un rendimiento excelente en la detección de esta actividad.

Sentado (AUC = 1.00): Al igual que en las clases Parado y Caminar, el clasificador alcanza una tasa de verdaderos positivos perfecta y una tasa de falsos positivos nula.

Acostado (AUC = 1.00): El clasificador también logra una tasa de verdaderos positivos perfecta y una tasa de falsos positivos nula. Nuevamente, muestra un rendimiento excelente.
Se muestra un rendimiento general muy bueno en la mayoría de las clases de actividad, especialmente en Caminar, Parado, Sentado y Acostado.

4.3.2. Árbol de Decisión Continuando con nuestro proceso, se realizó la aplicación del clasificador de Árbol de Decisión utilizando la metodología de validación cruzada. Los resultados obtenidos se detallan a continuación:

Clase	Precisión	Recall	F1-Score	Support
Caminar	1.00	1.00	1.00	25212
Arrastrar	0.62	0.62	0.62	47
Subir escaleras	0.06	0.08	0.07	13
Bajar escaleras	0.00	0.00	0.00	1
Parado	0.99	0.99	0.99	7267
Sentado	1.00	1.00	1.00	8980
Acostado	1.00	1.00	1.00	5989
Accuracy			1.00	47509
Macro Avg	0.67	0.67	0.67	47509
Weighted Avg	1.00	1.00	1.00	47509

Cuadro 8. Informe de Clasificación - Árbol de Decisión con validación cruzada

El modelo tiene una precisión promedio del 100 %. En general, clasifica correctamente todas las actividades. Igualmente, tiene dificultades en las actividades de 'arrastrar' y 'subir escaleras' con una precisión y recall más bajos. Por otra parte, tiene excelente en otras actividades como 'caminar', 'parado', 'sentado' y 'acostado'.
La Figura 11 ilustra la matriz de confusión resultante de la aplicación del modelo de Árbol de Decisión con validación cruzada.

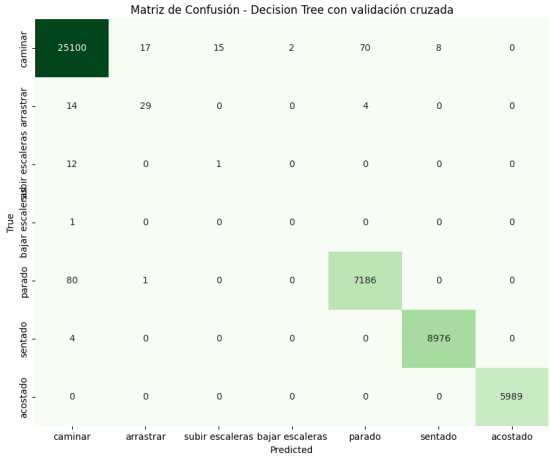

Figura 11. Matriz de confusión - Árbol de Decisión con validación cruzada

Observamos que el modelo logró una alta precisión en las clases 'caminar', 'parado', 'sentado' y 'acostado', con una gran cantidad de predicciones correctas (valores altos en la diagonal principal). Sin embargo, hubo cierta confusión en las clases 'arrastrar' y 'subir escaleras', donde se realizaron algunas predicciones incorrectas (valores fuera de la diagonal principal). Además, el modelo no pudo predecir correctamente las clases 'escaleras (subiendo)' y 'escaleras (bajando)', lo que se refleja en la fila correspondiente a esas clases, donde todos los valores son cero, indicando que no se realizaron predicciones correctas para esas clases. En general, el modelo tuvo un buen rendimiento.

A continuación, se presentan las curvas ROC y los valores AUC correspondientes.

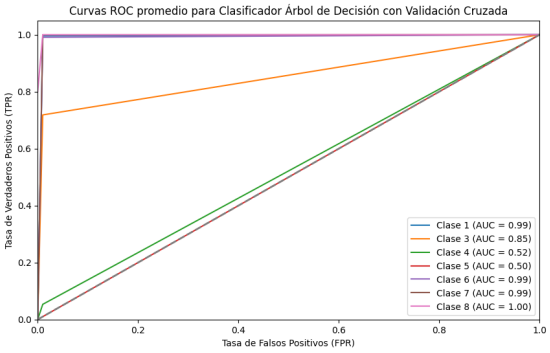

Figura 12. Curvas ROC - Árbol de Decisión con Validación Cruzada

Caminar (AUC = 0.99): El modelo tiene un excelente rendimiento al distinguir la actividad de caminar de otras actividades, con una AUC cercana a 1.

Arrastrar (AUC = 0.85): La capacidad del modelo para identificar la actividad de arrastrar es moderadamente buena, con un AUC de 0.85.

Subir escaleras (AUC = 0.52): Dificultades significativas para diferenciar la actividad de subir escaleras de otras actividades, ya que el AUC es cercano a 0.5, lo que indica un rendimiento pobre.

Bajar escaleras (AUC = 0.50): Similar a la actividad de subir escaleras, el modelo no es efectivo para distinguir la actividad de bajar escaleras, ya que el AUC es de 0.50, lo que representa un rendimiento aleatorio.

Parado (AUC = 0.99): El modelo es altamente preciso al identificar la actividad de estar parado, con una AUC cercana a 1.

Sentado (AUC = 0.99): Altamente preciso al identificar la actividad de estar parado, con una AUC cercana a 1.

Acostado (AUC = 1.00): Clasificación perfecta para la actividad de estar acostado, con un AUC de 1.0, lo que indica un rendimiento óptimo.

El modelo de Árbol de Decisión muestra un alto rendimiento en la mayoría de las clases, excepto en las actividades relacionadas con escaleras, donde tiene dificultades significativas para distinguirlas de otras actividades.

4.4. Modelo seleccionado

Si bien Decision Tree con validación cruzada proporcionó los resultados más sólidos, debemos adapatarnos a las particularidades de cada conjunto de datos y

problema de clasificación, por lo que no podemos inferir que este modelo y esta metodología son los más aptos para cualquier conjunto de datos. Nuestro trabajo está abierto a futuras investigaciones, en las que, de contar con el equipo de cómputo más adecuado, se podrían probar un sin fin de metodologías y modelos para describir a los 15 pacientes que se tienen en el conjunto de datos.

5. Conclusiones individuales

5.0.1. Salvador Mendoza En este proyecto de investigación, he tenido la oportunidad de crear un proceso de solución de un reto complejo. He aplicado y explorado diversas técnicas de clasificación avanzada y análisis exploratorio de datos para abordar el desafío de predecir las actividades físicas en personas mayores. Una de las facetas más intrigantes de este proyecto fue la gestión del desbalance en los datos, donde el uso de submuestreo demostró ser una estrategia efectiva. A lo largo de esta investigación he tenido la satisfacción de crear soluciones efectivas para cada una de las problemáticas planteadas, la evaluación y selección de los modelos de clasificación más precisos, así como la realización de procesos de validación cruzada y otros métodos de validación, me permitieron profundizar en el mundo de la estadística, la probabilidad y el análisis de datos de una manera práctica y enriquecedora.

5.0.2. Alfonso Pineda En este informe de investigación, se abordó la predicción de actividades físicas en personas mayores utilizando técnicas avanzadas de clasificación y análisis exploratorio de datos. Se aplicaron tres metodologías diferentes: una división 80/20, una división 50-50 y validación cruzada. Se evaluaron varios modelos de clasificación, pero se destacaron el Bosque Aleatorio y el Árbol de Decisión como los más efectivos.

En la metodología 80/20, se observó que ambos modelos lograron un rendimiento bastante sólido en la clasificación de la mayoría de las actividades, como 'Caminar', 'Parado', 'Sentado' y 'Acostado', con altos valores de precisión y recall. Sin embargo, hubo dificultades en la clasificación de actividades menos representadas, como 'Arrastrar' y 'Subir escaleras'. Además, el modelo de Árbol de Decisión tuvo un rendimiento inferior en general en comparación con el Bosque Aleatorio.

En la metodología 50-50, se utilizaron conjuntos de datos equilibrados para entrenar y probar los modelos, lo que permitió una mejor evaluación del rendimiento en actividades menos representadas. Se observaron mejoras en la clasificación de 'Arrastrar' y 'Subir escaleras', aunque aún hubo dificultades en la clasificación de 'Bajar escaleras'. En esta metodología igualmente pudimos notar que existió más ambigüedad al momento de predecir la acción de 'Caminar', lo cual atribuimos a la mayor cantidad de pacientes en el entrenado y evaluado del modelo; es decir, creemos que uno o más pacientes pudieron haber ocupado asistencias para caminar, modificando la tendencia en las lecturas de ambos sensores.

En la validación cruzada, ambos modelos demostraron un rendimiento excepcional en la mayoría de las clases, especialmente en 'Caminar', 'Parado', 'Sentado' y 'Acostado'. Sin embargo, continuaron teniendo dificultades en las actividades menos representadas, como 'Arrastrar', 'Subir escaleras' y 'Bajar escaleras'.

En general, los resultados sugieren que los modelos son efectivos para clasificar las actividades más comunes en personas mayores, lo que puede ser útil para monitorear su salud y mejorar su calidad de vida. Sin embargo, se necesitarían más datos y posiblemente estrategias adicionales para mejorar la clasificación de actividades menos representadas. Este trabajo proporciona una base sólida para futuras investigaciones y aplicaciones en el campo de la salud en personas de la tercera edad.

En general, puedo decir que este proyecto de investigación me sirvió para aprender a conocer e interpretar datos crudos, evaluar y comparar distintos modelos de clasificación y crear soluciones aplicables (modelos en producción) con ellos.

5.0.3. Mariana Rincón Para este reto utilizamos el lenguaje de programación Python, así como distintas librerías que nos permitieron trabajar, amoldar, y ajustar a nuestros datos para cada etapa. En el proceso de análisis exploratorio de datos (EDA) del conjunto HAR70+, se llevó a cabo un proceso metódico de exploración y comprensión de los datos utilizando estadísticas descriptivas y visualizaciones. Estas etapas permitieron identificar patrones, tendencias y relaciones importantes entre las variables, contribuyendo así a una base sólida para análisis, preprocesamiento y modelado posteriores.

Posteriormente, se pasó a la etapa de preprocesamiento, en donde se balancearon los datos, se definieron las tres metodologías para la etapa de modelación y se concluyó que los modelos de clasificación eran los más adecuados y convenientes para nuestra base de datos. Esta etapa fue sumamente importante para nosotros ya que nos hizo darnos cuenta que trabajar con un solo conjunto de datos (concatendado o no), no sería suficiente para describir correctamente a los datos y por consecuente, dar una solución a la problemática de identificar correctamente a cada clase.

En la etapa de modelado se implementó para cada metodología, 5 clasificadores distintos; esto con el objetivo de ver no sólo que modelo era el más adecuado, sino también, qué metodología. Además se añadieron gráficos para poder obtener una interpretación más visual del modelado de nuestros datos como: matrices de confusión, curvas ROC y valores AUC. Finalmente, comparando y analizando los reportes de clasificación y sus distintas métricas como: exactitud, precisión, recall y f1-score, además de los gráficos antes mencionados, se pudo llegar a una conlcusión en donde observamos un mejor desempeño en la metodología de validación cruzada, en cuanto al modelo, el que arrojó los mejores resultados en cada una de las categorías de evaluación planteadas, fue el de Árbol de Decisión, es por ello que decidimos escoger a este como nuestro modelo final.

Puedo concluir personalmente, que este reto me sirvió para reforzar mis habilidades en ciencia de datos, nos esforzamos en poder obtener una gran variedad de resultados de la cual pudieramos escoger al más óptimo, sin embargo, aún el que nosotros consideramos el mejor, está abierto a futuras investigaciones. Puedo decir que siempre hay una puerta abierta para analizar y modelar los datos de mil maneras y la única forma de hacerlo, es a prueba y error, comparando y viendo todas las opciones posibles.

5.0.4. Karla González En esta actividad, exploramos diversas metodologías de evaluación de modelos de aprendizaje automático, como 80/20, 50/50 y validación cruzada. Esta parte fue muy buena para mí ya que reforcé mis capacidades de análisis y comporación entre las diferentes técnicas al interpretar métricas cruciales como presición, recall, F1-Score, y el área bajo la curva ROC (AUC). Fue mi primer acercamiento en el mundo de las curvas ROC y AUC, que son herramientas muy versátiles y fundamentales que nos ayudaron a evaluar el desempeño de clasificadores. Finalmente, destaco la importancia de la selección de modelos en el éxito de un proyecto de aprendizaje automático, ya que son conocimientos y habilidades esenciales para abordar desafíos de clasificación.

Referencias

1. Hoyos Osorio, J. K. (2019). Metodología de clasificación de datos desbalanceados basado en métodos de submuestreo.
2. Guan, D., Yuan, W., Lee, Y. K., Lee, S. (2009). Nearest neighbor editing aided by unlabeled data. Information Sciences, 179(13), 2273-2282.
3. Narkhede, S. (2018). Understanding auc-roc curve. Towards Data Science, 26(1), 220-227.
4. Agarwal, S., Graepel, T., Herbrich, R., Har-Peled, S., Roth, D., Jordan, M. I. (2005). Generalization Bounds for the Area Under the ROC Curve. Journal of Machine Learning Research, 6(4).