Journal BioFab
https://www.biofab.com.pe/jour

Journal Biofab
Octubre 2022
Volumen 1 / No.1
ISSN: 2955-8654
pp. 143-162
https://doi.org/10.53673/jb.v1i1.5

Predicción de diabetes mellitus tipo 2 utilizando atributos médicos del Policlínico Leo SAC de San Juan de Lurigancho mediante el enfoque de Machine Learning

Prediction of type 2 diabetes mellitus using medical attributes of the Leo SAC Polyclinic in San Juan de Lurigancho through the Machine Learning approach

Previsão de diabetes mellitus tipo 2 usando atributos médicos da Policlínica Leo SAC em San Juan de Lurigancho através da abordagem de Machine Learning

ARTÍCULO GENERAL

Jaime Yelsin Rosales Malpartida
jrosalesm@uni.pe
https://orcid.org/0000-0003-4574-5172
Universidad Nacional de Ingeniería, Lima, Perú

Recibido 16 de Agosto 2022 | Arbitrado y aceptado 29 de Octubre 2022 | Publicado en 06 Diciembre 2022

RESUMEN

Las muertes por diabetes aumentaron en un 70% a nivel mundial entre los años 2000 y 2019, situándose entre las diez primeras causas de mortalidad. Fue la causa directa de 4.2 millones de muertes en 2019, además la cantidad de adultos (entre 20-79 años) que vivían con diabetes era de aproximadamente 463 millones y se espera que aumente a 700 millones en 2045. La diabetes es una enfermedad grave para la salud debido a la presencia de altos niveles de glucosa en el cuerpo humano, por lo que un diagnóstico temprano ayudará a tratarla y prevenir sus complicaciones. La necesidad de una manera fácil y rápida de diagnosticar la diabetes es crucial. Es esencial evaluar los impactos de los modelos de Machine Learning elegidos utilizando atributos médicos, por ello desarrollamos y probamos 13 métodos de Machine Learning de modelos clásicos, redes neuronales y modelos ensemble para predecir la diabetes mellitus tipo 2 en pacientes mayores de edad. el conjunto de datos se obtuvo del Policlínico Leo SAC de San Juan de Lurigancho. Los modelos con hiperparámetros óptimos se evaluaron mediante el accuracy, precisión, sensibilidad, especificidad, F1-score, tasa de clasificación errónea y el AUC en el conjunto de datos de entrenamiento y de prueba. En las siete medidas de rendimiento, el modelo que superó consistentemente a los demás fue LightGBM. Este estudio demuestra que la elección de modelos de Machine Learning tiene un efecto en los resultados de predicción.

Palabras clave: predicción de diabetes, Machine Learning, datos del Policlínico Leo SAC de San Juan de Lurigancho.

ABSTRACT

Deaths from diabetes increased by 70% globally between 2000 and 2019, ranking among the top ten causes of mortality. It was the direct cause of 4.2 million deaths in 2019, and the number of adults (ages 20-79) living with diabetes was approximately 463 million and is expected to rise to 700 million by 2045. Diabetes is a serious disease for health due to the presence of high levels of glucose in the human body, so an early diagnosis will help treat it and prevent its complications. The need for an easy and quick way to diagnose diabetes is crucial. It is essential to evaluate the impacts of the chosen Machine Learning models using medical attributes, therefore we developed and tested 13 Machine Learning methods of classical models, neural networks and ensemble models to predict type 2 diabetes mellitus in elderly patients. The data set was obtained from the Leo SAC Polyclinic in San Juan de Lurigancho. Optimal hyperparameter models were evaluated using accuracy, precision, sensitivity, specificity, F1-score, misclassification rate, and AUC on the training and test dataset. Across all seven performance measures, the model that consistently outperformed the others was LightGBM. This study demonstrates that the choice of machine learning models has an effect on the prediction results.

Keywords: diabetes prediction, Machine Learning, data from the Leo SAC Polyclinic in San Juan de Lurigancho.

RESUMO

As mortes por diabetes aumentaram 70% em todo o mundo entre 2000 e 2019, classificando-se entre as dez principais causas de mortalidade. Foi a causa direta de 4,2 milhões de mortes em 2019, e o número de adultos (20 a 79 anos) vivendo com diabetes foi de aproximadamente 463 milhões e deve aumentar para 700 milhões até 2045. O diabetes é uma doença grave para a saúde devido à a presença de altos níveis de glicose no corpo humano, portanto, um diagnóstico precoce ajudará a tratá-lo e prevenir suas complicações. A necessidade de uma maneira fácil e rápida de diagnosticar o diabetes é crucial. É essencial avaliar os impactos dos modelos de Machine Learning escolhidos usando atributos médicos, por isso desenvolvemos e testamos 13 métodos de Machine Learning de modelos clássicos, redes neurais e modelos ensemble para prever diabetes mellitus tipo 2 em pacientes idosos. O conjunto de dados foi obtido da Policlínica Leo SAC em San Juan de Lurigancho. Modelos de hiperparâmetros ideais foram avaliados usando exatidão, precisão, sensibilidade, especificidade, pontuação F1, taxa de classificação incorreta e AUC no conjunto de dados de treinamento e teste. Em todas as sete medidas de desempenho, o modelo que superou consistentemente os outros foi o LightGBM. Este estudo demonstra que a escolha de modelos de aprendizado de máquina tem efeito nos resultados de previsão.

Palavras-chave: previsão de diabetes, Machine Learning, dados da Policlínica Leo SAC em San Juan de Lurigancho.

1. Introduccion

La diabetes es una condición de salud crónica en la cual la capacidad de producir insulina disminuye, lo que resulta en serios problemas de salud como enfermedades del corazón, enfermedades renales, pérdida de la visión, etc. La diabetes mellitus tipo 2 es un trastorno metabólico que se caracteriza por hiperglucemia (nivel alto de azúcar en la sangre) en el contexto de resistencia a la insulina y falta relativa de insulina. El aumento de casos de prediabetes es un problema mundial que supondrá más cargas para la atención sanitaria en un futuro próximo. A pesar de esto, la prediabetes lo hará completamente inconsciente sin dar ninguna señal y gradualmente conduce a la diabetes.

Según el Centro para el Control y la Prevención de Enfermedades (2019), hacer cambios en la dieta y el ejercicio físico adecuado ralentizará el progreso de entrar en diabetes tipo 2. La capacidad de la inteligencia artificial y los algoritmos de Machine Learning para analizar conjuntos de datos complejos ayuda a los médicos en la predicción temprana de enfermedades. Esto también ayuda en la atención avanzada de los pacientes y mejora los resultados de la atención médica. La máxima utilización de la IA para las decisiones clínicas, la puntuación de riesgos y las alertas tempranas son las áreas más prometedoras del desarrollo del análisis de datos. Muchos investigadores han propuesto diferentes métodos de Machine Learning y Deep Learning para la clasificación y predicción de la diabetes temprana. Según Kristeen Cherney (2018) existe la probabilidad de desarrollar diabetes tipo 2 a partir de los 45 años, pero en algunos casos las personas padecen diabetes sin siquiera saber que padecen síntomas prediabéticos, por lo tanto, existe una gran variación entre la edad y el diagnóstico de diabetes.

Se han realizado varios enfoques para la clasificación y la predicción temprana de la diabetes utilizando métodos de inteligencia artificial, Machine Learning y Deep Learning. Rajput, M. R., & Khedgikar, S. S. (2022) mediante el análisis de diferentes atributos médicos de los datos de Mendeley predicen la diabetes utilizando cinco tipos diferentes de algoritmos de Machine Learning, mostrando que los algoritmos de aumento de gradiente estocástico y árbol de decisión superaron el rendimiento y lograron una mayor precisión en comparación de random forest, regresión logística multinomial y Naive Bayes. May, O. A. C. et al. (2018) procesaron datos de 768

pacientes para apoyar en la predicción de diabetes de las personas y mediante las técnicas de Machine Learning y sistemas expertos con aprendizaje supervisado para generar árboles de decisión, así como el análisis de resultados del algoritmo de predicción J48, con las herramientas BigML y Weka, respectivamente. ALSHARİ, H., & ODABAS, A (2016) utilizando algoritmos modernos de Machine Learning como XGBoost, LightGBM, CatBoost y redes neuronales artificiales para predecir la diabetes en función del comportamiento de salud del paciente mostrando que el algoritmo XGBoost funciona mejor con una puntuación de validación cruzada (10 veces) de 0,864 y una precisión general del 87,7 % para el conjunto de datos de validación y del 84,96 % para el conjunto de datos de prueba. Singh, et al., (2017) también trabajaron en el conjunto de datos de Pima. Han utilizado una técnica de selección de características basada en correlación para eliminar características irrelevantes. Emplearon el perceptrón multicapa (MLP) basado en funciones, Naive Bayes basado en probabilidades, algoritmos de random forest basados en árboles de decisión. Obtuvieron una precisión de predicción del 79,69 % con el método Naive Bayes. Mitushi Soni y Sunita Varma (2020) experimentaron con el conjunto de datos de diabetes de los indios pima con atributos de 768 pacientes. Desarrollaron un modelo para predecir la diabetes utilizando varias técnicas de Machine Learning como SVM, KNN, árbol de decisión, random forest, regresión logística y Gradient Boosting. Lograron una precisión de clasificación del 77 %. Nnamoko, N et al. (2018) utilizaron la capacidad del método de aprendizaje ensemble en el conjunto de datos de diabetes de UCI, y se utilizó un metaclasificador para agregar los resultados obtenidos de los clasificadores individuales. Produjeron una precisión del 83 % con la selección de subconjuntos de características. Barakat, N., Bradley, AP y Barakat, MNH (2010) emplearon un modelo híbrido basado en SVM para el diagnóstico y la predicción de la diabetes. Trabajaron en un conjunto de datos que constaba de 3014 pacientes con 11 atributos diferentes por paciente. Lograron una precisión de predicción del 94 %, una sensibilidad del 93 % y una especificidad del 94 %. Extrajeron reglas por SQR ex-SVM para el diagnóstico de diabetes. Nai-arun, N. y Moungmai, R. (2015) experimentaron con un conjunto de datos de 30 122 pacientes que constaba de 12 atributos por paciente. Implementaron una aplicación web utilizando redes neuronales artificiales, árboles de decisión, regresión logística, Naive Bayes y algoritmo de random forest. Esta aplicación clasifica los datos de los pacientes en diabetes o grupo normal. Consiguieron una precisión del 85,55 % con el random forest. Joshi, TN y Chawan, PPM (2018) implementaron un modelo de predicción temprana de

diabetes mediante regresión logística, SVM y redes neuronales artificiales. Los autores afirmaron que obtuvieron una mayor precisión utilizando estas técnicas. Sisodia, D. y Sisodia, DS (2018) realizaron experimentos en el conjunto de datos PIDD utilizando Naive Bayes, SVM y árbol de decisión. Naive Bayes superó con una precisión del 76,30 %. Majji, R. y Bramaramba, R. (2018) crearon un sitio web que da la probabilidad de desarrollar diabetes en un futuro próximo en función de la puntuación de riesgo. Los diferentes parámetros utilizados para el cálculo del riesgo incluyeron la edad, el tabaquismo, el uso de esteroides, el IMC (índice de masa corporal), el origen étnico, etc. Han utilizado la herramienta WEKA para la clasificación. Shetty, D. et al. (2017) desarrollaron un sistema inteligente de predicción diabética utilizando algoritmos KNN y bayesianos. Chowdary, PBK y Kumar, DRU (2021) propone un enfoque para mejorar la predicción de la diabetes utilizando técnicas de Deep Learning, basado en la memoria convolucional a largo plazo (CLSTM) en la base de datos de diabetes de los indios Pima (PIDD) logrando una precisión del 95.6%. Yahyaoui, A., et al. (2019) al comparar la CNN de Deep Learning con los enfoques de Machine Learning en la base de datos de diabetes de los indios Pima (PIDD), los autores obtuvieron una precisión del 76.81 % con CNN, que es bastante menor que la precisión del 83.67 % con el método Random Forest.

2. Materiales y métodos

2.1. Fuente de datos

El conjunto de datos proviene del Policlínico Leo SAC de San Juan de Lurigancho en Perú y contiene diferentes atributos médicos de 1000 pacientes mayores de edad. La tabla 1 muestra los detalles sobre este conjunto de datos. Se debe hacer una selección cuidadosa de estos atributos (o características), ya que cualquier atributo irrelevante puede inducir a error en los resultados.

N°	Atributo	Descripción
1	Sexo	Género del paciente (Masculino o Femenino)
2	Edad	Edad del paciente en años
3	TGO /AST	Aspartato-aminotransferasa en UI/L
4	TGP/ALT	Alanina-aminotransferasa en UI/L
5	Talla	Tamaño del paciente desde la coronilla de la cabeza hasta los pies (talones)
6	Peso	Peso del paciente en Kg
7	IMC	Índice de masa corporal en kg/altura (m^2)
8	HDL	Lipoproteínas de alta densidad en mg/dL
9	LDL	Lipoproteínas de baja densidad en mg/dL
10	CT	Colesterol total en mg/dL
11	TG	Triglicéridos en mg/dL
12	DM2	Diabetes mellitus tipo 2, No diabetes mellitus tipo 2

Tabla 1. Descripción de los atributos médicos.

2.2. Metodología

En esta sección se explicará la metodología utilizada para desarrollar y evaluar los diferentes modelos de Machine Learning para la predicción de diabetes mellitus tipo 2 en pacientes mayores de edad utilizando los atributos médicos explicados. Los procesos generales involucrados en este estudio se representan en la Figura 1. En las secciones siguientes se proporciona una explicación detallada de cada proceso.

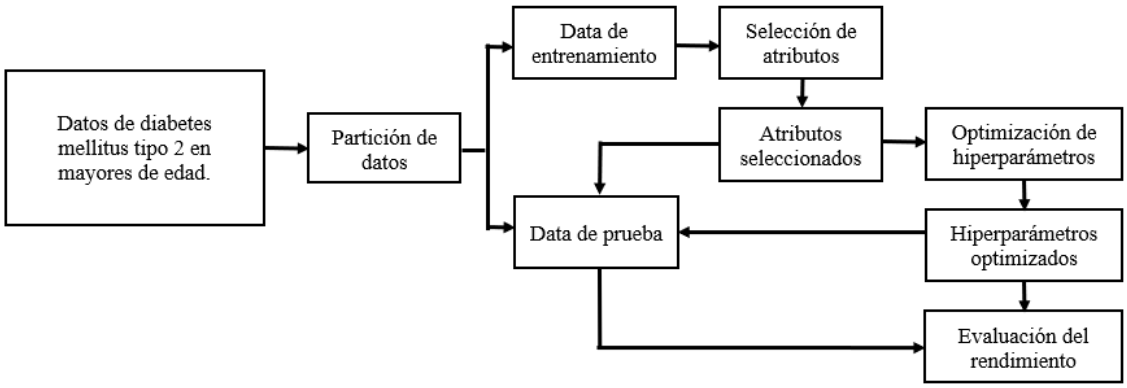

Figura 1. Procesos generales involucrados en el estudio.

2.2.1. Particionamiento de datos

En el sistema propuesto, se implementan los algoritmos de Machine Learning de clasificación supervisada para entrenar un modelo de predicción de diabetes mellitus tipo 2. La implementación se realiza en el software libre de Python. Para el conjunto de datos, la variable objetivo es "Y = DM2" y los otros atributos independientes se almacenan en X. Luego, X e Y se dividieron en el 80% para el entrenamiento y el 20% para la prueba. Para hacer esto, se utiliza el método train_test_split() de la selección del

modelo que proporciona la biblioteca Sklearn de Machine Learning. El conjunto de entrenamiento se utiliza para entrenar nuestro modelo. Y la evaluación de la precisión se ha realizado utilizando el conjunto de prueba, que son datos no vistos para el modelo entrenado, por Pedregosa et al. (2011).

2.2.2. Procedimiento de selección de características

Según Wang, S., Tang, J., & Liu, H. (2017), Jiang, S. Y., & Wang, L. X. (2016) las funciones irrelevantes o redundantes confunden el algoritmo de Machine Learning, lo que genera resultados de aprendizaje y minería deficientes. Hsu, H. H., & Hsieh, C. W. (2010) menciona que la selección de atributos es beneficiosa para mejorar la eficiencia y la previsibilidad del aprendizaje al eliminar información innecesaria o duplicada. En nuestro estudio utilizamos el coeficiente de correlación para determinar la dependencia de los atributos y eliminar los atributos redundantes. Los atributos de alta correlación (ya sea positiva o negativa) son más linealmente dependientes y, por lo tanto, tienen aproximadamente la misma influencia en la variable dependiente. Cuando dos atributos tienen una fuerte correlación, una de ellas puede descartarse. Los valores más cercanos a cero indican que no existe una relación lineal entre los dos atributos, mientras que los valores más cercanos a uno sugieren relaciones lineales sólidas, donde +1 indica que los dos atributos están correlacionados positivamente y -1 indica que están negativamente correlacionados.

2.2.3. Modelos de Machine Learning

En el presente trabajo, se desarrolla y se ajusta 13 modelos de Machine Learning de tres categorías: modelo clásico (regresión logística, árbol de decisión, Naive Bayes, KNN y SVM), red neuronal (perceptrón multicapa), y modelo ensemble (Random Forest, AdaBoost, LogitBoost, Gradient Boosting, XGBoost, LightGBM, y CatBoost) para predecir la diabetes mellitus tipo 2. Los diferentes modelos de Machine Learning tienen diferentes hiperparámetros, y es esencial ajustar el correcto para la salida. En consecuencia, dependiendo del conjunto de datos, los hiperparámetros óptimos pueden variar. Hay varios enfoques para el ajuste de hiperparámetros como el Manual Tuning, Grid Search y Randomized Search según Amin, MM. et al. (2021). En este este estudio los hiperparámetros de los modelos se establecieron durante el entrenamiento mediante el enfoque de Grid search.

- Regresión logística

Este es el modelo de Machine Learning más básico y ampliamente utilizado para la clasificación binaria, que puede extenderse fácilmente a problemas de clasificación de etiquetas múltiples. La técnica de regresión logística utiliza la función sigmoidea para construir un modelo de regresión que predice la posibilidad de que una entrada pertenezca a una categoría particular.

- Árbol de decisión

Según Breiman, L. et al. (2017) esta técnica separa repetidamente el conjunto de datos de acuerdo con un criterio de separación de datos óptimo, lo que da como resultado una estructura similar a un árbol. Los criterios de división más populares utilizados incluyen la ganancia de información, el índice de Gini y la relación de ganancia. La división tiene como objetivo crear nodos puros, es decir, reducir la impureza de un nodo.

- AdaBoost

AdaBoost o Adaptive Boosting es el primer algoritmo de impulso práctico por Freund, Y., & Schapire, R. E. (1997). Este es un algoritmo de ML de conjunto en el que los pesos se asignan de forma adaptativa, con mayores pesos asignados a instancias clasificadas incorrectamente. Cuando el posterior aprendizaje crece a partir de los aprendizajes previamente desarrollados, los aprendizajes débiles se transforman en aprendizajes fuertes.

- Random Forest

Random Forest es un modelo de aprendizaje de ensemble basado en árboles de decisión compuesto por múltiples árboles de decisión. Cada Árbol de Decisión en el Random Forest eventualmente producirá un nodo hoja. Random Forest hace predicciones basadas en la salida elegida por la mayoría de los nodos hoja del árbol de decisiones.

- LogitBoost

LogitBoost es una variación de AdaBoost. El método LogitBoost se desarrolló como una alternativa a AdaBoost para resolver las deficiencias de AdaBoost en el tratamiento del ruido y los valores atípicos.

- K-vecino más cercano (KNN)

Un algoritmo KNN asume que los objetos similares están cerca unos de otros. La similitud se expresa en KNN determinando la distancia entre dos puntos en un gráfico.

La clasificación de un punto de datos se basa en el voto mayoritario de su K-vecino más cercano en una función de distancia.

- Naive Bayes

Naive Bayes es un algoritmo probabilístico basado en el teorema de Bayes. Afirma que la presencia de una característica en una clase no influye en la presencia de ninguna otra característica.

- Máquina de vectores de soporte (SVM)

El método SVM clasifica los datos mediante la construcción de un hiperplano multidimensional que separa mejor dos clases al encontrar el margen máximo entre dos grupos de datos. Este enfoque logra un alto nivel de discriminación al cambiar el espacio de entrada en un espacio multidimensional mediante el uso de funciones no lineales únicas denominadas kernel.

- Perceptrón multicapa (MLP)

Una MLP es una red neuronal artificial (ANN) feedforward que tiene capas de entrada, ocultas y de salida. La capa de entrada acepta señales, mientras que la capa de salida las clasifica o las predice. Hay un número arbitrario de capas ocultas entre las capas de entrada y salida. Estas capas ocultas son el verdadero motor computacional de MLP. La función de activación no lineal se utiliza en las capas ocultas y la capa de salida. El procedimiento de retropropagación se utiliza para entrenar modelos MLP.

- Gradient Boosting

Este es un modelo de aprendizaje ensemble que hace predicciones al combinar muchos modelos de aprendizaje débiles, a menudo árboles de decisión. Gradient Boosting funciona reduciendo iterativamente la función de pérdida seleccionando una función que apunta a un gradiente negativo, es decir, una hipótesis débil.

- XGBoost

Según Chen, T., & Guestrin, C. (2016) XGBoost es la abreviatura de Extreme Gradient Boosting desarrollada. Es un modelo de Machine Learning de conjunto basado en el Árbol de decisiones que aprovecha el marco de Gradient Boosting. Usando impulso paralelo, se agrega un nuevo modelo de árbol de decisiones para compensar las debilidades del modelo anterior. Está diseñado para la velocidad y el rendimiento.

- LightGBM

Según Ke, G. et al. (2017) LightGBM, o Light Gradient Boosting Machine, es un marco de Gradient Boosting comparable a XGBoost que emplea enfoques de aprendizaje Decision Tree. Microsoft lo diseñó en 2017 para aumentar la velocidad. Los atributos se ordenan y categorizan en contenedores utilizando un método de aprendizaje de árbol de decisiones basado en histogramas, y las hojas se cultivan en forma sabia, lo que produce una mayor eficiencia y ventajas en el uso de la memoria en comparación con XGBoost.

- CatBoost

Según Prokhorenkova, L. et al. (2018) el modelo CatBoost que es la abreviatura de Category Boosting, al igual que XGBoost y LightGBM, se basa en un marco Gradient Boosting que utiliza la técnica de aprendizaje Decision Tree. Desarrollado en 2017, CatBoost intenta resolver características categóricas mediante técnicas de permutación.

2.2.4. Métricas de rendimiento

Las predicciones de los modelos desarrollados en este estudio pueden generar cuatro posibles resultados: Verdadero Positivo (TP), Verdadero Negativo (TN), Falso Positivo (FP) y Falso Negativo (FN). Los individuos positivos en nuestro estudio tenían diabetes mellitus tipo 2, mientras que los pacientes negativos no lo tenían. TP y TN son predicciones correctas. Los resultados de FP son predicciones positivas cuando en realidad son negativas. Por otro lado, los resultados de FN son predicciones que son negativas cuando en realidad son positivas. Evaluamos los modelos de predicción con las siguientes métricas de rendimiento:

- Sensibilidad, o Recall o tasa de TP: esta métrica indica la proporción de verdaderos positivos pronosticados de todos los positivos en un conjunto de datos:

$$Sensibilidad = \frac{TP}{TP+FN} \quad (1)$$

- Especificidad o tasa de falsos negativos: este es el número de casos negativos que el algoritmo identifica correctamente:

$$Especificidad = \frac{TN}{TN+FP} \quad (2)$$

- Accuracy: esta es una métrica utilizada para determinar cuántas predicciones correctas produjo un modelo a través de todo el conjunto de datos de prueba:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

- Precisión: esto indica la exactitud de la predicción correcta:

$$Precisión = \frac{TP}{TP+FP} \quad (4)$$

- F1-score: esta métrica mide la precisión del modelo en función de la sensibilidad y la precisión. Un valor más alto de F1-score indica que el modelo es más preciso:

$$F1 - score = 2 \times \frac{Precisión \times Recall}{Precisión + Recall} \quad (5)$$

- Tasa de clasificación errónea o error de clasificación (Misclassification rate): un indicador de rendimiento que indica la proporción de predicciones incorrectas sin diferenciar entre predicciones positivas y negativas:

$$Misclassification\ rate = 1 - Accuracy \quad (6)$$

- El área bajo la curva característica operativa del receptor (AUC): se utiliza para evaluar la precisión de la predicción del modelo. Esta estadística evalúa la capacidad del algoritmo para distinguir entre pacientes con diabetes mellitus tipo 2 o sin diabetes mellitus tipo 2.

3. Resultados

3.1. Selección de atributos

El conjunto de datos original consta de 11 atributos: sexo, edad, aspartato-aminotransferasa (TGO/AST), alanina-aminotransferasa (TGP/ALT), talla, peso, índice de masa corporal (IMC), lipoproteínas de alta densidad (HDL), lipoproteínas de baja densidad (LDL), colesterol total (CT) y triglicéridos (TG). La Figura 2 ilustra los coeficientes de correlación de los atributos mediante el uso de un mapa de calor. Se observa que aspartato-aminotransferasa y alanina-aminotransferasa tienen una dependencia muy alta aproximándose a la unidad, por lo tanto, podemos eliminar el aspartato-aminotransferasa del conjunto de datos y vemos nuevamente el mapa de calor de los coeficientes de correlación resultante en la Figura 3. Luego los siguientes atributos fueron elegidos por el proceso de selección de atributos: sexo, edad, alanina-aminotransferasa (TGP/ALT), talla, peso, índice de masa corporal (IMC), lipoproteínas de alta densidad (HDL), lipoproteínas de baja densidad (LDL), colesterol total (CT) y triglicéridos (TG).

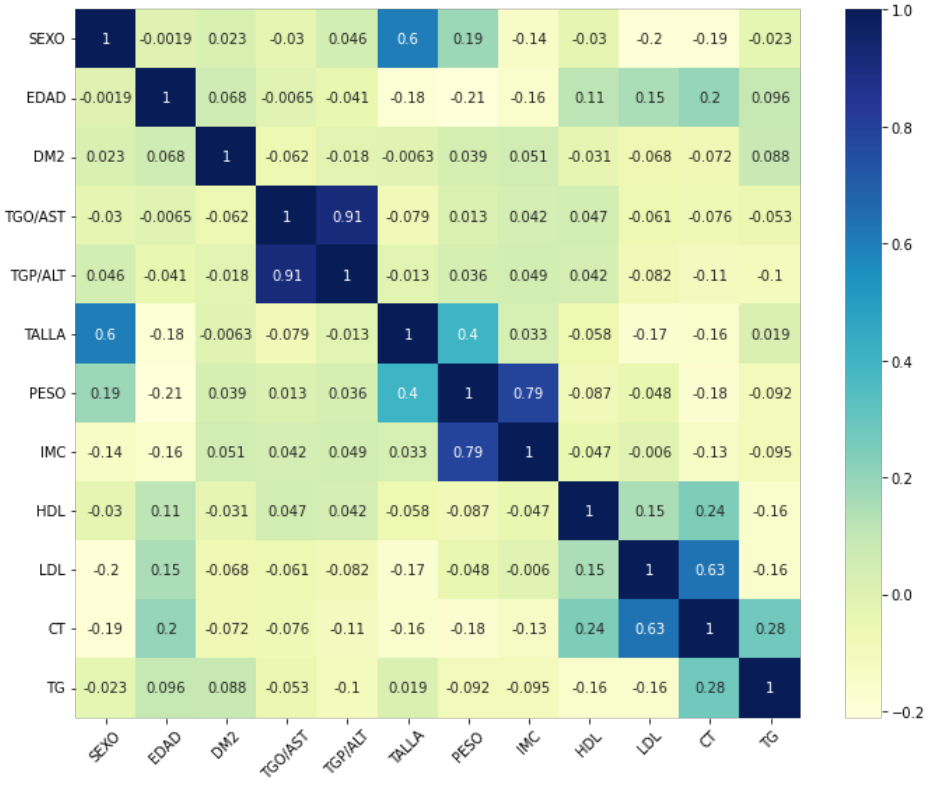

Figura 2. Mapa de calor que muestra el coeficiente de correlación de los 11 atributos y la variable objetivo: sexo, edad, TGO/AST, TGP/ALT, talla, peso, IMC, HDL, LDL, CT, TG y DM2.

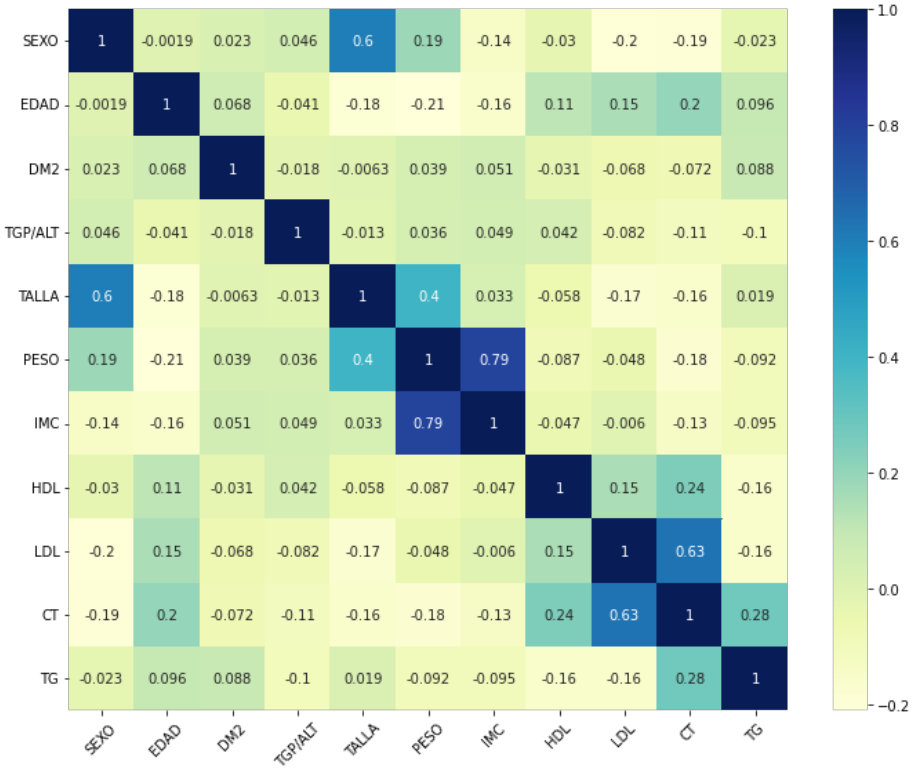

Figura 3. Mapa de calor que muestra el coeficiente de correlación de los 10 atributos seleccionados y la variable objetivo: sexo, edad, TGP/ALT, talla, peso, IMC, HDL, LDL, CT, TG y DM2.

3.2. Modelos de Machine Learning

Para adaptar los modelos de Machine Learning al conjunto de datos se deben configurar los hiperparámetros. En la mayoría de los casos, los efectos de los hiperparámetros en un modelo se comprenden bien. La tarea de seleccionar la colección adecuada de hiperparámetros y combinaciones de hiperparámetros que interactúan para el conjunto de datos, por otro lado, es difícil. En este trabajo, el procedimiento de Grid search se usa para explorar objetivamente múltiples valores para los hiperparámetros del modelo de Machine Learning y elegir un subconjunto que dé como resultado un modelo que logre el mayor rendimiento en el conjunto de datos utilizado. El Grid search funciona realizando una búsqueda completa en un subconjunto específico del espacio de hiperparámetros del algoritmo de entrenamiento. La Tabla 2 presenta los hiperparámetros óptimos descubiertos utilizando el enfoque Grid search para cada modelo.

Modelo	Hiperparámetros
Regresión Logística	C: 1.0, penalty: l2, solver: lbfgs
Árbol de decisión	max_depth: 3, criterion='entropy', min_samples_leaf: 1, min_samples_split: 2
Adaboost	learning_rate: 1, n_estimator: 100
Random Forest	max_depth: 5, min_samples_leaf: 1, n_estimator: 60
LogitBoost	C: 1.0
KNN	n_neighbors: 7
Naive Bayes	var_smoothing: $1.056x10^{-7}$
SVM	C: 1.0, gamma: scale, kernel: rbf
MLP	activation: sigmoid, alpha: 0.0001, hidden_layer_sizes: 4, optimizer: adam
Gradient Boosting	learning_rate: 0.1, n_estimators: 100
LightGBM	learning_rate: 0.003, n_estimators: 100, num_leaves: 50, max_depth: -1
XGBoost	learning_rate: 0.01, n_estimators: 200, max_depth: 3
CatBoost	max_depth: 5, iterations: 100, learning_rate: 0.01

Tabla 2. Hiperparámetros óptimos obtenidos para cada modelo utilizando grid search.

El modelo entrenado se validó con el conjunto de datos de prueba utilizando los mejores hiperparámetros descubiertos a través de Grid search. Para hacer más clara la comparación, la Tabla 3 muestra los resultados obtenidos del conjunto de entrenamiento y prueba. De acuerdo a dicha tabla, los modelos funcionan mejor durante la etapa de

entrenamiento que durante la etapa de prueba del modelo.

Modelo	Datos	Accuracy	Precisión	Sensibilidad	Especificidad	F1-Score	Misclassification Rate	AUC
Regresión Logística	Training	0.6763	0.5136	0.4264	0.8000	0.4660	0.3238	0.6549
	Testing	0.6650	0.6140	0.4375	0.8167	0.5109	0.3350	0.6497
Árbol de decisión	Training	0.5538	0.9000	0.3715	0.9176	0.5259	0.4463	0.6612
	Testing	0.5300	0.8596	0.3630	0.8769	0.5104	0.4700	0.6291
Adaboost	Training	0.7675	0.7409	0.5582	0.8878	0.6367	0.2325	0.7592
	Testing	0.7400	0.5789	0.5410	0.8273	0.5593	0.2600	0.6916
Random Forest	Training	0.8163	0.3364	0.9867	0.7986	0.5017	0.1838	0.9555
	Testing	0.7600	0.1930	0.8462	0.7540	0.3143	0.2400	0.8987
LogitBoost	Training	0.9075	0.9136	0.7852	0.9651	0.8445	0.0925	0.9094
	Testing	0.8350	0.6491	0.7400	0.8667	0.6916	0.1650	0.8178
KNN	Training	0.9613	0.9227	0.9355	0.9708	0.9291	0.0388	0.9493
	Testing	0.9150	0.8421	0.8571	0.9375	0.8496	0.0850	0.8931
Naive Bayes	Training	0.6063	0.7182	0.3844	0.8406	0.5008	0.3938	0.6410
	Testing	0.5750	0.6316	0.3600	0.7900	0.4586	0.4250	0.5920
SVM	Training	0.8950	0.6864	0.9096	0.8912	0.7824	0.1050	0.8303
	Testing	0.8350	0.5088	0.8529	0.8313	0.6374	0.1650	0.7369
MLP	Training	0.9188	0.9636	0.7881	0.9849	0.8671	0.0813	0.9485
	Testing	0.8450	0.8246	0.6912	0.9242	0.7520	0.1550	0.8389
Gradient Boosting	Training	0.9725	0.9273	0.9714	0.9729	0.9488	0.0275	0.9980
	Testing	0.8750	0.6140	0.9211	0.8642	0.7368	0.1250	0.9564
LightGBM	Training	0.9963	0.9999	0.9865	0.9999	0.9932	0.0038	0.9999
	Testing	0.9600	0.9298	0.9298	0.9720	0.9298	0.0400	0.9872
XGBoost	Training	0.9688	0.9409	0.9452	0.9776	0.9431	0.0313	0.9951
	Testing	0.8722	0.4865	0.8182	0.8797	0.6102	0.1278	0.9503
CatBoost	Training	0.8513	0.9727	0.6544	0.9873	0.7824	0.1488	0.8889
	Testing	0.8250	0.8947	0.6375	0.9500	0.7445	0.1750	0.8441

Tabla 3. Rendimientos de los modelos con la data de entrenamiento y de prueba utilizando los hiperparámetros óptimos obtenidos.

Se evaluaron los rendimientos para cada métrica en los modelos de Machine Learning mediante el conjunto de datos de entrenamiento (Training) y de prueba (Testing). El Accuracy, precisión, sensibilidad, especificidad, F1-score, tasa de clasificación errónea y el AUC de LightGBM resultaron ser los mejores con respecto a los demás modelos. Por otro lado, el árbol de decisión fue el modelo de menor rendimiento en la mayoría de las siete métricas en la parte de entrenamiento con respecto a los demás modelos, asimismo Random Forest fue el modelo de menor rendimiento en la parte de prueba. KNN fue el segundo mejor modelo en la mayoría de las siete métricas de rendimiento con respecto a los demás modelos en la parte de prueba. Con los datos de entrenamiento XGBoost supera a CatBoost en términos de Accuracy, Sensibilidad, F1-Score, Misclassification Rate y AUC; mientras que con los datos de prueba XGBoost supera a CatBoost en términos de Accuracy, Sensibilidad, Misclassification Rate y AUC. Los otros modelos de Machine Learning no muestran un patrón consistente en la clasificación de rendimiento. Cuando se utilizó AUC para comparar LightGBM con Gradient Boosting y XGBoost, las diferencias fueron 0.0308 y 0.0369, respectivamente.

4. Discusión

Las técnicas de Machine Learning se utilizan cada vez más para predecir la diabetes. Sin embargo, la comparabilidad y la eficacia de los modelos en aplicaciones del mundo real se han visto obstaculizadas por la incorporación de diversas atributos y técnicas de aprendizaje. En este trabajo también examinamos los efectos de varios enfoques de Machine Learning en esta predicción utilizando atributos médicos.

De los 13 modelos de Machine Learning el modelo LightGBM obtuvo una buena puntuación en las siete métricas de rendimiento: Accuracy, precisión, sensibilidad, especificidad, F1-score, tasa de clasificación errónea y AUC; El modelo KNN fue el segundo mejor modelo en la mayoría de las siete métricas de rendimiento con respecto a los demás modelos en la parte de prueba.

Por otro lado, el árbol de decisión tiene el rendimiento más bajo en la mayoría de medidas de rendimiento en la parte de entrenamiento y Random en la parte de prueba. En este estudio, investigamos tres tipos diferentes de algoritmos de aprendizaje supervisado: modelos clásicos, redes neuronales y modelos ensemble. Los modelos en cada una de estas categorías se enumeran en la Tabla 4.

El modelo KNN superó a los otros modelos clásicos en las 7 métricas de rendimientos en el entrenamiento, sin embargo, superó a los modelos clásicos en (Accuracy, sensibilidad, especificidad, F1-score, tasa de clasificación errónea y el AUC) en la parte de prueba. Si bien se sabe que los modelos ensemble producen resultados más precisos que los modelos clásicos, su desempeño para ciertos tipos de estos modelos no es tan bueno como el de los modelos clásicos. Por ejemplo, LogitBoost y AdaBoost no supera a KNN en términos de AUC. Por otro lado, el modelo MLP tiene un desempeño modesto en todos los ámbitos, aunque es un modelo interesante para la especificidad en la parte de entrenamiento y de prueba.

Los resultados revelaron que cada modelo superó a los demás en términos de las numerosas métricas de rendimiento utilizadas. Por esta razón, seleccionar el modelo más adecuado para la aplicación práctica puede ser un desafío.

Categoría	Modelo
Modelo clásico	Regresión Logística, Árbol de decisión, Naive Bayes, KNN y SVM
Redes Neuronales	Perceptrón multicapa (MLP)
Modelo Ensemble	AdaBoost, Random Forest, LogitBoost, Gradient Boosting, XGBoost, LightGBM y CatBoost.

Tabla 4. Diferentes modelos de aprendizaje en las tres categorías diferentes de Machine Learning supervisado.

5. Conclusiones

En este estudio, logramos utilizar atributos médicos para la predicción de la diabetes mellitus tipo 2 en mayores de edad en el Policlínico Leo SAC de San Juan de Lurigancho utilizando 13 modelos de Machine Learning. Hemos desarrollado algoritmos de Machine Learning a partir de tres categorías diferentes de Machine Learning supervisado; a saber, modelos clásicos, redes neuronales y modelos ensemble. La dependencia de atributos se evaluó utilizando el coeficiente de correlación para eliminar atributos redundantes. Durante el desarrollo de los modelos Machine Learning, se utilizó el Grid search para encontrar los hiperparámetros óptimos. Los modelos se entrenaron y se probaron usando el conjunto de datos de prueba. Se utilizaron siete métricas de rendimiento para evaluar el modelo entrenado. Según los resultados del estudio, el modelo con mejor rendimiento fue LightGBM, mientras que el modelo con el rendimiento más bajo en la mayoría de las siete métricas fue el árbol de decisión y Random Forest en la parte de entrenamiento y de prueba respectivamente.

Se puede notar que la mayoría de los modelos ensemble superaron a los modelos clásicos, pero también varios modelos ensemble tuvieron un rendimiento inferior a los modelos clásicos en diferentes tipos de métricas. El modelo, por otro lado, podría servir como un sistema de alerta temprana para predecir la diabetes mellitus tipo 2. También mostramos que existe una diferencia considerable entre los resultados obtenidos a partir de los diferentes modelos de predicción utilizados. Este estudio allanará el camino para que futuros investigadores proporcionen una mejor técnica generando una forma simple, económica, directa y confiable de predecir la diabetes mellitus tipo 2 basada en los atributos médicos. Esperamos que más experimentos en una gran variedad de conjuntos de datos de diabetes mellitus tipo 2 confirmen nuestros hallazgos.

REFERENCIAS

[1] Diabetes, Centers for Disease Control and Prevention, https://www.cdc.gov.

[2] Kristeen Cherney, Age of Onset for Type 2 Diabetes: Know Your Risk, online article https://www.healthline.com/health/type-2-diabetes-age-ofonset.

[3] Rajput, M. R., & Khedgikar, S. S. Diabetes prediction and analysis using medical attributes: A Machine learning approach.

[4] May, O. A. C., Koo, J. J. P., Kinani, J. M. V., & Encalada, M. A. Z. (2018). Construcción De Un Modelo De Predicción Para Apoyo Al Diagnóstico De Diabetes (Construction of a Prediction Model To Support the Diabetes Diagnosis). Pistas Educativas, 40(130).

[5] ALSHARİ, H., & ODABAS, A. Machine Learning Model to Diagnose Diabetes Type 2 Based on Health Behavior. Gazi University Journal of Science, 1-1.

[6] Singh, D. A. A. G., Leavline, E. J., & Baig, B. S. (2017). Diabetes prediction using medical data. Journal of Computational Intelligence in Bioinformatics, 10(1), 1-8.

[7] Mitushi Soni, Dr. Sunita Varma, 2020, Diabetes Prediction using Machine Learning Techniques, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 09, Issue 09 (September 2020).

[8] Nnamoko, N., Hussain, A., & England, D. (2018, July). Predicting diabetes onset: an ensemble supervised learning approach. In 2018 IEEE Congress on Evolutionary Computation (CEC) (pp. 1-7). IEEE.

[9] Barakat, N., Bradley, A. P., & Barakat, M. N. H. (2010). Intelligible support vector machines for diagnosis of diabetes mellitus. IEEE transactions on information technology in biomedicine, 14(4), 1114-1120.

[10] Nai-arun, N., & Moungmai, R. (2015). Comparison of classifiers for the risk of diabetes prediction. Procedia Computer Science, 69, 132-142.

[11] Joshi, T. N., & Chawan, P. P. M. (2018). Diabetes prediction using machine learning techniques. Ijera, 8(1), 9-13.

[12] Sisodia, D., & Sisodia, D. S. (2018). Prediction of diabetes using classification algorithms. Procedia computer science, 132, 1578-1585.

[13] Majji, Ramachandro y Bhramaramba Ravi. (2018). Type 2 Diabetes Classification and Prediction Using Risk Score. International Journal of Pure and Applied Mathematics Volume 119 No. 15 2018, 1099-1111.

[14] Shetty, D., Rit, K., Shaikh, S., & Patil, N. (2017, March). Diabetes disease prediction using data mining. In 2017 international conference on innovations in information, embedded and communication systems (ICIIECS) (pp. 1-5). IEEE.

[15] Chowdary, P. B. K., & Kumar, D. R. U. (2021). An Effective Approach for Detecting Diabetes using Deep Learning Techniques based on Convolutional LSTM Networks. IJACSA) International Journal of Advanced Computer Science and Applications, 12(4).

[16] Yahyaoui, A., Jamil, A., Rasheed, J., & Yesiltepe, M. (2019, November). A decision support system for diabetes prediction using machine learning and deep learning techniques. In 2019 1st International Informatics and Software Engineering Conference (UBMYK) (pp. 1-4). IEEE.

[17] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, 2825-2830.

[18] Wang, S., Tang, J., & Liu, H. (2017). Feature Selection.

[19] Jiang, S. Y., & Wang, L. X. (2016). Efficient feature selection based on correlation measure between continuous and discrete features. Information Processing Letters, 116(2), 203-215.

[20] Hsu, H. H., & Hsieh, C. W. (2010). Feature Selection via Correlation Coefficient Clustering. J. Softw., 5(12), 1371-1377.

[21] Amin, M. M., Gomes, P. M., Gomes, J. P., & Tasneem, F. (2021). Developing a machine learning based prognostic model and a supporting web-based application for predicting the possibility of early diabetes and diabetic kidney disease (Doctoral dissertation, Brac University).

[22] Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (2017). Classification and regression trees. Routledge.

[23] Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences, 55(1), 119-139.

[24] Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).

[25] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, 30.

[26] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. Advances in neural information processing systems, 31.