Diseño de algoritmo compuesto por Machine Learning y un modelo probabilístico para la detección de diabetes

H. García Chávez[1*], C.E. Cañedo Figueroa[2]
[1,2]Universidad Autónoma de Chihuahua, Chihuahua, México.
a323782@uach.mx[1], ccanedo@uach.mx[2]

Abstract— **Diabetes mellitus (DM) is a type of metabolic disorder which causes chronic hyperglycemia. This alteration usually occurs due to an inadequate secretion of insulin. In the present work, a set of algorithms for the detection and prediction of diabetes was carried out using Pimas database. This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. In particular, all patients here are females at least 21 years old of Pima Indian heritage. The algorithms used were: a Naive Bayesian algorithm with an 79.67% F1, a K-nearest neighbors (KNN) algorithm with an 79.64% F1, an Artificial Neural Network (ANN) with an 74.07% F1 and an algorithm composed of the three previous algorithms with an 80.32% F1.**

Key words—**Bayesian, Diabetes, KNN, ANN.**

I. Introducción

La diabetes mellitus (DM) es un tipo de alteración metabólica la cual provoca una hiperglucemia crónica. Dicha alteración suele ocurrir por una inadecuada secreción de la hormona llamada insulina [1]. La insulina ayuda a que la glucosa obtenida en la digestión de los alimentos pueda ser transportada al interior de las células y posteriormente ser convertida en adenosín trifosfato (ATP). Sin embargo, cuando una persona tiene diabetes el cuerpo no logra captar la energía necesaria obtenida de los alimentos, por lo que esa glucosa no utilizada se acumula causando un aumento considerable de glucemia en la sangre, llamado hiperglucemia [2]. Por lo tanto, se puede considerar a una persona con DM cuando tiene un valor mayor o igual a 126 mL/dL de glucemia en la sangre. Provocando a largo plazo daño o disfunción en varios órganos o sistemas, principalmente ojos, riñones y sistema cardiovascular [3]. La DM puede ser hereditaria y a pesar de que no se puede curar, se puede controlar, sin embargo, sigue siendo una de las principales causas de muertes en México ocupando el tercer lugar [4].

La DM es una enfermedad crónica que no siempre se diagnostica a tiempo y en forma correcta, ya que no se da el seguimiento adecuado a personas que suelen tener síntomas o padecer principios de esta enfermedad, lo que resultar ser perjudicial para la salud de quien la pueda padecer. Gracias a los avances tecnológicos ha sido posible estudiar la DM, analizando su comportamiento y patrones que se muestran en las personas que ya la padecen.

Si bien las pruebas de laboratorio tienen un importante impacto en la detección de DM como lo son el análisis de glucosa en la prueba de tolerancia, prueba de glucemia capilar y hemoglobina glucosilada, se han desarrollado diversos sistemas de inteligencia artificial como auxiliar al personal médico y de laboratorio para poder aumentar la fiabilidad de su detección con el mínimo error posible.

Los algoritmos desarrollados pueden incluir características de una persona como edad, índice de masa corporal (IMC), embarazos, concentración de glucosa en la sangre, hipertensión y predisposición genética a la enfermedad. Dichos algoritmos pueden ser redes Bayesianas con una precisión del 50% [5], K-vecinos más próximos (KNN) con una precisión del 64.86% [6], Árboles de Decisión y Bosques aleatorios con una precisión del 64.868% [7][8]. De igual manera se han generado algoritmos compuestos por redes neuronales, máquinas de soporte vectorial (SVM) y algoritmos bayesianos con una precisión promedio del 74.4% [9]. Con los resultados previos se consideró la hipótesis de que se pueden incrementar estos porcentajes de precisión.

II. Metodología

A. Base de datos

La base de datos utilizada en este sistema de Machine Learning se obtuvo de una investigación realizada por el Instituto Nacional de Diabetes y Enfermedades Digestivas y Renales, la cual contiene información anónima de mujeres mayores a 21 años provenientes de los Indios Pima [10]. Los datos obtenidos en esta base de datos fueron filtrados para obtener un mejor resultado y menor rango de error, ya que contaba con datos incorrectos o vacíos dentro de las ocho características existentes. Las categorías que contaban con espacios vacíos fueron eliminadas de este estudio, por lo que las seleccionadas a utilizar fueron cinco, número de embarazos, prueba de tolerancia oral a la glucosa después de dos horas, índice de masa corporal, predisposición genética a la DM, edad y por último nuestra salida o resultado a evaluar es si padece diabetes o no. De los datos filtrados se obtuvieron 754 muestras en total, 488 sin diabetes y 264 con diabetes. Para que de ese total se pudieran obtener aleatoriamente 200 muestras de personas sin diabetes (S_DM) y 200 muestras de personas con diabetes (C_DM) y por

dx.doi.org/10.24254/CNIB.21.2

último obtener 64 datos más con diabetes (X_D) y 64 sin diabetes (X_ND) para poder realizar una prueba de verificación para los diferentes algoritmos realizados e implementados en el software de MATLAB R2020b, los cuales se utilizaron para obtener un algoritmo compuesto por un algoritmo bayesiano, un KNN y una red neuronal con la finalidad de comprobar el incremento de la precisión en comparación con los algoritmos individuales.

B. Algoritmo bayesiano ingenuo

El algoritmo bayesiano ingenuo, tiene una funcionalidad en relación a la probabilidad de que una muestra pueda pertenecer a una clase en específico, considerando las características por clase [5].

Para el desarrollo del algoritmo bayesiano ingenuo, se consideraron los datos de S_DM y C_DM para determinar las clases sin diabetes y con diabetes respectivamente.

Se calculó la probabilidad seleccionando un dato aleatorio que pudiera pertenecer a una determinada clase. Lo cual se realizó utilizando las ecuaciones (1) y (2), en donde N_D y N_{ND} son la cantidad de vectores característicos para cada una de las clases, P_D y P_{ND} se refieren a la probabilidad aleatoria de que una muestra pueda ser de la clase Diabetes o No diabetes respectivamente.

$$(1)\quad P_D = \frac{N_D}{N_D + N_{ND}}$$

$$(2)\quad P_{ND} = \frac{N_{ND}}{N_D + N_{ND}}$$

Se determinaron los valores de media $\overline{X_j}$ y varianza σ_j^2 de cada una de las características contenidas en S_DM y C_DM siguiendo las ecuaciones (3) y (4), en donde j corresponde a cada una de las características, i a cada uno de los valores y N a la cantidad de valores de cada característica, en este caso, toma el valor de 200.

$$(3)\quad \overline{X_j} = \frac{\sum(S\_DM[j]_i)}{N}$$

$$(4)\quad \sigma_j^2 = \frac{\sum(S\_DM[j]_i - \overline{X})^2}{N-1}$$

Posteriormente se aplicó la ecuación (5) para poder determinar las probabilidades de una muestra de pertenecer a una clase según la característica que se esté analizando. En donde X_{ji} se refiere a una muestra nueva que contenía la misma cantidad de características y $C_{D,ND}$ se refiere a las clases diabetes y no diabetes.

$$(5)\quad P(j \mid C_{D,ND}) = \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-\frac{(X_{ji}-\overline{X}_j)^2}{2\sigma_j^2}}$$

Para continuar, se aplicó la ecuación (6) en la cual se calculó la relación de todas las probabilidades obtenidas por (5), (1) y (2), dadas por ambas clases.

$$(6)\quad P\_r_{D,ND} = (P_{D,ND})(P(j \mid C_{D,ND}))$$

Aplicando la ecuación (7), se calculó la evidencia, la cual se utilizó para calcular finalmente las probabilidades dadas por la ecuación (8) para cada una de las clases.

$$(7)\quad Evidencia = P\_r_D + P\_r_{ND}$$

$$(8)\quad Ps_{D,ND} = \frac{P\_r_{D,ND}}{Evidencia}$$

Para determinar la clase ganadora, según el algoritmo bayesiano, se consideró a Ps_D sí $Ps_D > Ps_{ND}$ o bien a Ps_{ND} sí $Ps_{ND} > Ps_D$.

C. Algoritmo KNN

El algoritmo K-vecinos más cercanos (KNN) es un método de clasificación por agrupamiento, que permite clasificar una muestra nueva a partir de la distancia entre dicha muestra y los datos pertenecientes a cada clase dentro del conjunto de entrenamiento ST, el cual se define con la ecuación (9).

$$(9)\quad ST = [S\_DM, C\_DM]$$

Para poder obtener la distancia de la muestra se aplicó la ecuación (10), donde ST_{ij} indica el valor de cada una de las muestras i por característica j y x_{ij} se refiere a la muestra nueva.

$$(10)\quad d_{ij} = \sqrt{\sum(ST_{ij} - x_{ij})^2}$$

Para el desarrollo del algoritmo se implementó de manera experimental utilizar las k=5 distancias más cercanas a la muestra, en donde K es el número de vecinos a considerar como cercanos. Para poder obtener un resultado se aplicaron las ecuaciones (11) y (12) en donde KC_D y KC_{ND} se incrementan en 1, siempre y cuando d_{ij} sea obtenida de una muestra de ST que pertenezca a C_DM o S_DM respectivamente.

$$(11)\quad KC_D = \begin{cases} KC_D + 1 & Si \quad d_{ij} \mid ST \in C\_DM \\ KC_D & Si \quad d_{ij} \mid ST \notin C\_DM \end{cases}$$

$$(12)\quad KC_{ND} = \begin{cases} KC_{ND} + 1 & Si \quad d_{ij} \mid ST \in S\_DM \\ KC_{ND} & Si \quad d_{ij} \mid ST \notin S\_DM \end{cases}$$

La salida del algoritmo, determina como clase ganadora a KC_D sí $KC_D > KC_{ND}$ o a KC_{ND} sí $KC_{ND} > KC_D$.

D. *Red neuronal*

Una red neuronal es un algoritmo basado en un conjunto de unidades llamadas neuronas, las cuales tienen una función similar a la de una neurona biológica, las cuales reciben un estímulo, una función de activación y una salida.

Así mismo se diseñó una red neuronal totalmente conectada hacía adelante (Fig. 1) con 5 entradas, 2 dos capas ocultas con 9 neuronas cada una con una función de activación tangencial sigmoidal y dos salidas con la función SOFTMAX.

La cantidad de neuronas, número de capas ocultas y funciones de activación se consideraron después de haber realizado entrenamientos de forma experimental.

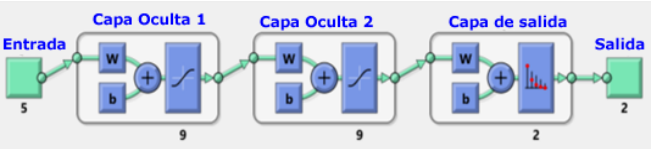


Fig. 1. Diseño da la red neuronal con 5 entradas, 2 capas ocultas de 9 neuronas con función de activación tangencial sigmoidal y una capa de salida con función SOFTMAX para dos clases resultantes.

El entrenamiento de la red se realizó con el algoritmo de Levenberg-Marquardt backpropagation con los siguientes hiperparámetros seleccionados de acuerdo a los resultados obtenidos de forma experimental:

- Factor de aprendizaje = 0.01
- Número de épocas = 5000
- Error mínimo= 1e-25.

Previo al entrenamiento se dividió de forma aleatoria la base de datos de S_DM y C_DM dejando un 92% como conjunto de entrenamiento, 4% para validación y 4% para prueba.

La salida de la red neuronal se determinó por la ecuación (13), en donde ANN se refiere a la red neuronal previamente entrenada y x_i' es la muestra nueva con todas las características transpuestas.

$$AC_{D,ND} = ANN(x_i') \quad (13)$$

Se consideró a AC_D como clase ganadora sí $AC_D > AC_{ND}$ o a AC_{ND} sí $AC_{ND} > AC_D$.

E. *Algoritmo compuesto*

Se realizó de forma experimental, un algoritmo compuesto por la salida de la red neuronal, bayesiano ingenuo y KNN, el cual consideró que, si se obtuvo una clase de diabetes en por lo menos dos de ellos, el resultado final será diabetes, de lo contrario se podrá emitir una recomendación.

El algoritmo se basa en la Tabla 1. En donde las clases ganadoras ya sean Diabetes o No diabetes de cada uno de los algoritmos, tienen una votación la cual se comparó con el resto de ellos, otorgando una salida de diabetes, si dos o más de ellos resultaron ser de la clase diabetes, si dos de ellos fueron de la clase no diabetes y uno de la clase diabetes o bien, si todos son de la clase no diabetes, se determinó el caso como no diabetes.

TABLA 1.
EJEMPLO DE LA COMPOSICIÓN DEL ALGORITMO COMPUESTO, EN DONDE EL RESULTADO ESTA DADO COMO POSITIVO SI DOS O MÁS RESPUESTAS SON POSITIVAS, Y COMO NEGATIVO SI SOLO UNA O NINGUNA ES POSITIVA.

Bayesiano	KNN	ANN	Resultado
Ps_D	KC_D	AC_D	Diabetes
Ps_{ND}	KC_D	AC_D	Diabetes
Ps_D	KC_{ND}	AC_D	Diabetes
Ps_D	KC_D	AC_{ND}	Diabetes
Ps_{ND}	KC_{ND}	AC_D	No diabetes
Ps_D	KC_{ND}	AC_{ND}	No diabetes
Ps_{ND}	KC_D	AC_{ND}	No diabetes
Ps_{ND}	KC_{ND}	AC_{ND}	No diabetes

III. RESULTADOS

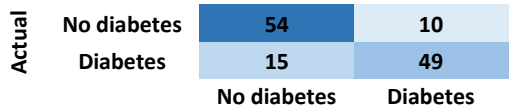
En los algoritmos bayesiano, KNN y Red neuronal se obtuvieron los resultados al analizar los datos de X_D y X_ND.

De los cuales en el algoritmo bayesiano se observan en la de la Fig. 2, teniendo una precisión del 83%, exhaustividad del 76.56%, una exactitud del 80.46% y un F1 del 79.67%.

En el algoritmo KNN los resultados obtenidos se observan en la Fig. 3, teniendo una precisión del 91.83%, una exhaustividad del 70.31%, una exactitud del 82.03% y un F1 del 79.64%.

En la Red neuronal se obtuvieron los datos de la Fig. 4, pudiendo calcular y obtener una precisión del 70.42%, exhaustividad del 78.12%, exactitud del 72.65% y un F1 del 74.07%.

Para finalizar se realizó el mismo análisis del algoritmo compuesto, incluyendo el bayesiano, KNN y red neuronal, de acuerdo a los datos de la Fig. 5, se obtuvo una precisión del 84.48%, exhaustividad de 76.56%, exactitud de 81.25% y un F1 de 80.32%.



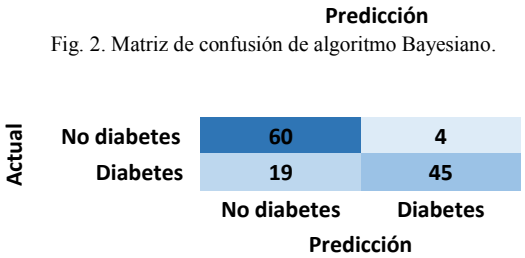


Fig. 2. Matriz de confusión de algoritmo Bayesiano.

Fig. 3. Matriz de confusión de algoritmo KNN.

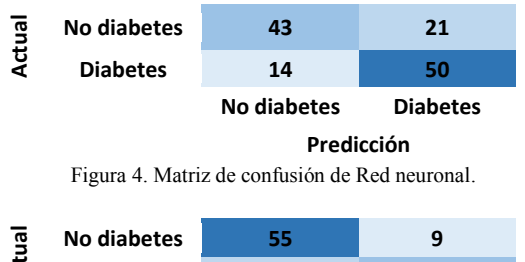


Figura 4. Matriz de confusión de Red neuronal.

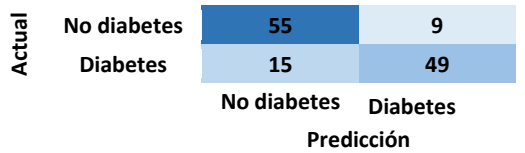


Figura 5. Matriz de confusión de algoritmo compuesto.

IV. Discusión

Los algoritmos diseñados para la base de datos diabetes Pimas, utilizan todas las características que se tienen, incluyendo los valores no completos utilizado regresión para completar dichos datos o bien rellenando con ceros, dichos algoritmos tienen precisiones muy dispersas como: 50% para Bayesiano, 64.86% para KNN, 64.86% para arboles de decisión y un 74.4 % para algoritmos compuestos. En este trabajo se eliminaron los datos faltantes, derivado de ello se utilizaron 5 características de esta base de datos.

Se realizaron 4 algoritmos, en donde el que tuvo mayor precisión fue el algoritmo KNN con un 91.83% mientras que el algoritmo compuesto desarrollado muestra una precisión del 84.48%, sin embargo, considerando otros parámetros como el F1 el algoritmo compuesto fue mejor obteniendo un 80.32% mientras que el KNN obtuvo 79.64%.

Cabe resaltar que si el algoritmo se pone en marcha, este algoritmo es para el apoyo médico, lo cual requiere la valoración del experto de la salud para validar los resultados obtenidos. Se plantea desarrollar en un trabajo a futuro analizar costos de cada clase si las ponderaciones obtenidas en los diversos algoritmos se encuentran muy cerca del punto medio, por lo que se deberá realizar un estudio más profundo para dar un resultado más eficaz.

V. Conclusiones

Los algoritmos compuestos son una buena herramienta para la clasificación de datos, sin embargo, no necesariamente deben ser mejores que un algoritmo "puro" ya que la diferencia que el F1 que se muestra en este trabajo entre el KNN y el algoritmo compuesto se tiene una diferencia de 0.68% siendo superior el algoritmo compuesto. De igual manera es importante detectar las características que mejor separen a las clases y no utilizar datos incompletos para la creación de algoritmos.

Reconocimiento

Se le agradece a la Universidad Autónoma de Chihuahua por el apoyo en la prestación de los recursos digitales y físicos para la realización de esta investigación, al igual que a la coordinadora del programa de Ing. Biomédica, la Ing. Natalia Gabriela Sámano Lira y al director de la Facultad de Medicina y Ciencias Biomédicas el Dr. Luis Carlos Hinojos Gallardo por la gestión en pro de la publicación de esta investigación.

Referencias

[1] D. I. Conget, "Diagnosis, classification and pathogenesis of diabetes mellitus," *Rev. Esp. Cardiol.*, vol. 55, no. 5, pp. 528–535, 2002, doi: 10.1016/S0300-8932(02)76646-3.

[2] M. Hull and P. A. Bruno, "STUDENTS with DIABETES.," *Nurse.com Mag.*, vol. 5, no. 9, pp. 34-39 6p, 2014, [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=c8h&AN=107828498&amp%5Cnlang=ja&site=ehost-live.

[3] E. Rojas de P, R. Molina, and C. Rodríguez, "Definición, clasificación y diagnóstico de la diabetes mellitus," *Rev. Venez. Endocrinol. y Metab.*, vol. 10, no. 1, pp. 7–12, 2012.

[4] Instituto Nacional de Estadística y Geografía, "Características de las defunciones registradas en México durante enero a agosto de 2020 [Comunicado de prensa]," vol. 1, no. 2, p. 45, 2021, [Online]. Available: https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2021/EstSociodemo/DefuncionesRegistradas2020_Pnles.pdf.

[5] O. D. Castrillón, W. Sarache, and E. Castaño, "Sistema bayesiano para la predicción de la diabetes," *Inf. Tecnol.*, vol. 28, no. 6, pp. 161–168, 2017, doi: 10.4067/S0718-07642017000600017.

[6] M. Panwar, A. Acharyya, R. A. Shafik, and D. Biswas, "K-nearest neighbor based methodology for accurate diagnosis of diabetes mellitus," *Proc. - 2016 6th Int. Symp. Embed. Comput. Syst. Des. ISED 2016*, no. November 2018, pp. 132–136, 2017, doi: 10.1109/ISED.2016.7977069.

[7] H. Lai, H. Huang, K. Keshavjee, A. Guergachi, and X. Gao, "Predictive models for diabetes mellitus using machine learning techniques," *BMC Endocr. Disord.*, vol. 19, no. 1, pp. 1–9, 2019, doi: 10.1186/s12902-019-0436-6.

[8] A. Singh, M. N., and R. Lakshmiganthan, "Impact of Different Data Types on Classifier Performance of Random Forest, Naïve Bayes, and K-Nearest Neighbors Algorithms," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 12, 2017, doi: 10.14569/ijacsa.2017.081201.

[9] G. Rani and P. K. Tiwari, "Handbook of Research on Disease Prediction Through Data Analytics and Machine Learning." 2020, [Online]. Available: https://books.google.ae/books?id=SrgIEAAAQBAJ.

[10] "Diabetes Data Set | Kaggle." https://www.kaggle.com/vikasukani/diabetes-data-set.%0A (accessed Jun. 13, 2021).