

AUTORIZACIÓN DE USO DE DERECHOS DE AUTOR OTORGADO POR

José Enrique Olvera Vázquez mayor de edad, con domicilio ubicado en calle segunda #230 Sector Benito Juárez, en mi calidad de titular y autor de la tesis denominada **Aplicación de métodos de pronósticos híbridos en base a métodos clásicos y de optimización inteligente en problemas emergentes tipo COVID 19** quien para todos los fines del presente documento se denominará **EL AUTOR Y/O TITULAR**, suscribo el presente documento de autorización de uso de derechos patrimoniales de autor a favor del Instituto Tecnológico de Ciudad Madero el cual se registrá por clausulas siguientes:

PRIMERA – AUTORIZACIÓN: EL AUTOR Y/O TITULAR, mediante el presente documento autoriza la utilización de los derechos patrimoniales de autor al Instituto Tecnológico de Ciudad Madero, de la tesis denominada **Aplicación de métodos de pronósticos híbridos en base a métodos clásicos y de optimización inteligente en problemas emergentes tipo COVID 19**, a través del Repositorio Institucional del Tecnológico Nacional de México (en lo sucesivo TecNM) y en el Repositorio Nacional, que puede ser consultado en la liga electrónica: (<https://www.repositorionacionalcti.mx/>).

SEGUNDA - OBJETO: Por medio del presente escrito, **EL AUTOR Y/O TITULAR** Autoriza al Instituto Tecnológico de Ciudad Madero, a través del Repositorio Institucional del Tecnológico Nacional de México (en lo sucesivo TecNM) y en el Repositorio Nacional para que de conformidad con la Ley Federal del Derecho de Autor y la Ley de la Propiedad Industrial, use los derechos del documento antes referido, con fines exclusivamente académicos.

TERCERA - TERRITORIO: Los derechos aquí Autorizados se dan sin limitación geográfica o territorial alguna.

CUARTA – ALCANCE: La presente autorización se da tanto para formato o soporte material, y se extiende a la utilización en medio óptico, magnético, electrónico, en red, mensajes de datos o similar conocido o por conocer, del ejemplar o número respectivo de la publicación.

QUINTA – EXCLUSIVIDAD: La autorización de uso aquí establecida no implica exclusividad en favor del Instituto Tecnológico de Ciudad Madero. Por lo tanto, **EL AUTOR Y/O TITULAR** en su carácter de autor de la obra objeto del presente documento se reserva el derecho de publicar directamente, u otorgar a cualquier tercero, autorizaciones de uso similares o en los mismos términos aquí acordados.

SEXTA - DERECHOS MORALES (Créditos y mención): La Autorización de los derechos antes mencionados no implica la cesión de los derechos morales sobre los mismos por cuanto en conformidad con lo establecido en los artículos 18, 19, 20, 21, 22 y 23 de la Ley Federal de Derechos de Autor, dada la cuenta que estos derechos son inalienables, imprescriptibles, irrenunciables e inembargables. Por lo tanto, los mencionados derechos seguirán radicados en cabeza de **EL AUTOR Y/O TITULAR**, y siempre deberá mencionarse su nombre cuando se utilice la obra.

SÉPTIMA - AUTORIA: **EL AUTOR Y/O TITULAR**, declara y ratifica que el material objeto de la presente y fue realizada por él (o ella) sin violar o usurpar derechos de Propiedad Intelectual de terceros.

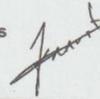
Ciudad Madero, Tamps. a Lunes del 1 día del mes de Diciembre del 2020.

Autor de la Tesis

Asesor de la Tesis



Ing. José Enrique Olvera Vázquez
CVU: 932607



Dr. Juan Frausto Solís
CURP: FASJ500708HBCRLN04

INSTITUTO TECNOLÓGICO DE CIUDAD MADERO

DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

MAESTRÍA EN CIENCIAS DE LA INGENIERÍA



TESIS

**“APLICACIÓN DE MÉTODOS DE PRONÓSTICOS HÍBRIDOS EN BASE A
MÉTODOS CLÁSICOS Y DE OPTIMIZACIÓN INTELIGENTE EN PROBLEMAS
EMERGENTES TIPO COVID 19”**

Que para obtener el grado de:

MAESTRO EN CIENCIAS DE LA INGENIERÍA

Presenta:

Ing. José Enrique Olvera Vázquez

G12071265

Director de Tesis:

Dr. Juan Frausto Solís

Co-Director de Tesis:

Dr. Juan Javier González Barbosa

Ciudad Madero, Tamaulipas. 1 de Diciembre de 2020.

OFICIO DE AUTORIZACIÓN DE IMPRESIÓN (OFICIO ESCANEADO)



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



Instituto Tecnológico de Ciudad Madero
División de estudios de Posgrado e Investigación

"2020, Año de Leona Vicario, Benemérita Madre de la Patria"

Cd. Madero, Tam. 05 de noviembre de 2020

OFICIO No. : U.047/20
ÁREA: DIVISIÓN DE ESTUDIOS
DE POSGRADO E INVESTIGACIÓN
ASUNTO: AUTORIZACIÓN DE IMPRESIÓN DE TESIS

ING. JOSÉ ENRIQUE OLVERA VÁZQUEZ
No. DE CONTROL G12071265
P R E S E N T E

Me es grato comunicarle que después de la revisión realizada por el Jurado designado para su Examen de Grado de Maestra en Ciencias de la Ingeniería, se acordó autorizar la impresión de su tesis titulada:

"APLICACIÓN DE MÉTODOS DE PRONÓSTICOS HÍBRIDOS EN BASE A MÉTODOS CLÁSICOS Y DE OPTIMIZACIÓN INTELIGENTE EN PROBLEMAS EMERGENTES TIPO COVID 19"

El Jurado está integrado por los siguientes catedráticos:

PRESIDENTE:	DR. JUAN FRAUSTO SOLÍS
SECRETARIO:	DR. JUAN JAVIER GONZÁLEZ BARBOSA
VOCAL:	DRA. MARÍA LUCILA MORALES RODRÍGUEZ
SUPLENTE:	DR. LUCIANO AGUILERA VÁZQUEZ

DIRECTOR DE TESIS:	DR. JUAN FRAUSTO SOLÍS
CO-DIRECTOR DE TESIS:	DR. JUAN JAVIER GONZÁLEZ BARBOSA

Es muy satisfactorio para la División de Estudios de Posgrado e Investigación compartir con Usted el logro de esta meta. Espero que continúe con éxito su desarrollo profesional y dedique su experiencia e inteligencia en beneficio de México.

ATENTAMENTE

Excelencia en Educación Tecnológica®
"Por mi patria y por mi bien"®

JOSÉ AARÓN MELO BANDA
JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN



c.c.p.- Archivo

JAMB 'MCI 'ALMS''



Av. 1º de Mayo y Sor Juana I. de la Cruz Col. Los Mangos, C.P. 89440, Cd. Madero, Tam.

Tel. 01 (833) 357 48 20 ext. 3110, e-mail: direccion@tecnm.mx
www.tecnm.mx | www.cdmadero.tecnm.mx



DEDICATORIA

A mis padres, María Martiniana Vázquez Galicia y José Luis Olvera Pérez, por darme la vida, por siempre ver lo mejor y creer en mí, incluso en los momentos en los que yo no lo hacía y por enseñarme el valor del esfuerzo y la constancia.

A mis hermanos, Luis Javier Olvera Vázquez y Jorge Gil Olvera Vázquez, por su apoyo incondicional, por ser ejemplo de superación, dedicación y perseverancia.

A mis amigos, por sus palabras de ánimo, su interés en mis proyectos y su amistad que es una fuente de inspiración humana.

AGRADECIMIENTOS

Al Consejo Nacional de Ciencia y Tecnología (CONACyT), A la División de Estudios de Posgrado e Investigación (DEPI) del Instituto Tecnológico de Ciudad Madero, por el apoyo económico y académico proporcionado para la realización de la maestría y mejorar mi nivel como estudiante.

A mi director, el Dr. Juan Frausto Solís, y Co-director el Dr. Juan Javier González Barbosa, por su gran apoyo académico, su enorme paciencia y conocimiento para la realización de esta Tesis.

A mi comité tutorial Dra. María Lucila Morales Rodríguez, Dr. Luciano Aguilera Vázquez, Dra. Guadalupe Castilla Valdez y el Dr. Guillermo Santamaría Bonfil, por su tiempo y sus valiosas aportaciones al proyecto.

A mis compañeros de la maestría, por estar siempre presentes y por compartir sus experiencias y aprendizajes.

Por su puesto: A Dios, quien me dio la vida y me dio la oportunidad de llegar hasta donde estoy.

“Aplicación de métodos de pronósticos híbridos en base a métodos clásicos y de optimización inteligente en problemas emergentes tipo COVID 19”

José Enrique Olvera Vázquez

Resumen

En la presente tesis muestra la teoría general de pronóstico de series de tiempo y métodos de pronóstico clásicos como la familia ARIMA y los diferentes modelos de Suavizamiento exponencial. Existen diferentes maneras de encontrar las constantes para los modelos, las cuales se muestran las principales en este trabajo. También se habla sobre los métodos de optimización inteligente como son las Maquinas de Soporte Vectorial y los diferentes formas de sintonizar sus parámetros. Se muestra una aplicación en problemas emergentes combinando métodos clásicos de pronóstico y de optimización inteligente.

Se desarrolla la metodología propuesta SVR-ESAR explicando paso a paso los diferentes algoritmos utilizados, el cual incluye un método híbrido entre Maquinas de Soporte Vectorial sintonizada con un algoritmo genético como método base y posteriormente se realiza un ajuste en los residuales con el modelo de Suavizamiento Exponencial de Holt (HSE) y los modelos de Promedio Móvil Integrados Autorregresivos (ARIMA), con la finalidad de disminuir el Error Absoluto Porcentual Medio del pronóstico en series de tiempo.

Finalmente aplica la metodología propuesta SVR-ESAR a las series de tiempo generadas por el número de casos infectados de la enfermedad del COVID-19 obteniendo buenos resultados en la experimentación.

Application of hybrid forecasting methods based on classic methods and intelligent optimization in emerging problems type COVID 19

José Enrique Olvera Vázquez

Abstract

This thesis shows the general forecast theory of classic time series and forecasting methods such as the ARIMA family and the different exponential smoothing models. There are different ways to find the constants for the models, which are shown as the main ones in this work. There is also talk about intelligent optimization methods such as Vector Support Machines and the different ways to tune their parameters. An application is displayed in emerging issues by combining classic forecasting and intelligent optimization methods.

The proposed SVR-ESAR methodology is developed explaining step by step the different algorithms used which includes a hybrid method between Vector Support Machines tuned with a genetic algorithm as the base method and subsequently an adjustment is made to the residuals with the Holt Exponential Smoothing (HES) model and Autorregresive Integrated Moving Average (ARIMA) models, in order to decrease the Mean Absolute Percentage Error of the forecast in time series.

Finally, it applies the proposed SVR-ESAR methodology to the time series generated by the number of cases infected with COVID-19 disease, obtaining good results in the experimentation.

Índice General

Resumen	VI
Abstract.....	VII
Índice Tablas.....	XI
Índice de Figuras	XII
Abreviaturas.....	XIII
1 Introducción	1
1.1 Planteamiento del problema.....	2
1.2 Objetivos	3
1.2.1 Objetivo General.....	3
1.2.2 Objetivos Específicos	3
1.3 Justificación del estudio.....	3
1.4 Organización de la tesis	4
2 Marco Teórico.....	5
2.1 Introducción a los modelos ARIMA (p, d, q)	7
2.1.1 Modelos Autorregresivos AR (p)	7
2.1.2 Modelos de Promedio Móvil MA(q)	8
2.1.3 Modelo de Promedio Móvil Autorregresivo ARMA (p, q).....	9
2.1.4 Modelos de Promedio Móvil Integrados Autorregresivos ARIMA (p, d, q) ...	9
2.2 Suavizamiento Exponencial.....	10
2.2.1 El método de suavizamiento exponencial simple (SE)	10
2.2.2 El método de Holt para suavizamiento exponencial (HSE)	12

2.2.3	El método de Holt-Winters para suavizamiento exponencial	13
2.3	Máquinas de Soporte Vectorial.....	17
2.3.1	Máquinas de Soporte Vectorial para Regresión	19
2.4	Algoritmo Genéticos.....	20
3	Método propuesto: SVR-ESAR.....	22
4	Experimentación, Análisis y Resultados.....	26
5	Conclusiones.....	30
5.1	Contribución del trabajo	30
5.2	Conclusiones	31
5.3	Trabajos publicados	31
	Bibliografía.....	32
	Apéndice A	35
	A CÓDIGOS UTILIZADOS	35
A.1	Código del algoritmo GA-SVR y el algoritmo Aplicar_SVR.....	35
A.2	Código para pronosticar con ARIMA	43
	A continuación, presentamos el código en Python de manera general, así se obtiene el pronóstico de los residuos con ARIMA.	43
A.3	Código para pronosticar Suavizamiento Exponencial de Holt (HSE).....	46
	Apéndice B	47
	B CASOS INFECTADOS	47

B.1 Conjunto de datos de China	47
B.2 Conjunto de datos de Todo el Mundo.....	49
B.3 Conjunto de datos de USA.....	51
B.4 Conjunto de datos de México	53
Apéndice C	55
C Resultados de los pronósticos	55

Índice Tablas

Tabla 2.1	Clasificación de los métodos de suavizamiento exponencial.....	16
Tabla 2.2	Nombre corto y común de los métodos de suavizamiento exponencial.....	16
Tabla 4.1	MAPE del pronóstico de los casos de infectados en todo el mundo y China usando el método de SVR-ESAR con las tres técnicas de ajuste (SVR, HSE y ARIMA)	27
Tabla 4.2	MAE del pronóstico de los casos de infectados en USA y México usando el método de SVR-ESAR con las tres técnicas de ajuste (SVR, HSE y ARIMA).....	29
Tabla B.1	Casos infectados de China del 22 de Enero al 9 de Marzo del 2020.....	47
Tabla B.2	Casos confirmados de China del 10 de Marzo al 25 de Abril del 2020.....	48
Tabla B.3	Casos confirmados de todo el mundo del 22 de Enero al 9 de Marzo del 2020.	49
Tabla B.4	Casos confirmados de todo el mundo del 10 de Marzo al 25 de Abril del 2020.	50
Tabla B.5	Casos confirmados de USA del 22 de Enero al 9 de Marzo del 2020.....	53
Tabla B.6	Casos confirmados de USA del 10 de Marzo al 25 de Abril del 2020	54
Tabla B.7	Casos confirmados de México del 22 de Enero al 9 de Marzo del 2020.....	51
Tabla B.8	Casos confirmados de México del 10 de Marzo al 25 de Abril del 2020.....	52
Tabla C.1	Resultado de los pronósticos del número de infectados de China.....	55
Tabla C.2	Resultado de los pronósticos del número de infectados de todo el mundo.....	55
Tabla C.3	Resultado de los pronósticos del número de infectados de USA.....	56
Tabla C.4	Resultado de los pronósticos del número de infectados de México.....	56

Índice de Figuras

Figura 2.1 Tipos de pronósticos	6
Figura 1.2 Clases linealmente separables.....	17
Figura 1.3 Clases linealmente separables con las distancias y el vector.....	18
Figura 3.1 Estructura del cromosoma.....	22
Figura 3.2 Representación del método de pronóstico híbrido SVR-ESAR	24
Figura 4.1 Pronóstico del número de infectados de todo el mundo y China.....	28
Figura 4.2 Pronóstico del número de infectados de USA y México	29

Abreviaturas

AR	Modelos Autorregresivos
MA	Modelos de Promedio Móvil
ARMA	Modelos de Promedio Móvil Autorregresivos
ARIMA	Modelos de Promedio Móvil Integrados Autorregresivos
SVM	Máquinas de Soporte Vectorial
SVR	Regresión de Vectores de Soporte
SE	Suavizamiento Exponencial
HSE	Suavizamiento Exponencial de Holt
SVR-ESAR	Método Propuesto Híbrido
COVID-19	Enfermedad provocada por el virus SARS-CoV-2
SARS-CoV-2	Virus del SARS
ICTV	International Committee on Taxonomy of Viruses
OMS	Organización Mundial de la Salud
SIRs	Modelos de Ecuaciones diferenciales de compartimientos (Susceptibles, Infectados y Recuperados).
SARS	Síndrome Respiratorio Agudo Grave
GARCH	Modelo Autorregresivo Generalizado Condicional Heterocedástico
NN	Redes Neuronales
MSE	Error Cuadrático Medio
AG	Algoritmo Genético
GA-SVR	Método Híbrido de Sintonización
MAPE	Error Absoluto Porcentual Medio
USA	Estados Unidos de América

1 Introducción

Pronosticar valores futuros representa una ventaja para la toma de decisiones en diferentes áreas de la sociedad. Para producir más es necesario estimar la demanda futura en un periodo de tiempo, ya sea en una panadería o en una fábrica de automóviles es conveniente e importante para no generar demasiado stock y no tener pérdidas financieras.

Existen procedimientos cuantitativos y cualitativos [1] que nos ayudan a estimar valores futuros de una serie de datos que transcurren en el tiempo, utilizando observaciones recopiladas en el pasado, la presente investigación tiene como base utilizar métodos cuantitativos como los modelos de Promedio Móvil Autorregresivos Integrados (ARIMA) [2], Suavizamiento Exponencial (SE) [3] para mejorar los resultados con la metodología propuesta de sintonización basados en algoritmos genéticos para la Regresión por Vectores de Soporte (SVR) [4].

Una de las áreas más importantes en las que es de utilidad el pronóstico de series de tiempo es la de inversiones financieras aplicados en los Portafolios de Inversión, ya que es necesario predecir el rendimiento y riesgo esperados para seleccionar los mejores activos y encontrar una distribución óptima del capital [5]. Otra de las áreas de gran interés es en la predicción del comportamiento de una enfermedad, conocer el número de muertes, contagiados o recuperados es prioridad para saber responder a las necesidades que la población pueda tener.

En marzo del 2020 la Organización Mundial de la Salud (OMS) presentó la enfermedad del COVID-19 como una pandemia, por el factor de propagación y las muertes que había generado en todo el mundo, es por eso que esta tesis titulada **“Aplicación de métodos de pronóstico híbridos en base a métodos clásicos y de optimización inteligente en problemas emergentes tipo COVID-19”** se centró en aplicar la metodología propuesta en el pronóstico del número de casos infectados por dicha enfermedad. Utilizando las librerías y el ambiente de Python como una herramienta importante para el desarrollo del proyecto.

1.1 Planteamiento del problema

A finales del 2019 en un mercado de Wuhan China, se experimentó un brote de un nuevo tipo de coronavirus que mató a más de mil ochocientos e infectó a más de setenta mil en los primeros 50 días [5] con una tasa de mortalidad del 2.9%. La enfermedad fue nombrada como COVID-19 y el virus que la causa como SARS-CoV-2 por el “International Committee on Taxonomy of Viruses (ICTV)”. Actualmente es considerada como una pandemia por la Organización Mundial de la Salud (OMS) y está siendo estudiada por diferentes científicos de todo el mundo. El número de casos de personas infectadas, fallecidas o recuperadas debido a la enfermedad se pueden utilizar para formar series de tiempo y ser analizadas por métodos de pronóstico eficientes que logren explicar su comportamiento y utilizar los resultados para la toma de decisiones adecuada por parte del gobierno. Cabe mencionar que no es la primera enfermedad en la que se han aplicado métodos de pronóstico de series de tiempo como por ejemplo la enfermedad hepatitis A.

El problema de pronosticar los valores futuros de una serie de tiempo ha sido estudiado desde hace mucho tiempo y se ha demostrado que no es posible encontrar el valor real con anterioridad, pero si aproximarlos con un nivel de incertidumbre bajo.

1.2 Objetivos

1.2.1 Objetivo General

Desarrollar un método híbrido de pronóstico en base a métodos clásicos y de optimización inteligente aplicado en series de tiempo generadas por problemas emergentes.

1.2.2 Objetivos Específicos

- **Pronosticar series de tiempo con los modelos de Promedio Móvil Autorregresivos Integrados (ARIMA) y de Suavizamiento Exponencial.**
- **Utilizar un Algoritmo Genético en la sintonización de parámetros que integran una Máquina de Soporte Vectorial para Regresión.**
- **Desarrollar un método híbrido de pronóstico para series de tiempo que mejore o iguale los métodos clásicos.**
- **Aplicar la metodología propuesta en el pronóstico de series de tiempo generadas por problemas emergentes.**

1.3 Justificación del estudio

Predecir el comportamiento de una enfermedad es de suma importancia para tomar las medidas de mitigación adecuadas para que los servicios de salud no se saturen. Siendo muy importante para el gobierno ayudando a predecir costos y el número de vacunas necesarias para la población. A demás al no prever adecuadamente el tamaño del problema se puede relajar las medidas de mitigación y dar como resultado un gran número de personas contagiadas, por el contrario, es importante no entrar en un miedo excesivo como en el caso del virus SARS en el 2003 que ciertos países compraron vacunas que finalmente no fueron necesarias aplicar.

Existen dos enfoques principales para describir el comportamiento de una epidemia el primero es con modelos matemáticos aplicando ecuaciones diferenciales (SIRs) propuestos

por Roland Ross en 1902 y ajustado en 1927 por Kermack y McKendrick [3]. El segundo enfoque es con modelos de predicción utilizando series de tiempo. Por ejemplo, en [6] muestra diferentes métodos aplicados en epidemias. Existe antecedente de técnicas de predicción aplicadas en el comportamiento de epidemias que sirven como base en la toma de decisiones con bajos errores de precisión. Es por eso que en el presente trabajo se propone una metodología híbrida con métodos clásicos y de computo inteligentes para la predicción.

1.4 Organización de la tesis

El desarrollo de la tesis se divide en cinco capítulos. En el capítulo I, se habla sobre el planteamiento del problema, los objetivos de la tesis y la justificación. Para el capítulo II se muestra los trabajos relacionados, los métodos clásicos y de optimización inteligente aplicados en la predicción de series de tiempo utilizados en la metodología propuesta, la clasificación de los diferentes modelos, los métodos clásicos de pronóstico y la teoría relacionada a la metodología de pronóstico propuesta.

El capítulo III se desarrolla la metodología propuesta, describiendo los pasos a seguir e incluyendo el pseudocódigo aplicado en la sintonización de parámetros para el modelo de pronóstico. En el capítulo IV se muestra el diseño de la experimentación, los análisis y resultados para finalizar con el capítulo V con las conclusiones del trabajo.

2 Marco Teórico

El desarrollar pronósticos en el siglo XVII fue una gran innovación para la toma decisiones y en los negocios; durante los siguientes 300 años hubo avances en los métodos de pronóstico basados en datos y utilizando análisis de regresión, descomposición, suavizamiento, y promedios móviles.

La figura 2.1, muestra una taxonomía de métodos de pronóstico diseñada a partir de una clasificación previa de Hanke [1], la cual es auto explicada y comprende los principales métodos de pronóstico y puede ser ampliada en cada categoría, por ejemplo, los métodos de Pronóstico por la Naturaleza podrían incluir, tipos de pronósticos financieros, de energía, industrial y biológicos; los métodos cuantitativos pueden incluir, métodos híbridos y de ensamble. La figura 2.1 nos sirve para dar un panorama general de los métodos de pronóstico.

Pronosticar valores futuros es necesario en cualquier área, a pesar de las imprecisiones inherentes al intentar predecir el futuro, ya que guían el establecimiento de políticas y la planeación [1].

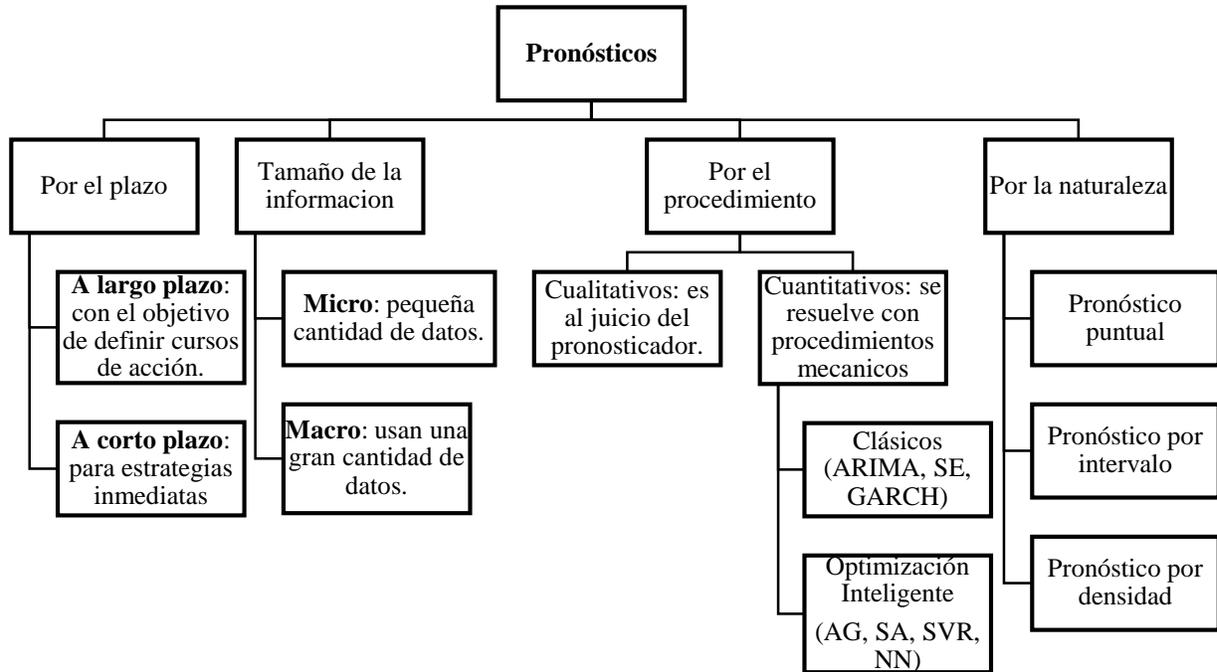


Figura 2.1 Tipos de pronósticos

Basado en: J. E. Hanke and D. W. Wichern [1]

Un factor importante que influye en la selección de la técnica de elaboración del pronóstico es la identificación y comprensión de patrones históricos en los datos. Si se pueden reconocer patrones de tendencia, cíclicos o estacionales, entonces se deben seleccionar las técnicas que sean capaces de extrapolar efectivamente tales patrones. Es importante saber que se pueden combinar diferentes métodos formando modelos híbridos de predicción.

2.1 Introducción a los modelos ARIMA (p, d, q)

Los modelos ARIMA (p, d, q) son una clase de modelos lineales que son capaces de representar tanto series de tiempo estacionarias como no estacionarias.

Una serie de tiempo consiste en datos que se recolectan, se registran o se observan durante incrementos de tiempo sucesivos.

Una serie estacionaria es aquella cuyas propiedades estadísticas básicas, como la media y la varianza, permanecen constantes a través del tiempo [1].

Dos especialistas en estadística, G. E. P. Box y G. M. Jenkins impulsaron una metodología para encontrar el mejor modelo de pronóstico, el cual fue aplicado a los modelos de Promedio Móvil Integrados Autorregresivos (ARIMA), se trata de una estrategia iterativa para la construcción del modelo que consta de 4 pasos [1]. Aplicado en los modelos ARIMA el primer paso es la propuesta de modelo, aquí es importante verificar los parámetros como la autocorrelación y autocorrelación parcial para identificar si se aplican los Modelos Autorregresivos (AR), Modelos de Promedio Móvil (MA) o Modelos de Promedio Móvil Autorregresivos (ARMA) además de analizar si es una serie estacionaria. Para series no estacionarias se pueden convertir a estacionarias por diferenciación, el número de diferenciaciones se denota con la letra d y a estos modelos de series no estacionarias se aplica el modelo denotado por ARIMA (p, d, q). El segundo paso es la estimación de los parámetros para el modelo propuesto y posteriormente en el tercer paso verificar el modelo comparando con los valores históricos y si los residuos son mínimos y no pueden mejorar el pronóstico se pasa a la última etapa o cuarto paso la cual es la elaboración de pronósticos con el modelo.

2.1.1 Modelos Autorregresivos AR (p)

Los modelos AR son modelos Autorregresivos de orden P que tienen la apariencia de un modelo de regresión lineal donde las variables independientes son los valores retrasados de la variable dependiente, de aquí el nombre de modelo autorregresivo [1]. A continuación, se muestra en la ecuación 2.1.

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \varepsilon_t \quad (2.1)$$

Dónde:

Y_t = variable de respuesta (o dependiente) en el tiempo t .

$Y_{t-1}, Y_{t-2}, \dots, Y_{t-p}$ = variable de respuesta en los retrasos de tiempo $t - 1, t - 2, \dots, t - p$, respectivamente; estas Y desempeñan el papel de variables independientes.

$\phi_0, \phi_1, \phi_2, \dots, \phi_p$ = coeficientes estimados.

ε_t = término de error en el tiempo t , el cual representa los efectos de variables no explicadas por el modelo.

Y_t es la variable de respuesta estimada con los valores retrasados en p tiempos valores conocidos de la serie de tiempo. Los valores de ϕ_0, ϕ_1 hasta ϕ_p se pueden encontrar con el modelo de regresión lineal o múltiple y el ε_t es el error que existe entre la respuesta estimada del pronóstico y el valor real de la serie.

2.1.2 Modelos de Promedio Móvil MA(q)

Los modelos de promedio móvil (MA) se refieren a la desviación entre el valor real y la media de los datos representados como Y_t y μ respectivamente, es una combinación lineal de los errores actuales y pasados y que, conforme el tiempo avanza, los errores implicados en esta combinación lineal también se mueven hacia delante [1] como se muestra en la ecuación 2.2. Las variables ε y ω son los errores y pesos por cada error retrasado q veces.

$$Y_t = \mu + \varepsilon_t - \omega_1 \varepsilon_{t-1} - \omega_2 \varepsilon_{t-2} - \dots - \omega_q \varepsilon_{t-q} \quad (2.2)$$

Donde:

Y_t = variable de respuesta (dependiente) en el tiempo t

μ = promedio constante en el proceso

$\omega_1, \omega_2, \dots, \omega_q$ = coeficientes estimados

ε_t = término de error, el cual representa los efectos de las variables no explicadas por el modelo

$\varepsilon_{t-1}, \varepsilon_{t-2}, \dots, \varepsilon_{t-q}$ = errores en periodos anteriores que, para el tiempo t, se incorporan a la respuesta, Y_t

2.1.3 Modelo de Promedio Móvil Autorregresivo ARMA (p, q)

Los modelos de promedio móvil autorregresivo (ARMA) es la combinación entre los modelos AR y MA con orden (p, q) como se muestra en la ecuación 2.3. Las variables descritas son iguales a los modelos AR y MA mostrados anteriormente estos pueden describir una gran variedad de comportamientos para series estacionarias.

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \varepsilon_t - \omega_1 \varepsilon_{t-1} - \omega_2 \varepsilon_{t-2} - \dots - \omega_q \varepsilon_{t-q} \quad (2.3)$$

2.1.4 Modelos de Promedio Móvil Integrados Autorregresivos ARIMA (p, d, q)

Si combinamos la diferenciación con la autorregresión y un modelo de media móvil, obtenemos un modelo ARIMA no estacional. ARIMA es un acrónimo de Promedio Móvil Integrado Autorregresivo (en este contexto, "integración" es el inverso de la diferenciación). El modelo completo se puede escribir en la ecuación 2.4.

$$Y'_t = \phi_0 + \phi_1 Y'_{t-1} + \phi_2 Y'_{t-2} + \dots + \phi_p Y'_{t-p} + \varepsilon_t - \omega_1 \varepsilon_{t-1} - \omega_2 \varepsilon_{t-2} - \dots - \omega_q \varepsilon_{t-q} \quad (2.4)$$

Donde Y'_t es la serie diferenciada (puede haber sido diferenciada más de una vez). Los "predictores" en el lado derecho incluyen valores retardados de Y_t errores retrasados.

Las mismas condiciones de estacionariedad e invertibilidad que se utilizan para los modelos de media móvil y autorregresiva también se aplican a un modelo ARIMA.

2.2 Suavizamiento Exponencial

2.2.1 El método de suavizamiento exponencial simple (SE)

El método de suavizamiento exponencial simple genera pronósticos con base en el promedio de valores pasados de una serie con una curva decreciente (exponencial) de ponderación o pesos [1]. Este modelo es adecuado para datos que no tienen una tendencia clara ascendente o descendente o patrón estacional. A final de la década de 1950 Brown en 1959; Holt en 1957 y Winters en 1960 propusieron el método de suavizamiento exponencial [2]. La estimación de los valores futuros es dependiente de los pesos que se otorgan a los valores pasados, el valor mayor del peso α donde ($0 < \alpha < 1$) es otorgado a la observación más reciente. Y cada vez disminuye el peso para las observaciones pasadas $\alpha(1-\alpha)$, $\alpha(1-\alpha)^2$, $\alpha(1-\alpha)^3$ y así sucesivamente.

El valor futuro resultado del pronóstico es la sumatoria de la multiplicación de los pesos por los valores anteriores como se muestra a continuación:

Pronóstico = [α x (nueva observación)] + [(1- α) x (último pronóstico)]

Formalmente la ecuación 2.5 se representa de la siguiente manera.

$$\hat{Y}_{t+1} = \alpha Y_t + (1 - \alpha) \hat{Y}_t \quad (2.5)$$

donde

\hat{Y}_{t+1} = nuevo valor suavizado o el valor del pronóstico para el siguiente periodo

α = constante de suavización ($0 < \alpha < 1$)

Y_t = nueva observación o el valor real de la serie en el periodo t

\hat{Y}_t = último valor suavizado o el pronóstico del periodo t

Esta ecuación se puede modificar $\hat{Y}_{t+1} = \hat{Y}_t + \alpha(Y_t - \hat{Y}_t)$. De esta forma, el nuevo pronóstico (\hat{Y}_{t+1}) será igual al pronóstico anterior (\hat{Y}_t) más α veces el error en el pronóstico antiguo ($Y_t - \hat{Y}_t$).

Para la ecuación 2.5 si el valor de α se acerca a 1 significa que existe mayor influencia en el pronóstico actual por parte de la observación anterior, esto significa que los pronósticos anteriores tendrán menor influencia en el resultado. La ecuación 2.5 se puede explicar de otra manera, dejándola como dependiente de los pronósticos anteriores (t-n).

Para el tiempo t, $\hat{Y}_t = \alpha Y_{t-1} + (1 - \alpha)\hat{Y}_{t-1}$ y al sustituir en la ecuación 2.5 se obtiene la ecuación 2.6.

$$\hat{Y}_{t+1} = \alpha Y_t + (1 - \alpha)[\alpha Y_{t-1} + (1 - \alpha)\hat{Y}_{t-1}]$$

$$\hat{Y}_{t+1} = \alpha Y_t + \alpha(1 - \alpha)Y_{t-1} + (1 - \alpha)^2\hat{Y}_{t-1} \quad (2.6)$$

La sustitución continua de \hat{Y}_{t-n} sucesivamente indica que \hat{Y}_{t+1} puede escribirse como la suma de la Y actual y las anteriores con pesos que descienden exponencialmente como se muestra en la ecuación 2.7.

$$\hat{Y}_{t+1} = \alpha Y_t + \alpha(1 - \alpha)Y_{t-1} + \alpha(1 - \alpha)^2Y_{t-2} + \alpha(1 - \alpha)^3Y_{t-3} \dots + \alpha(1 - \alpha)^nY_{t-n} \quad (2.7)$$

El valor asignado a α es la clave del análisis. Si se desea que las predicciones sean estables y las variaciones aleatorias se suavicen, se requiere un valor pequeño de α . Si se desea una respuesta rápida a un cambio real en el patrón de observaciones, un valor más grande de α es el apropiado.

Un método para estimar α es un procedimiento iterativo que minimiza el error cuadrático medio (MSE) dado por la ecuación 2.8. Los pronósticos se calculan para $\alpha = 0.1, 0.2, \dots, 0.9$

y se calcula la suma de los errores cuadráticos del pronóstico de cada uno de ellos [2] dejando el de menor MSE.

$$MSE = \frac{1}{n} \sum_{t=1}^n (Y_t - \hat{Y}_t)^2 \quad (2.8)$$

2.2.2 El método de Holt para suavizamiento exponencial (HSE)

Holt en el 2004 desarrolló un método de suavizamiento exponencial que se ajusta a la tendencia local de una serie generando así pronósticos [3]. Este método implica dos ecuaciones, una para el nivel y la otra para la tendencia, las cuales son suavizadas por constantes. Al método de suavizamiento exponencial simple se agregan L_t y T_t que son el nivel actual estimado y el estimado de tendencia respectivamente.

Las ecuaciones del método son:

1.- La serie suavizada exponencialmente o nivel actual estimado.

$$L_t = \alpha Y_t + (1 - \alpha) (L_{t-1} + T_{t-1}) \quad (2.9)$$

2.- El estimado de la tendencia.

$$T_t = \beta (L_t - L_{t-1}) + (1 - \beta) T_{t-1} \quad (2.10)$$

3.- El pronóstico para los p periodos del futuro

$$\hat{Y}_{t+p} = L_t + p T_t \quad (2.11)$$

Donde:

L_t = nuevo valor suavizado (estimado del nivel actual)

$\alpha = 1$, constante de suavización para el nivel ($0 < \alpha < 1$)

Y_t = nueva observación o valor real de la serie en el periodo t

β = constante de suavización para el estimado de la tendencia ($0 < \beta < 1$)

T_t = estimado de tendencia

p = periodos a pronosticar en el futuro

\hat{Y}_{t+p} = pronóstico para el periodo p en el futuro

2.2.3 El método de Holt-Winters para suavizamiento exponencial

El método de suavización exponencial lineal y estacional de tres parámetros de Winters es una extensión del método de Holt el cual emplea una ecuación para estimar la estacionalidad de la serie. Existen dos métodos que dependen de las variaciones estacionales, para aquellas que tienden a ser constantes se aplica el método aditivo, en cambio si las variaciones cambian proporcionalmente al nivel de la serie se aplica el método multiplicativo. En el método multiplicativo, la estimación de estacionalidad está dada por un índice estacional como se muestra en la ecuación.

Esta última indica que para calcular el componente estacional actual, S_t , el producto de γ y un estimado del índice estacional dado por Y_t / L_t se suma $(1 - \gamma)$ veces al componente estacional previo S_{t-s} . Este procedimiento es equivalente a suavizar los valores previos y actuales de Y_t / L_t . Y_t se divide entre el nivel actual estimado L_t , para crear un índice (razón) que pueda usarse de forma multiplicativa para ajustar un pronóstico que tome en cuenta los picos y valles estacionales.

Las cuatro ecuaciones usadas en la suavización (multiplicativa) de Winters son:

1.- Series suavizadas exponencialmente o nivel estimado:

$$L_t = \alpha \frac{Y_t}{S_{t-s}} + (1 - \alpha) (L_{t-1} + T_{t-1}) \quad (2.12)$$

2.- Estimación de la tendencia:

$$T_t = \beta(L_t - L_{t-1}) + (1 - \beta) T_{t-1} \quad (2.13)$$

3.- Estimado de estacionalidad:

$$S_t = \gamma \frac{Y_t}{L_t} + (1 - \gamma) S_{t-s} \quad (2.14)$$

4.- Pronóstico de p periodos futuros:

$$\hat{Y}_{t+p} = (L_t + p T_t) S_{t-s+p} \quad (2.15)$$

Donde:

L_t = nuevo valor suavizado (estimado de nivel actual)

α = constante de suavización del nivel

Y_t = nueva observación o valor real en el periodo t

β = constante de suavización para el estimado de tendencia

T_t = estimado de tendencia

γ = constante de suavización para el estimado de estacionalidad

S_t = estimado de estacionalidad

p = periodos futuros a pronosticarse

s = longitud de la estacionalidad

\hat{Y}_{t+p} = el pronóstico para el periodo p en el futuro

Al igual que en la suavización exponencial lineal de Holt, los pesos α , β y γ pueden seleccionarse subjetivamente o generarse al minimizar una medida de error de pronóstico, como el MSE [1].

Con el método aditivo, el componente estacional se expresa en términos absolutos en la escala de la serie observada, y en la ecuación de nivel, la serie se ajusta estacionalmente restando el componente estacional. Dentro de cada año, el componente estacional se sumará aproximadamente a cero.

Las ecuaciones del método aditivo son:

1.- Series suavizadas exponencialmente o nivel estimado:

$$L_t = \alpha(Y_t - S_{t-s}) + (1 - \alpha) (L_{t-1} + T_{t-1}) \quad (2.16)$$

2.- Estimación de la tendencia:

$$T_t = \beta(L_t - L_{t-1}) + (1 - \beta) T_{t-1} \quad (2.17)$$

3.- Estimado de estacionalidad:

$$S_t = \gamma(Y_t - L_{t-1} - T_{t-1}) + (1 - \gamma)S_{t-s} \quad (2.18)$$

4.- Pronóstico de p periodos futuros:

$$\hat{Y}_{t+p} = L_t + p T_t + S_{t-s+p} \quad (2.19)$$

Existen más métodos de suavizamiento exponencial que también dependen de las variaciones en la estacionalidad y su tendencia haciendo posible la combinación entre diferentes métodos de suavizamiento. [7] incluyó un método con una tendencia multiplicativo, [8] con tendencia amortiguada aditiva y [9] una tendencia amortiguada multiplicativa. En total sin considerar métodos que incluyen modelos espaciales de estado son 9 entre sus diferentes combinaciones de los antes ya mencionados. Como se muestra en las tablas 2.1 y 2.2.

Tabla 2.1 Clasificación de los métodos de suavizamiento exponencial

Basado en: R. Hyndman and G. Athanasopoulos [2]

Componente de tendencia	Componente estacional		
	N(Ninguno)	A(Aditivo)	M(multiplicativo)
N(Ninguno)	(N,N)	(N,A)	(N,M)
A(Aditivo)	(A,N)	(A,A)	(A,M)
A(Aditivo humedecido) _d	(A _d	(A _d	(A _d

Tabla 2.2 Nombre corto y común de los métodos de suavizamiento exponencial

Basado en: R. Hyndman and G. Athanasopoulos [2]

Nombre corto	Nombre común
(N,N)	Suavizamiento exponencial simple.
(A,N)	Método lineal de Holt
(A _d	Método de tendencia amortiguada aditiva
(A,A)	Método de Holt-Winters aditivo
(A,M)	Método de Holt-Winters multiplicativo.
(A _d	Método humedecido de Holt-Winters

2.3 Máquinas de Soporte Vectorial

Las Máquinas de Soporte Vectorial (SVM) es una técnica de aprendizaje supervisado desarrollada por Vapnik en 1995 [10]. Esta técnica funciona para clasificar, los datos de entrada son transformados a una dimensión nueva, normalmente de una dimensión superior con el objetivo de clasificarlos con hiperplanos de separación [11].

El objetivo de las Máquinas de Soporte Vectorial es separar las clases mediante una función lineal o hiperplano. Para un clasificador lineal aplicado en clases linealmente separables. Se utiliza una estructura binaria de entradas $(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)$ donde $x \in \mathbb{R}^2$ y $y \in \{+1, -1\}$, si x_k pertenece a la clase A entonces $y_k = 1$ o Si x_k pertenece a la clase B entonces $y_k = -1$ como se muestra en la figura 2.2.

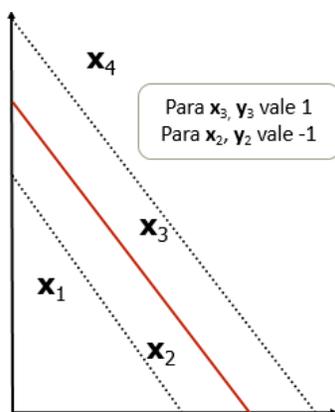


Figura 2.2 Clases linealmente separables

La formulación inicial del caso lineal de SVR se puede deducir de la figura 2.3 encontrando primeramente el margen con la ecuación 2.20 para después sustituir la ecuación 2.21 deducida de las razones trigonométricas aplicadas al triángulo rectángulo que se observa en la figura 2.3 dando como resultado la ecuación 2.22.

$$M = D_1 - D_2 \quad (2.20)$$

$$\cos(\alpha) = \frac{D_1}{\|X_1\|} \quad (2.21)$$

$$M = \|X_1\| \cos(\alpha) - \|X_2\| \cos(\beta) \quad (2.22)$$

Donde:

M es el margen o distancia del tubo,

D la distancia entre los vectores de soporte al origen

X los puntos a clasificar

α y β los ángulos entre el vector y los puntos a clasificar.

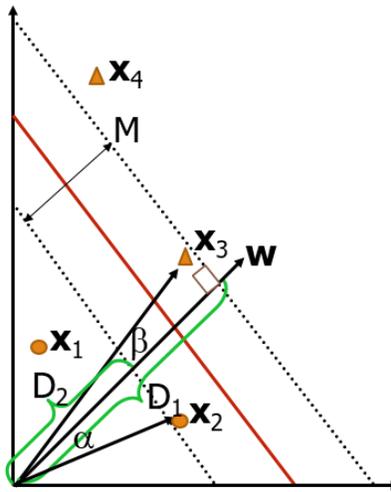


Figura 2.3 Clases linealmente separables con las distancias y el vector.

En la ecuación 2.22 se sustituye los valores de la ecuación 2.23 y 2.24 obteniendo la ecuación 2.25 al reducir los términos y operaciones

$$\cos(\alpha) = \frac{X_1^T W}{\|X_1\| \|W\|} \quad (2.23)$$

$$\cos(\beta) = \frac{X_2^T W}{\|X_2\| \|W\|} \quad (2.24)$$

$$M = \frac{2}{\|W\|} \quad (2.25)$$

Para obtener el menor margen posible podemos deducir de la ecuación 2.25 se debe minimizar $\|W\| = w^T w = \|w^2\|$ por lo tanto la función objetivo y restricciones se puede observar a continuación en las ecuaciones 2.26, 2.27 y 2.28.

$$\text{Minimizar} \quad \|w^2\| \quad (2.26)$$

$$\text{Sujeto a} \quad y_i[w^T x_i + b] \geq 1 \quad (2.27)$$

$$i = 1, k \quad (2.28)$$

Donde:

k es el número de ejemplos de entrenamiento

b es la abscisa al origen

w son los vectores de peso

2.3.1 Máquinas de Soporte Vectorial para Regresión

El método de Regresión de Vectores de Soporte (SVR) es una ampliación de las Máquinas de Vectores de Soporte (SVM) el cual sirve para clasificar un conjunto de datos mediante vectores de soporte. Aunque es una ampliación utiliza los mismos principios que el SVM: minimizar el error, individualizar el hiperplano que maximiza el margen, teniendo en cuenta que se tolera parte del error. El SVR sirve para datos lineales o no lineales, se puede representar de la siguiente forma [12].

$$\text{Minimizar} \quad \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^{\ell} (\xi_i + \xi_i^*) \quad (2.29)$$

$$y_i - \langle \omega, x_i \rangle - b \leq \varepsilon + \xi_i \quad (2.30)$$

$$\text{Sujeto a} \quad \langle \omega, x_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \quad (2.31)$$

$$\xi_i, \xi_i^* \geq 0 \quad (2.32)$$

En donde:

ω vector de pesos

C es una constante y debe ser mayor a 0, determina el equilibrio entre la regularidad de la función y la cuantía hasta la cual toleramos desviaciones mayores que las bandas de soporte.

ξ_i, ξ_i^* son las variables que controlan el error cometido por la función de regresión al aproximar a las bandas.

El objetivo de SVR es encontrar una función $f(x)$ que tenga una desviación de hasta ε de las observaciones reales y_i , y que al mismo tiempo conserve la horizontalidad de la función decisoria [13] que a continuación se muestran en las ecuaciones (2.33) y (2.34).

$$W^* = \sum_{i=1}^s (\alpha_i - \alpha_i^*) k(x_i, x_j) \quad (2.33)$$

$$f(x, \alpha_i, \alpha_i^*) = \sum_{i=1}^s (\alpha_i - \alpha_i^*) k(x_i, x_j) \quad (2.34)$$

2.4 Algoritmo Genéticos

Los algoritmos heurísticos encuentran soluciones de calidad para problemas combinatorios complejos como los de tiempo no polinomial (NP) [14]. Tienen ventajas en relación a los algoritmos polinomiales por ejemplo en la dificultad de la aplicabilidad y además en el

esfuerzo computacional, aunque teóricamente no se está seguro de llegar al óptimo global del problema. Este tipo de algoritmos se pueden clasificar en constructivos (golosos), algoritmos de descomposición y división, algoritmos de reducción, algoritmos de manipulación del modelo y algoritmos de búsqueda usando vecindad [15]. Los algoritmos genéticos entran en la última clasificación

Los algoritmos genéticos se inspiran en las teorías evolutivas de Darwin sobre selección natural introducidos por Holland en los años cincuenta y publicada en su libro “Adaptation in Natural and Artificial Systems” en 1975 [16]. De grandes poblaciones van sobreviviendo las características más fuertes de cada individuo de manera natural mediante la facilidad de adaptación de cada descendiente y aquellos con las peores características se van eliminando. A estas características se les conoce como genes y para un Algoritmo Genético (AG) los individuos serán los cromosomas, no necesariamente sucede de esta forma, siempre cabe la posibilidad de la supervivencia de individuos con genes no tan fuertes mediante la mutación o en la selección del individuo [17].

Un algoritmo genético está compuesto por 4 fases principalmente:

- 1.- Selección: se eligen con mayor probabilidad a los mejores individuos.
- 2.- Cruce: se combinan los genes de cada individuo seleccionados anteriormente.
- 3.- Mutación: se cambia los genes de un individuo con cierta probabilidad ya sean de posición o por algún otro al azar, cuidando que no sea muy seguido.
- 4.- Eliminación: Es necesario eliminar individuos de la población para que sea más eficiente el manejo por lo cual se van los peor adaptados.

Encontrar parámetros óptimos en un algoritmo puede ser un problema de combinatoria complejo es por eso que los AG son considerados en [13]. En el presente trabajo se utiliza para encontrar los mejores parámetros de la Máquina de Soporte Vectorial para Regresión.

3 Método propuesto: SVR-ESAR

En este trabajo se presenta un método de pronóstico híbrido que utiliza un algoritmo genético sintonizando los parámetros de una máquina de soporte (GA-SVR) como se muestra en el algoritmo 1 GA-SVR y una fase de ajuste al cual llamaremos SVR-ESAR la cual se muestra en la figura 3.2. La Arquitectura SVR-ESAR tiene dos fases.

❖ La Fase 1 obtiene un pronóstico $F(t)$ de la serie de tiempo con el algoritmo 1 GA-SVR, el cual inicia generando una población con individuos que tengan la estructura de la figura 3.1, después para cada individuo se ejecuta la SVR y se obtiene el Error Absoluto Porcentual Medio (MAPE) del pronóstico, como se muestra en la línea 4, para así ordenarlos de menor a mayor MAPE y obtener el mejor individuo. Después de evaluar la primera población, inicia un ciclo cuya condición de paro es el número de generaciones o la no mejora, en este ciclo se selecciona a los padres mediante un torneo (línea 8), se realiza una cruce agregando los hijos a la población (línea 9) y se hace una mutación tipo perturbación en los hijos generados para agregarlos a la población en la línea 10.

Kernel	C	Épsilon
Gen 1	Gen 2	Gen 3

Figura 3.1 Estructura del cromosoma

ALGORITMO 1 GA-SVR

```

1: Inicio
2: Generar Población_Inicial_Aleatoria
3: Para Cada_Individuo_Población
4:   MAPE(Individuo) = Aplicar_SVR(individuo) // para obtener el MAPE (Fitness) utilizar entrenamiento
5:   Ordenar población de menor a mayor MAPE
6:   Mejor_Individuo_Global = Mejor_Individuo_Población_Inicial
7:   Mientras (Num_Gen ≤ Lim_Gen) y (Num_Gen_Sin_Mejora ≤ Lim_Sin_Mejora)
8:     Seleccionar por torneo
9:     Cruzar padres seleccionados y actualizar población //agregar hijos a población
10:    Mutar hijos y actualizar población //agregar hijos a población
11:    Para Cada_Individuo_Población
12:      MAPE(Individuo) = Aplicar_SVR(individuo) // para obtener el MAPE (Fitness) utilizar entrenamiento
13:      Mejor_Individuo_Generación = Ordenar(Población) // de menor a mayor MAPE
14:      if (Mejor_Individuo_Generación ≥ Mejor_Individuo_Global):
15:        Num_Gen_Sin_Mejora = Num_Gen_Sin_Mejora +1
16:      else:
17:        Num_Gen_Sin_Mejora = 0
18:        Mejor_Individuo_Global = Mejor_Individuo_Generación
19:   Fin Mientras
20: Parametros_Sintonizados_SVR = Mejor_Individuo_Global
21: Fin

```

Para cada individuo del proceso anterior se ejecuta la SVR, se obtiene su MAPE y se ordena para obtener el mejor individuo de la generación comparando con el mejor individuo global. Si el MAPE del mejor individuo de la generación es menor o igual al mejor individuo global, aumenta el contador de generaciones sin mejora, de lo contrario, se actualiza el mejor individuo global. Cuando se cumplan las condiciones de paro, se obtienen los mejores parámetros (el mejor individuo global) línea 20.

Con los mejores parámetros se utiliza el algoritmo Aplicar_SVR y se obtiene el pronóstico $F(t)$ para el periodo de validación y prueba de una serie en específico.

ALGORITMO 2 Aplicar_SVR

```

1: Inicio
2: Leer Datos
3: X = Datos [Días]
4: Y = Datos [Serie de tiempo]
5: X_train, X_test = (90%)X, (10%)X
6: y_train, y_test = (90%)Y, (10%)Y
7: svr = SVR(kernel = K, C = c, epsilon = E, gamma='auto', degree=2, coef0 = 1.0)
8: svr.fit(X_train, y_train)
9: Y_pred = svr.predict(X_test)
10: MAPE = evaluar_MAPE(Y_pred, y_test)
11: Return (Y_pred, MAPE)
12: Fin

```

El algoritmo Aplicar_SVR inicia leyendo la serie de tiempo (número de infectados) y separando los datos en entrenamiento y verificación en las líneas 5 y 6. El algoritmo recibe el cromosoma (kernel, C y Épsilon), además de otros parámetros, en la línea 7, para después entrenar el modelo en la línea 8 y hacer la predicción en la línea 9. Se calcula el MAPE y regresa el pronóstico y el valor del MAPE. En el apéndice A se muestra de manera general el código en Python del Algoritmo GA-SVR y del Aplicar_SVR.

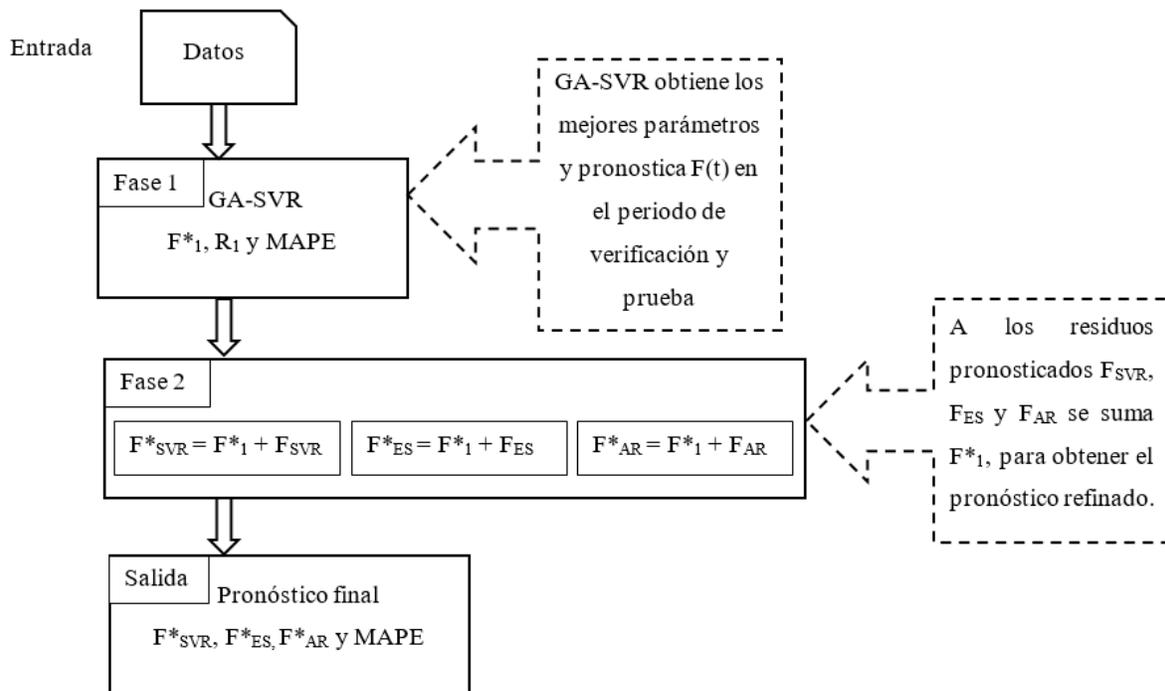


Figura 3.2 Representación del método de pronóstico híbrido SVR-ESAR

El pronóstico $F(t)$ es mejorado ($F(t) \rightarrow F^*_1(t)$) utilizando suavizamiento exponencial simple. El suavizamiento exponencial simple genera pronósticos con base en el promedio de valores pasados de una serie con una curva decreciente (exponencial) de ponderación o pesos [1]. Estos pesos son estimados de manera iterativa hasta encontrar el mejor pronóstico mejorado que arroje un error bajo.

Finalmente se obtiene el pronóstico mejorado $F^*_1(t)$ su MAPE y los residuos R_1 del periodo de validación. Así como, $F^*_1(t)$ y su MAPE del periodo de prueba. Los residuos R_1 son la diferencia entre el valor real y $F^*_1(t)$.

❖ Fase 2: En esta fase se efectúa un refinamiento al pronóstico mejorado $F^*_1(t)$, para lo cual se le aplican tres técnicas alternativas (GA-SVR, Suavizamiento Exponencial de Holt (HSE), y ARIMA [3]) para pronosticar los residuos (R1) en el periodo de prueba (F_{SVR} , F_{ES} y F_{AR}). El algoritmo GA-SVR fue descrito en la fase 1, HSE y ARIMA han sido descritos en [9], también se muestra el código general para realizar el pronóstico en el apéndice A.

A los residuos pronosticados (F_{SVR} , F_{HSE} y F_{ARIMA}) se suma $F^*_1(t)$, dando como resultado un pronóstico refinado (F^*_{SVR} , F^*_{HES} , F^*_{AR}), paso seguido se obtiene el MAPE de F^*_{SVR} , F^*_{ES} y F^*_{AR} .

4 Experimentación, Análisis y Resultados

Este trabajo propone un método para el pronóstico de series de tiempo, y dada la contingencia actual de la Pandemia originada por la enfermedad del COVID-19, se prueba el método propuesto en las series de tiempo generadas por el registro de los casos infectados relacionados a la enfermedad.

El método de pronóstico presentado se aplicó a personas infectadas con la enfermedad del COVID-19. El período analizado de las series temporales es del 22 de enero al 25 de abril de 2020 [18], las tablas de cada registro se muestra en el apéndice B; hemos tomado los datos del repositorio de GitHub (<https://github.com/CSSEGISandData/COVID19/>). En [19] y [20], el pronóstico se aplicó para los últimos diez días. Por lo tanto, utilizamos los mismos parámetros para comparar sus resultados con nuestro pronóstico, cabe mencionar que solo se toma en cuenta el registro de cada escenario. Teniendo cuatro escenarios:

- El mundo entero: los resultados de pronóstico obtenidos para las personas infectadas [15]. Se comparó los resultados obtenidos con nuestro método de pronóstico propuesto utilizando los casos infectados.

- China: este país fue el origen de la pandemia. Hay resultados publicados para este caso [4], que usamos en esta sección para validar el método propuesto.
- Estados Unidos de América (USA): Este país tiene el mayor número de incidentes del continente; además, es importante porque tiene mucha comunicación con el resto del mundo.
- México implementó algunas políticas particulares y tiene muchas relaciones con los países, particularmente con los Estados Unidos de América (USA) y Canadá.

La tabla 4.1 presenta los resultados para todo el mundo y China, obtenidos con la aplicación del método propuesto para el caso de pronóstico confirmado. En esta tabla, mostramos los resultados de SVR-ESAR utilizando las técnicas de ajuste SVR, Suavizamiento Exponencial de Holt (HSE) y ARIMA. Notamos que todos estos tienen un pequeño valor de error MAPE. También presentamos los resultados obtenidos para los casos en [20] y [19]. Observamos que SVR-ESAR con ARIMA logró los mejores resultados para estos casos. La figura 4.1 muestra los resultados obtenidos por SVR-ESAR para los escenarios 1 y 2.

Tabla 4.1 MAPE del pronóstico de los casos infectados en todo el mundo y China usando el método de SVR-ESAR con las tres técnicas de ajuste (SVR, HSE y ARIMA)

Escenario 1: Infectados en todo el mundo				
Métodos de pronóstico	Suavizamiento Exponencial [3][20]	SVR-ESAR		
	Técnicas de ajuste:	Técnicas de ajuste:		
	Modelo multiplicativo de Holt	SVR	HSE	ARIMA
MAPE error (%)	6.9	2.46	0.77	0.56
Escenario 2: Infectados en China				
Métodos de pronóstico	Neural Networks [19]	SVR-ESAR		
	Técnicas de ajuste:	Técnicas de ajuste:		
	Desconocida	SVR	HSE	ARIMA
MAPE error (%)	4.79	1.90	3.53	1.41

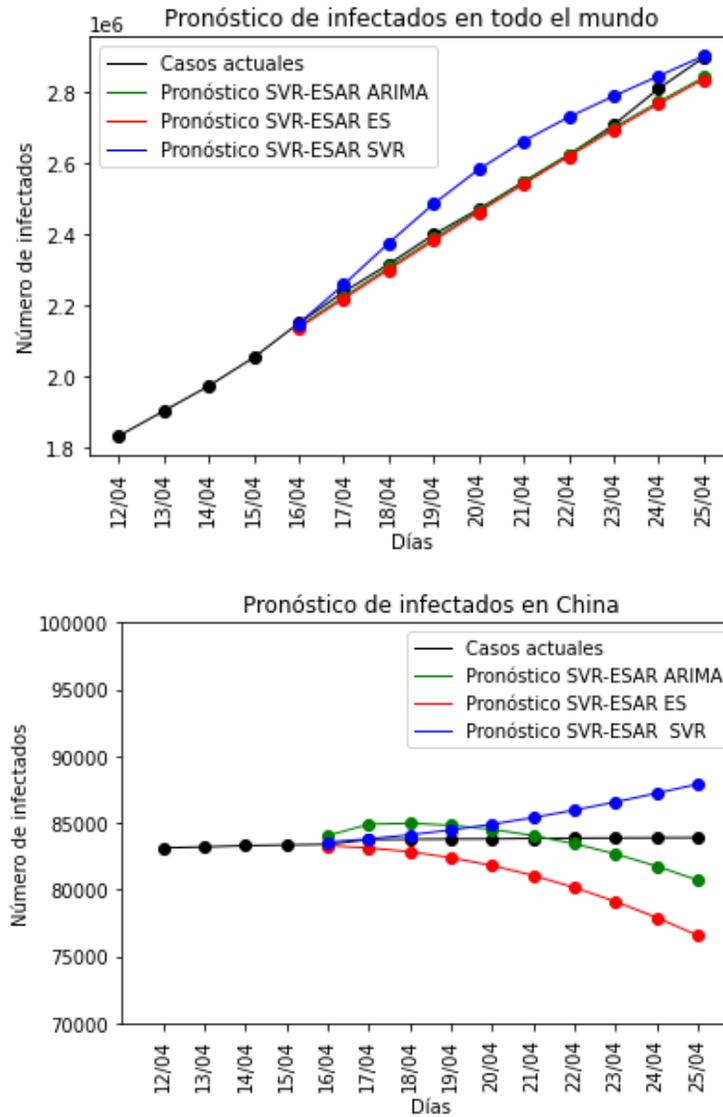


Figura 4.1 Pronóstico del número de infectados de todo el mundo y China

En la tabla 4.2 presentamos los resultados confirmados de pronóstico para los Estados Unidos y México obtenidos con SVR-ESAR. Mostramos los resultados del método propuesto utilizando el ajuste obtenido por SVR, HSE y ARIMA (figura 4.2). Señalamos que, en el caso de USA, la técnica de ARIMA de ajuste logra el mejor resultado. Esta tabla también muestra los resultados obtenidos por SVR-ESAR para México. Esta vez, el método propuesto logra un modesto resultado de pronóstico con las tres técnicas de ajuste.

Tabla 4.2 MAPE del pronóstico de los casos de infectados en USA y México usando el método de SVR-ESAR con las tres técnicas de ajuste (SVR, HSE y ARIMA)

Escenario 3 y 4: Infectados en Estados Unidos de América y México			
Métodos de pronóstico	SVR-ESAR		
	Técnica de ajuste		
	SVR	HSE	ARIMA
MAPE USA (%)	5.09	5.39	0.90
MAPE México (%)	12.44	13.25	12.57

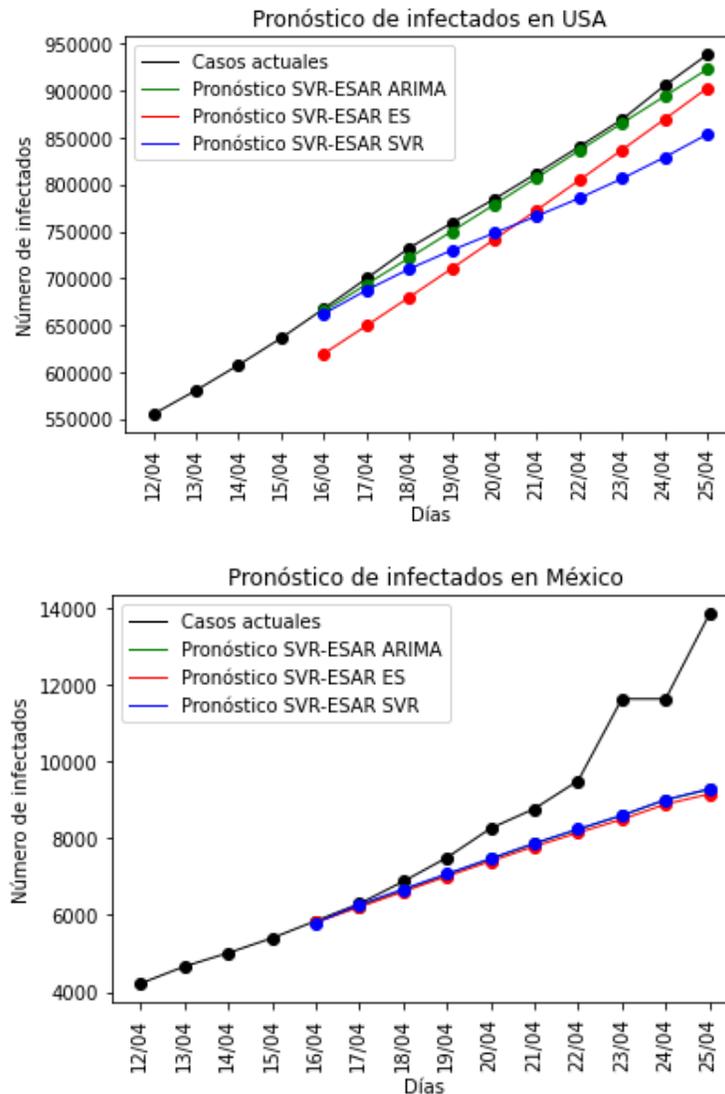


Figura 4.2 Pronóstico del número de infectados de USA y México

5 Conclusiones

Se presenta un método de pronóstico denominado SVR-ESAR (Regresión de vectores de soporte con suavizado exponencial y ARIMA), que se aplicó para la estimación de casos confirmados (infectados) de la enfermedad del COVID-19 en cuatro escenarios (Todo el Mundo, China, USA y México). Utilizamos casos infectados que hemos tomado del repositorio de Github (<https://github.com/CSSEGISandData/COVID19>)

Este método es sencillo y usa SVR con una sintonización de parámetros iterativo y una fase de ajuste. En la fase de ajuste, aplicamos tres algoritmos a los residuos de SVR: ES, SVR, y ARIMA descritos en el método propuesto.

5.1 Contribución del trabajo

En [21] y [22] presentaron el enfoque de pronósticos en dos fases mejorando el pronóstico base con los residuales para series temporales de la competencia M3 con muy buenos resultados. Además, entre los mejores métodos para la competencia M4 están los métodos híbridos como SVR con HSE y ARIMA [23] es por eso que se decidió aplicar este enfoque pero con la contribución de que los parámetros se sintonizan mediante un Algoritmo Genético y en los residuales se aplican las tres técnicas AG-SVR, HSE y ARIMA ajustando con el pronóstico de los residuales el pronóstico base.

5.2 Conclusiones

Calculando la variación porcentual del MAPE del pronóstico de infectados en todo el mundo del método propuesto con el MAPE reportado por [20], se obtuvo una mejora del **64.34%** en el peor caso con una técnica de ajuste de SVR y un **91.88%** en el mejor resultado como se muestra en la **tabla 4.1** con la técnica de ajuste de ARIMA. También en la misma tabla se muestra la comparación con el MAPE del pronóstico de los casos infectados de China publicados en [19] nuevamente obteniendo mejores resultados finales con la técnica de ajuste con ARIMA con una mejora del **70.56%**.

El valor del MAPE obtenido es satisfactorio con respecto a los del estado del arte mostrados en la tabla 4.1. A demás para los escenarios 3 y 4 se obtuvo un MAPE de 0.9 %y 12.44% los cuales son errores bajos o aceptables para ese tipo de series. Se muestran los resultados en el apéndice C.

5.3 Trabajos publicados

12° Congreso Internacional de la Academia Mexicana Multidisciplinaria el extenso con título “Pronóstico de Precios de Activos de la Bolsa Mexicana de Valores con ARIMA”.

Artículo enviado a la revista internacional de Combinatoria e Informática de Problemas de Optimización, “The Hybrid Forecasting Method SVR-ESAR for Covid-19” el cual está en revisión, además se publicó en <https://www.medrxiv.org/content/10.1101/2020.05.20.20103200v1> un servidor gratuito que publica manuscritos clínicamente relevantes como parte de una iniciativa entre PLOS (y otros editores) y la Organización Mundial de la Salud (OMS).

Actualmente se está apoyando en otro artículo aplicando Redes Neuronales ajustando los residuales con ARIMA o Suavizamiento Exponencial de Holt para pronosticar el número de casos infectados diarios por la enfermedad del COVID-19.

Bibliografía

- [1] J. E. Hanke and D. W. Wichern, *Pronósticos en los negocios*, 9th ed. Mexico, 2010.
- [2] R. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*, Melbourne. OTexts, 2018.
- [3] D. C. Montgomery, C. L. Jennings, and M. Kulahci, *Introducion to Time Series Analysis ans Forecasting*. New Jersey, 2008.
- [4] H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik, “Support vector regression machines,” *Adv. Neural Inf. Process. Syst.*, vol. 1, pp. 155–161, 1997.
- [5] M. A. Shereen, S. Khan, A. Kazmi, N. Bashir, and R. Siddique, “COVID-19 infection: Origin, transmission, and characteristics of human coronaviruses,” *J. Adv. Res.*, vol. 24, pp. 91–98, 2020, doi: 10.1016/j.jare.2020.03.005.
- [6] M. Ture and I. Kurt, “Comparison of four different time series methods to forecast hepatitis A virus infection,” *Expert Syst. Appl.*, vol. 31, no. 1, pp. 41–46, 2006, doi: 10.1016/j.eswa.2005.09.002.
- [7] G. P. McCormick, “Communications to the Editor—Exponential Forecasting: Some New Variations,” *Manage. Sci.*, vol. 15, no. 5, pp. 311–320, 1969, doi: 10.1287/mnsc.15.5.311.
- [8] E. S. Gardner, “Exponential Smoothing: The State of the Art,” vol. 4, no. August 1984, pp. 1–28, 1985.
- [9] J. W. Taylor, “Exponential smoothing with a damped multiplicative trend,” *Int. J. Forecast.*, vol. 19, no. 4, pp. 715–725, 2003, doi: 10.1016/S0169-2070(03)00003-7.
- [10] V. N. Vapnik, *The Nature Of Statistical Learning Theory*, 1st ed. 1995.
- [11] A. G. Morales, M. Sc, and G. Hernández, “regresión : m de construcción en Bogotá Using support vector machines for regression : m building in Bogotá,” vol. 6, no. 2, 2009.
- [12] A. J. Smola and B. Schölkopf, “A tutorial on support vector regression*,” pp. 199–222, 2004.
- [13] G. Santamaría-Bonfil, J. Frausto-Solís, and M. Chi-chim, “Una metodología basada en Máquinas de Vectores de Soporte para el pronóstico de la volatilidad del Índice de Precios y Cotizaciones,” *Komput. Sapiens*, vol. III, no. 4, pp. 12–17, 2010.

- [14] S. A. Cook, “The Complexity of Theorem-Proving Procedures,” 1971.
- [15] R. Hincapié Isaza, R. Gallego Rendón, and C. Ríos Porras, “Técnicas Heurísticas Aplicadas Al Problema Del Cartero Viajante (TSP).,” *Sci. Tech.*, vol. 1, no. 24, pp. 1–6, 2004, doi: 10.22517/23447214.7279.
- [16] M. Gestal, D. Rivero, J. Ramón Rabuñal, J. Dorado, and A. Pazos, *Introducción a los Algoritmos Genéticos y la Programación Genética*. Coruña: CONSORCIO EDITORIAL GALEGO, 2010.
- [17] A. García, *Inteligencia Artificial*, 2da ed. CDMX, 2017.
- [18] “No Title,” *Center for Systems Science and Engineering (CSSE) at Johns Hopkins University*. [Online]. Available: <https://github.com/CSSEGISandData/COVID-19>. [Accessed: 26-Apr-2020].
- [19] 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, 2020, doi: 10.3390/JCM9030674.
- [20] F. Petropoulos and S. Makridakis, “Forecasting the novel coronavirus COVID-19,” *PLoS One*, vol. 15, no. 3, pp. 1–8, 2020, doi: 10.1371/journal.pone.0231236.
- [21] A. J. Koning, P. H. Franses, M. Hibon, and H. O. Stekler, “The M3 competition: Statistical tests of the results,” *Int. J. Forecast.*, vol. 21, no. 3, pp. 397–409, 2005, doi: 10.1016/j.ijforecast.2004.10.003.
- [22] S. Makridakis and M. Hibon, “The M3-competition: Results, conclusions and implications,” *Int. J. Forecast.*, vol. 16, no. 4, pp. 451–476, 2000, doi: 10.1016/S0169-2070(00)00057-1.
- [23] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, “The M4 Competition: Results, findings, conclusion and way forward,” *Int. J. Forecast.*, vol. 34, no. 4, pp. 802–808, 2018, doi: 10.1016/j.ijforecast.2018.06.001.

Anexos

Apéndice A

A CÓDIGOS UTILIZADOS

A.1 Código del algoritmo GA-SVR y el algoritmo Aplicar_SVR

A continuación, presentamos el código en Python de manera general, así se obtiene el pronóstico con los algoritmos GA- SVR y el algoritmo Aplicar_SVR.

```
import numpy as np
import pandas as pd
from sklearn.svm import SVR
from sklearn.metrics import mean_absolute_error
import math
from sklearn.preprocessing import StandardScaler
def sumar(lista):
    suma=0
    for elemento in lista:
        suma += elemento
    return suma
def evaluar_MAPE(test, predictions):
    Mape=0.0
    errores=[]
    for i in range(0, len(test)):
        if(test[i]!=0):
            errores.append((math.fabs(test[i]-predictions[i])/math.fabs(test[i])))
    if(len(errores)!=0):
        Mape=100*(sumar(errores)/len(errores))
    return Mape
#Construir una poblacion(generar soluciones aleatorias)
class Solutions:
    def __init__(self, n):#constructor de solucion ##guines bajos dobles para ocultar, no mover esa parte
        #crear binario aleatorio de tamaño n
        kernel_random = kernel[np.random.randint(0, len(kernel))]
        C_svr_random = C[np.random.randint(0, len(C))]
        E_svr_random = epsilon[np.random.randint(0, len(epsilon))]
        self.value = [kernel_random, C_svr_random , E_svr_random]
        self.mape = 0.0
```

```

#Evaluar la poblacion dependiendo el problema (un metodo)
def evaluarmape(ind):
    K=ind.value[0]
    c=ind.value[1]
    E=ind.value[2]
    size1 = int(len(X_svr)-dias_pronosicados)
    size2 = int(len(y_svr)-dias_pronosicados)
    X_train, X_test, y_train, y_test = X_svr[0:size1], X_svr[size1:size1+size2], y_svr[0:size2], y_svr[size2:size2+size2]
    if (K == 'poly'):
        svr = SVR(kernel = K, C = c, epsilon = E, gamma='auto', degree=2, coef0= 1.0)
        svr.fit(X_train, y_train)
        Y_pred = svr.predict(X_test)
        ind.mape=evaluar_MAPE(sc_y.inverse_transform(Y_pred), sc_y.inverse_transform(y_test))
    else:
        if (K == 'linear'):
            svr = SVR(kernel = K, C = c, epsilon = E, gamma='auto')
            svr.fit(X_train, y_train)
            Y_pred = svr.predict(X_test)
            ind.mape = evaluar_MAPE(sc_y.inverse_transform(Y_pred), sc_y.inverse_transform(y_test))
        else:
            if (K == 'rbf'):
                svr = SVR(kernel = K, C = c, epsilon = E, gamma='auto', max_iter = -1)
                svr.fit(X_train, y_train)
                Y_pred = svr.predict(X_test)
                ind.mape = evaluar_MAPE(sc_y.inverse_transform(Y_pred), sc_y.inverse_transform(y_test))
            else:
                svr = SVR(kernel = K, C = c, epsilon = E, gamma='scale')
                svr.fit(X_train, y_train)
                Y_pred = svr.predict(X_test)
                ind.mape = evaluar_MAPE(sc_y.inverse_transform(Y_pred), sc_y.inverse_transform(y_test))
    return ind.mape
def selection(populat, n_pairs):
    padres1=[]
    padres2=[]
    parents_pairs=[]
    for i in range(n_pairs):
        parents= np.random.choice(populat, 4,replace=False) #selecciono 4 padres de la lista de soluciones
        if(parents[0].mape < parents[1].mape):
            p1=parents[0]
        else:
            p1=parents[1]
        if(parents[2].mape < parents[3].mape):
            p2=parents[2]

```

```

else:
    p2=parents[3]
    parents_pairs.append((p1,p2))
    padres1.append(p1)
    padres2.append(p2)
return parents_pairs, padres1, padres2
def crossover(parents_pairs):
    childs=[]
    for p1,p2 in parents_pairs:
        mask=np.random.randint(0,2,n)
        c1=Solutions(n)
        c2=Solutions(n)
        for i in range(len(mask)):
            if (mask[i]==0):
                c1.value[i]=p1.value[i]
                c2.value[i]=p2.value[i]
            else:
                c1.value[i]=p2.value[i]
                c2.value[i]=p1.value[i]
        if(c1.value==p1.value):
            c1.mape=p1.mape
        else:
            if(c1.value==p2.value):
                c1.mape=p2.mape
            else:
                evaluarmape(c1)
        if(c2.value==p1.value):
            c2.mape=p1.mape
        else:
            if(c2.value==p2.value):
                c2.mape=p2.mape
            else:
                evaluarmape(c2)
        childs.append(c1)
        childs.append(c2)
    return childs
def mutation(populat):
    for sol in np.random.choice(populat, mutations, replace=True):
        kernel_random = kernel[np.random.randint(0, len(kernel))]
        C_svr_random = C[np.random.randint(0, len(C))]
        E_svr_random = epsilon[np.random.randint(0, len(epsilon))]
        sol.value = [kernel_random, C_svr_random , E_svr_random]
        evaluarmape(sol)

```

```

def sintonizacion_SVR_AG(pob_evaluada):
    suma_fit_gen = []
    fitpromedio_gen = []
    mejor_sol_global = []
    mejor_fit_global = 1e+21
    con_gen_sinmejora = 0
    sujeto_campeon = []
    MAPE=0
    MAPE1=0
    n_generacions = 1
    itr = 1
    while (itr <= n_generacions and con_gen_sinmejora <= 100):
        #seleccion de padres
        parents, padres1, padres2=selection(pob_evaluada, individuos//2)
        #crear hijos con los padres
        childs=crossover(parents)
        pob_evaluada.extend(childs)
        #mutar hijos (probabilidad)
        mutation(childs)
        #unir hijos con padres (nueva poblacion)
        pob_evaluada.extend(childs) #concatenamos las listas
        #ordenar poblacion
        pob_evaluada.sort(key=lambda x: x.mape, reverse=False)
        #acotar poblacion
        pob_evaluada= pob_evaluada[:individuos*3]
        suma_fit= 0
        fitpromedio = 0
        for sol in population: #evaluo para calcular su fitness en cada generacion
            suma_fit += sol.mape
            fitpromedio = suma_fit/len(population)
        suma_fit_gen.append(suma_fit)
        fitpromedio_gen.append(fitpromedio) #fitness promedio de la generacion
        sujeto_campeon= population[0]
        if (sujeto_campeon.mape >= mejor_fit_global):
            con_gen_sinmejora = con_gen_sinmejora+1
        else:
            con_gen_sinmejora = 0
            mejor_sol_global = sujeto_campeon.value
            mejor_fit_global = sujeto_campeon.mape
    print('Generacion: ', n_generacions, len(population))
    for sol in population: #imprime cada generacion
        print(sol.value,sol.mape)
    print("Parametros y MAPE del sujeto campeon: ", sujeto_campeon.value, sujeto_campeon.mape)

```

```

    itr += 1
    n_generacions +=1
    print("Mejores parametros y MAPE global: ", mejor_sol_global, mejor_fit_global)
    print('numero de generaciones sin mejora: ', con_gen_sinmejora)
    size1 = int(len(X_svr)-dias_pronosicados)
    size2 = int(len(y_svr)-dias_pronosicados)
    X_train, X_test, y_train, y_test = X_svr[0:size1], X_svr[0:size1], y_svr[0:size2], y_svr[0:size2]
    if(mejor_sol_global[0] == 'poly'):
        svr = SVR(kernel = mejor_sol_global[0], C = mejor_sol_global[1], epsilon = mejor_sol_global[2],
gamma='auto', degree=2, coef0 = 1.0)
        svr.fit(X_train, y_train)
        Y_pred = svr.predict(X_svr[len(X_svr)-dias_pronosicados:len(X_svr)])
        MAPE = evaluar_MAPE(sc_y.inverse_transform(Y_pred), sc_y.inverse_transform(y_svr[len(y_svr)-
dias_pronosicados:len(y_svr)]))
    else:
        if(mejor_sol_global[0] == 'linear'):
            svr = SVR(kernel = mejor_sol_global[0], C = mejor_sol_global[1], epsilon = mejor_sol_global[2],
gamma='auto')
            svr.fit(X_train, y_train)
            Y_pred = svr.predict(X_svr[len(X_svr)-dias_pronosicados:len(X_svr)])
            MAPE = evaluar_MAPE(sc_y.inverse_transform(Y_pred), sc_y.inverse_transform(y_svr[len(y_svr)-
dias_pronosicados:len(y_svr)]))
        else:
            if (mejor_sol_global[0] == 'rbf'):
                svr = SVR(kernel = mejor_sol_global[0], C = mejor_sol_global[1], epsilon = mejor_sol_global[2],
gamma='auto', max_iter = -1)
                svr.fit(X_train, y_train)
                Y_pred = svr.predict(X_svr[len(X_svr)-dias_pronosicados:len(X_svr)])
                MAPE = evaluar_MAPE(sc_y.inverse_transform(Y_pred),
sc_y.inverse_transform(y_svr[len(y_svr)-dias_pronosicados:len(y_svr)]))
            else:
                svr = SVR(kernel = mejor_sol_global[0], C = mejor_sol_global[1], epsilon = mejor_sol_global[2],
gamma='scale')
                svr.fit(X_train, y_train)
                Y_pred = svr.predict(X_svr[len(X_svr)-dias_pronosicados:len(X_svr)])
                MAPE = evaluar_MAPE(sc_y.inverse_transform(Y_pred),
sc_y.inverse_transform(y_svr[len(y_svr)-dias_pronosicados:len(y_svr)]))
                size1 = int(len(X_svr)-dias_pronosicados)
                size2 = int(len(y_svr)-dias_pronosicados)
                X_train, X_test, y_train, y_test = X_svr[0:size1], X_svr[int((size1)*0.90):size1], y_svr[0:size2],
y_svr[int((size2)*0.90):size2]#[int((size1)*0.70):size1],int((size2)*0.70):size2
                if(mejor_sol_global[0] == 'poly'):

```

```

    svr = SVR(kernel = mejor_sol_global[0], C = mejor_sol_global[1], epsilon = mejor_sol_global[2],
gamma='auto', degree=2, coef0 = 1.0)
    svr.fit(X_train, y_train)
    Y_entren_pred = svr.predict(X_test)
    MAPE1 = evaluar_MAPE(sc_y.inverse_transform(Y_entren_pred), sc_y.inverse_transform(y_test))
else:
    if(mejor_sol_global[0] == 'linear'):
        svr = SVR(kernel = mejor_sol_global[0], C = mejor_sol_global[1], epsilon = mejor_sol_global[2],
gamma='auto')
        svr.fit(X_train, y_train)
        Y_entren_pred = svr.predict(X_test)
        MAPE1 = evaluar_MAPE(sc_y.inverse_transform(Y_entren_pred), sc_y.inverse_transform(y_test))

    else:
        if (mejor_sol_global[0] == 'rbf'):
            svr = SVR(kernel = mejor_sol_global[0], C = mejor_sol_global[1], epsilon = mejor_sol_global[2],
gamma='auto', max_iter = -1)
            svr.fit(X_train, y_train)
            Y_entren_pred = svr.predict(X_test)
            MAPE1 = evaluar_MAPE(sc_y.inverse_transform(Y_entren_pred), sc_y.inverse_transform(y_test))
        else:
            svr = SVR(kernel = mejor_sol_global[0], C = mejor_sol_global[1], epsilon = mejor_sol_global[2],
gamma='scale')
            svr.fit(X_train, y_train)
            Y_entren_pred = svr.predict(X_test)
            MAPE1 = evaluar_MAPE(sc_y.inverse_transform(Y_entren_pred), sc_y.inverse_transform(y_test))

print()
print('DATOS DEL MEJOR MODELO VECTORES DE SOPORTE REGRESIÓN')
print()
print('Precisión del mejor modelo:')
print('score del mejor modelo: ', svr.score(X_train, y_train))
print('MAE del mejor modelo: ', mean_absolute_error(sc_y.inverse_transform(Y_pred),
sc_y.inverse_transform(y_svr[len(y_svr)-dias_pronosicados:len(y_svr)])))
print('MAPE del mejor modelo: ', MAPE)
print("Mejores parametros y MAPE global: ", mejor_sol_global, mejor_fit_global)
print('Valores reales', sc_y.inverse_transform(y_test))
print('Valores pronosticados', sc_y.inverse_transform(Y_pred))
print('Valores pronosticados', sc_y.inverse_transform(Y_entren_pred))
return mejor_sol_global, mejor_fit_global, sc_y.inverse_transform(y_test),
sc_y.inverse_transform(Y_pred), sc_y.inverse_transform(Y_entren_pred), MAPE, MAPE1

if __name__ == "__main__":
    #####Parametros#####

```

```

documento = 'time_series_covid_19_confirmed4.csv'
n=3
individuos = 100
mutations = int(individuos*100/100)
n_generations = 200
kernel=['rbf', 'linear', 'sigmoid', 'poly']
C=[[0.001, 0.002, ..., 0.009], [0.01, 0.02, 0.03, ..., 10.0], [15.0, 20.0, 25.0, ..., 1000.0], [1100.0, 1200.0,
1300.0, ..., 3000.0]]
epsilon=[[0.001, 0.002, ..., 0.009], [0.01, 0.02, 0.03, ..., 10.0], [15.0, 20.0, 25.0, ..., 1000.0], [1100.0, 1200.0,
1300.0, ..., 3000.0]]
#####
df = pd.read_csv(documento)
dias_pronosicados=10
mejores_parametros=[]
MAPE_del_test=[]
valores_reales=[]
valores_pronosticados=[]
conjunt_entrenamiento_pronost=[]
Country_Region=[]
indice=[]
MAPE_entrenamientos = []
mape_del_pronostico = []
X = df.iloc[0, 5:df.shape[1]].values
Y = df.iloc[83, 5:df.shape[1]].values
sc_X = StandardScaler()
sc_y = StandardScaler()
X_svr = sc_X.fit_transform(X.reshape(-1, 1))
Y_svr = sc_y.fit_transform(Y.reshape(-1, 1))
y_svr= Y_svr
Country_Region.append(df.iloc[83, 2])
indice.append(df.iloc[83, 0])
"""Generar poblacion inicial"""
population_size=individuos
population = []
for i in range(population_size):
    population.append(Solutions(n))
print("Poblacion sin evaluar: ")
for sol in population:
    print(sol.value, sol.mape)
    """Evaluar poblacion inicial"""
print("Primera poblacion: Generacion 0 ")
for sol in population:
    evaluarmape(sol)

```

```

    print(sol.value, sol.mape)
fit_prom_poblacion_inicial=0
for sol in population:
    fit_prom_poblacion_inicial+=sol.mape/len(population)
    print("Ultimos parametros y fit promedio")
    print(sol.value, fit_prom_poblacion_inicial)
    parametros, MAPE, valoresreales, valorespronosticados, entrenamiento_pronost,
MAPE_DEL_PRONOSTICO, MAPE_entrenamiento = sintonizacion_SVR_AG(population)
    MAPE_entrenamientos.append(MAPE_entrenamiento)
    mejores_parametros.append(parametros)
    MAPE_del_test.append(MAPE)
    valores_reales.append(valoresreales)
    valores_pronosticados.append(valorespronosticados)
    conjunt_entrenamiento_pronost.append(entrenamiento_pronost)
    mape_del_pronostico.append(MAPE_DEL_PRONOSTICO)
    Indice=pd.DataFrame(indice)
    Indice.columns=['INDICE']
    Country=pd.DataFrame(Country_Region)
    Country.columns=['Country/Region']
    Parametros_del_modelo=pd.DataFrame(mejores_parametros) #CONVIERTO LISTA NUMPY A
DATAFRAME(PANDAS)
    Parametros_del_modelo.columns=['Kernel', 'C', 'Epsilon'] #ASIGNO NOMBRE DE A LA COLUMNA
MAPES=pd.DataFrame(MAPE_del_test)
    MAPES.columns=['MAPE']
    MAPE_ENTRENAMIENTO=pd.DataFrame(MAPE_entrenamientos)
    MAPE_ENTRENAMIENTO.columns=['MAPE ENTRENAMIENTO']
    MAPE_PRONOSTICO=pd.DataFrame(mape_del_pronostico)
    MAPE_PRONOSTICO.columns=['MAPE DEL PRONOSTICO']
    entrenamientopronosticado=pd.DataFrame(conjunt_entrenamiento_pronost)
    Pronostico=pd.DataFrame(valores_pronosticados)
    Soluciones = pd.concat([Indice, Country, Parametros_del_modelo, MAPES, MAPE_ENTRENAMIENTO,
MAPE_PRONOSTICO, entrenamientopronosticado, Pronostico], axis=1, sort=False)
    Soluciones.to_csv("C:\\Users\\usuario\\Desktop\\Maestria\\Covid\\PRONOSTICOS_COVID_19\\PRONOST
ICO_DE_TODOS_LOS_PAISES_SVR\\SVR_CONFIRMED\\CHINA\\CHINACORREGIDO\\pretest\\time
_series_covid_19_CONFIRMED-resultadosCHINA.csv",header=True,index=True)

```

A.2 Código para pronosticar con ARIMA

A continuación, presentamos el código en Python de manera general, así se obtiene el pronóstico de los residuos con ARIMA.

```
import math
import pandas as pd
from matplotlib import pyplot
from pandas.plotting import autocorrelation_plot
from statsmodels.tsa.arima_model import ARIMA
from pandas import DataFrame
from sklearn.metrics import mean_squared_error
def sumar(lista):
    suma=0
    for elemento in lista:
        suma += elemento
    return suma
documento = 'time_series_covid_19_confirmed-Residuales.csv'
df = pd.read_csv(documento)
dfreal = pd.read_csv('time_series_covid_19_confirmed.csv')
dias_pronosicados=10
prueba=3
print("Francia")
x = dfreal.iloc[0, (dfreal.shape[1]-(df.shape[1]-3)-9):dfreal.shape[1]-9]
y = df.iloc[0, 3:df.shape[1]]
y.plot()
autocorrelation_plot(y)
pyplot.show()
parametros= [[2, 0, 0]]
y = df.iloc[0, 3:df.shape[1]].values
# fit model
model = ARIMA(y, order=(parametros[0][0], parametros[0][1], parametros[0][2]))
model_fit = model.fit(dispatch=0)
print(model_fit.summary())
# plot residual errors
residuals = DataFrame(model_fit.resid)
residuals.plot()
pyplot.show()
residuals.plot(kind='kde')
pyplot.show()
```

```

print(residuals.describe())
z = y[0:len(y)-dias_pronosicados]
size = int(len(z)-prueba)
train, test = z[0:size], z[size:len(z)]
history = [z for z in train]
predictions = []
for t in range(len(test)):
    model = ARIMA(history, order=(parametros[0][0], parametros[0][1], parametros[0][2]))
    model_fit = model.fit(dispatch=0)
    output = model_fit.forecast()
    yhat = output[0]
    predictions.append(yhat)
    obs = yhat
    history.append(obs)
    print('predicted=%f, expected=%f' % (yhat, test[t]))
error = mean_squared_error(test, predictions)
print("Test MSE: %.3f" % error)
errores=[]
for i in range(0, len(test)):
    errores.append(math.fabs(test[i]-predictions[i])/math.fabs(test[i]))
Mape1=[]
Mape1.append(100*sumar(errores)/len(test))
print("Test MAPE: %.3f", Mape1)
# plot
pyplot.plot(test)
pyplot.plot(predictions, color='green')
pyplot.show()
z = y
size = int(len(z)-dias_pronosicados)
train, test = z[0:size], z[size:len(z)]
history = [z for z in train]
predictions = []
for t in range(len(test)):
    model = ARIMA(history, order=(parametros[0][0], parametros[0][1], parametros[0][2]))
    model_fit = model.fit(dispatch=0)
    output = model_fit.forecast()
    yhat = output[0]
    predictions.append(yhat)
    obs = yhat
    history.append(obs)
    print('predicted=%f, expected=%f' % (yhat, test[t]))
error = mean_squared_error(test, predictions)
print("Test MSE: %.3f" % error)

```

```

errores=[]
for i in range(0, len(test)):
    errores.append(math.fabs(test[i]-predictions[i])/math.fabs(test[i]))
Mape=[]
Mape.append(100*sumar(errores)/len(test))
print("Test MAPE: %.3f" % Mape)
# plot
pyplot.plot(test)
pyplot.plot(predictions, color='green')
pyplot.show()
forecast = model_fit.forecast()[0]
print("Forecast: %f" % forecast)
Country_Region=[]
indice=[]
Country_Region.append('Francia')
indice.append(0)
Indice=pd.DataFrame(indice)
Indice.columns=['INDICE']
Country=pd.DataFrame(Country_Region)
Country.columns=['Country/Region']
Parametros_del_modelo=pd.DataFrame(parametros) #CONVIERTO LISTA NUMPY A
DATAFRAME(PANDAS)
Parametros_del_modelo.columns=['p', 'd', 'q'] #ASIGNO NOMBRE DE LA COLUMNA
MAPE_PRONOSTICO=pd.DataFrame(Mape)
MAPE_PRONOSTICO.columns=['MAPE DEL PRONOSTICO']
Pronostico=pd.DataFrame(predictions).transpose()
Soluciones = pd.concat([Indice, Country, Parametros_del_modelo, MAPE_PRONOSTICO, Pronostico],
axis=1, sort=False)
Soluciones.to_csv("C:\\Users\\usuario\\Desktop\\Maestria\\Covid\\PRONOSTICOS_COVID_19\\PRONOST
ICO_DE_TODOS_LOS_PAISES_SVR\\SVR_CONFIRMED\\Francia
2\\pretest\\ARIMA\\time_series_covid_19_confirmed-resultados de residuos
pronest.csv",header=True,index=True)

```

A.3 Código para pronosticar Suavizamiento Exponencial de Holt (HSE)

A continuación, presentamos el código en Python de manera general, así se obtiene el pronóstico de los residuos con Suavizamiento Exponencial.

```
import pandas as pd
import matplotlib.pyplot as plt
import math
from statsmodels.tsa.api import Holt
def sumar(lista):
    suma=0
    for elemento in lista:
        suma += elemento
    return suma
def evaluar_MAPE(test, predictions):
    Mape=0.0
    errores=[]
    for i in range(0, len(test)):
        if(test[i]!=0):
            errores.append((math.fabs(test[i]-predictions[i])/math.fabs(test[i])))
    if(len(errores)!=0):
        Mape=100*(sumar(errores)/len(errores))
    return Mape
series = pd.read_csv('residuosFrancia.csv')
x = series.iloc [0:series.shape[0]-10, 0].values
y = series.iloc [0:series.shape[0]-10, 1].values
fig,ax=plt.subplots()
series.plot(x='dias', y='residuos', ax=ax, style=':')
modeloholt=Holt(y).fit()
resultadosholt=modeloholt.forecast(10)
alpha2=modeloholt.model.params['smoothing_level']
beta2=modeloholt.model.params['smoothing_slope']
amortiguacion2=modeloholt.model.params['damping_slope']
pronosticos=[]
pronosticos.append(resultadosholt)
Pronostico=pd.DataFrame(pronosticos)
Pronostico.to_csv("C:\\Users\\usuario\\Desktop\\Maestria\\Covid\\PRONOSTICOS_COVID_19\\PRONOSTI
CO_DE_TODOS_LOS_PAISES_SVR\\SVR_CONFIRMED\\Francia      2\\pretest\\SUAVIZAMIENTO
EXPONENCIAL\\time_series_covid_19_confirmed-resultados      de      residuos
pron.csv",header=True,index=True)
```

Apéndice B

B CASOS INFECTADOS

B.1 Conjunto de datos de China

Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados
22/01/2020	548	07/02/2020	34110	23/02/2020	77022
23/01/2020	643	08/02/2020	36814	24/02/2020	77241
24/01/2020	920	09/02/2020	39829	25/02/2020	77754
25/01/2020	1406	10/02/2020	42354	26/02/2020	78166
26/01/2020	2075	11/02/2020	44386	27/02/2020	78600
27/01/2020	2877	12/02/2020	44759	28/02/2020	78928
28/01/2020	5509	13/02/2020	59895	29/02/2020	79356
29/01/2020	6087	14/02/2020	66358	01/03/2020	79932
30/01/2020	8141	15/02/2020	68413	02/03/2020	80136
31/01/2020	9802	16/02/2020	70513	03/03/2020	80261
01/02/2020	11891	17/02/2020	72434	04/03/2020	80386
02/02/2020	16630	18/02/2020	74211	05/03/2020	80537
03/02/2020	19716	19/02/2020	74619	06/03/2020	80690
04/02/2020	23707	20/02/2020	75077	07/03/2020	80770
05/02/2020	27440	21/02/2020	75550	08/03/2020	80823
06/02/2020	30587	22/02/2020	77001	09/03/2020	80860

Tabla B.1 Casos infectados de China del 22 de Enero al 9 de Marzo del 2020.

Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados
10/03/2020	80887	26/03/2020	81782	11/04/2020	83014
11/03/2020	80921	27/03/2020	81897	12/04/2020	83134
12/03/2020	80932	28/03/2020	81999	13/04/2020	83213
13/03/2020	80945	29/03/2020	82122	14/04/2020	83306
14/03/2020	80977	30/03/2020	82198	15/04/2020	83356
15/03/2020	81003	31/03/2020	82279	16/04/2020	83403
16/03/2020	81033	01/04/2020	82361	17/04/2020	83760
17/03/2020	81058	02/04/2020	82432	18/04/2020	83787
18/03/2020	81102	03/04/2020	82511	19/04/2020	83805
19/03/2020	81156	04/04/2020	82543	20/04/2020	83817
20/03/2020	81250	05/04/2020	82602	21/04/2020	83853
21/03/2020	81305	06/04/2020	82665	22/04/2020	83868
22/03/2020	81435	07/04/2020	82718	23/04/2020	83884
23/03/2020	81498	08/04/2020	82809	24/04/2020	83899
24/03/2020	81591	09/04/2020	82883	25/04/2020	83909
25/03/2020	81661	10/04/2020	82941		

Tabla B.2 Casos confirmados de China del 10 de Marzo al 25 de Abril del 2020.

B.2 Conjunto de datos de Todo el Mundo

Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados
22/01/2020	556	07/02/2020	34408	23/02/2020	78991
23/01/2020	656	08/02/2020	37138	24/02/2020	79595
24/01/2020	944	09/02/2020	40169	25/02/2020	80441
25/01/2020	1438	10/02/2020	42782	26/02/2020	81424
26/01/2020	2123	11/02/2020	44823	27/02/2020	82783
27/01/2020	2933	12/02/2020	45243	28/02/2020	84150
28/01/2020	5585	13/02/2020	60391	29/02/2020	86050
29/01/2020	6174	14/02/2020	66909	01/03/2020	88409
30/01/2020	8243	15/02/2020	69055	02/03/2020	90347
31/01/2020	9937	16/02/2020	71250	03/03/2020	92882
01/02/2020	12049	17/02/2020	73285	04/03/2020	95163
02/02/2020	16799	18/02/2020	75164	05/03/2020	97930
03/02/2020	19894	19/02/2020	75668	06/03/2020	101846
04/02/2020	23906	20/02/2020	76227	07/03/2020	105893
05/02/2020	27650	21/02/2020	76850	08/03/2020	109868
06/02/2020	30810	22/02/2020	78604	09/03/2020	113638

Tabla B.3 Casos confirmados de todo el mundo del 22 de Enero al 9 de Marzo del 2020.

Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados
10/03/2020	118669	26/03/2020	529656	11/04/2020	1735731
11/03/2020	125925	27/03/2020	593357	12/04/2020	1834803
12/03/2020	128403	28/03/2020	660760	13/04/2020	1904921
13/03/2020	145257	29/03/2020	720208	14/04/2020	1975279
14/03/2020	156154	30/03/2020	782458	15/04/2020	2055142
15/03/2020	167508	31/03/2020	857557	16/04/2020	2151507
16/03/2020	181629	01/04/2020	932546	17/04/2020	2239231
17/03/2020	197158	02/04/2020	1013398	18/04/2020	2316804
18/03/2020	214878	03/04/2020	1095771	19/04/2020	2400167
19/03/2020	242628	04/04/2020	1175931	20/04/2020	2471337
20/03/2020	272267	05/04/2020	1249559	21/04/2020	2548182
21/03/2020	304567	06/04/2020	1321207	22/04/2020	2623141
22/03/2020	337014	07/04/2020	1396169	23/04/2020	2707821
23/03/2020	378293	08/04/2020	1479882	24/04/2020	2809593
24/03/2020	418104	09/04/2020	1565357	25/04/2020	2896839
25/03/2020	467717	10/04/2020	1657606		

Tabla B.4 Casos confirmados de todo el mundo del 10 de Marzo al 25 de Abril del 2020.

B.3 Conjunto de datos de USA

Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados
22/01/2020	1	07/02/2020	11	23/02/2020	15
23/01/2020	1	08/02/2020	11	24/02/2020	51
24/01/2020	2	09/02/2020	11	25/02/2020	51
25/01/2020	2	10/02/2020	11	26/02/2020	57
26/01/2020	5	11/02/2020	12	27/02/2020	58
27/01/2020	5	12/02/2020	12	28/02/2020	60
28/01/2020	5	13/02/2020	13	29/02/2020	68
29/01/2020	5	14/02/2020	13	01/03/2020	74
30/01/2020	5	15/02/2020	13	02/03/2020	98
31/01/2020	7	16/02/2020	13	03/03/2020	118
01/02/2020	8	17/02/2020	13	04/03/2020	149
02/02/2020	8	18/02/2020	13	05/03/2020	217
03/02/2020	11	19/02/2020	13	06/03/2020	262
04/02/2020	11	20/02/2020	13	07/03/2020	402
05/02/2020	11	21/02/2020	15	08/03/2020	518
06/02/2020	11	22/02/2020	15	09/03/2020	583

Tabla B.5 Casos confirmados de México del 22 de Enero al 9 de Marzo del 2020.

Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados
10/03/2020	959	26/03/2020	83836	11/04/2020	526396
11/03/2020	1281	27/03/2020	101657	12/04/2020	555313
12/03/2020	1663	28/03/2020	121465	13/04/2020	580619
13/03/2020	2179	29/03/2020	140909	14/04/2020	607670
14/03/2020	2727	30/03/2020	161831	15/04/2020	636350
15/03/2020	3499	31/03/2020	188172	16/04/2020	667592
16/03/2020	4632	01/04/2020	213242	17/04/2020	699706
17/03/2020	6421	02/04/2020	243622	18/04/2020	732197
18/03/2020	7783	03/04/2020	275367	19/04/2020	758809
19/03/2020	13747	04/04/2020	308650	20/04/2020	784326
20/03/2020	19273	05/04/2020	336802	21/04/2020	811865
21/03/2020	25600	06/04/2020	366317	22/04/2020	840351
22/03/2020	33276	07/04/2020	397121	23/04/2020	869170
23/03/2020	43843	08/04/2020	428654	24/04/2020	905358
24/03/2020	53736	09/04/2020	462780	25/04/2020	938154
25/03/2020	65778	10/04/2020	496535		

Tabla B.6 Casos confirmados de USA del 10 de Marzo al 25 de Abril del 2020

B.4 Conjunto de datos de México

Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados
22/01/2020	0	07/02/2020	0	23/02/2020	0
23/01/2020	0	08/02/2020	0	24/02/2020	0
24/01/2020	0	09/02/2020	0	25/02/2020	0
25/01/2020	0	10/02/2020	0	26/02/2020	0
26/01/2020	0	11/02/2020	0	27/02/2020	0
27/01/2020	0	12/02/2020	0	28/02/2020	1
28/01/2020	0	13/02/2020	0	29/02/2020	4
29/01/2020	0	14/02/2020	0	01/03/2020	5
30/01/2020	0	15/02/2020	0	02/03/2020	5
31/01/2020	0	16/02/2020	0	03/03/2020	5
01/02/2020	0	17/02/2020	0	04/03/2020	5
02/02/2020	0	18/02/2020	0	05/03/2020	5
03/02/2020	0	19/02/2020	0	06/03/2020	6
04/02/2020	0	20/02/2020	0	07/03/2020	6
05/02/2020	0	21/02/2020	0	08/03/2020	7
06/02/2020	0	22/02/2020	0	09/03/2020	7

Tabla B.7 Casos confirmados de México del 22 de Enero al 9 de Marzo del 2020.

Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados	Fecha (D/M/A)	Infectados
10/03/2020	7	26/03/2020	475	11/04/2020	3844
11/03/2020	8	27/03/2020	585	12/04/2020	4219
12/03/2020	12	28/03/2020	717	13/04/2020	4661
13/03/2020	12	29/03/2020	848	14/04/2020	5014
14/03/2020	26	30/03/2020	993	15/04/2020	5399
15/03/2020	41	31/03/2020	1094	16/04/2020	5847
16/03/2020	53	01/04/2020	1215	17/04/2020	6297
17/03/2020	82	02/04/2020	1378	18/04/2020	6875
18/03/2020	93	03/04/2020	1510	19/04/2020	7497
19/03/2020	118	04/04/2020	1688	20/04/2020	8261
20/03/2020	164	05/04/2020	1890	21/04/2020	8772
21/03/2020	203	06/04/2020	2143	22/04/2020	9501
22/03/2020	251	07/04/2020	2439	23/04/2020	11633
23/03/2020	316	08/04/2020	2785	24/04/2020	11633
24/03/2020	367	09/04/2020	3181	25/04/2020	13842
25/03/2020	405	10/04/2020	3441		

Tabla B.8 Casos confirmados de México del 10 de Marzo al 25 de Abril del 2020

Apéndice C

C Resultados de los pronósticos

Fecha (D/M/A)	Técnicas de ajuste		
	SVR	ARIMA	HSE
16/04/2020	83574.4059	84057.6409	83303.1953
17/04/2020	83807.306	84908.2036	83133.3327
18/04/2020	84118.6827	84982.2471	82848.4172
19/04/2020	84480.7949	84831.6052	82405.7374
20/04/2020	84911.0257	84525.4455	81813.7892
21/04/2020	85407.7069	84065.4157	81068.4464
22/04/2020	85967.2587	83452.4635	80170.2302
23/04/2020	86579.0964	82686.6217	79119.1336
24/04/2020	87232.8552	81774.2893	77921.5576
25/04/2020	87914.0753	80723.5057	76585.5222

Tabla C.1 Resultado de los pronósticos del número de infectados de China

Fecha (D/M/A)	Técnicas de ajuste		
	SVR	ARIMA	HSE
16/04/2020	2149873.56	2141986.37	2138190.8
17/04/2020	2259899.23	2225695.96	2220053.72
18/04/2020	2376053.95	2307279.69	2301639.9
19/04/2020	2486596.28	2388364.05	2382710.93
20/04/2020	2582949.07	2468599.33	2462954.94
21/04/2020	2662858.35	2547620.95	2541984.29
22/04/2020	2729587.36	2624970.19	2619336.71
23/04/2020	2788746	2700116.69	2694480.06
24/04/2020	2845174.17	2772459.37	2766820.33
25/04/2020	2901362.68	2841350.72	2835712.93

Tabla C.2 Resultado de los pronósticos del número de infectados de todo el mundo

Apéndice C. Resultados de los pronósticos

Fecha (D/M/A)	Técnicas de ajuste		
	SVR	ARIMA	HSE
16/04/2020	662250.97	664993.056	619715.955
17/04/2020	687348.455	693588.571	649491.559
18/04/2020	709858.576	721598.672	679679.778
19/04/2020	729818.278	749972.754	710280.614
20/04/2020	748176.68	778454.1	741294.065
21/04/2020	766303.374	807254.058	772720.132
22/04/2020	785417.342	836343.884	804558.814
23/04/2020	806239.872	865193.054	836810.113
24/04/2020	828953.307	893927.367	869474.027
25/04/2020	853365.761	922628.776	902550.557

Tabla C.3 Resultado de los pronósticos del número de infectados de USA

Fecha (D/M/A)	Técnicas de ajuste		
	SVR	ARIMA	HSE
16/04/2020	5794.18266	5822.6633	5810.88281
17/04/2020	6273.9269	6232.30617	6212.21174
18/04/2020	6676.31079	6644.82113	6611.32748
19/04/2020	7072.94193	7057.21658	7010.95874
20/04/2020	7472.59545	7462.34108	7403.31643
21/04/2020	7870.24689	7861.54664	7789.75661
22/04/2020	8239.92529	8231.64691	8147.09265
23/04/2020	8601.08623	8592.92268	8495.60508
24/04/2020	9009.70862	9001.57789	8891.49758
25/04/2020	9286.21054	9278.09069	9155.24814

Tabla C.4 Resultado de los pronósticos del número de infectados de México