UN MODELO MATEMÁTICO DE CLASIFICACIÓN DE PACIENTES CON SINTOMAS COVID-19

TECNOCIENCIA

A MATHEMATICAL MODEL FOR CLASSIFICATION OF PATIENTS WITH COVID-19 SYMPTOMS

Trujillo González, Julio Enrique; Martínez Valderrama, Iveth Verónica

Julio Enrique Trujillo González
julio.trujillo@up.ac.pa
Universidad de Panamá, Facultad de Ciencias Naturales, Exactas y Tecnología, Departamento de Matemática, Panamá
Iveth Verónica Martínez Valderrama
iveth.martinez@up.ac.pa
Universidad de Panamá, Facultad de Ciencias Naturales, Exactas y Tecnología, Departamento de Matemática, Panamá

Tecnociencia
Universidad de Panamá, Panamá
ISSN: 1609-8102
ISSN-e: 2415-0940
Periodicidad: Semestral
vol. 24, núm. 2, 2022
Luis.rodriguez@up.ac.pa

Recepción: 01 Abril 2022
Aprobación: 30 Mayo 2022

URL: http://portal.amelica.org/ameli/journal/224/2243351006/

Esta obra está bajo una Licencia Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional.

Resumen: El objetivo de este trabajo es clasificar pacientes con síntomas de COVID-19 utilizando la estructura de una red neuronal artificial, optimizándola mediante el método de Stochastic gradient descent (SGD) y el método Adam. Los paquetes *Tensorflow* fueron de gran utilidad para obtener el modelo de forma secuencial y los resultados que aseguran que es posible modificar o insertar datos nuevos. La incorporación de la red neuronal artificial nos permitirá hacer una clasificación que toma en cuenta el 99% de la variabilidad de la(s) variable(es) dependiente(s) dentro del conjunto de datos.

Palabras clave: Perceptrón, Red neuronal artificial, COVID-19.

Abstract: The objective of this work is to classify patients with COVID-19 symptoms using the structural an artificial neural network, optimizing it using the Stochastic gradient descent (SGD) method and the Adam method. Tensorflow packages were very useful to obtain the model sequentially and the results that ensure that it is possible to modify or insert new data. The incorporation of the artificial neural network will allow us to make a classification that takes into account 99% of the variability of the dependent variable(s) within the data set.

Keywords: Perceptron, Artificial Neural Network, COVID-19.

INTRODUCCIÓN

La inteligencia artificial (IA) ha intervenido en nuestras vidas de forma espectacular. Es indiscutible que, en diversas actividades, la mano de obra puede ser reemplazada por IA que realiza el trabajo en un periodo corto de tiempo. Su intervención se ha extendido al empleo de técnicas de aprendizaje automático, en aplicaciones clínicas para apoyar un sistema inteligente para una detección temprana y un método de diagnóstico de la enfermedad (Tama & Lim, 2020), entre otras. Además, ayuda a los médicos proporcionando una segunda opinión sobre un diagnóstico clínico eficaz y más efectivo, así como en otras tareas relacionadas que evite el error humano de manera que se evite el riesgo de perder la vida del paciente.

Muchos investigadores han desarrollados diferentes algoritmos de aprendizaje automático para el diagnóstico de enfermedades y han aceptado que los algoritmos funcionan bien en el diagnóstico de diferentes

enfermedades. En la Fig. 1 se muestra algunos ejemplos de enfermedades diagnosticadas por técnicas de aprendizaje automático.

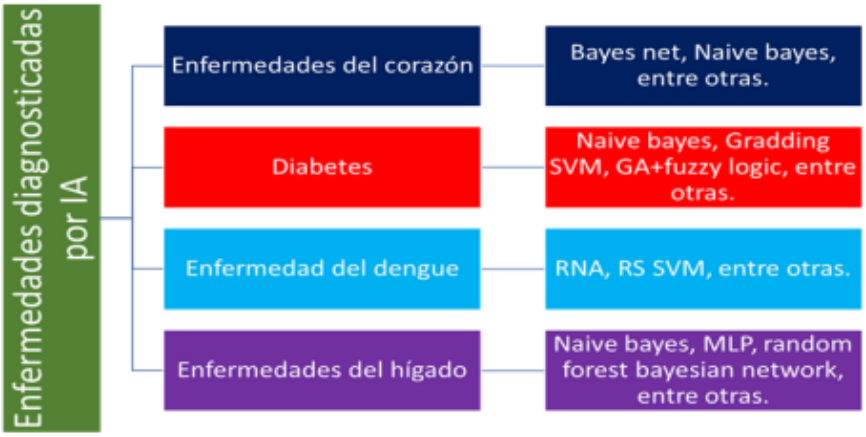

FIGURA. 1
Enfermedades diagnosticadas por Aprendizaje Automático

Este trabajo se centra en implementar el algoritmo de clasificación propuesto en la tesis "Modelización de la predicción de riesgo de la dinámica de transmisión de la influenza en la provincia de Panamá" (Trujillo, 2020) en el caso particular para el COVID, donde podamos observar el índice de efectividad de este algoritmo, cuya estructura es una red neuronal artificial (RNA).

1. Antecedentes de las redes neuronales artificiales

El desarrollo de las ciencias de la computación ha motivado que los modelos matemáticos sean automatizados y utilizados como herramientas para el desarrollo de la humanidad. Como he de esperarse existen modelos que intentan simular el modo que el ser humano aprende, lo que se conoce como modelos de Inteligencia artificial (IA).

Es así, que tomar como modelo una neurona cerebral y abstraer sus partes nos permite describir su comportamiento. Esto originó una nueva área de estudio llamada redes neuronales artificiales o simplemente redes neuronales (RN).

Nuestro punto de partida va a ser el modelo propuesto por los fisiólogos y biofísicos ingleses, Alan Lloyd Hodgkin y Andrew Fielding Huxley que, en 1952, describieron los mecanismos iónicos por los que producían y propagaban los pulsos a lo largo del axón gigante de un calamar. Además, explicaremos el modelo presentado por (Berzal, 2019).

El mecanismo del potencial de acción de la RN depende de la apertura de canales de iones en la membrana celular de la neurona. Esos canales permiten que los iones de sodio, Na+, penetren el axón y causen un aumento del potencial de la membrana, que al ser de grasa actúa como un aislante, descrito en la Fig. 2.

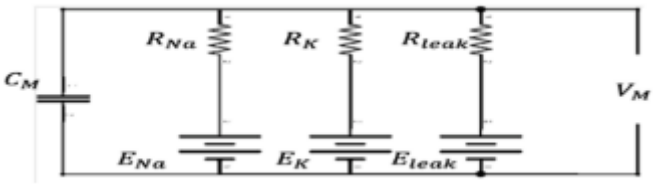

FIGURA 2
Modelo Hodgkin-Huxley

En este modelo el potencial de la membrada ## está dado por

$$C_M \frac{DV_M}{dt} = -g_{leak}(V_M - E_{leak}) - g_{Na}(V_j - E_{Na}) - g_K(V_M - E_K) \quad (1)$$

donde

i) g indica la conductividad de los canales de sodio, potasio y pérdida o escape (leak).
ii) Los potenciales E corresponde a los gradientes electroquímicos asociados al flujo de los iones: ENa (sodio), Ek potasio y Elek (Corriente perdida).
iii) CM es la capacidad asociada a la membrana como condensador.

Cuando se conectan múltiples segmentos del axón, aparecen el fenómeno de la propagación del potencial de acción o pulso (spike). En el caso de un circuito el pulso se movería en las dos direcciones, pero en una neurona biológica sólo se movería en un sentido.

Ignorando los canales de iones, obtenemos el siguiente modelo «Leaky integration and fire neuron model»

$$C_M \frac{DV_M}{dt} = -g_{leak}(V_j - E_{leak}) + g_{int} \sum_{i=1}^{n} w_{ij} x_i \quad (2)$$

donde

i) Vj representa el potencial de la membrana para la neurona j −ésima.
ii) xi la salida actual de la neurona j −ésima.
iii) wij el peso de la conexión desde la neurona i hasta la neurona j.

Para el modelo presentado, asumiremos que Eleak = 0, gleak = 1, gint = 1 y exigiremos que la neurona no tenga memoria, obteniendo

$$V_j = \sum_{i=1}^{n} w_{ij} x_i \quad (3)$$

En ocasiones nos interesa que el comportamiento de la neurona tenga un sesgo, que es similar a las corrientes de pérdidas en una neurona biológica. El sesgo va a depender del estado de la neurona, descrito de la forma:

$$\Delta V_j = -V_j + \sum_{i=1}^{n} w_{ij} x_i \quad (4)$$

donde ΔVj dota a la neurona de una capacidad de memoria. Gran parte de los modelos consideran un sesgo externo o bias, bj

$$V_j = \sum_{i=1}^{n} w_{ij} x_i + b_j \quad (5)$$

De manera que bj = w0j, está asociado a una entrada fija #0 = 1.

En la entrada, una neurona combina las diferentes entradas xI con sus pesos asociados, para determinar la entrada total zj, descrita por

$$z_j = \sum_i w_{ij} x_i \quad (6)$$

En la activación, utiliza los valores de entrada para generar una salida
yj, obteniendo un modelo numérico

$$y_j = f(z_j) = f\left(\sum_i w_{ij} x_i\right) \quad (7)$$

donde f es una función de activación, usualmente no lineal. Lo podemos escribir como

$$y_j = f(w^T x) \quad (8)$$

donde wes un vector de pesos wij y x el vector de entradas xi, cuando se toman varias neuronas que interactúan conjuntamente se forma una capa de neuronas.

La notación vectorial simplificada al diseñar la red neuronal compuesta por varias capas nos da la expresión.

$$y = f(Wx) \quad (9)$$

donde y es el vector de salida de la capa m × 1; # es el vector de entradas n × 1 y W es la matriz de pesos m × n.

Es relativamente sencillo incorporar los sesgos de las neuronas a la expresión (9), obteniendo la expresión.

$$y = f(Wx) + b \quad (10)$$

2. Perceptrón

El origen del Deep learning se remonta a 1943, cuando Warren McCulloch y Walter Pitts describieron en su artículo el primer modelo de red neuronal artificial.

Fueron los primeros en estudiar el cerebro desde un enfoque computacional, en donde con sus ingenios resolvieron problemas sin interpretar el funcionamiento del ordenador de forma algorítmica. Fue así, que plantearon un algoritmo de aprendizaje que ajustara los parámetros de la red automáticamente, de manera similar a una máquina de Turing.

Alan Turing, fue el primero en considerar el entrenamiento de redes, conectadas de forma aleatoria demostrado, en su publicación de 1948 "Unorganized machine".

Posteriormente, fueron muchos los investigadores que contribuyeron al desarrollo de la Inteligencia Artificial (IA.), de los cuales destacamos a Frank Rosenblatt, quien diseñó el algoritmo de aprendizaje del perceptrón como se muestra en Fig. 3 y quien utilizó el término conexionista para distinguir su aproximación a IA (Rosenblatt, 1960).

Algorithm 1 Algoritmo del Perceptrón

```
procedure PERCEPTRON(w_0)
    w_1 ← w_0                                   ▷ Usualmente w_0 = 0
    for t ← 1, T do
        RECIBIR(x_t)
        z̄_t ← sgn(w_t · x_t)
        RECIBIR(z_t)
        error = |z̄_t − z_t|
        if error > 0 then
            w_{t+1} ← w_t + z_t x_t
        else
            w_{t+1} ← w_t
        end if
    end for
    return w_{T+1};
end procedure
```

FIGURA 3
Algoritmo del Perceptrón

Para la tarea de clasificación de zt es usual usar +1 o -1. La regla de aprendizaje del perceptrón es relativamente simple y la describimos de la siguiente manera:

i) Comenzar con pesos aleatorios para las conexiones,
ii) Seleccionar el vector ## para el entrenamiento,
iii) Si |zt − zt/ > 0, modificar todas las conexiones w(i) de acuerdo con: zt(i)xt(i)
iv) Regresar al paso ii.

Para la regla de aprendizaje del perceptrón existe un teorema de convergencia, el cual establece lo siguiente

Teorema. Si existe un conjunto de pesos w el cual es capaz de realizar la transformación zt, la regla de aprendizaje del perceptrón convergerá a una solución en una cantidad finita de pasos para cualquier elección inicial de los pesos.

Prueba. La prueba de este teorema fue realizada por Trujillo, (2020)

3. Modelo de clasificación de pacientes

Las variables que consideraremos, para el estudio, serán las mismas utilizadas para pronosticar cuando un paciente contrae COVID-19, entre las cuales tenemos: la fatiga, expectoración, nariz tapada, fiebre, tos seca, entre otras (Michelen y col., 2020).

Por lo tanto, para predecir la severidad de la enfermedad, es razonable considerar los síntomas como datos de entrada. En este estudio se utilizará, para la prueba, el conjunto de datos del COVID-19 Symptoms Checker que consta de 31680 registros para todas las variables. En este estudio utilizamos el 70% en el modelo de predicción y el 30% restante para la verificación del modelo.

El modelo RN es desarrollado para predecir que personas, infectadas, desarrollaran una situación de gravedad de la enfermedad.

Es importante destacar que el modelo utilizado es una RN multicapa (Gulli, Kappor & Pal, 2019) de manera que se desarrollen las combinaciones no lineales de las variables de entrada. Se empleará funciones de activación RELU y Softmax en las capas ocultas, y en el nodo de salida se utiliza también RELU.

Las RN cuentan con seis capas ocultas, ver en Fig. 4, y se entrena utilizando un algoritmo de gradiente descendiente estocástico (SGD) y términos de momento. El número de nodos varía entre 5 y 20 en la que en cada iteración se calcula el error cuadrático medio (MSE) entre la salida del modelo y los datos.

Para fines de predicción, la propiedad más importante de un modelo es su competencia para generalizar. Si bien la competencia de generalización indica el poder de un modelo en el buen desempeño aún con datos no utilizados para entrenarlo. El sobreajuste evita la generalización del modelo frente a nuevas situaciones (Bishop, 1995), así para evitar esto, se emplea la interrupción temprana (EarlyStopping) de la técnica de regularización más utilizada y con el uso de la capa de abandono (Dropout).

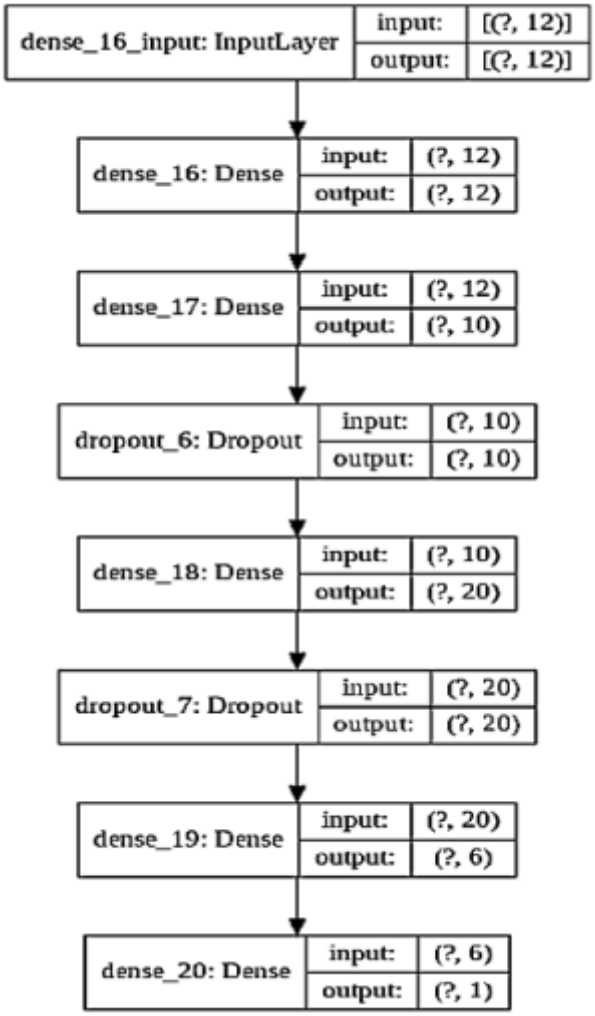

FIGURA 4
Arquitectura de la red neuronal artificial

El análisis del modelo de redes neuronales, en Fig. 5, correspondiente al entrenamiento del modelo. Es fácil observar como el patrón en el entrenamiento tiene un comportamiento parecido a la prueba. Además, cómo la convergencia de ambos coincide en un valor cercano a la iteración 315 y 68, respectivamente.

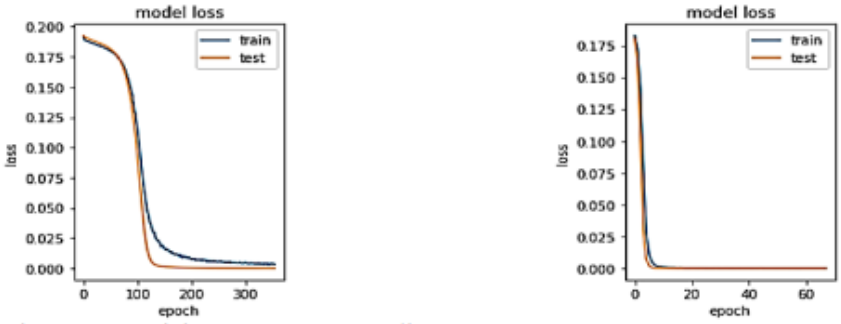

FIGURA 5
Precisión y error promedio

Lo anterior nos evidencia que el modelo propuesto tendría una buena estimación al utilizar datos reales.

Tomando en cuenta que para la evaluación de cada uno de los algoritmos de optimización la data se obtuvo de COVID-19 Symptoms Checker, mostrando la eficiencia de cado uno en la tabla 1.

TABLA 1
Comparación de los modelos SGD y Adam

Optimizador	□2	MSE	MAE
SGD	0.999778805	4.19011988 2e- 05	3.28848371 1e- 03
Adam	0.999999913	1.64512201 9e- 08	6.46279917 8e- 05

CONCLUSIONES

El propósito del modelo de RN era identificar de forma temprana las personas que podrían presentar cuadros graves de COVID-19 y con esta información elaborar una estrategia y solicitar los recursos mínimos para su atención oportuna. Debido a la falta de datos y la dificultad de adquisición, este estudio proporciona una RN general para la clasificación de los pacientes. Este estudio facilitó la producción de pruebas analíticas rápidas y repetidas con la utilización de datos que nos permitieron verificar el rendimiento de los modelos. Se examinaron la red utilizando dos optimizadores el SGD y Adam, con las mismas entradas y salidas.

Se comprobó que el rendimiento de las redes, las dos diseñadas, proporcionan un buen modelo para la clasificación. Específicamente, con el optimizador Adam, se produce la mejor bondad de ajuste (R2 = 0.999999913) y el error de predicción más bajo (MSE = 1.645122019e- 08).

No existe una regla para seleccionar el número de capas intermedias y el número de neuronas en las RN. La mejor manera de definir estos números es probando diferentes estructuras de modelos con diferentes números de intermedios hasta lograr un resultado que se ajuste a la realidad.

En el estudio, el desempeño de cada red se comprobó y la estructura utilizada contaba con el número mínimo de capas, seleccionando neuronas para evitar cualquier redundancia.

Los modelos presentados sirven como complementos para otros modelos basados en ecuaciones diferenciales o análisis estadísticos, sin intentar reemplazarlos.

REFERENCIAS

Berzal, F. (2019). Redes neuronales y Deep learning. España, Editorial Universidad de Granada.

Bishop, C. (1995). Neuronal networks for pattern recognition. Gran Bretaña, Oxford university Press.

Gulli, A., Kapoor, A. & Pal, S. (2019). Deep Learning with TensorFlow 2 and Keras: Regression, ConvNets, GANs, RNNs, NLP, and More with TensorFlow 2 and the Keras API. Gran Bretaña, Editorial Pack Publishing.

Michelen, M., Jones, N. & Stavropoulou, C. (2020). In patients of COVID-19, what are the symptoms and clinical features of mild and moderate cases? The Centre for Evidence-Based Medicine develops, promotes and disseminates better evidence for healthcare. https://www.cebm.net/covid-19/in- patients- ofcovid- 19- what-are- the- symptoms- and- clinical - features- of - mild- andmoderate-case/

Rosenblatt, F. (1960). Perceptron simulation experiments, Proceedings of the IRE, vol. 48, no. 3, pp. 301–309.

Tama, B. & Lim, S. (2020). A Comparative Performance Evaluation of Classification Algorithms for Clinical Decision Support Systems, Mathematics, vol. 8, no. 10, 1814.

Trujillo, J. (2020). Modelización de la predicción de riesgo de la dinámica de transmisión de la influenza en la provincia de panamá (tesis de postgrado). Universidad de Panamá, Panamá.