ISBN: 978-628-95207-8-1. ISSN: 2414-6390. Digital Object Identifier: 10.18687/LACCEI2024.1.1.1787

Increasing Quality in Deep Learning Models through Stratification.

Darwin Patiño-Pérez,Ph.D[1], Alex Collantes-Farah,MSIG[1],
Maylee Ordoñez-Valencia,MSc[1], Luisa Herrera-Rivas,Ph.D[2],

[1]Universidad de Guayaquil, Facultad de Ciencias Matemáticas y Física, Ecuador,darwin.patinop@ug.edu.ec, alex.collantesf@ug.edu.ec,maylee.ordonezv@ug.edu.ec

[2]Universidad de Guayaquil, Facultad de Ciencias Administrativas, Ecuador, luisa.herrerar@ug.edu.ec

Abstract. – In this study, increasing quality in deep learning models through data stratification was investigated, focusing on a dataset of patients with diabetes mellitus. An artificial neural network (ANN) was created with the objective of predicting the probability of diabetes complications based on multiple clinical and biomedical characteristics of the patients. To improve model quality, a data stratification approach was implemented, dividing the data set into homogeneous groups to ensure balanced representation of the classes of interest in the training and testing data sets. Data stratification proved to be an effective strategy to improve the quality of the deep learning model. By maintaining a balanced distribution of diabetes complication classes in the training and testing data sets, the model was able to learn more representative patterns and better generalize to unseen data. This resulted in a significant improvement in the accuracy and generalizability of the model, resulting in more accurate and useful predictions for early identification of complications in patients with diabetes mellitus. Together, these findings highlight the importance of data stratification in increasing quality in deep learning models, especially in clinical applications such as predicting medical complications. This approach has the potential to improve healthcare by enabling more accurate and earlier identification of risks for patients, which can lead to more effective interventions and better overall health outcomes.

Keywords: Deep Learning, Stratification, Models, Diabetes, Artificial Neural Networks.

ISBN: 978-628-95207-8-1. ISSN: 2414-6390. Digital Object Identifier: 10.18687/LACCEI2024.1.1.1787

Aumento de la Calidad en Modelos de Aprendizaje Profundo a través de Estratificación. Increasing Quality in Deep Learning Models through Stratification.

Darwin Patiño-Pérez,Ph.D[1], Alex Collantes-Farah,MSIG[1],
Maylee Ordoñez-Valencia,MSc[1], Luisa Herrera-Rivas,Ph.D[2],
[1]Universidad de Guayaquil, Facultad de Ciencias Matemáticas y Física, Ecuador,darwin.patinop@ug.edu.ec,
alex.collantesf@ug.edu.ec,maylee.ordonezv@ug.edu.ec
[2]Universidad de Guayaquil, Facultad de Ciencias Administrativas, Ecuador, luisa.herrerar@ug.edu.ec

Resumen. – En este estudio, se investigó el aumento de la calidad en modelos de aprendizaje profundo a través de la estratificación de datos, centrándose en un conjunto de datos de pacientes con diabetes mellitus. Se creó una red neuronal artificial (RNA) con el objetivo de predecir la probabilidad de complicaciones de la diabetes en función de múltiples características clínicas y biomédicas de los pacientes. Para mejorar la calidad del modelo, se implementó un enfoque de estratificación de datos, dividiendo el conjunto de datos en grupos homogéneos para garantizar una representación equilibrada de las clases de interés en los conjuntos de datos de entrenamiento y prueba. La estratificación de datos demostró ser una estrategia efectiva para mejorar la calidad del modelo de aprendizaje profundo. Al mantener una distribución equilibrada de las clases de complicaciones de la diabetes en los conjuntos de datos de entrenamiento y prueba, el modelo pudo aprender patrones más representativos y generalizar mejor a datos no vistos. Esto se tradujo en una mejora significativa en la precisión y la capacidad de generalización del modelo, lo que resultó en predicciones más precisas y útiles para la identificación temprana de complicaciones en pacientes con diabetes mellitus. En conjunto, estos hallazgos destacan la importancia de la estratificación de datos en el aumento de la calidad en modelos de aprendizaje profundo, especialmente en aplicaciones clínicas como la predicción de complicaciones médicas. Este enfoque tiene el potencial de mejorar la atención médica al permitir una identificación más precisa y temprana de riesgos para los pacientes, lo que puede conducir a intervenciones más efectivas y mejores resultados de salud en general.

Palabras Claves: Aprendizaje Profundo, Estratificación, Modelos, Diabetes, Red Neuronal Artificial.

Abstract. – In this study, increasing quality in deep learning models through data stratification was investigated, focusing on a dataset of patients with diabetes mellitus. An artificial neural network (ANN) was created with the objective of predicting the probability of diabetes complications based on multiple clinical and biomedical characteristics of the patients. To improve model quality, a data stratification approach was implemented, dividing the data set into homogeneous groups to ensure balanced representation of the classes of interest in the training and testing data sets. Data stratification proved to be an effective strategy to improve the quality of the deep learning model. By maintaining a balanced distribution of diabetes complication classes in the training and testing data sets, the model was able to learn more representative patterns and better generalize to unseen data. This resulted in a significant improvement in the accuracy and generalizability of the model, resulting in more accurate and useful predictions for early identification of complications in patients with diabetes mellitus. Together, these findings highlight the importance of data stratification in increasing quality in deep learning models, especially in clinical applications such as predicting medical complications. This approach has the potential to improve healthcare by enabling more accurate and earlier identification of risks for patients, which can lead to more effective interventions and better overall health outcomes.

Keywords: Deep Learning, Stratification, Models, Diabetes, Artificial Neural Networks.

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).

I. INTRODUCCION

El aprendizaje profundo ha revolucionado la inteligencia artificial al permitir que los sistemas informáticos aprendan representaciones complejas de datos de forma automática [1], lo que ha llevado a avances significativos en una amplia gama de aplicaciones, desde el reconocimiento de imágenes [2] hasta el procesamiento del lenguaje natural [3]. Sin embargo, la calidad y la capacidad de generalización de los modelos de aprendizaje profundo pueden variar considerablemente según la naturaleza de los datos utilizados para entrenarlos. En este contexto, la estratificación de los datos [4] emerge como una estrategia fundamental para mejorar la calidad de estos modelos, asegurando una representación equilibrada de las clases o características importantes en los conjuntos de datos de entrenamiento y prueba.

En esta revisión, se explora la importancia de la estratificación en el aumento de la calidad en modelos de aprendizaje profundo. Se examina la influencia de la estratificación en la calidad y el rendimiento de los modelos de aprendizaje profundo y se analizan diferentes técnicas para implementarla en el entrenamiento de los modelos. Además, se discuten los desafíos asociados con la estratificación de datos en el contexto del aprendizaje profundo, así como las posibles áreas de investigación para mejorar más esta estrategia.

El aprendizaje profundo, también conocido como *Deep Learning* en inglés, es una rama del aprendizaje automático (*machine learning*) que se centra en el entrenamiento de modelos de inteligencia artificial (IA) llamados redes neuronales artificiales, los cuales están inspirados en la estructura y el funcionamiento del cerebro humano. Estas redes neuronales consisten en múltiples capas de unidades de procesamiento interconectadas, que se organizan en una arquitectura jerárquica [5].

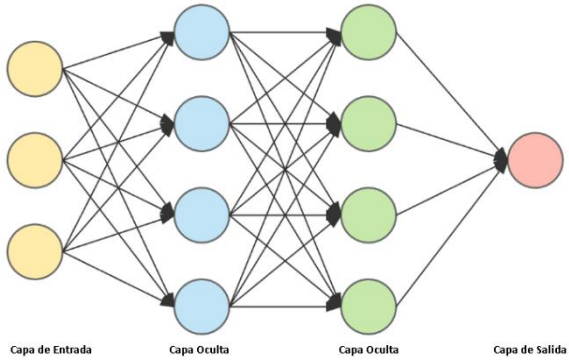

Fig. 1 Red Neuronal Artificial

Lo que distingue al aprendizaje profundo es su capacidad para aprender representaciones de datos de manera automática y progresiva, a través de múltiples capas de abstracción [6]. Esto significa que los modelos de aprendizaje profundo pueden aprender características cada vez más complejas y abstractas de los datos, lo que les permite capturar patrones y relaciones difíciles de detectar con enfoques más tradicionales de aprendizaje automático.

El aprendizaje profundo o *Deep Learning* (DL) es un área del *machine learning* (ML) o aprendizaje automático qué es un subcampo de la inteligencia artificial (IA) y se enfoca en el desarrollo de modelos basados en redes neuronales artificiales o *artificial neural networks* (ANN) como se ve en la Fig. 1, que le permiten a la máquina aprender y mejorar a través de la experiencia [7] y el procesamiento de los datos a través de sus capas ya que en lugar de ser programadas específicamente para realizar una tarea estas aprenden utilizando a las redes neuronales artificiales que utilizan datos y algoritmos para aprender por sí misma y mejorar su rendimiento en una tarea específica [8].

El proceso de aprendizaje en el aprendizaje automático se basa en identificar patrones y relaciones en los datos de entrenamiento, y luego utilizar esos patrones para hacer predicciones o tomar decisiones en nuevas situaciones [9].

El aprendizaje profundo ha demostrado ser extremadamente efectivo en una variedad de tareas de IA, incluyendo el reconocimiento de imágenes, el procesamiento del lenguaje natural, la traducción automática, la generación de texto y el reconocimiento de voz, entre otros. Su aplicación en el diagnostico de enfermedades como covid19 [10] así como la diabetes mellitus [11], entre otras ha arrojado resultados muy prometedores. Hay varios tipos de aprendizaje en inteligencia artificial, los cuales se pueden clasificar en tres categorías principales: aprendizaje supervisado, aprendizaje no supervisado y aprendizaje por refuerzo [12].

El aprendizaje supervisado es un enfoque en el que el modelo aprende a partir de un conjunto de datos etiquetados. El conjunto de datos consiste en entradas y salidas esperadas. El objetivo del modelo es aprender una función que pueda predecir la salida esperada para una entrada dada [13].

Los algoritmos de aprendizaje supervisado incluyen regresión, clasificación y redes neuronales. Sin embargo, el aprendizaje no supervisado es un enfoque en el que el modelo aprende de un conjunto de datos no etiquetados [14]. El modelo busca patrones en los datos y los utiliza para crear grupos o clústeres de datos similares. El objetivo del modelo es encontrar patrones interesantes y estructuras en los datos que puedan ayudar a los humanos a entenderlos mejor.

Los algoritmos de aprendizaje no supervisado incluyen agrupamiento, reducción de dimensionalidad y análisis de componentes principales. Por otra parte, el aprendizaje por refuerzo es un enfoque en el que el modelo aprende a través de la interacción con un entorno [15]. El modelo recibe recompensas o castigos por tomar ciertas acciones y su objetivo es aprender una política que maximice la recompensa acumulada.

En general, el tipo de aprendizaje que se utiliza depende del problema que se está intentando resolver y de la naturaleza de los datos disponibles. Cada tipo de aprendizaje tiene sus propias fortalezas y debilidades, y es importante seleccionar el enfoque adecuado para obtener los mejores resultados.

Fig. 2 Estratificación

Para este estudio se usaron datos relacionados con una de las principales enfermedades en el Ecuador, como es la diabetes mellitus (DM) la misma que ha afectado la vida de más de 1.3 millones de ecuatorianos y que representan más del 7.5 % de la población; puesto que en un estudio previo [16] destacó el rendimiento de un modelo de predicción basado en una red neuronal artificial o *artificial neural network* (ANN) la misma que alcanzó una exactitud del 93.3%. Se aplicó estratificación que usa una técnica de muestreo de datos para garantizar que las distribuciones de las clases (etiquetas) estén representadas de forma proporcional en los conjuntos de entrenamiento y prueba ver la Fig. 2; esto se lo hizo con el objetivo de poder mejorar la calidad del modelo y que en la puesta en marcha sea lo más preciso posible.

II.MATERIALES Y METODOS

A. Dataset

Se recopiló un conjunto de datos para el diagnóstico de pacientes con diabetes mellitus tipo-2 de varios centros de salud privados en Guayaquil, Ecuador. Se creó el dataset con 2768 registros de pacientes con las variables: pregnant_times, glucose, blood_pressure, tst, insulin, bmi, dpf, age, is_diabetic(etiqueta). La Tabla I muestra una descripción de las unidades e intervalos de las características de riesgo del conjunto de datos.

TABLA I
CARACTERÍSTICAS Y ETIQUETA

Variables	Descripción	Unidad
Embarazos	numero de veces de embarazo	-
Sexo	sexo M(1),F(0)	-
Glucosa	concentración de la glucosa	ml/dl
PresionSanguinea	presion arterial diastólica	mm.Hg
PliegueCutaneo	grosor pliegue cutaneo triceps	mm
Insulina	insulina sérica a 2-horas	µU/ml
IndiceDeMasaCorporal	índice de masa corporal	kg/m^2
PedigriDiabetesFuncion	función pedigrí diabetes	-
Edad	edad de la persona en años	-
Etiqueta		
DiabetesResultado	Si(1),No(0)	-

Para la implementación de las técnicas de aprendizaje de maquina supervisado se utilizará Python como herramienta de programación que se ejecuta sobre una máquina virtual de Google llamada Colab que está configurada con todas las librerías requeridas para el uso de *machine learning* y *deep learning*, además se necesitará una conexión a internet con un amplio ancho de banda para la interacción con colab.

B. Estratificación

La estratificación se aplica durante la partición de los datos en conjuntos de entrenamiento y prueba. En lugar de simplemente dividir aleatoriamente los datos en un conjunto de entrenamiento y un conjunto de prueba, la estratificación asegura que la proporción de cada clase en los datos originales se mantenga en ambos conjuntos. En vista que la estratificación de los datos implica dividir el conjunto de datos en grupos homogéneos, o estratos, con el objetivo de garantizar una representación adecuada de cada clase o característica relevante.

Este enfoque juega un papel crucial en la mitigación de desequilibrios en la distribución de clases, la reducción del sobreajuste y la mejora de la capacidad de generalización del modelo de aprendizaje profundo. Al mantener una distribución equilibrada de clases en los conjuntos de datos de entrenamiento y prueba, la estratificación permite que el modelo aprenda patrones más representativos y mejore su capacidad para hacer predicciones precisas sobre datos no vistos.

Por ejemplo, si un conjunto de datos tiene un 80% de instancias de la clase A y un 20% de instancias de la clase B, la estratificación asegura que el conjunto de entrenamiento y el conjunto de prueba también tengan aproximadamente un 80% de instancias de la clase A y un 20% de instancias de la clase B. La estratificación es importante en situaciones en las que una clase minoritaria es de interés especial y es importante garantizar que el modelo tenga suficientes ejemplos de esta clase para aprender y generalizar correctamente. La estratificación también ayuda a prevenir la sobreestimación de la precisión del modelo en casos de clases desequilibradas.

La estratificación se basa en la definición de una función de densidad de probabilidad conjunta que describe la distribución de las variables de entrada y de la variable de salida. Esta función se puede escribir como:

$$f(x, y) = f(x) * f(y \mid x) \quad (1)$$

donde:

x es el conjunto de variables de entrada

y es la variable de salida

f(x, y) es la función de densidad de probabilidad conjunta de x e y

f(x) es la función de densidad de probabilidad marginal de x

f(y | x) es la función de densidad de probabilidad condicional de y dado x.

La estratificación se puede lograr aplicando esta función de densidad de probabilidad conjunta en cada conjunto de datos de entrenamiento y prueba. Para cada conjunto, se debe asegurar que las proporciones de las clases en la variable de salida sean similares a las proporciones en el conjunto original.

Para lograr esto, se puede utilizar un enfoque de muestreo estratificado, en el cual se divide el conjunto de datos original en diferentes estratos basados en la variable de salida y se extrae una muestra de cada estrato para formar los conjuntos de entrenamiento y prueba. La proporción de cada clase en la variable de salida se mantiene en los dos conjuntos.

C. Marco Teórico

Neurona Artificial

Una neurona artificial según se muestra en la Fig. 3 es una unidad básica de procesamiento en una red neuronal artificial [17].

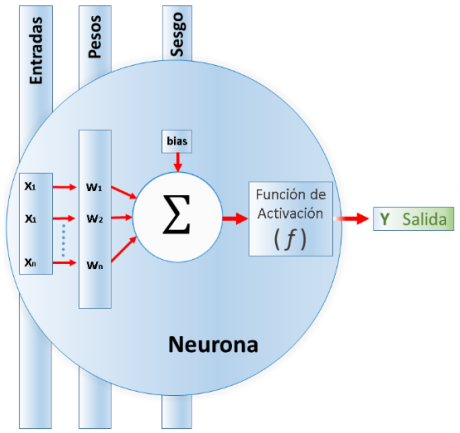

Fig. 3 Neurona Artificial

Se modela matemáticamente como una función que recibe una o más entradas, realiza un cálculo y produce una salida según (2). La entrada a una neurona artificial es ponderada por un conjunto de pesos, que representan la fuerza relativa de cada entrada en el cálculo de la salida. Además, la neurona puede tener un sesgo (bias) que permite ajustar la salida en función de un valor de referencia [18].

$$y = f(w1x1 + w2x2 + ... + wnxn + b) \quad (2)$$

donde:

x1, x2, ..., xn son las entradas a la neurona.
w1, w2, ..., wn son los pesos asignados a cada entrada.
b es el sesgo (bias) de la neurona.
f es la función de activación que se aplica al resultado de la suma ponderada de las entradas y los pesos, más el sesgo.

La salida de una neurona artificial generalmente se procesa por otras neuronas en la red neuronal artificial, y así sucesivamente hasta llegar a la salida final de la red. Este proceso se conoce como propagación hacia adelante (*forward propagation*).

Existen diferentes tipos de funciones de activación que se pueden utilizar en las neuronas artificiales, como la función sigmoide, la función ReLU (Rectified Linear Unit), entre otras. Cada una de estas funciones de activación tiene sus propias ventajas y desventajas en términos de la capacidad de la red para aprender y generalizar a partir de los datos de entrenamiento.

La función de activación se utiliza para introducir no-linealidad en el cálculo de la salida de la neurona como por ejemplo según (2) o (3), lo que permite a la red neuronal aprender relaciones más complejas entre las entradas y las salidas.

La función de activación más comúnmente utilizada es la función sigmoide, que se define como:

$$f(z) = 1 / (1 + e^{(-z)}) \quad (3)$$

donde z es la suma ponderada de las entradas y los pesos más el sesgo.

Otra función de activación popular es la función ReLU (Rectified Linear Unit), que se define como:

$$f(z) = \max(0, z) \quad (4)$$

donde z es la suma ponderada de las entradas y los pesos más el sesgo.

La Red Neuronal Artificial (ANN)

Una red neuronal artificial o ANN es un modelo computacional inspirado en la estructura y funcionamiento del cerebro humano. Consiste en un conjunto de unidades básicas de procesamiento llamadas neuronas artificiales ver Fig.1, conectadas en capas, que trabajan juntas para resolver problemas de aprendizaje y clasificación [19].

Cada neurona artificial recibe una o varias entradas, que son ponderadas por un conjunto de pesos y sumadas con un sesgo (bias). La salida de cada neurona se procesa como entrada para las neuronas de la capa siguiente, y así sucesivamente, hasta que se obtiene la salida final de la red.

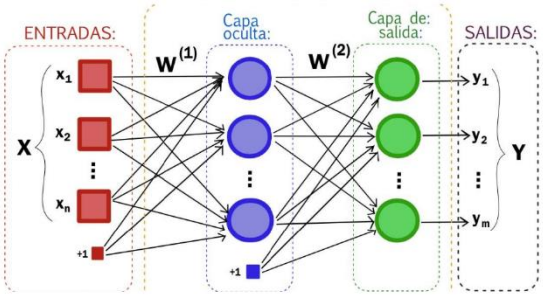

Fig. 4 Red Neuronal Artificial

La estructura de la red neuronal se define por el número y la disposición de las capas, así como por el número de neuronas en cada capa y la forma en que se conectan entre sí. Las redes neuronales pueden tener varias capas ocultas, que les permiten aprender características más complejas y abstractas de los datos de entrada [20].

Las redes neuronales se entrenan mediante un proceso de aprendizaje supervisado, en el que se presenta un conjunto de datos de entrenamiento etiquetados a la red, y se ajustan los pesos y sesgos de las neuronas para minimizar la diferencia entre las salidas de la red y las etiquetas de los datos de entrenamiento.

Una vez que se entrena la red, se puede utilizar para hacer predicciones sobre datos nuevos y no vistos [21]. Las redes neuronales se han utilizado con éxito en una variedad de aplicaciones, como reconocimiento de voz, reconocimiento de objetos en imágenes, procesamiento de lenguaje natural, y muchas otras áreas de la inteligencia artificial y la ciencia de datos.

Tipos de Redes Neuronales Artificiales (ANN)

Redes neuronales recurrentes: estas redes tienen conexiones bidireccionales que permiten que la información fluya hacia adelante y hacia atrás. Son útiles para procesar secuencias de datos, como en el procesamiento del lenguaje natural, el reconocimiento del habla y la predicción de series temporales [22].

Redes neuronales convolucionales: son especialmente adecuadas para el procesamiento de datos estructurados, como imágenes y videos. Utilizan capas convolucionales para extraer características importantes de las imágenes y reducir su complejidad [23].

Redes neuronales de propagación hacia atrás: se utilizan para el aprendizaje supervisado y se basan en la optimización de una función de pérdida mediante el ajuste de los pesos de las neuronas [24].

Redes neuronales generativas adversarias (GANs): se utilizan para generar datos sintéticos que se parecen a los datos de entrada. Las GANs utilizan dos redes neuronales que compiten entre sí para mejorar la calidad de los datos generados [25].

Redes neuronales *feedforward*: son las redes neuronales más simples y comunes. La información fluye en una dirección, desde la entrada a la salida, a través de una o más capas ocultas. Estas redes se utilizan para tareas como la clasificación, la regresión y el reconocimiento de patrones; se puede representar una red neuronal *feedforward* como una composición de funciones matemáticas [26].

Sea X un vector de entrada, Y un vector de salida, y f(x) una función que transforma x en una representación interna z:

$$z = f(x) \quad (5)$$

Luego, se pueden aplicar capas adicionales de funciones de transformación lineales y no lineales, denotadas como W y g, respectivamente, para producir la salida:

$$y = g(Wz) \quad (6)$$

donde W es la matriz de pesos que conecta las neuronas de la capa anterior con las de la capa actual, y g es la función de activación que se aplica a la suma ponderada de las entradas.

Estas capas se organizan en una estructura de capas, con la capa de entrada como la primera capa, la capa de salida como la última capa, y las capas ocultas en el medio. Cada capa puede tener un número variable de neuronas, y las conexiones entre las capas están completamente conectadas.

El proceso de entrenamiento de una red neuronal *feedforward* implica ajustar los pesos de las conexiones para minimizar una función de pérdida, que mide la diferencia entre la salida producida por la red neuronal y la salida deseada. Esto se realiza mediante un algoritmo de optimización, como el descenso del gradiente, que ajusta gradualmente los pesos de las conexiones para mejorar la precisión de la red neuronal.

Redes neuronales autoencoder: son redes *feedforward* que se utilizan para aprender representaciones de datos. Son útiles para la compresión de datos, la eliminación de ruido y la reconstrucción de datos faltantes [27].

Cada tipo de red neuronal tiene sus propias fortalezas y debilidades, y es importante seleccionar el tipo correcto para la tarea en cuestión. Además, muchas veces se combinan diferentes tipos de redes neuronales para crear sistemas más complejos y robustos.

Métricas de Evaluación

Matriz de Confusión. Según la Tabla II, la matriz de confusión [28] es el elemento sobre el cual se basan todas las métricas de clasificación, en ella se agrupan los valores clasificados por un determinado modelo (0) es *Negative* y (1) es *Positive*. Cuando el modelo clasifica adecuadamente se tienen dos valores. Verdaderos Positivos, cuando el modelo ha predicho que SI y en realidad SI. Verdaderos Negativos, aquí el modelo ha predicho que NO y en realidad es un NO. Cuando no ha clasificado adecuadamente se tienen los siguientes valores. Falso Positivo, cuando el modelo ha predicho que SI y en realidad es un NO. Falso Negativo, es cuando el modelo ha predicho que NO pero en realidad es SI.

TABLA II
MATRIZ DE CONFUSIÓN

		VALOR-PREDICCION	
		0 (Negativo)	**1** (Positivo)
VALOR-REAL	**0**	**(TN)** Verdadero Negativo El valor real es negativo y predijo un valor negativo.	**(FP)** Falso Positivo El valor real es negativo y predijo un valor positivo.
	1	**(FN)** Falso Negativo El valor real es positivo y predijo un valor negativo.	**(TP)** Verdadero Positivo El valor real es positivo y predijo un valor positivo.

En este estudio, se emplean dos técnicas de evaluación para determinar el desempeño de cada modelo de aprendizaje predictivo desarrollado basado en varios algoritmos de ML supervisados [29]. Estas técnicas incluyen lo siguiente:
La precisión se utiliza para evaluar los modelos predictivos de ML supervisados. La precisión tiene la siguiente definición:

1)Accuracy.- *Accuracy* o exactitud es la cantidad de predicciones positivas que fueron correctas.

$$\text{Accuracy} = \frac{(TN+TP)}{(FP+TP)+(TN+FN)} \quad (7)$$

2) La curva ROC o curva Característica Operativa del Receptor es el gráfico ver Fig. 5, que expone el rendimiento de un clasificador binario en función del umbral de corte, exponiendo la tasa de verdaderos positivos (TPR) contra la tasa de falsos positivos (FPR) [30] .

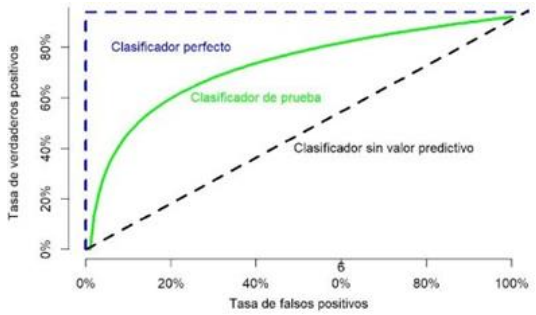

Fig. 5 Curva ROC

III.RESULTADO Y DISCUSION

El trabajo se realizó mediante un método de investigación experimental con un enfoque cuantitativo. Utilizando un método de inteligencia artificial basado en aprendizaje automático mediante aprendizaje profundo, probamos la efectividad de la estratificación a través de un proceso fijo con los siguientes pasos.:

1)Tratamiento de los Datos
2)Creación del Modelo
3)Fase de entrenamiento
4)Fase de Prueba
5)Evaluación del modelo con sus métricas
6)Aplicación del modelo

Paso 1)
Se tomo el dataset que contiene 2768 registros de información de pacientes repartidos en clases, hay 1816 pacientes con diabetes(1) y 952 pacientes sin diabetes(0), se procedió a usar un modelo con las nueve columnas de entrada sin ningún tipo de tratamiento y en otros modelos se tomaron las 6 columnas de entrada más significativas o relevantes las cuales fueron estandarizadas y escaladas, en todas las muestras se tomó el 80% para entrenamiento(train) y el 20% para pueba(test), además se tomó una muestra estratificada y otra no estratificada para realizar el comparativo.

Pasos 2) al 6)
En esta fase se exponen las etapas que van desde la creación del modelo, entrenamiento, prueba, evaluación y aplicación de estos, por lo que destaca de todos los modelos la red neuronal artificial.

Modelo de ANN-1: Para este modelo, se usó el dataset original que tiene las 9 columnas de características de entrada de donde se tomó el 80% para train y 20% para test, sin aplicarse ningún proceso de estandarización y/o normalización y sin haberse aplicado la estratificación de los datos según se aprecia en la Fig.6.

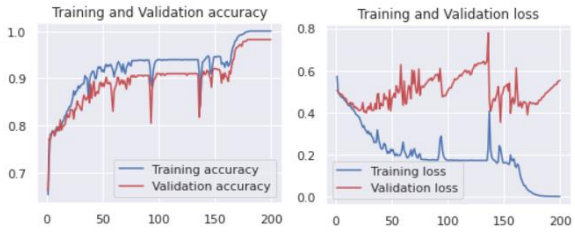

Fig. .6 Modelo de ANN-1

Se obtuvo una exactitud del 95.85% con una pérdida de 0.2955 y aunque la precisión es muy buena hay mucho ruido según la Fig. 6.

Modelo de ANN-2: Se seleccionaron de forma automática, las 6 columnas más relevantes ['Embarazos', 'Glucosa', 'Insulina','IndiceDMC','Pedigri','Edad'] y que son las que aportan de forma significativa al proceso de aprendizaje, las cuales fueron usadas como características de entrada, del total de registros se tomó el 80% para train y el 20% para test, los datos fueron estandarizados, pero no se les aplico el proceso de estratificación de los datos según la Fig.7.

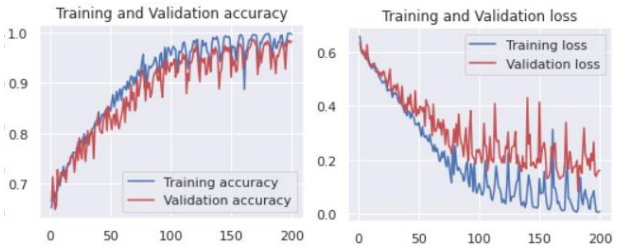

Fig. 7 Modelo de ANN-2

Se obtuvo una exactitud del 98.19% con una pérdida de 0.3748 y aunque la precisión es muy buena todavía hay ruido según la Fig.7.

Modelo de ANN-3: Se seleccionaron de forma automática, las 6 columnas más relevantes que aportan significativamente al proceso de aprendizaje, las cuales fueron usadas como características de entrada, del total de registros se tomó el 80% para *train* y el 20% para test, los datos fueron estandarizados, y aquí ya se aplicó el proceso de estratificación de los datos.

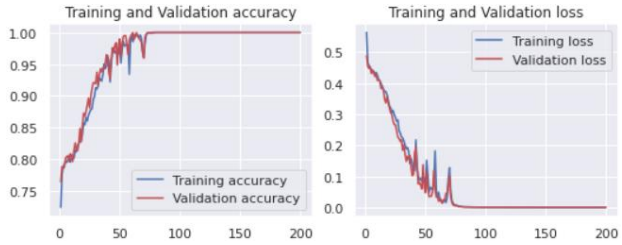

Fig. 8 Modelo de ANN-3

Se obtuvo una exactitud del 100% con una pérdida de 0.000002 en la etapa de *train* y una exactitud del 99.1% con una pérdida del 0.114 en la etapa de *test*, tal como se observa en la Fig. 8, por lo que el modelo a aprendido adecuadamente.

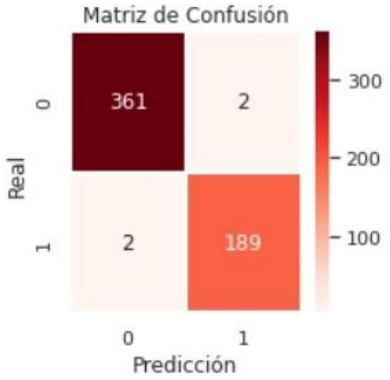

Fig. 9 Matriz de Confusión de la ANN

La matriz de confusión según la Fig. 9 fue muy importante para poder probar las métricas de clasificación, la matriz resultante está relacionada con el mejor de los modelos ANN-3.

IV. Discusion y Conclusion

La Fig. 8 refleja la exactitud o *accuracy* que se ha alcanzado con el modelo de red neuronal artificial o ANN-3 la cual tiene exactitud del 99.1% en su etapa de prueba con una pérdida del 0.114 sin que se vea algún tipo de sobre entrenamiento dado que la etapa de *train* la perdida fue del 0.000002.

Se puede concluir que de los 3 modelos de ANN el modelo de la Fig.10 ha aprendido a realizar adecuadamente la clasificación.

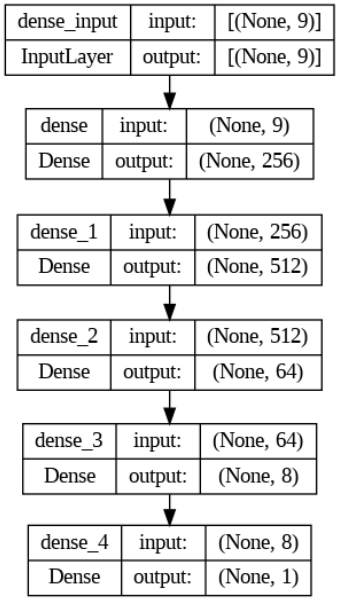

Fig. 10 Modelo ANN-3

Al realizarse el comparativo de los 3 modelos de ANN, se observa que el esquema de estatificación de datos se ha reducido la variabilidad dentro de cada clase, y que ha permitido mejorar la precisión de las estimaciones y reducir el margen de error de forma muy significativa.

El incremento de la exactitud entre la ANN-1 que tiene una exactitud del 95.85% y del último modelo ANN-3 que tiene una exactitud del 99.1%, se nota que la exactitud de predicción aumento en un 3.39%.

 En el modelo ANN-3 se aplicó estratificación y se puede determinar que la calidad del modelo aumento con un alto grado de confiabilidad, si un paciente es diabético o no; el uso del predictor será de gran ayuda en el sector de la salud porque puede diagnosticar la presencia o no de la diabetes en un paciente en cualquier momento y a cualquier hora por que se lo recomienda para su implementación.

Referencias

[1] A. Testolin and E. M. Zorzi, “The modern approach to artificial intelligence and the deep learning revolution,” *Giornale Italiano di Psicologia*, vol. 48, no. 2. 2021.

[2] Iberdrola, “Visión artificial: qué es, aplicaciones ,” *Iberdrola, S.A.* 2023.

[3] D. Khurana, A. Koli, K. Khatter, and S. Singh, “Natural language processing: state of the art, current trends and challenges,” *Multimed. Tools Appl.*, vol. 82, no. 3, 2023.

[4] M. H. Park, M. Ju, Y. H. Cho, and J. Y. Kim, “Data stratification toward advanced flood waste estimation: A case study in South Korea,” *Waste Manag.*, vol. 114, 2020.

[5] A. Krogh, “What are artificial neural networks?,” *Nature Biotechnology*, vol. 26, no. 2. 2008.

[6] T. J. Sejnowski, *The Deep Learning Revolution*. 2019.

[7] M. Varone, D. Mayer, and A. Melegari, “What is Machine Learning? A definition,” *Expert System*, 2020. .

[8] J. Cabanelas Omil, “Inteligencia artificial ¿Dr. Jekyll o Mr. Hyde?,” *Mercados y Negocios*, no. 40, pp. 5–22, 2019.

[9] A. Panesar, “What Is Machine Learning?,” in *Machine Learning and AI for Healthcare*, 2021.

[10] D. P. Pérez, R. S. Bustillos, C. M. Mora, and M. Botto-Tobar, “Prediction of Covid19 with the use of Random Forests Algorithm and Artificial Neural Networks,” *Ecuadorian Sci. J.*, vol. 4, no. 2, pp. 101–110, Sep. 2020.

[11] M. E. Baldeón *et al.*, “Prevalence of metabolic syndrome and diabetes mellitus type-2 and their association with intake of dairy and legume in Andean communities of Ecuador,” *PLoS One*, vol. 16, no. 7 July, 2021.

[12] J. Luna, “Tipos de aprendizaje automático,” *Medium*, 2018.

[13] V. Roman, “Aprendizaje No Supervisado en Machine Learning: Agrupación | by Victor Roman | Ciencia y Datos | Medium,” *Medium*, 2019.

[14] Mauricio Arango, “Introducción al Aprendizaje por Refuerzo,” *Oracle A-Team*, no. August, 2019.

[15] C. González-García, “En qué consiste el aprendizaje automático (machine learning) y qué está aportando a la Neurociencia Cognitiva,” *Cienc. Cogn.*, vol. 12, no. 2, 2018.

[16] D. Patiño-pérez *et al.*, “Modelos de Machine Learning basados en Aprendizaje Supervisado para la Detección de Diabetes Mellitus en la Ciudad de Guayaquil . Machine Learning Models based in Supervised Learning for the Detection of Diabetes Mellitus in the City of Guayaquil .,” *LACCEI*, pp. 1–8, 2022.

[17] J. Archila Rodriguez, “La neurona,” *LA Neurona*, 2017.

[18] J. A.-T. Barrera, “Redes Neuronales Artificiales,” *Univ. Guadalajara*, p. 276, 2016.

[19] B. Martín del Brío and C. Serrano Cinca, “Fundamentos de redes neuronales artificiales: hardware y software,” *Scire Represent. y Organ. del Conoc.*, 1995.

[20] O. Agasi, J. Anderson, A. Cole, M. Berthold, M. Cox, and D. Dimov, “What is an Artificial Neural Network (ANN)? - Definition from Techopedia,” *Techopedia*. 2018.

[21] D. Patiño Perez, R. Silva Bustillos, M. Botto-Tobar, and C. Munive Mora, “Análisis de Imágenes de Rayos X por Medio de Redes Neuronales Artificiales,” *Ecuadorian Sci. J.*, vol. 5, no. 1, 2021.

[22] C. Arana, “Redes Neuronales Recurrentes: Análisis de los Modelos Especializados en Datos Secuenciales,” *CEMA Work. Pap. Ser. Doc. Trab.*, 2021.

[23] A. Saxena, “An Introduction to Convolutional Neural Networks,” *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 12, 2022.

[24] T. Murooka, M. Hamaya, F. Von Drigalski, K. Tanaka, and Y. Ijiri, “Iterative Backpropagation Disturbance Observer with Forward Dynamics Model,” in *IEEE International Conference on Automation Science and Engineering*, 2021, vol. 2021-August.

[25] D. Saxena and J. Cao, “Generative Adversarial Networks (GANs),” *ACM Computing Surveys*, vol. 54, no. 3. 2021.

[26] J. Ilonen, J. K. Kamarainen, and J. Lampinen, “Differential evolution training algorithm for feed-forward neural networks,” *Neural Process. Lett.*, vol. 17, no. 1, 2003.

[27] T. A. Geddes *et al.*, “Autoencoder-based cluster ensembles for single-cell RNA-seq data analysis,” *BMC Bioinformatics*, vol. 20, 2019.

[28] P. Recuero de los Santos, “Machine Learning a tu alcance: La matriz de confusión - Think Big Empresas,” *Luca Telefonica Data Unit*, 2018. .

[29] Abhishek Sharma, “Confusion Matrix in Machine Learning,” *Www.Geeksforgeeks.Org*, 2018.

[30] A. A. Osi *et al.*, “A classification approach for predicting COVID-19 Patient’s survival outcome with machine learning techniques,” *medRxiv*, 2020.