INVESTIGACIÓN e INNOVACIÓN en INGENIERÍAS
ISSN2344-8652

Optimización de los hiperparámetros de una máquina de regresión de soporte vectorial utilizando enjambre de partículas para el pronóstico de casos de COVID-19

Vector Support Regression Machine Hyperparameters Optimization by utilizing Particle Swarms for COVID-19 cases forecasting

Norbey Danilo Muñoz Cañón **Jairo Andrés Romero Triana**

Universidad Distrital Francisco José de Caldas, Colombia

OPEN ACCESS

Recibido: 23/04/2021
Aceptado: 19/07/2021
Publicado: 10/09/2021

Correspondencia de autores:
ndmunozc@correo.udistrital.edu.co

Resumen

Objetivo: Optimizar los hiperparámetros de una máquina de regresión de soporte vectorial mediante la adaptación de la metaheurística de enjambre de partículas para pronosticar la serie de tiempo del total de casos positivos acumulados de la reciente enfermedad COVID-19 en la ciudad de Bogotá, Colombia. **Metodología:** se plantea un algoritmo híbrido de regresión de soporte vectorial y optimización por enjambre de partículas para encontrar el valor óptimo de los hiperparámetros de una máquina de regresión de soporte vectorial que mejor rendimiento muestre en el pronóstico de la serie de tiempo. Se valida a través de una comparación de los valores reales con los predichos obtenidos por una máquina de regresión sin hiperparámetros optimizados, en términos de métricas de desempeño como el error cuadrático medio, error absoluto medio y coeficiente de determinación. **Resultados:** cualitativamente se verifica el rendimiento mediante los pronósticos obtenidos en la serie de tiempo; cuantitativamente, con un valor en el error cuadrático medio de 0,000045 y un coeficiente de determinación de 0,998884, el método propuesto presenta un mayor desempeño. **Conclusiones:** el algoritmo presentado y aplicado es útil para el pronóstico de series de tiempo; con este algoritmo se aporta al campo de investigación; finalmente se discute sobre la implementación de este método en el contexto epidemiológico.

Palabras clave: covid-19, inteligencia de enjambres, máquina de soporte vectorial, optimización de enjambre de partículas, pronóstico.

Abstract

Objective: Hyper-parameter optimization of a vectorial-support regression machine via adaptation of metaheuristics of a particle swarm so that a prediction of the time series of the total amount of positive accumulated cases of COVID-19 in Bogotá, Colombia can be made. **Methodology:** a hybrid vectorial-support regression algorithm along with a particle-swarm based optimization are used to determine an optimal value for the hyper-parameters of a vectorial-support regression machine such that best performance is shown in the time series prediction. In order to validate the performance of the method, a comparison with a regression vectorial-support machine whose hyper-parameters have not been optimized will be made, being the metrics those of performance measurement like mean square error, mean absolute error, and determination coefficient. **Results:** The proposed method finds itself at a greater level of performance when the mean square error value is that of 0,000045 and the determination coefficient corresponds with the value of 0.998884. **Conclusions:** the presented and applied algorithm comes useful in order to predict time series; this algorithm is of ultimately great value to the research field; finally, implementation of this method in an epidemiological context is discussed.

Keywords: covid-19, swarm intelligence, support vector machine, particle swarm optimization, forecasting.

Como citar (IEEE): N. Miñoz-Cañón y J. Romero-Triana "Optimización de los hiperparámetros de una máquina de regresión de soporte vectorial utilizando enjambre de partículas para el pronóstico de casos de COVID-19", *Investigación e Innovación en Ingenierías*, vol. 9, n°2, 91-111, 2021. DOI: https://doi.org/10.17081/invinno.9.2.4475

Introducción

Con la aparición del síndrome respiratorio agudo severo coronavirus 2 (SARS-CoV-2), causante de la enfermedad Covid-19, el escenario de salud pública mundial entró en un estado de emergencia ante la amenaza global del nuevo tipo de coronavirus, generando pérdidas impredecibles en áreas como la economía y el empleo [1]. La enfermedad COVID-19 se ha expandido desde su epicentro en la provincia de Hubei en China, donde fue identificada por primera vez en diciembre de 2019, a prácticamente todo el mundo; desde el 11 de marzo de 2020 fue declarada pandemia mundial por la Organización Mundial de la Salud (OMS) [2,3]. Con menos de 30 casos a finales de diciembre de 2019 a más de 20.000.000 de casos confirmados (según datos existentes) al 10 de agosto de 2020 la enfermedad se expandió rápidamente en todo el mundo.

Al declararse pandemia mundial y surgir la emergencia sanitaria, resulta esencial el acceso a modelos precisos de predicción de brotes para obtener información sobre la probabilidad de propagación y las consecuencias de la reciente enfermedad infecciosa [4]. Los gobiernos y otras instituciones legislativas se basan en los conocimientos de los modelos de predicción para sugerir nuevas políticas y asimismo evaluar la eficacia de las estrategias aplicadas [5]. Debido a que la enfermedad ha exhibido una naturaleza no lineal y compleja [6], ha surgido un problema a gran escala sobre el desarrollo de modelos epidemiológicos, por lo cual el aprendizaje automático (ML) ha llamado la atención recientemente para construir modelos de predicción de brotes, pronóstico de casos, estimación de muertes y proyección de recuperaciones [4].

Los métodos de ML han sido utilizados para modelar pandemias anteriores, por ejemplo Ébola, Cólera, influenza H1N1, fiebre del dengue, Zika o norovirus de ostras [7,8,9,10,11]. Estas técnicas de ML se limitan a los métodos básicos de árbol de regresión, pronóstico aleatorio, redes neuronales, redes bayesianas, Naïve Bayes, y regresiones lineales simples o múltiples; sin embargo existen métodos de ML más sofisticados, por ejemplo híbridos o en conjunto, que otorgan mejores resultados al combinar varios algoritmos, utilizar técnicas de optimización o aplicar metaheurísticas. Durante los últimos años el interés por las metaheurísticas ha aumentado considerablemente en el campo de la optimización. Los mejores resultados encontrados para muchos problemas en la ciencia y la industria se obtienen combinando herramientas de optimización como las metaheurísticas y ML, proporcionando algoritmos eficientes [12].

Las metaheurísticas se han empleado ampliamente para mejorar las tareas del aprendizaje automático así como la optimización de parámetros o de configuraciones [13]. Algunas de estas metaheurísticas están basadas en la población y en los comportamientos colectivos en sistemas autoorganizados y descentralizados (distribuidos), por lo que representan una aplicación de la inteligencia de enjambres (SI). Los sistemas de SI están conformados por una población de agentes computacionales simples capaces de percibir y modificar su ambiente de manera local; tal capacidad hace posible la comunicación entre los individuos, que detectan los cambios en el ambiente generado por el comportamiento de sus semejantes [14]. La regresión y la clasificación son tareas de ML supervisada en la que se predice una categoría o clase predefinida a partir de un conjunto de atributos dado (variables continuas para regresión y variables discretas para clasificación) [15]. En particular, la regresión tiene como objetivo estimar las relaciones entre una variable de respuesta y una o más variables explicativas, y tiene una amplia gama de aplicaciones sobre las series de tiempo [13]. Normalmente, el uso del ML está relacionado con el entrenamiento de modelos de regresión avanzados por lo que resulta adecuado interpretar e implementar mejoras algorítmicas y de resultados con ayuda de otro tipo de técnicas de la inteligencia artificial o computacional, como los métodos de SI.

En este trabajo se presenta la aplicación de una técnica de optimización de inteligencia de enjambres sobre un modelo de regresión lineal de aprendizaje automático. El método planteado consiste en un modelo en conjunto (ensamblado) de una máquina de regresión de soporte vectorial (SVR) con la optimización del algoritmo de enjambre de partículas (PSO), denominado SVR+PSO. El propósito de la implementación del PSO es la optimización de los hiperparámetros de la máquina de regresión para obtener un mejor resultado, en términos de exactitud y rendimiento, del pronóstico de una serie de tiempo de casos confirmados de la enfermedad COVID-19 en la ciudad de Bogotá, Colombia. El rendimiento de la regresión depende en gran medida de la elección de los hiperparámetros [16]; en este documento el enfoque es sobre los siguientes hiperparámetros: ε que controla el ancho de la zona insensible del modelo, C un factor de regularización que penaliza los errores de restricción, y γ un parámetro de función del kernel. El documento continúa con una sección de trabajos relacionados que es útil como marco de referencia, la sección del método aplicado y las técnicas utilizadas, posteriormente la presentación de resultados y discusiones, y finalmente las conclusiones.

Trabajos relacionados

Las diferentes técnicas y herramientas proporcionadas por la inteligencia artificial permiten realizar predicciones en los ecosistemas actuales, que a diario presentan nuevos retos y escenarios inesperados para el hombre, quien se encuentra en la búsqueda constante de mecanismos que permitan reducir la incertidumbre con el objetivo de prepararse para dichos escenarios futuros. La inteligencia artificial no ha sido ajena a la aparición de diferentes virus que han desatado pandemias que a su vez han afectado a la humanidad y el caso actual del COVID-19 no es la excepción, teniendo en cuenta el limitante de la falta de información o el exceso de datos en otros casos, es importante la recopilación y análisis de los datos disponibles para poder entrenar las diferentes implementaciones en las que se utilizan herramientas propias de la inteligencia artificial y así realizar diferentes estimaciones que sean útiles para tomar decisiones que limiten daños e incluso permitan salvar vidas [17].

Paralelamente en la actualidad se han realizado diferentes trabajos investigativos sobre la dinámica y detección temprana de COVID-19 utilizando modelos matemáticos y técnicas de inteligencia artificial. Donde predomina la implementación de modelos Susceptible-Expuesto-Infectado-Recuperado (SEIR) y Susceptible-infectado-recuperado (SIR) con el uso de herramientas proporcionadas por la IA donde comúnmente se utilizan redes neuronales convolucionales (CNN) alimentadas por datos de casos de diagnósticos relacionados, imágenes médicas, estrategias de gestión, personal sanitario, demografía y movilidad. En gran variedad de propuestas se ha demostrado la eficacia en la aplicación de estas metodologías, herramientas y técnicas, quedando siempre la puerta abierta para la innovación, optimización y mejoramiento de modelos ya propuestos [18].

Del mismo modo, se han implementado otras técnicas menos utilizadas de predicción como regresión lineal y regresión vectorial para la anticipación del avance de esta pandemia, un claro ejemplo es el trabajo propuesto en India, donde se han realizado propuestas de aprendizaje de máquina para generar modelos de pronóstico de la pandemia en dicho país, empleando regresión lineal, perceptrón multicapa y método de regresión vectorial [19].

Además técnicas como sistemas de membranas estocásticas (sistemas P) han sido utilizadas en el pasado para el modelado de pandemias, como es el caso de la influenza pandémica A (H1N1) en regiones geográficas aisladas para la predicción de enfermedades infecciosas dentro de áreas predefinidas y la evaluación de estrategias de intervención [20].

Finalmente mecanismos de optimización bioinspirados han sido utilizados para pronosticar casos confirmados de COVID-19 en países como China, donde sistemas de inferencia neuro-difusos adaptativos han sido combinados con algoritmos de polinización de flores y algoritmos de enjambre de salpas para potenciar el sistema mencionado y lograr así mejores resultados a la hora de hacer el pronóstico [21].

Materiales y método

El planteamiento del método propuesto SVR+PSO consiste en la aplicación de una técnica de optimización basada en inteligencia de enjambres sobre el algoritmo de aprendizaje supervisado SVR, utilizado para problemas de clasificación y regresión. En esta sección se presentan los procedimientos que involucran la obtención del conjunto de datos, el preprocesamiento de los datos (herramientas de extracción, transformación y carga - ETL), técnicas utilizadas, la descripción de los hiperparámetros, el método propuesto y las métricas de medición.

Conjunto de datos

El desarrollo del presente trabajo implicó el uso de los datos del número de casos confirmados por el laboratorio de COVID-19 de la ciudad de Bogotá. Estos datos pertenecen a la secretaría de salud de la ciudad, son actualizados con los reportes diarios del laboratorio y se encuentran abiertos al público [22]. Para la fecha en la cual se consultó el conjunto de datos, este contaba con un total de 166.685 registros, donde cada registro representa a un individuo con diagnostico positivo de COVID-19 y diferentes atributos pertenecientes al mismo. Para efectos del presente trabajo solo se tiene en cuenta la fecha de diagnóstico, atributo que corresponde a la fecha en la cual se confirmó el positivo por parte del laboratorio.

Técnicas utilizadas

Para obtener un conjunto de datos apropiado que sea utilizado por la máquina de regresión de soporte vectorial y el algoritmo PSO, es necesario hacer un tratamiento a los datos originales mediante la creación de scripts en Python utilizando la librería Pandas. Se realiza una agrupación mediante la “fecha de diagnóstico” que permita contar las veces que se repite la misma fecha para obtener el total de casos diarios. Una vez obtenido el número de casos por día se crea el atributo “casos acumulados”, que registra el total de casos de COVID-19 por día, dato que será utilizado en la regresión.

Por otro lado, la normalización de datos en el contexto de aprendizaje de máquina es una técnica utilizada en la etapa de preparación de datos, que permite generar una escala común para un conjunto de datos numéricos [23]. Al ser las máquinas de soporte vectorial una técnica de aprendizaje automático se recomienda aplicar dicha normalización a los datos de casos acumulados de COVID-19. Esto mediante la aplicación de la siguiente ecuación.

$$X_{normalizado} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Donde:

$X_{normalizado}$: *Dato resultante de la normalización*

X: *Dato original al cual se le realiza normalización*

X_{min}: *Dato mínimo del conjunto de datos*

X_{max}: *Dato mínimo del conjunto de datos*

Así mismo, para modelar esta serie temporal es necesario transformar los datos de la manera adecuada con el propósito de alimentar el método, para esto se implementa una función de retraso (lag function), que tiene en cuenta los valores en pasos de tiempo anteriores, transformando un problema de predicción de series de tiempo en un problema de aprendizaje supervisado, donde se busca predecir el valor en el momento $(t+1)$, dado el momento anterior $(t-1)$ [24].

Una vez realizado todo lo anterior, el conjunto de datos ya está listo para alimentar el método SVR+PSO, donde la máquina cumple el papel de realizar el proceso de aprendizaje mediante los hiperparámetros optimizados por el algoritmo PSO y no los hiperparámetros definidos por defecto.

Hiperparámetros

Los hiperparámetros son parámetros ajustables que se eligen para entrenar un modelo y que rigen el propio proceso de entrenamiento. Estos valores suelen permanecer constantes durante el proceso de entrenamiento. En escenarios de aprendizaje profundo o aprendizaje automático, el rendimiento del modelo depende en gran medida de los valores de hiperparámetros seleccionados. Para la adecuación de los hiperparámetros se requiere de un espacio de búsqueda, el tipo de hiperparámetros (discreto o continuo) y una técnica de muestreo [25].

SVR

La máquina de regresión de soporte vectorial es una técnica de ML, que construye una predicción de modelo lineal minimizando simultáneamente el riesgo empírico y la complejidad del modelo [26]. SVR se caracteriza por el uso de kernels, solución dispersa y control VC (teoría de Vapnik-Chervonenkis) del margen, y el número de vectores de soporte; es una herramienta eficaz en la estimación de funciones de valor real. Como enfoque de aprendizaje supervisado, SVR entrena usando una función de pérdida simétrica, que penaliza igualmente las estimaciones erróneas altas y bajas [27].

El rendimiento de SVR depende de la selección adecuada de los hiperparámetros, los cuales son ε, C y γ. Estos tres hiperparámetros son continuos. El hiperparámetro ε controla el ancho de la zona insensible alrededor de la predicción del modelo; el número de vectores de soporte está relacionado directamente con el valor de ε. Si se elige un valor grande de ε, se seleccionan pocos vectores de soporte, lo que hace

que el modelo sea más plano. Mientras que un valor pequeño de ε permite seleccionar más vectores de soporte, lo que aumenta la complejidad del modelo. Para el hiperparámetro C, la elección de un valor bajo hace que la función sea plana. Sin embargo, tomando un valor C alto, el modelo selecciona más muestras como vectores de soporte para estimar correctamente todos los datos de entrenamiento. Por su parte, si el hiperparámetro del kernel γ es demasiado pequeño, la influencia de los vectores de soporte es demasiado fuerte y ninguna cantidad de regularización podrá evitar el sobreajuste. Cuando γ es muy grande, el modelo está demasiado restringido y no puede capturar la complejidad de los datos. Por tanto, cada hiperparámetro puede afectar la complejidad del modelo de forma diferente [16,26,28].

PSO

El uso del algoritmo PSO tiene como fundamento la optimización de los hiperparámetros C, ε y γ, propios de la máquina de regresión de soporte vectorial utilizada con miras a obtener resultados más precisos para el modelo de predicción.

Así mismo PSO se define como un algoritmo de optimización inspirado en el comportamiento óptimo de diferentes grupos de animales, donde la población social recibe el nombre de enjambre y cada individuo del enjambre se conoce como partícula [29], una partícula se compone por velocidades y posiciones de desplazamiento. Estos valores se van adaptando a medida que el proceso de aprendizaje se lleva a cabo con el objetivo de aproximarse a una posición deseada que implica la optimización de una función objetivo.

De manera análoga el algoritmo se compone resumidamente por la siguiente serie de pasos [30]:

- Determinar el tamaño del enjambre. Este valor no está predeterminado por alguna regla establecida, así pues, este valor se asigna como resultado de un proceso de ensayo y error.
- Crear la población inicial de manera aleatoria, es decir hacer una inicialización aleatoria de la posición de las partículas en el espacio de búsqueda.
- Inicializar y asumir las velocidades iniciales como $v = 0$.
- Búsqueda del P_{best} y el G_{best}, donde el P_{best} representa el vector de partículas que ha presentado las mejores soluciones de cada partícula y el G_{best} representa el vector de partículas que ha presentado la mejor solución.
- Hallar las nuevas velocidades para las partículas:

$$V_j(i) = V_j(i-1) + c_1 r_1 \left[P_{best} - x_j(i-1)\right] + c_2 r_2 \left[G_{best} - x_j(i-1)\right] \quad (2)$$
$$j = 1, 2, \ldots, N$$

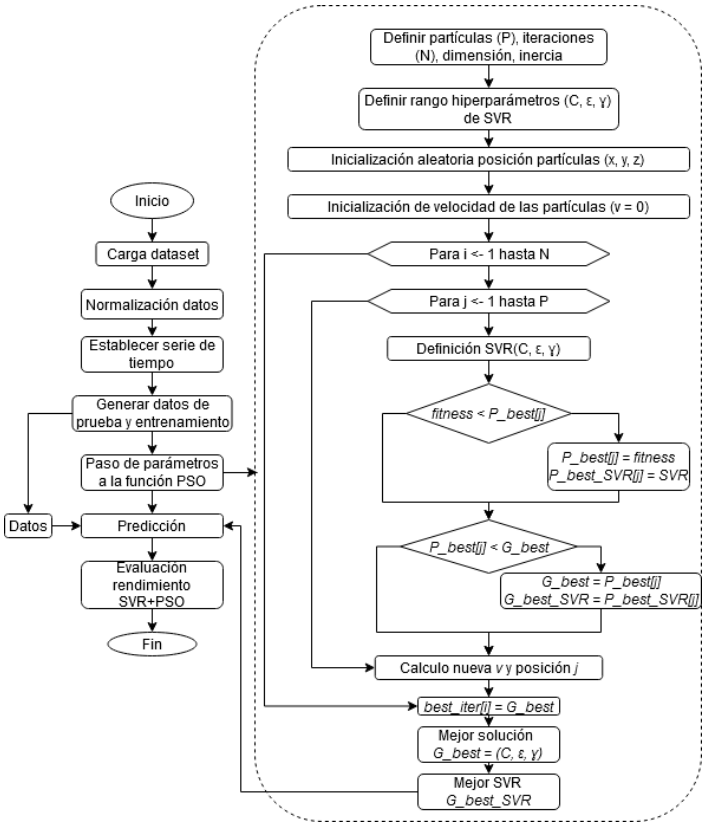
Donde:

- C_1 y C_2 son los ritmos de aprendizaje de la partícula y sociales respectivamente, este es un valor heurístico el cual es recomendable que tome el valor de 2.
- r_1 y r_2 son valores de probabilidad aleatorios.
- i es el número de la iteración actual.
- Generar el cambio en la posición de la partícula acorde a la tasa de cambio producida por la velocidad anteriormente calculada.

Método propuesto

La figura 1 muestra el diagrama de flujo del método SVR+PSO de series de tiempo para el pronóstico de los casos confirmados de COVID-19.

Figura 1. Método SVR+PSO.



Fuente: Elaboración propia

SVR+PSO utiliza el algoritmo de enjambre para evaluar la mejor configuración de hiperparámetros en el regresor en cada iteración y de esta manera obtener los valores óptimos que minimicen el error y aumenten el rendimiento. En la fase de optimización en la que se busca la mejor ubicación para cada partícula y la mejor partícula del enjambre, la función fitness para el entrenamiento del regresor se hace a través de la métrica del error cuadrático medio (MSE) existente entre el actual valor y el valor candidato, esto con el propósito de elegir la mejor solución del enjambre.

$$MSE = \frac{(p_1 - a_1)^2 + \cdots + (p_n - a_n)^2}{n} \qquad (3)$$

Donde p es el valor real, a el valor candidato y n la cantidad de observaciones del conjunto de datos.

Para la obtención de un rendimiento alto del método, se encontró que un valor apropiado de población inicial es 120 partículas. La dimensión del espacio de búsqueda está determinada por los hiperparámetros, para este caso es 3. La posición de cada partícula en el espacio de búsqueda está definida por el vector $(x, y, z) = (C, \varepsilon, \gamma)$ y se inicializa aleatoriamente. El rango para el espacio de búsqueda de los hiperparámetros se definió de la siguiente manera (ver Tabla 1).

Tabla 1. Rango para el espacio de búsqueda de los hiperparámetros.

C		ε		γ	
Min	Max	Min	Max	Min	Max
0,001	10000	0,00000001	0,1	0,001	1000

Fuente: Elaboración propia.

Cabe resaltar que el criterio de detención en este caso es el número de iteraciones, que ha sido fijado en 10, ya que con pocas iteraciones se logra el resultado esperado y se observó que aumentando el número de iteraciones, muy pocas veces se presentó un cambio significativo en las métricas.

En cada iteración se evalúan cada una de las partículas para seleccionar la mejor partícula del enjambre. Con la mejor partícula se evalúa el regresor de manera que se guarda hasta que se encuentre uno mejor. Al finalizar las iteraciones, el mejor regresor con los hiperparámetros optimizados es utilizado para la evaluación con los datos, obtener el rendimiento y finalmente alcanzar el pronóstico.

Comparación del método

Para establecer el rendimiento esperado del método propuesto e implementado, se realiza una comparación con una máquina de regresión de soporte vectorial con hiperparámetros por defecto a través de las medidas de rendimiento que se enuncian en la siguiente subsección. Esta comparación permite evaluar el funcionamiento y validar el rendimiento del método propuesto de manera que se logre satisfacer la idea de la mejora que brinda las metaheurísticas para la optimización de modelos de ML y su aplicación sobre una serie de tiempo.

Medidas de rendimiento

La calidad del método propuesto se evalúa utilizando un conjunto de métricas de rendimiento de la siguiente manera:

- MSE $$\sqrt{\frac{(p_1 - a_1)^2 + \cdots + (p_n - a_n)^2}{n}} \quad (4)$$
- RMSE $$\sqrt{\frac{(p_1 - a_1)^2 + \cdots + (p_n - a_n)^2}{n}} \quad (5)$$
- MAE $$\frac{|p_1 - a_1| + \cdots + |p_n - a_n|}{n} \quad (6)$$
- R2 $$\frac{\frac{\sum(p_i - \bar{p})(a_i - \bar{a})}{n-1}}{\sqrt{\frac{\sum(p_i - \bar{p})^2}{n-1}\frac{\sum(a_i - \bar{a})^2}{n-1}}} \quad (7)$$

Algoritmo 1 SVR+PSO propuesto.

Input: Conjunto de datos de histórico de COVID-19, cantidad de partículas P, número de iteraciones N, dimensión e inercia.
Normalización de datos.
Aplicación de función lag.
División de datos para entrenamiento y prueba.
Definir espacio de búsqueda de las partículas (rango para los hiperparámetros (C, ε, γ))
Inicialización aleatoria de posición de las partículas (x, y, z) en el espacio
Inicialización velocidades iniciales $v = 0$
$for\ i = 0\ to\ N\ do$
  $for\ j = 0\ to\ P\ do$
    Crear regresor con los hiperparámetros.
    Entrenamiento del regresor para ajustar curva.
    Calculo función fitness.
    $if\ fitness < P_{best}[j]\ then$
      Guardar fitness como mejor P_{best} para j.
      Guardar regresor como el mejor regresor para j.
    $end\ if$
    $if\ P_{best}[j] < G_{best}\ then$
      Guardar $P_{best}[j]$ como mejor G_{best} del enjambre.
      Guardar regresor de j como el mejor regresor.
    $end\ if$
    Actualizar velocidad y posición de la partícula.
  $end\ for$
Guardar la mejor partícula del enjambre de la iteración i.
Retornar la mejor solución que representa la mejor configuración.
Retornar el mejor regresor.
Aplicar el conjunto de datos de prueba al mejor regresor.
Evaluación de métricas.
Pronóstico de la serie de tiempo.

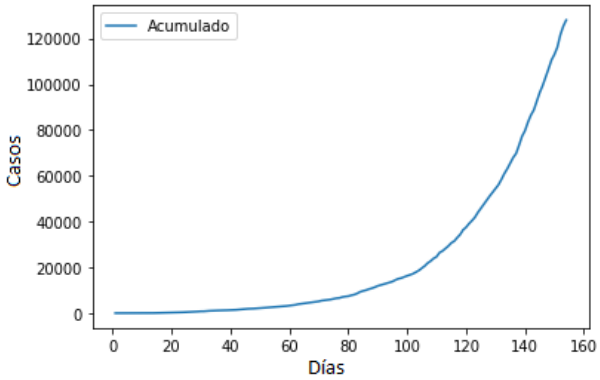
Fuente: elaboración propia.

Resultados y discusiones

En esta sección se presentan los rendimientos del método propuesto y la máquina de regresión de soporte vectorial para el pronóstico de la serie de tiempo de los casos de la enfermedad. En la parte final se presenta una discusión sobre algunas consideraciones que involucran este tipo de técnicas para el propósito presentado.

La figura 2 muestra la totalidad de los datos de la serie de tiempo; representan la cantidad de casos acumulados por día desde que se informó sobre el primer caso en la ciudad el 6 de marzo de 2020.

Figura 2. Casos acumulados diarios en Bogotá.



Fuente: Elaboración propia.

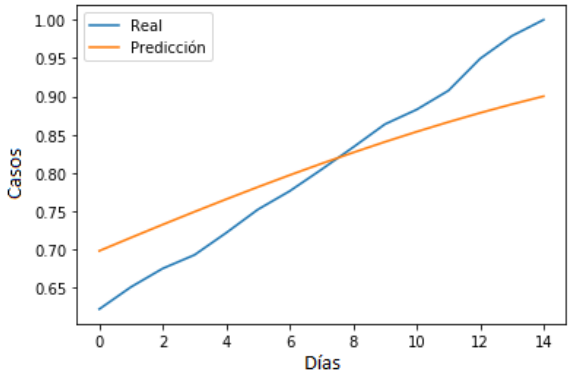
SVR

Con el objetivo de tener un punto de referencia con el cual poder realizar una comparación del método propuesto, se plantea la implementación de un algoritmo que únicamente utilice una máquina de regresión de soporte vectorial cuyos hiperparámetros no sean optimizados, es decir que serán utilizados con su configuración por defecto.

Para ello solamente se define el regresor sin necesidad de pasarle ningún parámetro; el valor por defecto de los hiperparámetros definido por la librería scikit learn es: $C = 1{,}0, \varepsilon = 0{,}1$ y $\gamma = 0{,}1$. Luego se entrena el modelo y se evalúan los resultados.

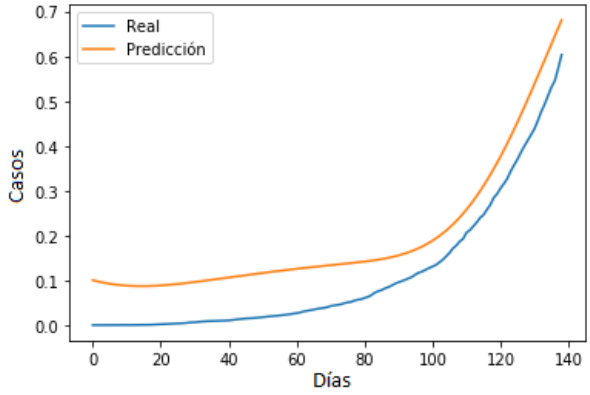
La figura 3 y 4 muestran la serie de tiempo de los valores reales y predichos para el conjunto de datos normalizados de prueba y de entrenamiento respectivamente.

Figura 3. Serie de tiempo de los valores reales y predichos para el conjunto de datos de prueba normalizados para SVR.



Fuente: Elaboración propia.

Figura 4. Serie de tiempo de los valores reales y predichos para el conjunto de datos de entrenamiento normalizados para SVR.



Fuente: Elaboración propia.

Las métricas del rendimiento del modelo SVR son presentadas en la tabla 2; incluye los resultados de las medidas para los datos de entrenamiento y los datos de prueba.

Tabla 2. Métricas del rendimiento del método SVR.

Métrica	Datos de prueba	Datos de entrenamiento
MSE	0,003047	0,007109
RMSE	0,055203	0,084315
MAE	0,047713	0,08294
R2	0,778894	0,681891

Fuente: Elaboración propia.

La tabla 3 presenta la configuración del regresor con los hiperparámetros establecidos por defecto.

Tabla 3. Configuración del regresor con los hiperparámetros por defecto.

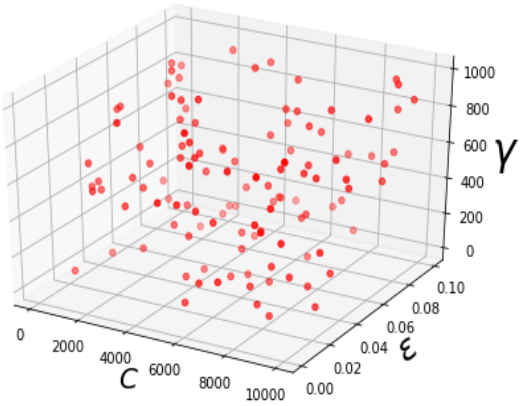
Hiperparámetro	Valor
C	1,0
ε	0,1
γ	0,1
$kernel$	Radial basis function
$cache\ size$	200
$degree$	3
$coef0$	0,0

Fuente: Elaboración propia.

SVR+PSO

De acuerdo con la definición del método propuesto, las partículas que optimizarán los hiperparámetros de la máquina de regresión se inicializan aleatoriamente en el espacio de búsqueda. Para cada partícula se tiene un conjunto de coordenadas (x, y, z), que le indican la ubicación en el espacio y que corresponden respectivamente a los mejores valores de cada uno de los hiperparámetros $(x, y, z) = (C, \varepsilon, \gamma)$. La figura 5 muestra la representación inicial aleatoria del enjambre de partículas en el espacio de búsqueda.

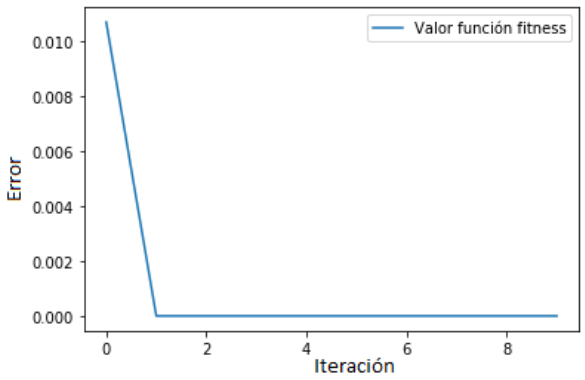
Figura 5. Distribución inicial aleatoria del enjambre de partículas en el espacio de búsqueda.



Fuente: Elaboración propia.

El resultado del comportamiento de la función fitness durante la optimización, determinada por el error cuadrático medio, se muestra en la figura 6. En la primera iteración el error es superior a 0,010, para la segunda el error decae a un valor cercano a cero y se mantiene hasta el final de la optimización. Esto describe que la aplicación del método propuesto mediante el enjambre de partículas permite llegar prontamente a un error mínimo. Para diferentes cantidades de iteraciones (10, 50, 100) el comportamiento es similar, variando únicamente por la naturaleza estocástica del método.

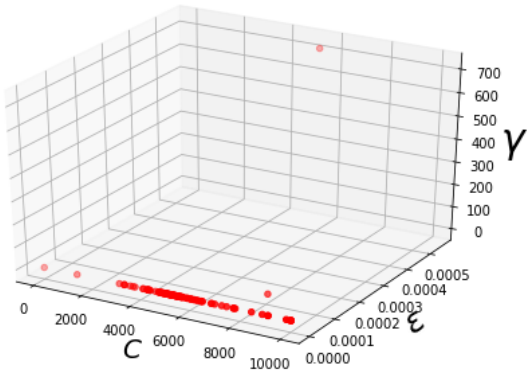
Figura 6. Comportamiento de la función fitness durante la optimización.



Fuente: Elaboración propia.

Tras la ejecución del método, las partículas se han ubicado sobre el óptimo del espacio de búsqueda. La distribución resultante del enjambre sobre el espacio, al finalizar la optimización, se muestra en la figura 7. Los valores de los hiperparámetros obtenidos por el método son $C = 5381{,}051481262206$, $\varepsilon = 1e-08$, $\gamma = 0{,}001$; el espacio de búsqueda corrobora gráficamente estos valores al encontrarse una densidad de partículas sobre C cercano a 5000, ε cercano a 0,0001 y γ a 0.

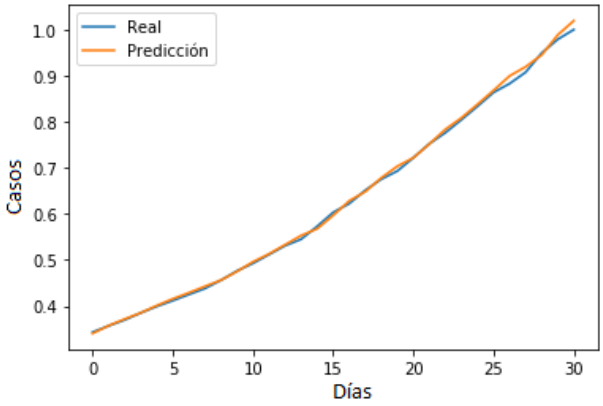
Figura 7. Distribución resultante del enjambre de partículas sobre el espacio de búsqueda.



Fuente: Elaboración propia

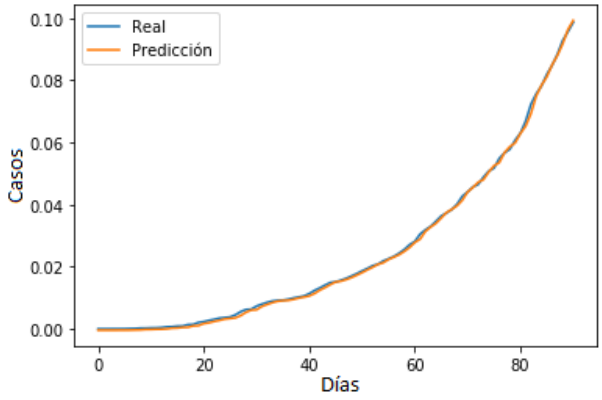
La figura 8 y 9 muestran la serie de tiempo de los valores reales y predichos para el conjunto de datos normalizados de prueba y de entrenamiento respectivamente.

Figura 8. Serie de tiempo de los valores reales y predichos para el conjunto de datos normalizados de prueba para SVR+PSO.



Fuente: Elaboración propia

Figura 9. Serie de tiempo de los valores reales y predichos para el conjunto de datos normalizados de entrenamiento para SVR+PSO.



Fuente: Elaboración propia

Las métricas del rendimiento del método son presentadas en la tabla 4; incluye los resultados de las medidas para los datos de entrenamiento y los datos de prueba.

Tabla 4. Métricas del rendimiento del método SVR+PSO.

Métrica	Datos de prueba	Datos de entrenamiento
MSE	0,000045	0,000001
RMSE	0,006712	0,00072
MAE	0,004917	0,00059
R2	0,998884	0,999279

Fuente: Elaboración propia

La tabla 5 presenta la configuración del mejor regresor obtenido con los hiperparámetros hallados por el enjambre y los establecidos por defecto.

Tabla 5. Configuración del mejor regresor obtenido con los hiperparámetros hallados por el enjambre.

Hiperparámetro	Valor
C	5381,51481262206
ε	1e-08
γ	0,001
$kernel$	Radial basis function
$cache\ size$	200
$degree$	3
$coef0$	0,0

Fuente: Elaboración propia.

Comparación datos de prueba por fecha

La tabla 6 presenta una comparativa de los datos con los cuales se realizó la prueba del método, estos datos han sido reservados previamente para verificar si el proceso de entrenamiento ha resultado un método válido para generar predicciones confiables en comparación con los datos reales registrados por el conjunto de datos original.

Tabla 6. Comparativa de las predicciones para datos de prueba.

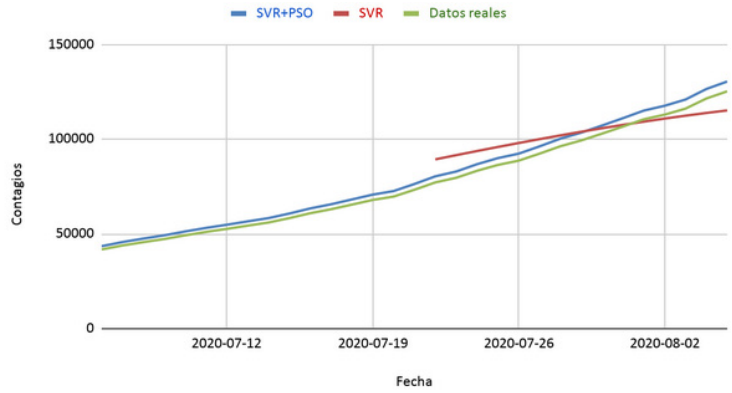
Fecha	SVR+PSO	SVR	Dato real
2020-07-06	43565		41841
2020-07-07	45746		43933
2020-07-08	47578		45691
2020-07-09	49292		47335
2020-07-10	51326		49287
2020-07-11	53204		51089
2020-07-12	54878		52695
2020-07-13	56657		54402
2020-07-14	58383		56058
2020-07-15	60794		58372
2020-07-16	63488		60957
2020-07-17	65686		63066
2020-07-18	68232		65510
2020-07-19	70791		67966
2020-07-20	72657		69757
2020-07-21	76398		73348
2020-07-22	80439	89401	77227
2020-07-23	82978	91618	79664
2020-07-24	86835	93801	83368
2020-07-25	90056	95942	86460
2020-07-26	92415	98038	88726
2020-07-27	96263	100081	92421
2020-07-28	100331	102066	96329
2020-07-29	103523	103988	99395
2020-07-30	107276	105841	103000
2020-07-31	111155	107620	106727
2020-08-01	115159	109321	110575
2020-08-02	117706	110938	113023
2020-08-03	121000	112469	116189
2020-08-04	126562	113908	121536
2020-08-05	130504	115252	125327

Fuente: Elaboración propia.

Es importante resaltar que para el caso propuesto de implementación del algoritmo SVR+PSO se utilizaron más datos de prueba, ya que el entrenamiento se realizó con menos datos en comparación con la implementación que solo utilizó SVR. Se aprecia que inicialmente SVR+PSO presenta un valor más cercano al valor real de contagios la gran mayoría de días, sin embargo al finalizar el mes de Julio la proyección del algoritmo que solo implementa SVR se acerca mucho más, pero a medida que transcurre el mes de agosto esta proyección nuevamente vuelve a alejarse.

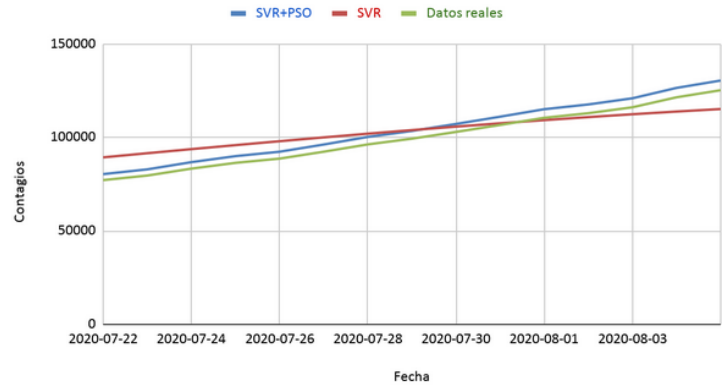
En la figura 10 se observa gráficamente la evolución de las proyecciones de estos algoritmos en comparación con los datos reales para los días correspondientes. Por otro lado, en la figura 11 se tienen en cuenta únicamente los días para los cuales se tienen predicciones tanto para SVR+PSO como para SVR.

Figura 10. Evolución de las proyecciones de la serie de tiempo para los métodos.



Fuente: Elaboración propia

Figura 11. Evolución de las proyecciones de la serie de tiempo para los métodos teniendo en cuenta la misma cantidad de días.



Fuente: Elaboración propia

En definitiva, se puede observar una proyección más regular y estable por parte del algoritmo que implementa SVR+PSO, mientras que el algoritmo que solo implementa SVR no logra una estabilidad en un periodo de tiempo prolongado; la proyección de contagiados inicialmente se encuentra considerablemente por encima, logra acercarse con el transcurso del tiempo, pero al cabo de un par de días nuevamente se aleja, esta vez por debajo del valor real de contagios.

Comparación para el pronóstico

Se propone realizar una predicción de diez días, utilizando estos dos algoritmos, sin tener conocimiento aún de los datos reales reportados por la secretaría de salud de Bogotá; esta predicción arroja los resultados que se encuentran en la tabla 7 para el periodo comprendido desde el día 6 de agosto de 2020, hasta el día 16 de agosto de 2020.

Tabla 7. Comparativa de los pronósticos para los dos métodos.

Fecha	SVR+PSO	SVR
2020-08-06	133274	116498
2020-08-07	137027	117642
2020-08-08	140596	118683
2020-08-09	143563	119618
2020-08-10	148516	120444
2020-08-11	153266	121162
2020-08-12	157492	121769
2020-08-13	162106	122266
2020-08-14	165895	122651
2020-08-15	170736	122925
2020-08-16	173469	123199

Fuente: Elaboración propia

Se puede apreciar que con el pasar de los días esta brecha de proyección entre ambos algoritmos aumenta significativamente.

Análisis comparativo de resultados

Con el propósito de satisfacer la hipótesis planteada acerca de la mejora que brinda el algoritmo PSO en el cálculo de los hiperparámetros y sus resultados sobre la serie de tiempo respecto al método SVR, se presenta en la tabla 8 la comparación de los resultados obtenidos para ambos métodos.

Tabla 8. Comparativa de las métricas para los dos métodos.

	SVR		SVR+PSO	
Métrica	Prueba	Entrenamiento	Prueba	Entrenamiento
MSE	0,003047	0,007109	0,000045	0,000001
RMSE	0,055203	0,084315	0,006712	0,00072
MAE	0,047713	0,08294	0,004917	0,00059
R2	0,778894	0,681891	0,998884	0,999279

Fuente: Elaboración propia.

Para el método propuesto SVR+PSO las métricas de medición del rendimiento MSE, RMSE y MAE, que deben acercarse a cero, se presentan con mejores resultados en comparación a las de SVR. Para la métrica R2, que entre más cercana a 1 es mejor, SVR+PSO presenta un valor de 0,998884 que es significativamente más alto en comparación con el 0,778894 obtenido por SVR.

Discusiones

- ¿Por qué utilizar PSO junto a SVR?

La revisión de la literatura realizada por [13] muestra la amplia aplicación de metaheurísticas para mejorar los métodos de ML y los resultados efectivos que los modelos en conjunto presentan en comparación con las técnicas aplicadas individualmente. En la presente investigación los resultados cuantitativos y cualitativos obtenidos al aplicar PSO junto a SVR son significativamente mejores respecto a la técnica SVR aplicada individualmente, esto en referencia a un conjunto de métricas de medición de rendimiento.

En particular, los hiperparámetros de SVR determinan el rendimiento del modelo predictivo. Sin el uso de PSO la configuración de SVR debe ser "manual", en el sentido que los hiperparámetros deben ser configurados a elección arbitraria y bajo la idea de ensayo-error. Esto implica tener un conocimiento a detalle de las SVR y la naturaleza de los datos que construyen la serie de tiempo. Al utilizar PSO junto a SVR la configuración de los hiperparámetros es ajustada por optimización del enjambre, siendo necesario únicamente el establecimiento del espacio de búsqueda que limite la búsqueda de las partículas.

Aunque el ajuste de los hiperparámetros mediante PSO es mejor que el método de ensayo-error propio de SVR, este sigue siendo dependiente de la configuración de parámetros propios de la metaheurística como la cantidad de partículas o la inercia para ajustar la velocidad de cada partícula, esto sin dejar de lado la naturaleza estocástica de la metaheurística que genera un resultado diferente en cada ejecución.

- ¿Por qué es importante la optimización de los hiperparámetros?

Los hiperparámetros son parámetros ajustables que se eligen para entrenar un modelo y que rigen el propio proceso de entrenamiento; en escenarios de aprendizaje profundo o aprendizaje automático, el rendimiento del modelo depende en gran medida de los valores de hiperparámetro seleccionados [25].

El objetivo de la exploración de los hiperparámetros es buscar entre diversas configuraciones de hiperparámetros hasta encontrar un resultado con un rendimiento aceptable. Normalmente, el proceso de exploración de hiperparámetros es un trabajo manual laborioso, dado que el espacio de búsqueda es muy extenso y la evaluación de cada configuración puede ser costosa. Con la optimización de los hiperparámetros se llega a una configuración que si bien es posible que no sea óptima, es una configuración aceptable y "buena" que resulta importante al momento de simplificar la labor de encontrar los hiperparámetros.

- ¿Por qué no es recomendable utilizar este método para el pronóstico en una pandemia?

Si bien el método SVR+PSO muestra resultados con alta exactitud en el ajuste y regresión, para el proceso de pronóstico en una pandemia no es el más indicado, sin dejar de ser útil, dadas las variables y condiciones que exige un modelo epidemiológico. En los trabajos de [21,31] métodos de regresión y optimización son aplicados para el modelamiento y pronóstico de casos de la enfermedad, sin embargo no se tienen en consideración modelos epidemiológicos. La pandemia de la enfermedad COVID-19 exhibe una naturaleza no lineal y compleja [4] por lo que un modelo de regresión no considera todas las variables necesarias. Los trabajos de [32,33,34] presentan ajustes de parámetros de modelos epidemiológicos SIR, SEIR y SEIJR mediante la interpretación de las variables, las ecuaciones diferenciales presentes y la optimización con PSO. Estos modelos, en términos epidemiológicos, consideran más variables por lo que son más apropiados.

Conclusiones

En este trabajo se aplicó un método de optimización de los hiperparámetros C, ε y γ, para una máquina de regresión de soporte vectorial utilizando el algoritmo bioinspirado de optimización de enjambre de partículas, con el propósito de mejorar el desempeño de la máquina en el pronóstico de una serie de tiempo. El método implementado se aplicó y validó para la predicción de los casos positivo de la enfermedad COVID-19 en la ciudad de Bogotá.

La optimización de los hiperparámetros mediante PSO permite una mejora significativa en la tasa de predicción de una máquina de regresión de soporte vectorial, generando proyecciones confiables para periodos de tiempo cortos. El método SVR+PSO presentó un error cuadrático medio de 0,000045 que supera el de SVR, el cual fue de 0,003047. Para el coeficiente de determinación el resultado obtenido para el método SVR+PSO fue 0,998884 mientras que para el SVR fue 0,778894. Evidenciando mejores resultados cuantitativos el método propuesto.

Si bien el método SVR+PSO presentó un rendimiento aceptable para la serie de tiempo respecto a la técnica de aprendizaje automático SVR, se considera que no es el más indicado para el pronóstico en una pandemia dado que la literatura presenta modelos epidemiológicos sofisticados que utilizan técnicas de inteligencia artificial y de optimización, y que tienen en cuenta la cantidad de variables que se ajustan a la naturaleza no lineal y compleja de la enfermedad.

Aunque el método SVR+PSO presenta un comportamiento aceptable para la optimización de los hiperparámetros y el pronóstico de la serie de tiempo, los resultados están determinados por la naturaleza estocástica de la metaheurística, esto es que para configuraciones diferentes de cantidad de partículas, número de iteraciones e inercia, los resultados presentarán una variación probabilística. Sin embargo, si bien cada prueba del método describe unos valores diferentes, estos no se dispersan de un rango definido que actúa como límite del espacio de búsqueda.

Referencias bibliográficas

1. J. Zheng, “SARS-coV-2: An emerging coronavirus that causes a global threat,” Int. J. Biol. Sci., vol. 16, no. 10, pp. 1678–1685, 2020. DOI: https://doi.org/10.7150/ijbs.45053.
2. A.-B. A. Al-Hussein and R. Tahir, “Epidemiological Characteristics of COVID-19 Ongoing Epidemic in Iraq,” Bull. World Heal. Organ., Apr. 2020 DOI: https://doi.org/10.2471/BLT.20.251561.
3. E. Estrada, “COVID-19 and SARS-CoV-2. Modeling the present, looking at the future,” Phys. Rep., vol. 869, pp. 1–51, Jul. 2020. DOI: https://doi.org/10.1016/j.physrep.2020.07.005.
4. S. Ardabili et al., “COVID-19 Outbreak Prediction with Machine Learning,” SSRN Electron. J., Apr. 2020. DOI: https://doi.org/10.1101/2020.04.17.20070094.
5. A. Remuzzi and G. Remuzzi, “COVID-19 and Italy: what next?,” Lancet, vol. 395, no. 10231, pp. 1225–1228, Apr. 2020. DOI: https://doi.org/10.1016/S0140-6736(20)30627-9.
6. D. Ivanov, “Predicting the impacts of epidemic outbreaks on global supply chains: A simulation-based analysis on the coronavirus outbreak (COVID-19/SARS-CoV-2) case,” Transp. Res. Part E Logist. Transp. Rev., vol. 136, p. 101922, Apr. 2020. DOI: https://doi.org/10.1016/j.tre.2020.101922.

7. F. Koike and N. Morimoto, “Supervised forecasting of the range expansion of novel non-indigenous organisms: Alien pest organisms and the 2009 H1N1 flu pandemic,” Glob. Ecol. Biogeogr., vol. 27, no. 8, pp. 991–1000, Aug. 2018, DOI: https://doi.org/10.1111/geb.12754.
8. N. Agarwal, S. Reddy Koti, S. Saran, and A. S. Kumar, “Data mining techniques for predicting dengue outbreak in geospatial domain using weather parameters for New Delhi, India,” Curr. Sci., vol. 114, no. 11, 2018, Accessed: Aug. 20, 2020. DOI: https://doi.org/10.18520/cs/v114/i11/2281-2291.
9. S. S. Chenar and Z. Deng, “Development of artificial intelligence approach to forecasting oyster norovirus outbreaks along Gulf of Mexico coast,” Environ. Int., vol. 111, pp. 212–223, Feb. 2018. DOI: https://doi.org/10.1016/j.envint.2017.11.032.
10. L. Tapak, O. Hamidi, M. Fathian, and M. Karami, “Comparative evaluation of time series models for predicting influenza outbreaks: Application of influenza-like illness data from sentinel sites of healthcare centers in Iran,” BMC Res. Notes, vol. 12, no. 1, pp. 1–6, Jun. 2019. DOI: https://doi.org/10.1186/s13104-019-4393-y.
11. R. Liang et al., “Prediction for global African swine fever outbreaks based on a combination of random forest algorithms and meteorological data,” Transbound. Emerg. Dis., vol. 67, no. 2, pp. 935–946, Mar. 2020. DOI: https://doi.org/10.1111/tbed.13424.
12. E.-G. Talbi, “Machine Learning for Metaheuristics - State of the Art and Perspectives,” in International Conference on Knowledge and Smart Technology (KST), Apr. 2019, pp. XXIII–XXIII. DOI: https://doi.org/10.1109/kst.2019.8687812.
13. L. Calvet, J. De Armas, D. Masip, and A. A. Juan, “Learnheuristics: Hybridizing metaheuristics with machine learning for optimization with dynamic inputs,” Open Math., vol. 15, no. 1, pp. 261–280, Jan. 2017. DOI: https://doi.org/10.1515/math-2017-0029.
14. M. A. Muñoz, J. A. López, and E. F. Caicedo, “Inteligencia de enjambres: sociedades para la solución de problemas (una revisión) Swarm intelligence: problem-solving societies (a review),” Rev. Ing. E Investig., vol. 28, no. 2, pp. 119–130, 2008.
15. E.-G. Talbi, “Machine learning into metaheuristics: A survey and taxonomy of data-driven metaheuristics,” 2020. Accessed: Aug. 20, 2020. [Online]. Available: https://hal.inria.fr/hal-02745295/document.
16. K. Smets, B. Verdonk, and E. M. Jordaan, “Evaluation of performance measures for SVR hyperparameter selection,” in IEEE International Conference on Neural Networks - Conference Proceedings, 2007, pp. 637–642. DOI: https://doi.org/10.1109/IJCNN.2007.4371031.
17. W. Naudé, “Artificial intelligence vs COVID-19: limitations, constraints and pitfalls,” AI Soc., vol. 1, p. 3, Apr. 2020. DOI: https://doi.org/10.1007/s00146-020-00978-0.
18. Y. Mohamadou, A. Halidou, and P. T. Kapen, “A review of mathematical modeling, artificial intelligence and datasets used in the study, prediction and management of COVID-19,” Appl. Intell., pp. 1–13, Jul. 2020. DOI: https://doi.org/10.1007/s10489-020-01770-9.
19. R. Sujath, J. M. Chatterjee, and A. E. Hassanien, “A machine learning forecasting model for COVID-19 pandemic in India,” Stoch. Environ. Res. Risk Assess., vol. 34, no. 7, pp. 959–972, Jul. 2020. DOI: https://doi.org/10.1007/s00477-020-01827-8.
20. L. Xu, “Modelling to contain pandemic influenza A (H1N1) with stochastic membrane systems: A work-in-progress paper,” in Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, 2012, vol. 87 LNICST, pp. 74–81, DOI: https://doi.org/10.1007/978-3-642-32615-8_10.

21. M. A. A. Al-Qaness, A. A. Ewees, H. Fan, and M. A. El Aziz, "Optimization method for forecasting confirmed cases of COVID-19 in China," Appl. Sci., vol. 9, no. 3, p. 674, Mar. 2020. DOI: https://doi.org/10.3390/JCM9030674.

22. Secretaría Distrital de Salud de Bogotá, "Número de casos confirmados por el laboratorio de COVID- 19 - Bogotá D.C. Datos Abiertos Bogotá," Apr. 07, 2020. https://datosabiertos.bogota.gov.co/dataset/44eacdb7-a535-45ed-be03-16dbbea6f6da (accessed Aug. 19, 2020).

23. "Normalizar datos: referencia para los módulos - Azure Machine Learning | Microsoft Docs," Feb. 22, 2020. https://docs.microsoft.com/es-es/azure/machine-learning/algorithm-module-reference/normalize-data (accessed Aug. 20, 2020).

24. R. Adhikari and R. K. Agrawal, "An Introductory Study on Time Series Modeling and Forecasting," L. Lambert Acad. Publ., Feb. 2013, Accessed: Aug. 20, 2020. [Online]. Available: http://arxiv.org/abs/1302.6613.

25. Microsoft, "Ajuste de los hiperparámetros de un modelo - Azure Machine Learning | Microsoft Docs," Documentación Microsoft, 2020. https://docs.microsoft.com/es-es/azure/machine-learning/how-to-tune-hyperparameters (accessed Aug. 20, 2020).

26. R. Laref, E. Losson, A. Sava, and M. Siadat, "On the optimization of the support vector machine regression hyperparameters setting for gas sensors array applications," Chemom. Intell. Lab. Syst., vol. 184, pp. 22–27, Jan. 2019. DOI: https://doi.org/10.1016/j.chemolab.2018.11.011.

27. M. Awad, R. Khanna, M. Awad, and R. Khanna, "Support Vector Regression," in Efficient Learning Machines, Apress, 2015, pp. 67–80.

28. G. Barrero, "Optimización de hiperparámetros de algoritmos de aprendizaj automático usados para el análisis de la calidad del software," Res. Gate, no. January, 2019. DOI: https://doi.org/10.13140/RG.2.2.15055.74405.

29. S. Kefi, N. Rokbani, and A. M. Alimi, "Impact of ant size on ant supervised by PSO, AS-PSO, performances," in Advances in Intelligent Systems and Computing, Nov. 2017, vol. 552, pp. 567–577. DOI: https://doi.org/10.1007/978-3-319-52941-7_56.

30. K. S. Raghuwanshi, "A Qualitative Review of Two Evolutionary Algorithms Inspired by Heuristic Population Based Search Methods: GA & PSO," in Lecture Notes in Networks and Systems, vol. 18, Singapore: Springer, 2018, pp. 169–175.

31. K. Demertzis, D. Tsiotas, and L. Magafas, "Modeling and forecasting the covid-19 temporal spread in Greece: An exploratory approach based on complex network defined splines," Int. J. Environ. Res. Public Health, vol. 17, no. 13, pp. 1–18, Jul. 2020. DOI: https://doi.org/10.3390/ijerph17134693.

32. A. Godio, F. Pace, and A. Vergnano, "Seir modeling of the italian epidemic of sars-cov-2 using computational swarm intelligence," Int. J. Environ. Res. Public Health, vol. 17, no. 10, May 2020. DOI: https://doi.org/10.3390/ijerph17103535.

33. M. Paggi, "Simulation of Covid-19 epidemic evolution: are compartmental models really predictive?," arXiv.org, Apr. 2020, Accessed: Aug. 20, 2020. [Online]. Available: http://arxiv.org/abs/2004.08207.

34. S. Sun and Y. Zheng, "Prediction of 2019-nCov in Italy based on PSO and inversion analysis," medRxiv, May 2020. DOI: https://doi.org/10.1101/2020.05.08.20095869.