



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Software

**Conservación de la precisión de un modelo de Machine
Learning por medio de un flujo de reentrenamiento
automatizado para predecir el índice de infección por
COVID 19 en el Perú**

TESIS

Para optar el Título Profesional de Ingeniero de Software

AUTOR

Erick Victor PALOMINO GUTIERREZ

ASESOR

Dr. Hugo David CALDERÓN VILCA

Lima, Perú

2024



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

Referencia bibliográfica

Palomino, E. (2024). *Conservación de la precisión de un modelo de Machine Learning por medio de un flujo de reentrenamiento automatizado para predecir el índice de infección por COVID 19 en el Perú*. [Tesis de pregrado, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática, Escuela Profesional de Ingeniería de Software]. Repositorio institucional Cybertesis UNMSM.

Metadatos complementarios

Datos de autor	
Nombres y apellidos	Erick Victor Palomino Gutierrez
Tipo de documento de identidad	DNI
Número de documento de identidad	75729110
URL de ORCID	https://orcid.org/0000-0002-2697-0529
Datos de asesor	
Nombres y apellidos	Hugo David Calderón Vilca
Tipo de documento de identidad	DNI
Número de documento de identidad	01317748
URL de ORCID	https://orcid.org/0000-0002-1177-4947
Datos del jurado	
Presidente del jurado	
Nombres y apellidos	Pedro Martin Lezama Gonzales
Tipo de documento	DNI
Número de documento de identidad	09656793
Miembro del jurado 1	
Nombres y apellidos	Juan Gamarra Moreno
Tipo de documento	DNI
Número de documento de identidad	20039857
Datos de investigación	
Línea de investigación	C.0.5.8 Inteligencia Artificial

Grupo de investigación	No aplica
Agencia de financiamiento	Sin financiamiento
Ubicación geográfica de la investigación	Universidad Nacional Mayor de San Marcos Edificio: Facultad de Ingeniería de Sistemas e Informática País: Perú Departamento: Lima Provincia: Lima Distrito: Cercado de Lima Calle: Av. Carlos Germán Amezaga 375 Latitud: -12.053697 Longitud: -77.085774
Año o rango de años en que se realizó la investigación	Marzo 2022 – Enero 2024
URL de disciplinas OCDE	Ciencias de la computación https://purl.org/pe-repo/ocde/ford#1.02.01



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERIA DE SISTEMAS E INFORMATICA
Escuela Profesional de Ingeniería de Software

Acta de Sustentación de Tesis

Siendo las once (11:00) horas del día doce (19) del mes diciembre de año 2024, se reunieron en el auditorio presidido por el Dr. Pedro Martin Lezama Gonzales (Presidente), Mg. Juan Gamarra Moreno (Miembro) y el Dr. Hugo David Calderón Vilca (Asesor), para la sustentación de la tesis Intitulada: **"CONSERVACIÓN DE LA PRECISIÓN DE UN MODELO DE MACHINE LEARNING POR MEDIO DE UN FLUJO DE REENTRENAMIENTO AUTOMATIZADO PARA PREDECIR EL ÍNDICE DE INFECCIÓN POR COVID 19 EN EL PERÚ"**, por el Bachiller Erick Victor Palomino Guterrez, para obtener el Título Profesional de Ingeniero de Software.

Acto seguido de la exposición de la Tesis, el Presidente invitó al Bachiller a responder las preguntas formuladas por los Miembros del Jurado.

El Bachiller, en el curso de sus intervenciones demostró pleno dominio del tema, al responder con acierto y fluidez las preguntas formuladas por los señores miembros del Jurado.

Finalmente habiéndose efectuado la calificación correspondiente por los miembros del Jurado, el bachiller obtuvo la nota de (16). *Muy bueno*

A continuación, el Presidente del Jurado Dr. Pedro Martin Lezama Gonzales, declara al Bachiller Ingeniero de Software.

Siendo horas, se levantó la sesión.


Dr. Pedro Martin Lezama Gonzales
Presidente


Mg. Juan Gamarra Moreno
Miembro


Dr. Hugo David Calderón Vilca
Asesor



CERTIFICADO DE SIMILITUD

Yo **HUGO DAVID CALDERÓN VILCA** en mi condición de asesor acreditado con la **Resolución Directoral N°000002-2024-EPISW-FISI/UNMSM** de la **tesis** académico, cuyo título es **CONSERVACIÓN DE LA PRECISIÓN DE UN MODELO DE MACHINE LEARNING POR MEDIO DE UN FLUJO DE REENTRENAMIENTO AUTOMATIZADO PARA PREDECIR EL ÍNDICE DE INFECCIÓN POR COVID 19 EN EL PERÚ**, presentado por el bachiller Erick **Victor PALOMINO GUTIERREZ**, para optar el **título de Ingeniero de Software**. CERTIFICO que se ha cumplido con lo establecido en la Directiva de Originalidad y de Similitud de Trabajos Académicos, de Investigación y Producción Intelectual. Según la revisión, análisis y evaluación mediante el software de similitud textual, el documento evaluado cuenta con el porcentaje de **14%** de similitud, nivel **PERMITIDO** para continuar con los trámites correspondientes y para su **publicación en el repositorio institucional**.

Se emite el presente certificado en cumplimiento de lo establecido en las normas vigentes, como uno de los requisitos para la obtención del **título** correspondiente.

Firma del Asesor _____

DNI: 01317748

Dr. Hugo David Calderón Vilca

Asesor



AGRADECIMIENTOS

Al Dr. Calderón Vilca, Hugo David por su orientación, consejo e impulso para la realización de este trabajo. A mi familia y amigos que me animaron durante todo el transcurso de la investigación.

RESUMEN

Título	Conservación de la precisión de un modelo de Machine Learning por medio de un flujo de reentrenamiento automatizado para predecir el índice de infección por COVID 19 en el Perú
Asesor de tesis	Hugo David Calderón Vilca
Fecha	05/05/2024

Con la llegada de la infección por COVID 19 en el mundo, se vio afectada la estabilidad económica y sanitaria de distintos países del mundo; y a pesar de las múltiples propuestas de predicción de infección por COVID 19, aún existe el problema de la precisión poca duradera de estos trabajos. Este trabajo presenta un enfoque de reentrenamiento automático de un modelo de predicción de índice de infección por COVID 19 en el Perú con el objetivo de tener una herramienta de planificación protocolar y estratégica para contrarrestar esta infección y que tenga una precisión que perdure a lo largo del tiempo. Para el desarrollo del modelo se utilizó una red neuronal recurrente LSTM y una fuente de datos constantemente actualizada, fundamental para la aplicación del reentrenamiento. Los modelos producidos por este enfoque flujo automatizado durante dos meses de reentrenamiento semanal cuentan con un coeficiente de determinación y raíz del error cuadrático medio en promedio de 0.994 y 260.944 respectivamente. Comparado con otros trabajos, los modelos producidos tienen la ventaja de ser entrenados semanalmente con datos actualizados, manteniendo una precisión duradera y se comprobó, además, la efectividad de un proceso automatizado de reentrenamiento de modelos de machine learning.

Palabras Clave: Predicción, Índice de infección, COVID 19, Redes Neuronales Recurrentes.

ÍNDICE

ÍNDICE	4
CAPÍTULO I: INTRODUCCIÓN	10
1.1. Antecedentes	10
1.2. Descripción del Problema	11
1.2.1. Formulación del Problema	13
1.3. Importancia.....	14
1.4. Motivación	14
1.5. Objetivos	15
1.6. Propuesta	16
1.7. Estructura de la Tesis	16
CAPÍTULO 2: MARCO TEÓRICO	18
2.1. Modelo de regresión polinomial	18
2.2. Modelo de Red Neuronal Multicapa	18
2.3. Modelo de Red neuronal recurrente:	19
2.3.1. Redes neuronales LSTM (Long short-term memory)	20
CAPÍTULO 3: ESTADO DEL ARTE	22
3.1. Dataset de datos de infección por COVID 19	25
3.2. Métodos Estadísticos para la predicción de infección por COVID 19.....	26

3.3. Métodos de Machine Learning para la predicción de infección por COVID 19

33

CAPÍTULO 4: METODOLOGÍA 39

4.1. Determinación de la mejor arquitectura de red neuronal para la predicción de
infección por COVID 19 en el Perú 39

4.2. Metodología de reentrenamiento de un modelo de predicción de infección por
COVID 19 en el Perú 42

4.2.1. Dataset de infección por COVID 19 44

4.2.2. Obtención y Filtrado de Datos 45

4.2.3. Preprocesamiento de los datos 45

4.2.4. Arquitectura de la Red Neuronal.....48

4.2.5. Validación del modelo 50

4.2.6. Modelo de regresión de infección de COVID 19 50

4.2.7. Reentrenamiento automático 51

CAPÍTULO 5: DISEÑO DE LA APLICACIÓN 52

5.1. Requerimientos.....52

5.1.1. Alcance.....52

5.1.2. Restricciones 52

5.1.3. Actores del sistema.....52

5.1.4. Historias de Usuario 53

5.2. Product Backlog 55

5.3. Sprints.....55

5.3.1. Sprint 0	55
5.3.2. Sprint 1	60
5.4. El sistema de predicción de infección por COVID 19 en el Perú	67
CAPÍTULO 6: RESULTADOS Y DISCUSIÓN	70
6.1. Características del conjunto de datos	70
6.2. Resultados del modelo de predicción por índice de infección por COVID 19 en el Perú	70
6.2.1. Primer Periodo (05/10/22 - 23/11/22)	71
6.2.2. Segundo Periodo (29/07/23 - 13/09/23)	74
6.2.3. Comparación entre periodos.....	77
6.3. Discusión	77
6.4. Conclusiones	79
REFERENCIAS	81

ÍNDICE DE FIGURAS

Figura 1: Casos positivos diarios abril 2020-Abril 2022 (Dong et al., 2020)	10
Figura 2: Fallecimientos diarios Abril 2020-Abril 2022(Dong et al., 2020)	10
Figura 3: Perceptrón de Rosenblatt (Haykin, 2016).....	18
Figura 4: Arquitectura de celda LSTM (Luo et al., 2021)	20
Figura 5: Distribución de bases de datos indexadas.....	23
Figura 6: Metodología de reentrenamiento de un modelo de predicción de infección por COVID 19 en el Perú	43
Figura 7: Entrada Modificada basada en (Luo et al., 2021).....	45
Figura 8: Transformación de entrada	46
Figura 9: Arquitectura LSTM basada en (Luo et al., 2021).....	48
Figura 10: Resumen Arquitectura	49
Figura 11: Flujo de Reentrenamiento.....	51
Figura 12: Prototipo y criterios de aceptación	55
Figura 13: Arquitectura Propuesta	56
Figura 14: Arquitectura API.....	58
Figura 15: Dimensiones de entrada.....	60
Figura 16: Arquitectura red neuronal	61
Figura 17: Sistema de Predicción de infección por COVID 19 en el Perú	67
Figura 18: Sistema de Predicción de infección por COVID 19 en el Perú paso 2.....	68
Figura 19: Función de pérdida de entrenamientos primer periodo	72
Figura 20: Función de pérdida de entrenamientos segundo periodo.....	75

ÍNDICE DE TABLAS

Tabla 1: Resumen del estado del arte	23
Tabla 2: Resumen de resultados (Kashif, 2021)	26
Tabla 3: Resumen resultados Crecimiento no lineal (Omara & Harby, 2021)	27
Tabla 4: Resumen resultados Regresión (Omara & Harby, 2021).....	27
Tabla 5: Regresión Polinomial Infección (Chanchí Golondrino et al., 2020).....	28
Tabla 6: Regresión Polinomial Muertes (Chanchí Golondrino et al., 2020).....	28
Tabla 7: Regresión Polinomial Recuperados (Chanchí Golondrino et al., 2020)	29
Tabla 8: Resultados Regresión (Singh & Dalmia, 2020)	31
Tabla 9: Resultados Regresión (Cortés Martínez & Becerra Huertas, 2022)	32
Tabla 10: Resumen resultados (Shawaqfah & Almomani, 2021)	34
Tabla 11: Resumen de Resultados (Dhamodharavadhani, Rathipriya, & Chatterjee, 2020).....	34
Tabla 12: Resumen de Resultados (Luo et al., 2021).....	35
Tabla 13: Resumen de Resultados (Nayan et al., 2022).....	36
Tabla 14: Resumen de Resultados (Torrealba-Rodriguez et al., 2020)	37
Tabla 15: Entrada Justificada (Wieczorek et al., 2020)	39
Tabla 16: Algoritmo de Optimización (Wieczorek et al., 2020).....	40
Tabla 17: Premisas de (Luo et al., 2021).....	42
Tabla 18: Grupos de atributos dataset (Hasell et al., 2020)	44
Tabla 19: Atributos de casos confirmados dataset (Hasell et al., 2020)	44
Tabla 20: Parámetros de arquitectura.....	49
Tabla 21: Historias de Usuario	53
Tabla 22: Tareas de Historias de Usuario	54
Tabla 23: Product Backlog	55

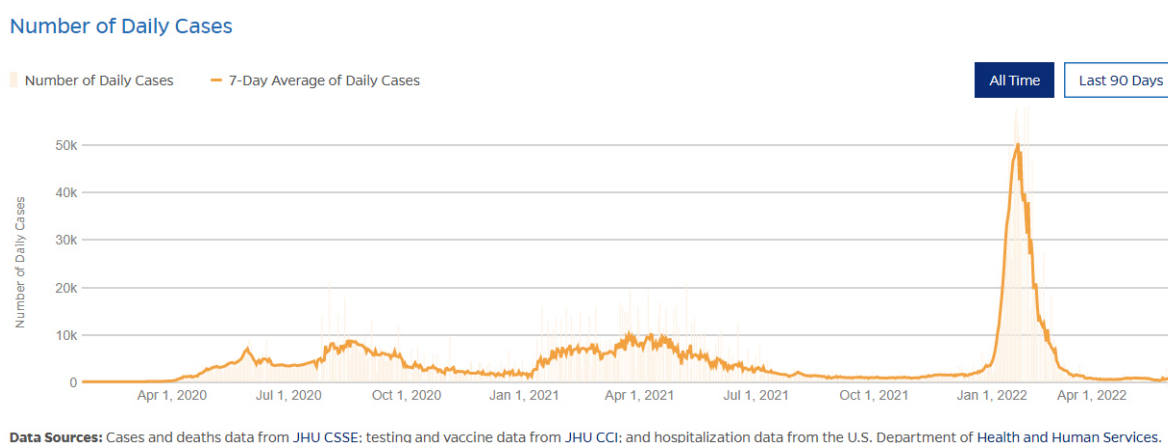
Tabla 24: Endpoint obtener infección 7 días.....	62
Tabla 25: Endpoint obtener infección 7 días y predicción.....	63
Tabla 26: Pruebas API de predicción de infección por COVID 19 en el Perú	64
Tabla 27: Endpoint actualización de modelo	65
Tabla 28: Pruebas Endpoint carga de modelo API de predicción de infección por COVID 19 en el Perú	65
Tabla 29: Pruebas aplicación de consulta de predicción de infección por COVID 19 en el Perú.....	66
Tabla 30: Pruebas aplicación de consulta de infección por COVID 19 en el Perú.....	67
Tabla 31: Distribución de datos de entrenamiento y prueba	70
Tabla 32: Resultados en 2 meses Perú primer periodo	71
Tabla 33: Resultados en 2 meses Perú segundo periodo.....	74
Tabla 34: Comparación entre periodos	77

CAPÍTULO I: INTRODUCCIÓN

1.1. Antecedentes

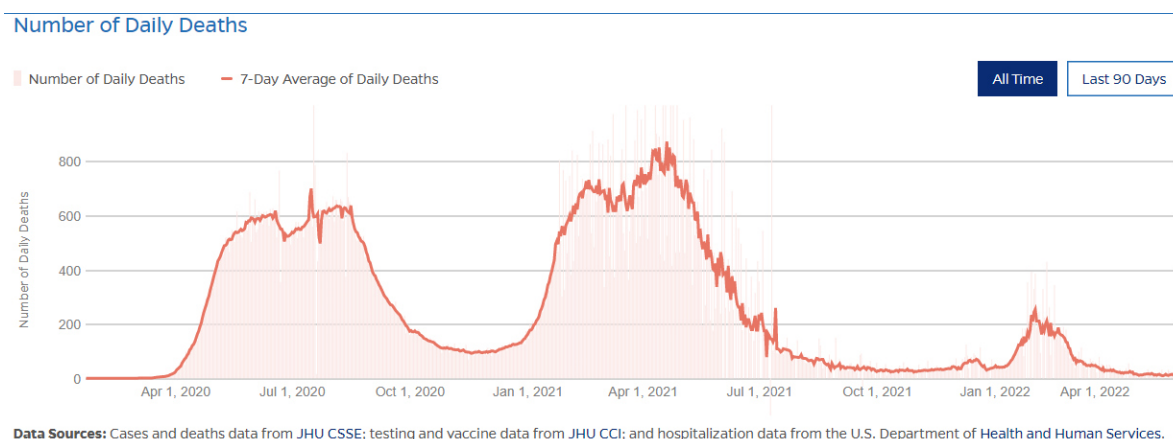
En estos 3 últimos años en el Perú se ha experimentado un gran impacto económico y sanitario a causa de la infección de COVID 19. En la siguiente imagen, extraída del Coronavirus Resource Center de la universidad Jhon Hopkins y elaborado por (Dong et al., 2020) se pudo observar y concluir lo siguiente:

Figura 1: Casos positivos diarios abril 2020-Abril 2022 (Dong et al., 2020)



A pesar de que se ha venido vacunando cada vez en mayor medida a la población peruana, los casos positivos en la primera y segunda ola son alrededor del 10% de los casos positivos en la 3ra ola, como podemos ver en la Figura 1. Esto se debió a la disminución de restricciones y la sobreconfianza de la población en las vacunas.

Figura 2: Fallecimientos diarios Abril 2020-Abril 2022(Dong et al., 2020)



(Dong et al., 2020) nos muestra también a través de la Figura 2 que hubo una disminución enorme en la cantidad de fallecidos diarios en la tercera ola en comparación de la primera y segunda, demostrando de esta forma la eficacia de las vacunas.

Sin embargo, el estado aún sigue haciendo esfuerzos para tratar de reducir el impacto de futuras olas de contagios, ya que estas traen consigo el saturamiento del sistema de salud estatal y privado.

1.2. Descripción del Problema

Con el fin de desarrollar una herramienta capaz de predecir el número de afectados por la infección, se han venido desarrollando modelos predictivos con técnicas estadísticas y de machine learning.

Existen trabajos peruanos con el objetivo de predecir el número de fallecidos e infectados por COVID 19 a nivel mundial a través de redes neuronales artificiales (Mayta et al., 2021), como también existen propuestas de técnicas estadísticas como el uso de modelo de HOLT con el mismo objetivo de predicción, ajustando el rango de predicción a solo el Perú y a solo predecir muertes (Díaz, 2020). Por otro lado, en el mundo se han visto propuestas similares tanto en Asia con el trabajo de (Dhamodharavadhani, Rathipriya, & Chatterjee, 2020) quienes proponen varios modelos entrenados con Probabilistic Neural Network (PNN), Radial Basis Function Neural Network (RBFNN), y Generalized Regression Neural Network (GRNN) con el objetivo

de predecir la cantidad de fallecidos por COVID 19 en el país de India. Como también podemos ver propuestas europeas como las de (Shawaqfah & Almomani, 2021) quienes proponen un modelo de regresión entrenado con redes neuronales para predecir la infección por COVID 19 en Qatar, España e Italia. Además, también existen propuestas norteamericanas como la de (Yu et al., 2021) quienes proponen un modelo estadístico recursivo llamado RLIM basado en el modelo SEIR (Susceptibles, Expuestos, Infectados y Recuperados) con el objetivo de predecir el índice de infección de COVID 19 en Estados Unidos. Finalmente, pero no menos importante, existen propuestas latinoamericanas además de las peruanas mencionadas anteriormente; tal es el caso de la propuesta de (Torrealba-Rodriguez et al., 2020) quienes proponen modelos estadísticos como el Gomperts y Logístico, además de un modelo de redes neuronales para predecir los casos positivos de COVID 19 en México.

Como es notorio, existe un gran interés por parte de la comunidad científica mundial por mantener controlado el comportamiento de los índices de infección y mortalidad por COVID 19. La gran parte de autores que fueron mencionados anteriormente, utilizan como fuente de datos un dataset estático que muestra el comportamiento del virus hasta el momento del desarrollo de la investigación; por otro lado, los autores afirman, además, que los resultados de sus propuestas son muy buenos, pero que la precisión obtenida de sus modelos no es duradera (Mayta et al., 2021; Omara & Harby, 2021). Esto es debido a que los factores de contagio varían conforme pasa el tiempo y por tanto los modelos entrenados con datos hasta cierto periodo no podrán predecir eficazmente más que en futuros cercanos a la fecha de la realización del entrenamiento.

Dicho esto, existe un artículo de (Hasell et al., 2020) que propone una fuente de datos centralizada con atributos como cantidad de infectados, cantidad de muertos, vacunados, nro. de personas con riesgo de muerte, y otros factores importantes organizados por países que además se actualiza casi diariamente. Sin embargo, no hay muchas propuestas que aprovechen

la constante actualización de este dataset; es decir, se está desaprovechando la ventaja que este dataset está ofreciendo para poder mantener un entrenamiento constante a un modelo de predicción de infectados o fallecidos por COVID 19 en cualquier lugar de mundo.

En resumen, el principal problema de los artículos relacionados a la predicción de infección por COVID 19 en cualquier parte del mundo es que la precisión de los modelos desarrollados se pierde conforme transcurre el tiempo. Por esto existe una necesidad de una propuesta de predicción de infección por COVID 19 en el Perú que mantenga su precisión a través del tiempo.

1.2.1. Formulación del Problema

- **Problema General**

- ¿Se puede diseñar una aplicación web capaz de predecir la cantidad de infectados por COVID 19 en el Perú y mantener su precisión a través de la automatización de un flujo de reentrenamiento?

- **Problemas Específicos**

- ¿Qué método de machine learning es el mejor para predecir la cantidad de infectados por COVID 19 en el Perú?
- ¿Qué tipo de arquitectura de red neuronal es más precisa para predecir la cantidad de infectados en el Perú?
- ¿Cómo podemos validar el modelo de predicción de infección propuesto?
- ¿Qué tipo de arquitectura de software es mejor para desarrollar y mantener la aplicación web anteriormente mencionada?
- ¿A través de qué mecanismo se puede automatizar el flujo de reentrenamiento del modelo de predicción de infección por COVID 19 en el Perú?

1.3. Importancia

A pesar de esta disminución en los fallecidos en las últimas olas, hemos podido notar la sobresaturación de la atención médica tanto en centros públicos como privados. (El Comercio Perú & Raffo, 2022) afirma que la tercera ola fue la que tuvo mayor cantidad de contagiados durante toda la pandemia, debido a esto aseguradoras particulares como Rimac Seguros informaron que la atención a consultas y emergencias sobrepasó los 5 mil casos por día; otra aseguradora particular como Pacífico EPS aseguró que la atención de médicos a domicilio pasó de 350 a 1200 al día.

Si bien es cierto, se trató de cubrir como se pudo la saturación de la atención médica; resulta complicado actuar sobre la marcha de una ola de contagios, además, no podemos dejar en manos de la vacuna el control de esta pandemia, para ello el estado necesita una herramienta que fundamente sus planes o estrategias para combatir la infección. Además, muchas herramientas parecidas ya existen en otros países, pero todas se encuentran con el mismo problema de perder precisión conforme transcurre el tiempo.

Es por esto por lo que se propone una herramienta de predicción de contagios por COVID19 en el Perú que conservará su precisión a través del entrenamiento continuo a lo largo del tiempo. Esto ayudará al estado a establecer un plan de acción con anticipación a las olas de contagio, evitando de esta manera la saturación de la atención médica ya sea: estableciendo cuarentenas estratégicas, medidas protocolares, aumento de recursos y demás estrategias para contrarrestar el impacto de una ola de contagios COVID 19. Así como también se establecerá un precedente estratégico para futuras pandemias.

1.4. Motivación

Alrededor del mundo existen muchas investigaciones y propuestas de predicción tanto de mortalidad e infección de COVID 19 en base al tiempo y a través de métodos de machine

learning como (Dhamodharavadhani et al., 2020; Shawaqfah & Almomani, 2021; M. & K., 2021; Chanchí Golondrino et al., 2020). Y cada uno de ellos arroja buenos resultados, pero distintos al fin y al cabo. Esto se debe al distinto ambiente en el que se realiza la investigación. Es decir, la infección no se comporta de igual forma en todos los rincones del mundo. Este comportamiento es afectado no solamente por el tiempo sino también por eventos nacionales e internacionales, cultura, costumbres y principalmente concientización y esfuerzo por parte del estado y principalmente la población.

En el Perú existen contadas propuestas de predicción de infección o muertes, como es el caso de (Mayta et al. 2021) quienes en su trabajo proponen un modelo de regresión a través de redes neuronales artificiales para poder predecir el índice de fallecidos por COVID 19 en el Perú; sin embargo, no resalta algún trabajo que proponga un modelo de predicción de infección por COVID 19 en el país que mantenga su precisión a lo largo del tiempo. Es por esta razón que este trabajo tiene como principal objetivo desarrollar un sistema web capaz de predecir el índice de infección por COVID19 en el Perú utilizando métodos de machine learning y utilizar un enfoque de reentrenamiento automático para mantener la precisión a lo largo del tiempo.

1.5. Objetivos

- **Objetivo General**

- Diseñar e implementar un sistema web de predicción de infectados por COVID 19 en el Perú y que mantenga su precisión a lo largo del tiempo a través de un flujo automático de reentrenamiento de Machine Learning.

- **Objetivos específicos**

- Determinar el método de machine learning que proporcione los resultados más precisos y duraderos para predecir la cantidad de infectados por COVID 19 en el Perú.

- Determinar la arquitectura de red neuronal que proporcione los resultados más precisos y duraderos de predicción de contagiados por COVID 19 en el Perú.
- Determinar un medio eficaz de validar los modelos de predicción de infección por COVID 19 en el Perú.
- Diseñar una arquitectura de aplicación web e inteligente para desarrollar y mantener el entrenamiento constante de sistema web de predicción.
- Diseñar un flujo de reentrenamiento automatizado para mantener la precisión del modelo de predicción de infección por COVID 19 en base al día seleccionado en el Perú.

1.6. Propuesta

Se propone un sistema web capaz de predecir la cantidad de casos positivos de COVID 19 en el Perú a través de métodos de machine learning y permitir que el modelo predictivo que utilice dicho sistema se actualice constantemente de manera automática conforme se va actualizando los datos de infección por COVID 19 en el Perú.

1.7. Estructura de la Tesis

La presente tesis está estructurada en 6 capítulos que se detallan a continuación.

- El capítulo 1 en el cual se detallaron los antecedentes, la problemática, la importancia, la motivación y los objetivos para esta investigación.
- En el capítulo 2, se describe el marco teórico en el cual define qué es el modelo de regresión polinomial, red neuronal multicapa, recurrente y redes neuronales LSTM.
- En el capítulo 3, se hace un detalle y análisis del estado del arte de investigaciones que han desarrollado modelos estadísticos y de machine learning para predecir el índice de infección por COVID 19 en distintos países.
- En el capítulo 4, se presenta el aporte en el cual se detalla el dataset a utilizar, el modelo propuesto para el índice de infección por COVID 19 en el Perú, además de los detalles

de implementación del enfoque de reentrenamiento automatizado de este modelo.

- En el capítulo 5 se analizan los resultados obtenidos del entrenamiento y reentrenamiento automatizado del modelo de predicción de infección por COVID 19 en el Perú, además de presentar las conclusiones y trabajos futuros.

CAPÍTULO 2: MARCO TEÓRICO

2.1. Modelo de regresión polinomial

(Chanchí Golondrino et al, 2020) Explica que la regresión polinomial es un caso especial de regresión lineal múltiple donde se busca predecir una variable dependiente a través de una variable independiente. Esta relación además tiene un modelo o patrón representado por una función polinomial.

Mediante la regresión es posible obtener una ecuación representada de la siguiente manera (1):

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \dots + \beta_p X^p + \varepsilon \quad (1)$$

En donde (1) representa a la ecuación de predicción de Y en base a X con un error de ε

2.2. Modelo de Red Neuronal Multicapa

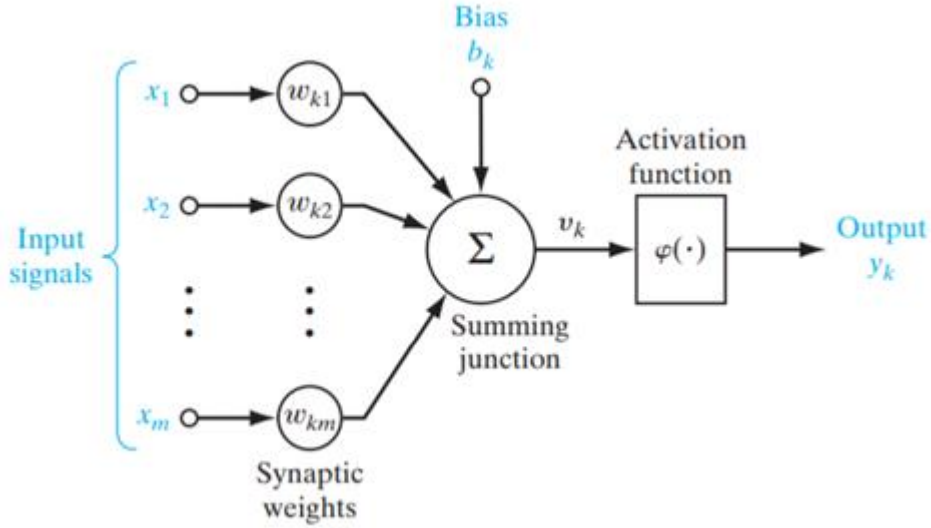
El modelo de red neuronal (Haykin, 2016) trata de imitar la fisiología de un conjunto de neuronas reales, cada neurona o perceptrón contiene ciertos elementos como:

Pesos sinápticos: Cada entrada va a ser multiplicada por un peso especificado para luego pasar a la unión sumadora.

Unión sumadora: Luego de que cada entrada sea multiplicada por el peso correspondiente, estos productos van a ser sumados dando un resultado que pasará a ser la entrada de una función de activación.

Función de activación: Esta función limitará la salida a los valores de 1 o 0 dependiendo de los requerimientos del caso.

Figura 3: Perceptrón de Rosenblatt (Haykin, 2016)



En la Figura 3 se puede observar de forma más clara el esquema planteado anteriormente.

Pasando a una notación más matemática se tiene que una neurona “k” se puede describir a través de la siguiente ecuación (2):

$$u_k = \sum_{j=1}^m w_{kj} x_k \quad (2)$$

Siendo u la salida de la unión sumadora y la entrada de la función de activación (3):

$$y_k = \phi(u_k + b_k) \quad (3)$$

Terminando en y como salida de la neurona y phi como función de activación.

Y un conjunto contiguo de estos perceptrones es el que forma una red neuronal.

2.3. Modelo de Red neuronal recurrente:

(Luo et al., 2021) define en pocas palabras que una red neuronal recurrente es un tipo de red neuronal artificial donde las conexiones entre los nodos forman un grafo dirigido en base a una secuencia temporal. Esto permite a la red neuronal contar con un comportamiento dinámico temporal ya que estas mismas pueden utilizar su estado interno o “memoria” para procesar secuencias de variables de longitud variable. Una RNN puede ser entendida como múltiples copias de la misma red neuronal cada una pasando su salida como entrada de la siguiente, logrando de esta forma conectar información pasada a una tarea actual.

2.3.1. Redes neuronales LSTM (Long short-term memory)

(Sun, 2021) explica que las redes neuronales LSTM es una arquitectura de Red Neuronal Recurrente usado en el campo de deep learning y es un algoritmo eficiente para la construcción de modelos de series de tiempo.

(Luo et al., 2021) menciona que este tipo de arquitectura encaja muy bien cuando se quiere clasificar, procesar y hacer predicciones basadas en datos de series de tiempo, Las redes neuronales LSTM tienen la forma de una cadena de repetición de módulos, pero el módulo de repetición tiene distinta estructura ya que, en vez de estar compuesto por una sola capa, esta está formada por 4 capas que interactúan de forma especial. Una unidad de LSTM está compuesta por:

- Celda de memoria: Encargada de recordar valores sobre intervalos de tiempo arbitrarios y las demás compuertas son las encargadas de controlar el flujo dentro y fuera de esta celda.
- Compuerta de olvido: Encargada de seleccionar qué información va a ser olvidada de la celda de memoria.
- Compuerta de entrada: Encargada de decidir qué información nueva va a ser guardada en la celda de memoria.
- Compuerta de salida: Encargada de seleccionar qué valor deseamos como salida

Las compuertas en LSTM procesan usando una función sigmoide de activación con salidas de 0 o 1. Siendo 0 el bloqueo de todo y 1 el permiso de todo.

Figura 4: Arquitectura de celda LSTM (Luo et al., 2021)



CAPÍTULO 3: ESTADO DEL ARTE

El comportamiento de la infección por COVID 19 ha sido de mucho interés estos últimos años, múltiples investigadores han aplicado distintas técnicas de machine learning para tratar de predecir el comportamiento de la infección en sus distintos países de residencia, origen o interés. Por esta razón se realizó una revisión de la literatura de artículos científicos para fundamentar y sustentar el problema, estos artículos en su mayoría fueron extraídos de las bases de datos Scopus y Web of Science. Se seleccionaron 21 artículos que trataron de resolver el problema de predicción de infección por COVID 19 en distintos países incluyendo el Perú y un artículo que propone un repositorio de datos centralizado y frecuentemente actualizado donde encontramos los datos de infección, muertes y vacunaciones en muchos países del mundo.

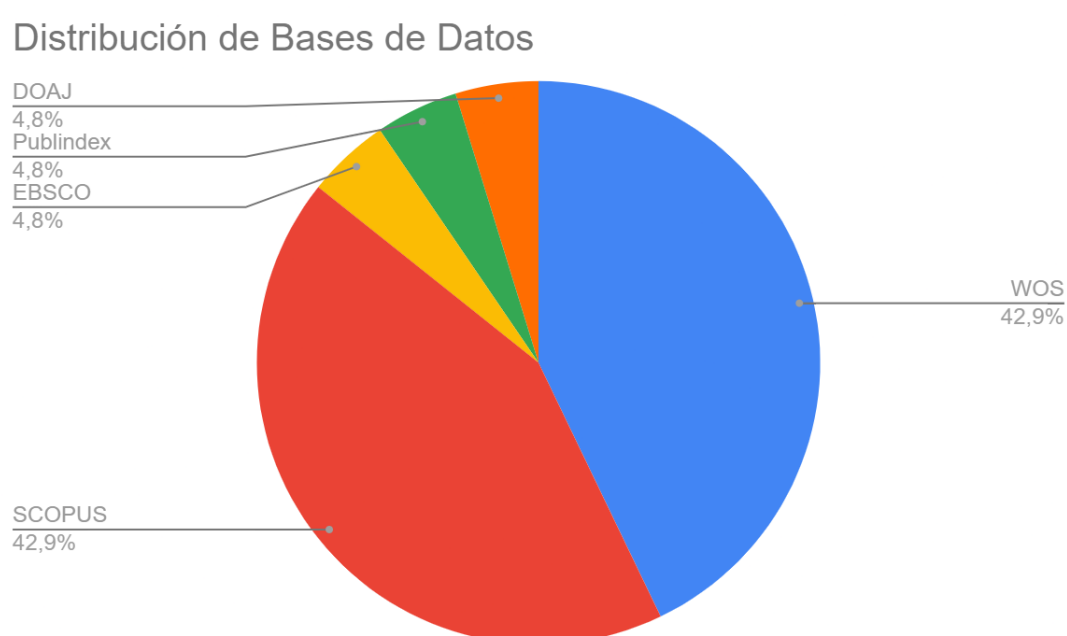
Para enfocar correctamente esta revisión literaria, primero se plantearon las siguientes preguntas:

- ¿Cuál es el método estadístico más utilizado para la predicción de infección o muerte por COVID 19 en el mundo?
- ¿Cuál es el método de machine learning más utilizado para la predicción de infección o muerte por COVID 19 en el mundo?
- ¿Qué métodos se utilizaron para validar los modelos estadísticos de predicción de infección o muerte por COVID 19 en el mundo?
- ¿Qué métodos se utilizaron para validar los modelos de machine learning de predicción de infección o muerte por COVID 19 en el mundo?
- ¿Existe algún dataset que indique el índice de infección por COVID 19 en el Perú y sea actualizado constantemente?

En base a estas preguntas que buscaban enfocar esta revisión acorde a los objetivos establecidos en el capítulo I, se logró obtener una cantidad aproximada de 50 papers, que fueron siendo filtrados en base a la claridad de sus resultados, la precisión de las propuestas y la complejidad del método utilizado.

Finalmente se obtuvo la siguiente distribución (Ver Figura 5) de Bases de Datos indexadas de las cuales fueron extraídos los artículos seleccionados:

Figura 5: Distribución de bases de datos indexadas



Los artículos seleccionados fueron clasificados por tres tipos: Métodos de predicción estadísticos, Métodos de predicción de Machine learning y Dataset (Ver Tabla 1)

Tabla 1: Resumen del estado del arte

	Tipo	Nombre	Autor	Año
1	Estadístico	Predicción de Covid-19 a nivel mundial para el año 2021	P. Jorge	2020
2	Machine Learning	Predicción de mortalidad a causa del Covid 19 en Perú utilizando redes neuronales artificiales	C. Mayta Avalos, J. C. Valdivia Mamani, F. Rosales Castilla y M. Gines Colana	2021

3	Machine Learning	COVID-19 Mortality Rate Prediction for India Using Statistical Neural Network Models	S. Dhamodharavadhani, R. Rathipriya y J. M. Chatterjee	2020
4	Machine Learning	Forecast of the outbreak of COVID-19 using artificial neural network: Case study Qatar, Spain, and Italy	M. Shawaqfah y F. Almomani	2021
5	Machine Learning	Neural network powered COVID-19 spread forecasting model	M. Wiczorek, J. Silka y M. Woźniak	2020
6	Machine Learning	The Prediction of COVID-19 Using LSTM Algorithms	M. H. Kim, J. H. Kim, K. Lee y G.-Y. Gim	2021
7	Estadístico	Analysis of COVID-19 Based on Several Machine Learning Techniques	T. Sun	2021
8	Estadístico	Estimation of Death and Recovery Rates using Covid-19 Data of Pakistan	m. kashif	2021
9	Estadístico	Using Mathematical and Statistical Model to Forecast the Path of Infection by Covid-19 in the Kingdom of Saudi Arabia	T. Omara y K. A. Harby	2021
10	Estadístico	Aplicación de la regresión polinomial para la caracterización de la curva del COVID-19, mediante técnicas de machine learning	G. E. Chanchí Golondrino, W. Y. Campo Muñoz y L. M. Sierra Martinez	2020
11	Estadístico	COVID-19 Prediction Models and Unexploited Data	K. C. Santosh	2020
12	Machine Learning	Time series prediction of COVID-19 transmission in America using LSTM and XGBoost algorithms	Junling Luo, Zhongliang Zhang, Yao Fu, Feng Rao	2021

13	Machine Learning	Coronavirus disease situation analysis and prediction using machine learning: a study on Bangladeshi population	Al-Akhir Nayan , Boonserm Kijirikul , Yuji Iwahori	2022
14	Estadístico	A Two-Step Polynomial and Nonlinear Growth Approach for Modeling COVID-19 Cases in Mexico	R. Pérez Abreu C., S. Estrada y H. de-la-Torre-Gutiérrez	2021
15	Estadístico	Polynomial Based Linear Regression Model to Predict COVID-19 Cases	Nikhil, A. Saini, S. Panday y N. Gupta	2021
16	Estadístico	Forecasting COVID-19 outbreak progression using hybrid polynomial-Bayesian ridge regression model	M. Saqib	2021
17	Estadístico	Prediction of number of fatalities due to Covid-19 using Machine Learning	Singh, M., & Dalmia, S.	2020
18	Machine Learning	Modeling and Prediction of COVID-19 in Mexico applying mathematical and computational models	O. Torrealb-Rodriguez, R.A. Conde-Gutiérrez y A.L. Hernández Javier	2020
19	Estadístico	RLIM: a recursive and latent infection model for the prediction of US COVID-19 infections and turning points	Xiang Yu · Lihua Lu · Jianyi Shen · Jiandun Li · Wei Xiao · Yangquan Chen	2021
20	Estadístico	Characterization of the COVID-19 trend in Colombia with polynomial regressions	Cortés Martínez, A. E., & Becerra Huertas, C. E.	2022
21	DataSet	A cross-country database of COVID-19 testing	Hasell, J., Mathieu, E., Beltekian, D. et al.	2020

3.1. Dataset de datos de infección por COVID 19

En las múltiples investigaciones mencionadas anteriormente, la mayor cantidad de fuentes de datos, son las instituciones estatales encargadas del área médica como del área estadística del país en el que son elaborados los artículos. Sin embargo, existe un repositorio de datos centralizado y frecuentemente actualizado, tal es el caso de la propuesta de (Hasell et al., 2020) que a través de scrapping de estadística de infección oficial de los 94 países de los que recolecta

información, crea un repositorio centralizado de datos importantes como son la cantidad de infectados nuevos por día, la cantidad de infectados acumulados, la cantidad de fallecidos por día, la cantidad de fallecidos acumulados, cantidad de vacunados, etc. Esta información por cada uno de los países que contempla este repositorio.

3.2. Métodos Estadísticos para la predicción de infección por COVID 19

En la literatura de la predicción de infección por COVID 19 ya existen varios métodos estadísticos propuestos, los cuales podemos encontrarlos en las siguientes investigaciones:

(Díaz Pinzón, 2020) propone en su artículo el desarrollo de un modelo HOLT o de suavizamiento exponencial doble al conjunto de datos extraídos del tablero COVID del Centro de Ciencias e Ingeniería de Sistema (CSSE) de la Universidad Johns Hopkins, de este conjunto de datos se utilizarían los campos de infectados, muertos y los días. Esta propuesta fue evaluada con el Coeficiente de Determinación o R^2 obteniendo un valor de 99.6% de precisión y a través del error porcentual se obtuvo un valor de 0.27%, siendo este un modelo muy preciso y concluyendo que posee la precisión necesaria para tomar decisiones que requieren un conocimiento aproximado, mas no exacto de la cantidad de infectados o fallecidos.

(Kashif, 2021) tiene como objetivo estimar los índices de recuperación y muerte de los pacientes de COVID 19 en Pakistán. Para ello utilizó modelos de regresión lineal y polinomial con data recolectada a lo largo de las provincias de Pakistán. Obtuvo como resultado el coeficiente de determinación de cada modelo propuesto en cada provincia:

Tabla 2: Resumen de resultados (Kashif, 2021)

Resultados Pakistan		
<i>Provincia</i>	<i>R2</i>	<i>Grado de polinomio</i>
Pakistan	99,7	3
Punjab	97,7	3
Sindh	99,4	3

Khyber Phaktunkhwa	99,8	3
Baluchistan	96,3	3
Gilgit Balstistan	76,9	3
Azzad Jammu Kashmir	99,8	3
Islamabad Capital Territory	99,4	3

Los autores, con los resultados obtenidos en la Tabla 2, concluyen que el modelo de regresión polinomial de grado 3, se ajusta bastante a la mayoría de las localidades en las que se evaluó. Sin embargo, en una Provincia se observó que el resultado fue por debajo del esperado (R^2 menor a 95%), por lo que se debería personalizar el modelo o encontrar el que se ajuste de mejor forma, por provincia y no en conjunto.

(Omara & Harby, 2021) con el objetivo de encontrar un modelo que se adapte mejor a la curva de contagios por COVID 19 en Arabia Saudita proponen distintos modelos, entre ellos: Modelos de crecimiento no lineal, modelo SEIR, modelos de regresión. Entre sus principales resultados para los modelos de crecimiento no lineal mostrado en la Tabla 3 se obtuvo:

Tabla 3: Resumen resultados Crecimiento no lineal (Omara & Harby, 2021)

Modelos de Crecimiento no lineal	
<i>Modelos</i>	<i>R²</i>
Logístico	0.558
Logístico Generalizado	0.862
Weibull	0.906
Mitscherlich	0.452
Gompertz	0.568

Para el modelo SEIR se obtuvo un coeficiente de determinación de 0.882 y finalmente los resultados de los modelos de regresión, mostrados en la Tabla 4, fueron los siguientes:

Tabla 4: Resumen resultados Regresión (Omara & Harby, 2021)

Modelos de Regresión	
<i>Modelos</i>	<i>R²</i>

Cuadrática	0.064
Polinomial n=2	0.803
Polinomial n=3	0.904
Polinomial n=4	0.910

Con estos resultados los autores concluyen que, de los modelos propuestos, los que más se ajustan a la curva de infección del COVID 19 en Arabia Saudita son los modelos de Weibull, el modelo SEIR (específicamente para predicciones a mayor plazo de tiempo) y el modelo polinomial de grado 4 para las secciones de modelos de crecimiento no lineal, SEIR y modelos de regresión respectivamente.

(Chanchí Golondrino et al., 2020) con el objetivo de caracterizar el comportamiento de la curva de casos positivos, muertes y personas recuperadas de COVID 19 en Colombia, los autores proponen la utilización de datos de reportes estatales del ministerio de Salud de Colombia para entrenar un modelo de regresión polinomial capaz de predecir la tendencia de los índices anteriormente mencionados. Los autores realizan el entrenamiento en los distintos tipos de datos objetivo: infectados, fallecidos y recuperados. Este entrenamiento lo repiten variando el grado polinomial yendo del 1 al 20. Obteniendo los tres mejores resultados, organizados en las Tablas 5, 6 y 7:

Tabla 5: Regresión Polinomial Infección (Chanchí Golondrino et al., 2020)

Modelos de Regresión Polinomial-Infección		
Grados polinomiales	RMSE	R2
9	53.91408897	0.999564117
10	52.84168399	0.999581285
11	52.17891103	0.999581285

Tabla 6: Regresión Polinomial Muertes (Chanchí Golondrino et al., 2020)

Modelos de Regresión Polinomial-Infección		
Grados polinomiales	RMSE	R2
8	2.532089425	0.999559795
9	2.540729209	0.999556786
10	2.529539736	0.999560681

Tabla 7: Regresión Polinomial Recuperados (Chanchí Golondrino et al., 2020)

Modelos de Regresión Polinomial-Infección		
Grados polinomiales	RMSE	R ²
11	19.58884614	0.999039444
12	18.79872151	0.99911537
13	19.87213183	0.999011461

Los autores concluyen que los valores obtenidos por las métricas de RMSE y R^2 nos permiten verificar que el modelo de regresión polinomial se ajusta exitosamente a la curva de infección, fallecimientos y recuperación de las personas afectadas por COVID 19.

(Santosh, 2020) propone la revisión de modelos SEIR/SIR, basados en agente, técnicas de curve-fitting para discutir sobre cómo los factores imprevistos afectan al diseño de las propuestas que utilizan los modelos anteriormente mencionados al ser utilizados para predecir el índice de infección por COVID 19 en las distintas localidades de papers analizados. Al finalizar la revisión de los modelos encontrados, el autor concluye que es importante tratar de enfocarse e incluir aquellos factores que desvían significativamente los resultados de las predicciones y se recomienda que los modelos realizados sean mejorados constantemente conforme pasa el tiempo.

(Pérez Abreu C. et al., 2021) con el objetivo de predecir los casos positivos de COVID 19 en México, los autores proponen el uso de un modelo de regresión polinomial y un modelo de crecimiento no lineal. El uso de estos modelos no fue en paralelo, sino que se utilizaron en serie, puesto que el modelo de regresión polinomial se utilizó para la predicción de infectados por COVID 19 hasta llegar al punto de inflexión o la caída de infectados, donde se comenzó a utilizar un modelo no lineal para predecir durante la mitad y final estimados de la pandemia. Los resultados obtenidos de todos los modelos propuestos fueron de un rango entre 300 a 600 de RMSE en un plazo de 30 días antes de un pico de infección a más de 20 días después del

pico de infección. Con los resultados obtenidos los autores concluyeron que no solo el modelo polinomial encajaba bien al comportamiento de la infección días antes de un pico de infección, sino que también lo hacían los modelos no lineales como Gompertz y Logístico. Sin embargo, afirman que esta herramienta no es de predicción exacta, sino solamente una herramienta de prevención y preparación frente a posibles olas de contagios.

(Nikhil et al., 2021) proponen el uso de un modelo lineal basado en regresión polinomial para la predicción del número de infectados por COVID 19 en el mundo. A través del entrenamiento con data extraída de una publicación de la Unión Europea sobre índice de infección y fallecimiento por COVID 19 a nivel mundial, se obtuvo un modelo de regresión polinomial con un R^2 de 0.98295. El artículo es finalizado con la conclusión del autor de que los métodos de machine learning en este caso el de regresión polinomial es bastante eficaz como apoyo de toma de decisiones de prevención de contagios por COVID 19.

(Saqib, 2020) con el objetivo de estimar la cantidad de infectados por COVID 19 en el mundo en base a una fecha seleccionada propone el uso de un modelo de predicción basado en Regresión lineal Bayesiana hibridizada con regresión polinomial y que utiliza distribución probabilística para estimar el valor predictivo. Esta propuesta tiene como resultados un coeficiente de determinación de 0.91 y un error cuadrático medio de 0.04. Finalmente, el autor concluye que su modelo posee una precisión bastante válida y que el error mostrado por su modelo es insignificante comparado a la población afectada por el virus, por esto menciona que su modelo puede ser utilizado como base para la toma de decisiones y reducir el impacto del virus en la población mundial.

(Singh & Dalmia, 2020) proponen un modelo de regresión lineal y otro de regresión polinomial para predecir el índice de infectados por COVID 19 en India. Adicionalmente estos autores utilizan el dataset (Hasell et al., 2020) descrito en este trabajo, filtrándolo en el país objetivo (India) y particionando los datos en un 66% para entrenamiento y un 33% para

verificación del modelo. Como resultados los autores obtuvieron los siguientes resultados mostrados en la Tabla 8:

Tabla 8: Resultados Regresión (Singh & Dalmia, 2020)

Modelos de Regresión		
<i>Modelos</i>	<i>R²</i>	<i>RMSE</i>
Lineal	0.9979	0.000124
Polinomial grado 3	0.9998	0.000010

Finalmente, los autores concluyen que para el desarrollo de una API que proporciona estas predicciones a entidades gubernamentales, se prefirió utilizar el modelo de regresión polinomial debido a que su resultado es superior, además comentan que el modelo propuesto podrá fundamentar las estrategias o decisiones tomadas para contrarrestar el impacto de la infección.

(Yu et al., 2021) proponen, para el desarrollo de un modelo recursivo de predicción de infección de COVID 19 en Estados Unidos, el entrenamiento de un modelo basado en el modelo SEIR de infección de COVID 19; a este modelo se le agrega un proceso recursivo para lograr la optimización de este mismo y esta propuesta es llamado como RLIM. A través de este método y modelo propuesto, los autores pudieron obtener como resultado la mejora del modelo en base al error porcentual o AFER desde 29% a 15%. Los autores concluyen que el modelo se ajusta a la infección en Estados Unidos, además ellos consideran como ventaja el hecho de que su modelo no incluya factores externos como son las cuarentenas, el clima o la capacidad hospitalaria. Esto se debe a que gracias a que no son considerados estos factores la predicción puede realizarse sin ninguna otra información necesaria además de la fecha que se busque.

(Cortés Martínez & Becerra Huertas, 2022) con el objetivo de desarrollar un grupo de modelos regresión polinomiales diferenciados por su grado polinomial para poder predecir el índice de infección y mortalidad por COVID 19 en Colombia, los autores utilizaron la herramienta Microsoft Office Excel para desarrollar los modelos matemáticos de regresión polinomial y los

datos fueron extraídos de los informes oficiales de infección y mortalidad estatales de Colombia. Los resultados obtenidos se muestran en la Tabla 9:

Tabla 9: Resultados Regresión (Cortés Martínez & Becerra Huertas, 2022)

Modelos de Regresión Polinomial-Infección		
<i>Objetivo</i>	<i>Fallecidos</i>	<i>Contagiados</i>
<i>Grados polinomiales</i>	<i>R2</i>	<i>R2</i>
1	0.96050	0.94426
2	0.98191	0.98834
3	0.98466	0.99657
4	0.99466	0.99667
5	0.99535	0.997096
6	0.9984	0.99777

Los autores concluyen que los resultados mostrados son bastante favorables para el método de desarrollo de estos modelos. Comentan que al utilizar una herramienta al alcance de cualquier profesional como lo es Microsoft Office Excel hay una gran oportunidad de desarrollar modelos fácilmente gracias a esta herramienta.

En resumen:

- (Cortés Martínez & Becerra Huertas, 2022; Singh & Dalmia, 2020; Nikhil et al., 2021; Pérez Abreu C. et al., 2021; Chanchí Golondrino et al., 2020; Omara & Harby, 2021; Kashif, 2021) Proponen un modelo de regresión polinomial en su mayoría con grados polinomiales entre 5 a 11, obteniendo con estos modelos resultados bastante precisos; esto nos indica que el modelo de regresión estadístico más popular o utilizado para este tipo de predicciones de infección es el polinomial. Sin embargo, (Omara & Harby, 2021) nos mencionan que estos modelos son útiles para predicciones a corto plazo y que para predicciones a mayor plazo es necesario considerar un modelo epidemiológico.
- En general se pudo encontrar que las propuestas de (Cortés Martínez & Becerra Huertas, 2022; Singh & Dalmia, 2020; Nikhil et al., 2021; Pérez Abreu C. et al., 2021; Chanchí Golondrino et al., 2020; Omara & Harby, 2021; Kashif, 2021; Yu et al., 2021; Saqib,

2020; Díaz Pinzón, 2020) fueron validadas a través de las métricas de RMSE, porcentaje de error o R^2 siendo estas métricas las más utilizadas y confiables para los autores.

- (Singh & Dalmia, 2020) proponen su modelo utilizando el dataset que se actualiza diariamente, sin embargo, no realizan un reentrenamiento constante, pero sí constatan que es de baja complejidad el desarrollo con el uso de herramientas como Microsoft Excel.
- Se puede observar que (Yu et al., 2021; Santosh, 2020; Omara & Harby, 2021) proponen modelos basados en el modelo SEIR, siendo este un modelo epidemiológico y que facilita la interpretación matemática del comportamiento de la infección de COVID 19 en cualquier parte del mundo.

3.3. Métodos de Machine Learning para la predicción de infección por COVID 19

En la literatura de predicción de infección de COVID 19 en distintas partes del mundo utilizan distintos métodos de Machine Learning para lograr desarrollar un modelo predictivo válido y con buenos resultados, entre algunas de estas investigaciones tenemos:

(Mayta et al., 2021) proponen una herramienta de predicción de mortalidad o cantidad de fallecidos por COVID 19 en función del tiempo. Los autores utilizaron redes neuronales artificiales y series temporales para el entrenamiento del modelo. Como datos de entrada utilizaron información proporcionada por el Ministerio de Salud del estado peruano a través del portal de datos abiertos. Y entrenaron su modelo en base a los fallecidos diariamente. Los resultados de la evaluación de este modelo fueron en base al error cuadrático medio: 0.0037 y la pérdida de 0.048. Concluyendo que la herramienta de predicción tiene validez, sin embargo, de forma temporal, puesto que es cuestión de tiempo que los factores para la predicción varíen.

(Shawaqfah & Almomani, 2021) proponen una arquitectura de una red neuronal artificial con el objetivo de predecir los efectos de la pandemia en Qatar, España e Italia. Obtuvieron sus

datos de la información estadística estatal de cada país, y con estos mismos datos se entrenó el modelo y se validó. Los resultados del modelo propuesto fueron en base al coeficiente de regresión R^2 y a la raíz del error medio cuadrado (RMS), obteniendo los siguientes resultados mostrados en la Tabla 10:

Tabla 10: Resumen resultados (Shawaqfah & Almomani, 2021)

Resumen resultados Shawaqfah & Almomani		
<i>País</i>	<i>R²</i>	<i>RMS</i>
Qatar	0.96	6.08
España	0.99	0.53
Italia	0.95	0.51

Concluyendo de esta forma que los modelos entrenados a través de redes neuronales son modelos bastante válidos para este tipo de casos; sin embargo, no es posible la introducción de ciertos factores exógenos que influyen en el contagio de la población.

(Dhamodharavadhani, Rathipriya, & Chatterjee, 2020) propusieron tres modelos de red neuronal: Red neuronal probabilística (PNN), Red neuronal de función de base radial (RBFNN) y una Red neuronal de regresión general (GRNN); para poder predecir el índice de muertos por COVID 19 en la India. Para el entrenamiento y evaluación utilizaron dos datasets conocidos D1 y D2 donde contienen series temporales de muertes por COVID 19, y casos confirmados de COVID 19 y muertes por COVID19 respectivamente. Los resultados de sus modelos fueron evaluados en base a la raíz del error medio cuadrado y del coeficiente de determinación R^2 ; y son mostrados en la Tabla 11.

Tabla 11: Resumen de Resultados (Dhamodharavadhani, Rathipriya, & Chatterjee, 2020)

Datasets	Resultados Dataset D1		Resultados Dataset D2	
<i>Modelos</i>	<i>R²</i>	<i>RMS</i>	<i>R²</i>	<i>RMS</i>
PNN	0.999	8.89	999	7.898
GRNN	0.999	9.714	999	8.389
RBFNN	0.999	8.528	999	9.505

Con estos resultados los autores concluyen que el modelo de PNN y RBFNN son los más precisos cuando son entrenados con el dataset D2.

(Wieczorek et al., 2020) proponen una red neuronal artificial para la predicción de infección por COVID 19 utilizando el algoritmo de entrenamiento NAdam. Para el entrenamiento y evaluación se utilizaron datos de repositorios abiertos como los provistos por la CSSE de la universidad de Jhons Hopkins. Como resultados obtienen que la precisión promedio entre la mayoría de países es de 87.7% y con esto concluyen que la precisión puede ser mayor si se entrena un predictor por cada región.

(Sun, 2021) a diferencia de los demás autores, analiza un grupo de algoritmos para encontrar cuál es el más preciso entre este grupo para ello realiza una comparación entre distintos métodos de predicción tales como regresión polinomial múltiple, modelo de crecimiento logístico y LSTM para la predicción de los efectos de la pandemia del COVID 19.

Se mide la precisión de los algoritmos a través del coeficiente de determinación y de la raíz del error medio cuadrado. Concluyen finalmente que el modelo LSTM es el más efectivo entre todos los modelos comparados con un coeficiente de determinación de $R^2 = 0.97$.

(Luo et al., 2021) con el objetivo de obtener un estimado de casos positivos de infección de COVID 19 en Estados Unidos propusieron y compararon dos modelos: un modelo LSTM y otro modelo XGBoost. De esta forma se obtuvieron los siguientes resultados; mostrados en la Tabla 12; en base al error medio (MAE), el error cuadrático medio (MSE), raíz del error cuadrático medio (RMSE) y error porcentual absoluto medio (MAPE):

Tabla 12: Resumen de Resultados (Luo et al., 2021)

Modelos LSTM y XGBoost				
<i>Modelos</i>	<i>MAE</i>	<i>MSE</i>	<i>RMSE</i>	<i>MAPE(%)</i>
LSTM	771	962577	981	2.32
XGBoost	2.658	11.964.681	3.459	7.21

Concluyendo que el mejor Algoritmo para predecir un estimado de contagios es el LSTM; sin embargo, mencionan que sus propuestas tienen muchas limitaciones, como por ejemplo la longitud de la muestra y que hace falta un suavizado en la curva para mejorar el entrenamiento de los modelos.

(Nayan et al., 2022) buscando predecir el número de infectados y fallecidos por COVID 19 en Bangladesh los autores proponen tres modelos: uno de Multi-Layer Perceptron (MLP), uno de Support Vector Regression y uno de regresión lineal. Se desarrollaron 5 regresores por cada tipo de modelo cambiando ciertos parámetros como por ejemplo: En el modelo MLP se utilizaron 1k, 5k y 10k iteraciones en los 3 primeros regresores y en los dos últimos se utilizó relu y tanh como funciones de activación respectivamente y lbfgs y sgd como optimizadores respectivamente también y con 1k iteraciones como máximo para ambos. En el modelo SVR se varió en Kernel RBF, grado polinomial 5, lineal, grado polinomial 2 y grado polinomial 7 para cada regresor respectivamente. Finalmente para el modelo de regresión lineal se utilizaron índices de aprendizaje 0.5, 0.1, 0.01, 0.001, 0.0001 e iteraciones de 2.5k, 3k, 3.5k, 5k y 10k respectivamente para cada regresor. De esta forma los autores obtuvieron los siguientes resultados mostrados en la Tabla 13:

Tabla 13: Resumen de Resultados (Nayan et al., 2022)

Coeficiente de determinación de Regresores						
	MLP		SVR		Regresión Lineal	
<i>Regresores</i>	<i>Infectados</i>	<i>Fallecidos</i>	<i>Infectados</i>	<i>Fallecidos</i>	<i>Infectados</i>	<i>Fallecidos</i>
1	0.8669	0.8812	0.8230	0.8322	0.7996	0.8090
2	0.8758	0.8976	0.8413	0.8633	0.6954	0.7796
3	0.9182	0.9341	0.8373	0.8701	0.6405	0.6819
4	0.8772	0.8571	0.7322	0.7798	0.7363	0.7430
5	0.7632	0.7581	0.6909	0.7154	0.7073	0.6448

Los autores concluyen que el resultado de su propuesta es válido tomando en cuenta el estado del arte y que los modelos pueden mejorarse, posteriormente se buscaría considerar también en los modelos el impacto de las vacunas.

(Torrealba-Rodriguez et al., 2020) proponen 3 modelos para predecir el número de casos positivos de COVID 19 en México. Estos modelos propuestos son estadísticos: Gompertz y Logístico; y computacionales a través de redes neuronales artificiales. Los resultados que los autores obtuvieron se muestran en la Tabla 14:

Tabla 14: Resumen de Resultados (Torrealba-Rodriguez et al., 2020)

Modelos de Regresión	
<i>Modelos</i>	<i>R2</i>
Gompertz	0.9998
Logístico	0.9996
ANN	0.9999

Los autores concluyen que los modelos propuestos poseen muy buenos resultados, sin embargo, indican que se debe considerar que los resultados pueden no ser muy correctos cuando en la fecha que se intenta predecir hay algún evento o sucede algún acontecimiento que influya con el índice de contagiados.

En resumen:

- En las propuestas de (Torrealba-Rodriguez et al., 2020; Nayan et al., 2022; Luo et al., 2021; Sun, 2021; Wiczorek, Siłka, & Woźniak, 2020 ; Shawaqfah & Almomani, 2021; Mayta et al., 2021) se puede observar siempre por lo menos una propuesta de modelo de Red Neuronal artificial multicapa. Sin embargo, algunos de ellos como (Luo et al., 2021; Sun, 2021; Dhamodharavadhani et al., 2020) proponen un modelo de red neuronal recurrente, específicamente el modelo LSTM, mostrando resultados bastante buenos como son un error de 2.37% en el modelo propuesto por (Luo et al., 2021) y un $R^2 = 0.97$ por parte de la propuesta de (Sun, 2021).

- El modelo de red neuronal propuesto por (Torrealba-Rodriguez et al., 2020; Nayan et al., 2022; Luo et al., 2021; Sun, 2021; Wiecezorek, Siłka, & Woźniak, 2020 ; Shawaqfah & Almomani, 2021; Mayta et al., 2021) muestran resultados que se ajustan bastante bien al comportamiento de la infección en las partes del mundo donde están contextualizadas estas investigaciones. El rango de valor del coeficiente de determinación en las investigaciones mencionadas se encuentra entre 0.7 a 0.999.
- Se puede observar que los modelos propuestos por (Torrealba-Rodriguez et al., 2020; Nayan et al., 2022; Sun, 2021; Wiecezorek, Siłka, & Woźniak, 2020 ; Shawaqfah & Almomani, 2021) son validados a través de su coeficiente de determinación o R^2 . Por otro lado (Luo et al., 2021; Mayta et al., 2021) lo hacen en base al RMSE.

CAPÍTULO 4: METODOLOGÍA

En este capítulo se describe la metodología utilizada en esta investigación para poder predecir el índice de infección por COVID 19 en el Perú y mantener la precisión a través de un flujo de reentrenamiento automatizado. Este procedimiento es iniciado por una elección de la arquitectura de red neuronal óptima para este tipo de problemas. Por ello primero se explicará la discusión entre las propuestas y luego se presentará la metodología utilizada en esta investigación.

4.1. Determinación de la mejor arquitectura de red neuronal para la predicción de infección por COVID 19 en el Perú

Entre las principales propuestas de modelos de regresión revisadas se encuentran:

(Wieczorek et al., 2020) quienes propusieron un modelo de regresión de Multi Layer Perceptron (MLP) para predecir la cantidad de infectados en distintos países del mundo utilizando el dataset provisto por la universidad de Jhon Hopkins, justificaron el diseño de su formato de entrada comparando distintos “timesteps” o conjunto de datos de infectados de un grupo de días y obtuvieron el siguiente resultado de la tabla 15:

Tabla 15: Entrada Justificada (Wieczorek et al., 2020)

TimeSteps	Precisión
6 días	80.45%
8 días	79.58%
10 días	79.39%
12 días	79.90%
14 días	80.62%
16 días	78.75%

Seleccionando de esta manera su formato de entrada de tal manera que se deben tener los números de infectados de los 14 días anteriores a la fecha que se busca.

Además (Wieczorek et al., 2020), nos muestran otra comparación para la elección del algoritmo de optimización utilizado en el entrenamiento de su modelo de regresión mediante la siguiente tabla 16:

Tabla 16: Algoritmo de Optimización (Wieczorek et al., 2020)

Algoritmo	Precisión
Adadelta	0%
SGD	9.80%
Adagrad	40.10%
Ftri	40.10%
Adamax	87.47%
Adam	87.53%
RMSprop	87.65%
NAdam	87.73%

De esta manera se comprobó que el algoritmo “NAdam” fue el que mejores resultados proporcionó en el entrenamiento de la propuesta de (Wieczorek et al., 2020). Ellos, además, nos mencionan que una red neuronal recurrente fue probada y obtuvo un 1% a 2% peores resultados que una MLP clásica y además fue 3 veces más lenta que esta última, por esto decidieron utilizar una arquitectura de red neuronal MLP, obteniendo finalmente casi un 0.9 de precisión en las predicciones.

(Nayan et al., 2022) propone igualmente modelo de regresión para predecir la infección y el índice de fallecidos por COVID 19 en Bangladesh, para ello utiliza una red neuronal con arquitectura MLP y también utiliza el método Support Vector Regression (SVR); sin embargo,

la arquitectura MLP resulta ser la más eficiente entre las dos, debido a que se obtiene un resultado de 0.91 a diferencia del 0.83 obtenido por el modelo SVR. Sin embargo, este resultado es obtenido realizando más de mil iteraciones o epochs.

Otra propuesta es la de (Torrealba-Rodriguez et al., 2020) quienes a través de un modelo MLP buscan predecir el índice de infección por COVID 19 en México. Su propuesta logra una precisión del 0.999, la cual resultaría ser la mejor de las arquitecturas propuestas si no fuera porque esta arquitectura solo considera la fecha en la que se está buscando predecir y para el entrenamiento utiliza como variable a predecir el índice de infección acumulada hasta el momento, que si bien muestra una precisión bastante alta en base al coeficiente de determinación; el error cuadrático medio muestra un valor bastante alto debido a la cantidad de infectados totales por día, lo que nos demuestra que el rango de error es bastante amplio y las predicciones no son muy exactas.

Por otro lado, (Sun, 2021), a diferencia de los demás propone analizar 3 modelos de predicción de infección de COVID 19: dos estadísticos (Crecimiento Logístico y Regresión Polinomial) y uno de red neuronal, siendo el modelo de red neuronal uno recurrente, específicamente Long short-term Memory(LSTM); este modelo obtuvo una precisión de 0.95 y fue el más preciso de entre los propuestos por el autor.

(Luo et al., 2021) también proponen un modelo de regresión para predecir el índice de infección por COVID 19 en Estados Unidos por medio de una arquitectura de red neuronal LSTM ajustada a una entrada especial. Esta entrada refuerza la idea de dividir la entrada en 14 “timesteps” mencionada por (Wieczorek et al., 2020) y el uso de otras métricas como la media y desviación estándar de estos 14 días y un indicador de si el día que se busca predecir es día de semana o no. (Luo et al., 2021) utiliza 4 premisas para fundamentar el diseño de esta entrada que se enlistan en la siguiente tabla 17.

Tabla 17: Premisas de (Luo et al., 2021)

Nro.	Premisas
1	La infección depende de la fuerza del contagio en el momento
2	Las series temporales del índice de infección otorgan información de la fuerza del contagio en el momento
3	Se puede asumir que un infectado solo puede contagiar durante 14 días porque es el tiempo promedio hasta su aislamiento
4	El área de trabajo es un medio crítico de transmisión del virus por lo que es más probable que una infección ocurra durante un día de semana

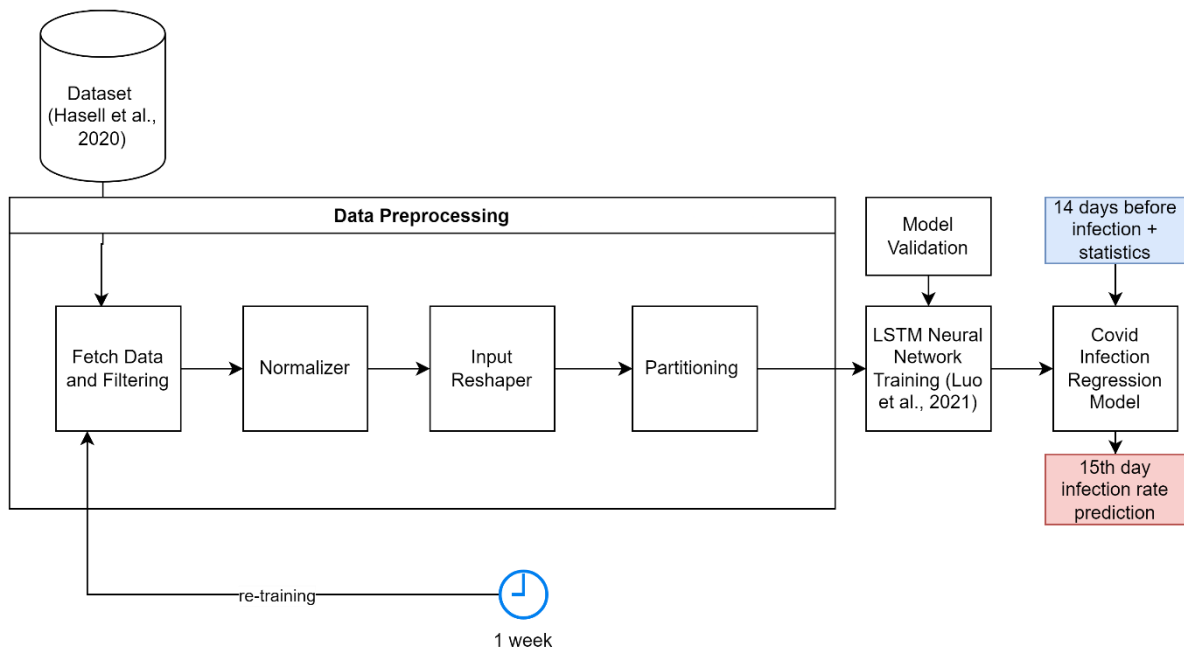
Esta propuesta además requiere poco tiempo de entrenamiento y obtiene una precisión bastante alta medida en base al error cuadrático medio que resulta 981.

En esta investigación se optó por utilizar como la arquitectura de red neuronal propuesta por (Luo et al., 2021), debido al uso de las premisas mencionadas en la tabla 17 que justifican de forma realista y lógica una entrada y , en base a esta entrada, una arquitectura precisa y de rápido entrenamiento. Esto permite que el modelo pueda considerar también, como variables de predicción implícitas, los eventos relevantes en la propagación del virus como son las medidas estatales, las festividades y actividades culturales del país. Así también, en esta investigación se seleccionó como algoritmo de optimización el algoritmo “NAdam” gracias a las comparaciones realizadas por (Wieczorek et al., 2020).

4.2. Metodología de reentrenamiento de un modelo de predicción de infección por COVID 19 en el Perú

En este punto se explicará el proceso de forma genérica de la metodología utilizada en esta investigación, esta también puede ser comprendida como la figura 6.

Figura 6: Metodología de reentrenamiento de un modelo de predicción de infección por COVID 19 en el Perú



Como podemos observar en la figura 6 el flujo comienza con la selección del mejor modelo de regresión basado en redes neuronales para poder predecir el número de infectados de un determinado día, evaluando características como robustez de fundamento, precisión y eficiencia en el entrenamiento se selecciona una propuesta de red neuronal. Luego se comienza con la extracción de los datos de la fuente de datos seleccionada para que estos sean procesados: es decir filtrados, normalizados, transformados y particionados para poder ser utilizados como datos de entrenamiento del modelo seleccionado en el primer paso; esto viene a ser el preprocesamiento de los datos. Una vez terminado el preprocesamiento se procede con el entrenamiento del modelo y su posterior validación con métodos seleccionados a partir de una revisión bibliográfica. Finalmente se obtiene un modelo de regresión que predice el índice de infección de COVID 19 en el Perú en base a la fuerza de infección del virus y se registra la fecha en la que se entrenó este modelo para luego poder ejecutar un reentrenamiento automático una vez pasada cierta cantidad de tiempo determinada.

4.2.1. Dataset de infección por COVID 19

Para el entrenamiento del modelo se utiliza el dataset propuesto por (Hasell et al., 2020) el cual contiene el índice de infectados nuevos por cada día de cada país. Este dataset cuenta además con los siguientes grupos de atributos mostrados en la Tabla 18:

Tabla 18: Grupos de atributos dataset (Hasell et al., 2020)

Metrics	Source	Updated	Countries
Vaccinations	Official data collated by the Our World in Data team	Daily	218
Tests & positivity	Official data collated by the Our World in Data team	Weekly	193
Hospital & ICU	Official data collated by the Our World in Data team	Daily	47
Confirmed cases	JHU CSSE COVID-19 Data	Daily	217
Confirmed deaths	JHU CSSE COVID-19 Data	Daily	217
Reproduction rate	Arroyo-Marioli F, Bullano F, Kucinskas S, Rondón-Moreno C	Daily	192
Policy responses	Oxford COVID-19 Government Response Tracker	Daily	187
Other variables of interest	International organizations (UN, World Bank, OECD, IHME...)	Fixed	241

Del cual el grupo que sirve como fuente de datos de entrenamiento será el grupo de “Confirmed cases” donde podemos encontrar los siguientes atributos en la Tabla 19:

Tabla 19: Atributos de casos confirmados dataset (Hasell et al., 2020)

Variable	Description
total_cases	Total confirmed cases of COVID-19. Counts can include probable cases, where reported.
new_cases	New confirmed cases of COVID-19. Counts can include probable cases, where reported. In rare cases where our source reports a negative daily change due to a data correction, we set this metric to NA.
new_cases_smoothed	New confirmed cases of COVID-19 (7-day smoothed). Counts can include probable cases, where reported.
total_cases_per_million	Total confirmed cases of COVID-19 per 1,000,000 people. Counts can include probable cases, where reported.
new_cases_per_million	New confirmed cases of COVID-19 per 1,000,000 people. Counts can include probable cases, where reported.
new_cases_smoothed_per_million	New confirmed cases of COVID-19 (7-day smoothed) per 1,000,000 people. Counts can include probable cases, where reported.

Donde podemos observar los datos los campos de “new_cases” y “new_cases_smoothed” siendo estos los más importantes para esta investigación. En este caso el campo que se utiliza como variable a predecir será “new_cases_smoothed”.

4.2.2. Obtención y Filtrado de Datos

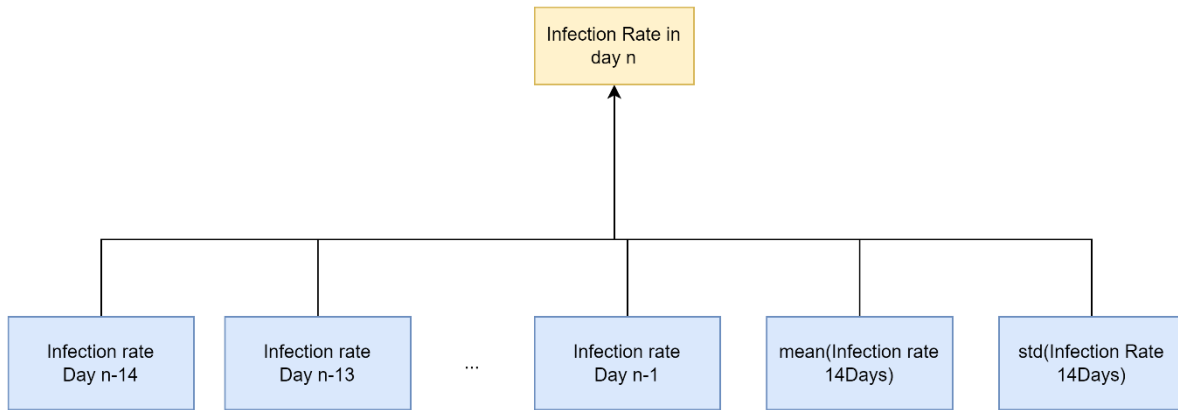
Para la obtención de datos se utiliza una url proporcionada por (Hasell et al., 2020) que contiene el acceso al csv mencionado anteriormente. Una vez obtenido el conjunto de datos, estos son filtrados por el atributo “location” con valor “Peru” recogiendo únicamente los datos de infección en el Perú. Finalmente, para aligerar y facilitar el preprocesamiento de los datos se eliminan las columnas que no son utilizadas dejando solo la columna “new_cases_smoothed” y “date” en el conjunto de datos.

4.2.3. Preprocesamiento de los datos

4.2.3.1. Diseño de entrada

Considerando las comparaciones realizadas al inicio del capítulo para determinar la arquitectura de red neuronal más óptima para predecir el índice de infección por COVID 19 en el Perú, se optó por utilizar para el entrenamiento el procedimiento de entrenamiento de red neuronal propuesto por (Luo et al., 2021) que nos indica que en base a las 4 premisas mencionadas en la tabla 17 se considera como una entrada a los infectados en los 14 días anteriores al día que se busca predecir; también se tomó como entrada a la media de los infectados en los 14 días anteriores a la fecha de predicción y finalmente la desviación estándar de los días antes mencionados para conocer el comportamiento de variación de la infección. Aunque (Luo et al., 2021) propone tomar como entrada un valor que indicará si el día es día de semana o no, en el contexto del Perú no se considera como una variable relevante para la predicción, por tanto se elimina de la entrada propuesta. Este esquema puede ser comprendido como en la Figura 7:

Figura 7: Entrada Modificada basada en (Luo et al., 2021)



4.2.3.2. Normalización

Para facilitar el entrenamiento a la red neuronal se realiza una normalización con la función MinMax para mantener los datos en un rango de 0 a 1, esto para todos los campos inicialmente principales del dataset: “new_cases_smoothed”. Luego se procede a transformar los datos para adaptarlos al input del modelo de regresión seleccionado.

4.2.3.3. Transformación de los datos de entrada

Para la obtención de los datos de entrada conforme a la entrada planteada por (Luo et al., 2021) es necesario hacer un tratamiento a los datos recopilados por la propuesta de (Hasell et al., 2020). Para este tratamiento se tiene un componente transformador encargado de dar una entrada preparada para el entrenamiento de la red neuronal. Para esto partir del dataset inicial se tomaron los campos “date” y “new_cases_smoothed” y se reorganizan como se muestra en la Figura 8 para poder generar otro dataset con la forma requerida.

Figura 8: Transformación de entrada

date	new_cases_smoothed
...	...
n-2	I(n-2)
n-1	I(n-1)
n	I(n)



Input
Reshape



date	Input[1]	Input[2]		Input[14]	Input[15]	Input[16]
15	I(1)	I(2)	...	I(14)	mean(I(1)...I(14))	std(I(1)...I(14))
...	...	...	...	...	...	...
n	I(n-14)	I(n-13)	...	I(n-1)	mean(I(n-14)...I(n-1))	std(I(n-14)...I(n-1))

Como se puede observar en la figura 8, este módulo se encarga de transformar la entrada del formato de (Hasell et al., 2020), donde n simboliza el número de día pasados desde el primer día de infección y $I(n)$ simboliza los infectados en dicho día, a un formato que acorde a la entrada de la propuesta de (Luo et al., 2021), siendo los valores de $I(n-14)$ al $I(n-1)$ los infectados de los 14 días anteriores y las funciones mean y std las funciones de media aritmética y desviación estándar respectivamente.

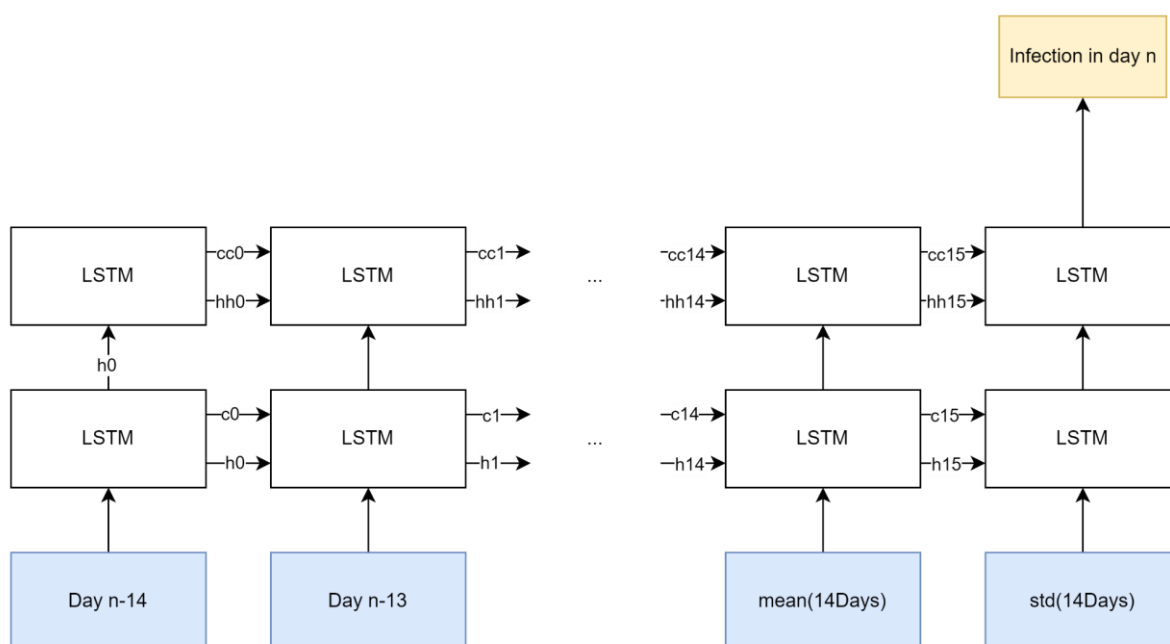
4.2.3.4. Partición del conjunto de datos

Finalmente luego de la transformación de entrada, se particiona el nuevo conjunto de datos normalizado en dos partes: una de entrenamiento que abarca el 70% del conjunto de datos y una partición de validación que abarca el 30% restante.

4.2.4. Arquitectura de la Red Neuronal

Continuando con la propuesta de (Luo et al., 2021) se dispone a utilizar la arquitectura de red neuronal recurrente mostrada en la Figura 9:

Figura 9: Arquitectura LSTM basada en (Luo et al., 2021)



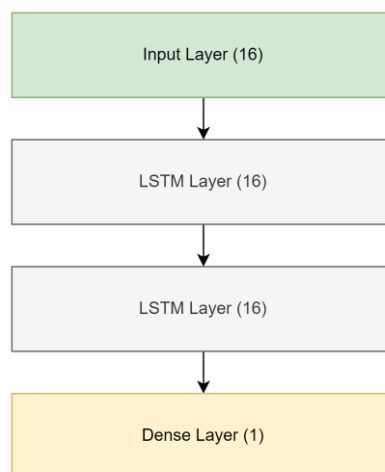
La cual comprende dos capas de nodos LSTM, cada una de las cuales contiene un nodo por cada entrada considerada anteriormente. Siendo, en resumen, dos capas de nodos LSTM con 16 nodos cada una y una salida que viene a ser el índice de infección de la fecha esperada.

El flujo de los datos de entrada es pasar por cada nodo LSTM en la primera capa LSTM donde podemos observar que el nodo que recibe el primer día no es afectado por el resultado de algún nodo anterior; sin embargo los siguientes nodos que reciben los siguientes elementos de la

entrada son afectados por los valores “c1,c2,c3, ... ,c15” y “h1,h2,h3, ... , h15” siendo estos valores los vectores de estado (c) y las salidas de los nodos (h), siendo estos últimos valores “h” el input para la siguiente capa de nodos LSTM y, al igual que en la primera capa, el primer nodo LSTM solo recibe la salida del primer nodo de la cada de entrada y los siguientes reciben la resultante del nodo anterior de la misma capa y el resultado del nodo en la misma posición que ellos pero de la capa anterior. Finalmente todo este flujo es comprimido en una salida de un nodo de una capa densa que termina siendo la predicción de infección del día “n”.

La Figura 10 nos muestra las capas consideradas para la arquitectura con sus respectivos nodos en cada una de estas capas.

Figura 10: Resumen Arquitectura



A continuación tenemos la tabla 20 con la configuración del modelo mencionando los hiper parámetros de cada capa.

Tabla 20: Parámetros de arquitectura

Capa	Hiper Parámetros
Input Layer	Nodos: 16
LSTM 1	Nodos: 16 Función de activación: tanh Función de activación recurrente: sigmoide Output: secuencia de datos (16)

LSTM 2	Nodos: 16 Función de activación: tanh Función de activación recurrente: sigmoide
Dense Layer	Nodos: 1 Función de activación: Lineal

Finalmente es utilizado “Nadam” como algoritmo de optimización tal como propone (Wieczorek et al., 2020) y la función de pérdida será en base al MSE para tratar de obtener los resultados más precisos posibles; así como también se considerarán 100 epochs para obtener un resultado de forma rápida.

4.2.5. Validación del modelo

En base a las propuestas de (Torrealba-Rodriguez et al., 2020; Nayan et al., 2022; Sun, 2021; Wieczorek, Silka, & Woźniak, 2020 ; Shawaqfah & Almomani, 2021; Cortés Martínez & Becerra Huertas, 2022; Singh & Dalmia, 2020; Nikhil et al., 2021; Pérez Abreu C. et al., 2021; Chanchí Golondrino et al., 2020; Omara & Harby, 2021; Kashif, 2021; Yu et al., 2021; Saqib, 2020; Díaz Pinzón, 2020; Luo et al., 2021; Mayta et al., 2021) para validar el modelo propuesto se utilizan como métricas: el coeficiente de determinación o R^2 y el error cuadrático medio o RMSE para determinar la precisión del modelo,

4.2.6. Modelo de regresión de infección de COVID 19

Una vez realizada la validación del modelo a través de las métricas establecidas, se tiene como resultado del proceso de entrenamiento un modelo de regresión de infección de COVID 19 en el Perú entrenado con los datos recolectados por (Hasell et al., 2020) y con una arquitectura basada en la propuesta de (Luo et al., 2021) teniendo una capa de entrada de 16 nodos, dos

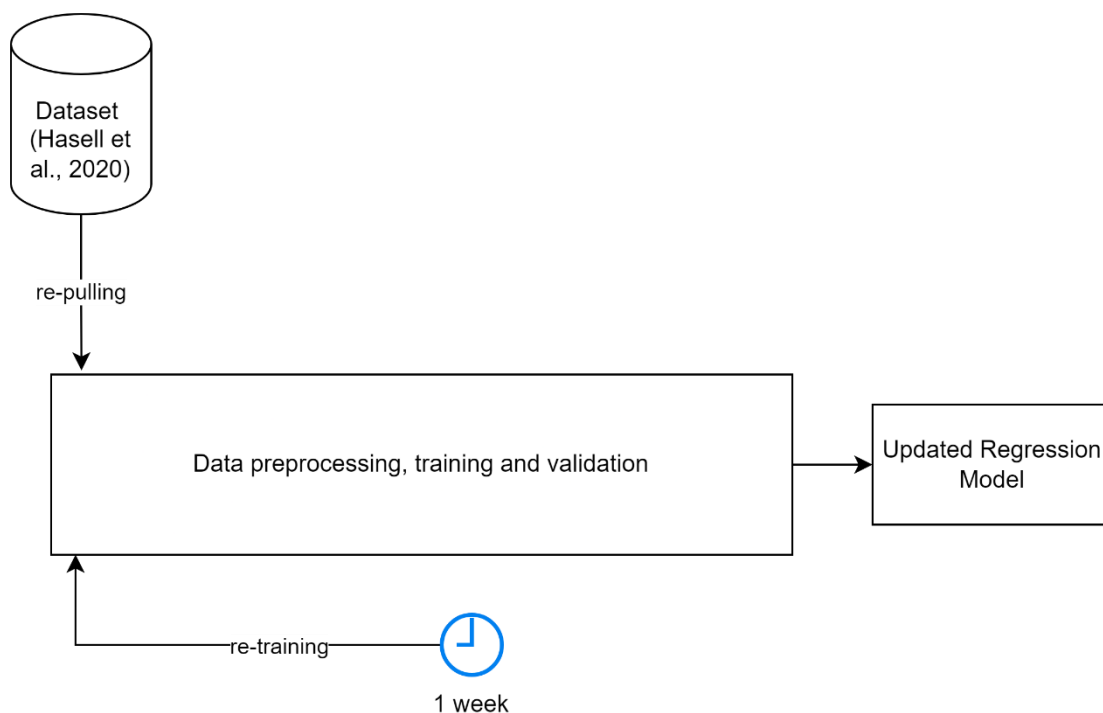
capas LSTM de 16 nodos cada capa y finalmente una capa densa de 1 solo nodo de salida como se describe con más detalle en la figura 9. Este modelo entrenado tiene como entrada 14 valores reales que representan la infección en los 14 días anteriores al día que se busca predecir, también se tiene como entrada la media y la desviación estándar de este conjunto de datos de infección de los 14 días anteriores sumando un total de 16 entradas de números reales y teniendo como salida un valor que puede ser igualmente un número real siendo esta salida el índice de infección estimado del día que se busca predecir.

Finalmente este modelo entrenado es empaquetado para su uso en sistemas externos.

4.2.7. Reentrenamiento automático

Para asegurar la conservación del modelo resultante del proceso, este será re-entrenado cada 2 semanas para de esta forma obtener mínimamente una entrada totalmente desconocida por el modelo. De forma que se realizarán todos los pasos descritos dentro de este capítulo dándonos como resultado el flujo mostrado en la Figura 11:

Figura 11: Flujo de Reentrenamiento



Es decir, el sistema volverá a realizar el preprocesamiento, esta vez con los nuevos datos, y realizar el entrenamiento del modelo nuevamente. El resultado del entrenamiento será validado con el mismo procedimiento realizado la primera vez y finalmente empaquetado para su uso en predicciones. Además, la fecha en la que se realizó este proceso se registrará para el posterior reentrenamiento 1 semana después.

CAPÍTULO 5: DISEÑO DE LA APLICACIÓN

El presente capítulo tiene como finalidad abordar los componentes referentes al proceso de diseño, desarrollo y despliegue de la aplicación de predicción de infección por COVID 19 en el Perú.

5.1. Requerimientos

5.1.1. Alcance

Desarrollo de un sistema web de predicción de infección por COVID 19 en el Perú que indica el índice de infección por COVID 19 estimado para el día posterior a la consulta.

5.1.2. Restricciones

- El sistema solo predecirá el índice de infección por COVID 19 en el Perú, la predicción de este u otro índice o algún otro país no entra en el alcance de la investigación.

5.1.3. Actores del sistema

- Personas interesadas en el sistema
- Sistema
- Entorno de ejecución de entrenamiento

5.1.4. Historias de Usuario

Para la documentación de los requisitos funcionales que debe cumplir el software se utilizaron los conceptos de Épica e Historias de Usuario para tener descripciones de las funcionalidades de sistema acordes al formato de una metodología Ágil, que en este caso se utiliza Scrum. A continuación, se puede observar las Historias de Usuario y sus detalles en la Tabla 21.

Tabla 21: Historias de Usuario

Épica		Historias de Usuario		
Código	Nombre	Código	Nombre	Descripción
EP-01	Módulo de Predicción de infección por COVID 19 en el Perú	HU-001	Predecir la cantidad de Infectados del día siguiente	Como sistema quiero predecir el índice de infección por COVID 19 en el Perú del día de mañana para poder mostrar la predicción al usuario
		HU-002	Re-entrenar modelo automáticamente	Como entorno de ejecución de entrenamiento quiero iniciar automáticamente el proceso de entrenamiento del modelo de predicción para que el modelo conozca los nuevos datos reales y mejore su ajuste a la realidad pasadas 2 semanas.
EP-02	Módulo Web de consulta de predicción de infección por COVID 19 en el Perú	HU-003	Consultar la predicción de infección del día de mañana del índice de infección por COVID 19 en el Perú	Como usuario interesado en el sistema quiero consultar la predicción del índice de infección por COVID 19 en el Perú para poder

				realizar un plan o estrategia de contramedida de la infección.
		HU-004	Consultar el índice de infección por COVID 19 en el Perú en los últimos 7 días	Como usuario interesado en el sistema quiero consultar el índice de infección por COVID 19 en el Perú en los últimos 7 días para poder conocer la situación actual de la infección.

De forma detallada se desacoplan las historias de usuario en las siguientes tareas mostradas en la Tabla 22:

Tabla 22: Tareas de Historias de Usuario

Historias de Usuario	Tareas
HU-001	● Extraer y preprocesar los datos de infección por COVID 19 en el Perú. ● Entrenar modelo de predicción de infección por COVID 19 en el Perú. ● Serializar y guardar el modelo de predicción de infección por COVID 19 en el Perú. ● Desarrollar API que utiliza el modelo para predecir el índice de infección por COVID 19 en el Perú al día siguiente del día de consulta. ● Ejecutar pruebas ● Desplegar backend
HU-002	● Desarrollar endpoint para recibir el modelo re-entrenado. ● Ejecutar pruebas.
HU-003	● Desarrollar interfaz de consulta de predicción de infección por COVID 19 en el Perú. ● Ejecutar pruebas. ● Desplegar la aplicación cliente.
HU-004	● Desarrollar interfaz de consulta de índice de infección por COVID 19 en el Perú en los ultimos 7 días. ● Ejecutar pruebas.

5.2. Product Backlog

En el Producto backlog se lista las historias de usuario y cómo estas mismas fueron agrupadas en un único sprint, dejando el sprint backlog de la siguiente manera (Ver tabla 23):

Tabla 23: Product Backlog

Identificador de la HU	Sprint	Tiempo estimado	Prioridad
HU-001	Sprint 1	3 días	Alta
HU-002		2 días	Media
HU-003		2 días	Alta
HU-004		2 días	Media

5.3. Sprints

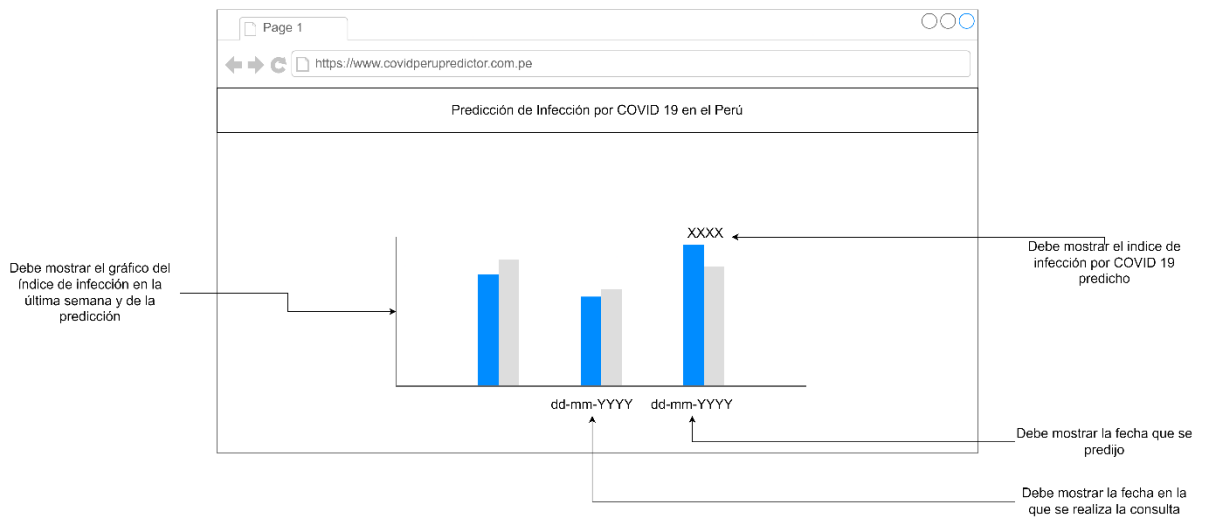
5.3.1. Sprint 0

En este sprint se define los conceptos previos al desarrollo del sistema, tales como:

5.3.1.1. Prototipo

Como primer punto, se definió el prototipo de la aplicación visual o cliente que utilizará el usuario para realizar las consultas de predicción del índice de infección por COVID 19 en el Perú como se puede observar en la Figura 12.

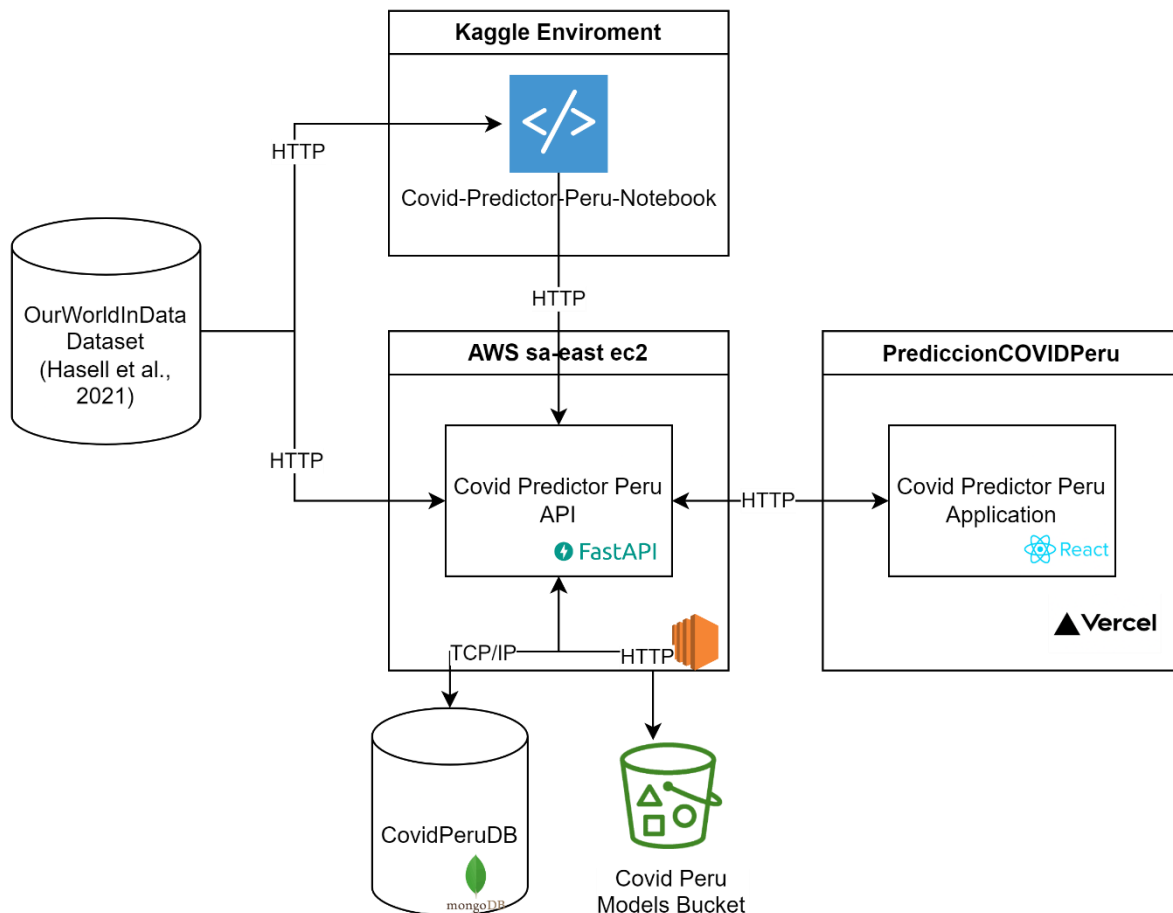
Figura 12: Prototipo y criterios de aceptación



5.3.1.2. Arquitectura del sistema

El sistema está compuesto principalmente por 5 componentes, que vienen a ser el dataset propuesto por (Hasell et al., 2020), el script de reentrenamiento del modelo de predicción de infección por COVID 19 en el Perú, la API de predicción de infección por COVID 19 en el Perú, la base de datos de infección por COVID 19 en el Perú y finalmente la aplicación web de consulta de predicción de infección por COVID 19 en el Perú como podemos observar en la siguiente Figura 13.

Figura 13: Arquitectura Propuesta



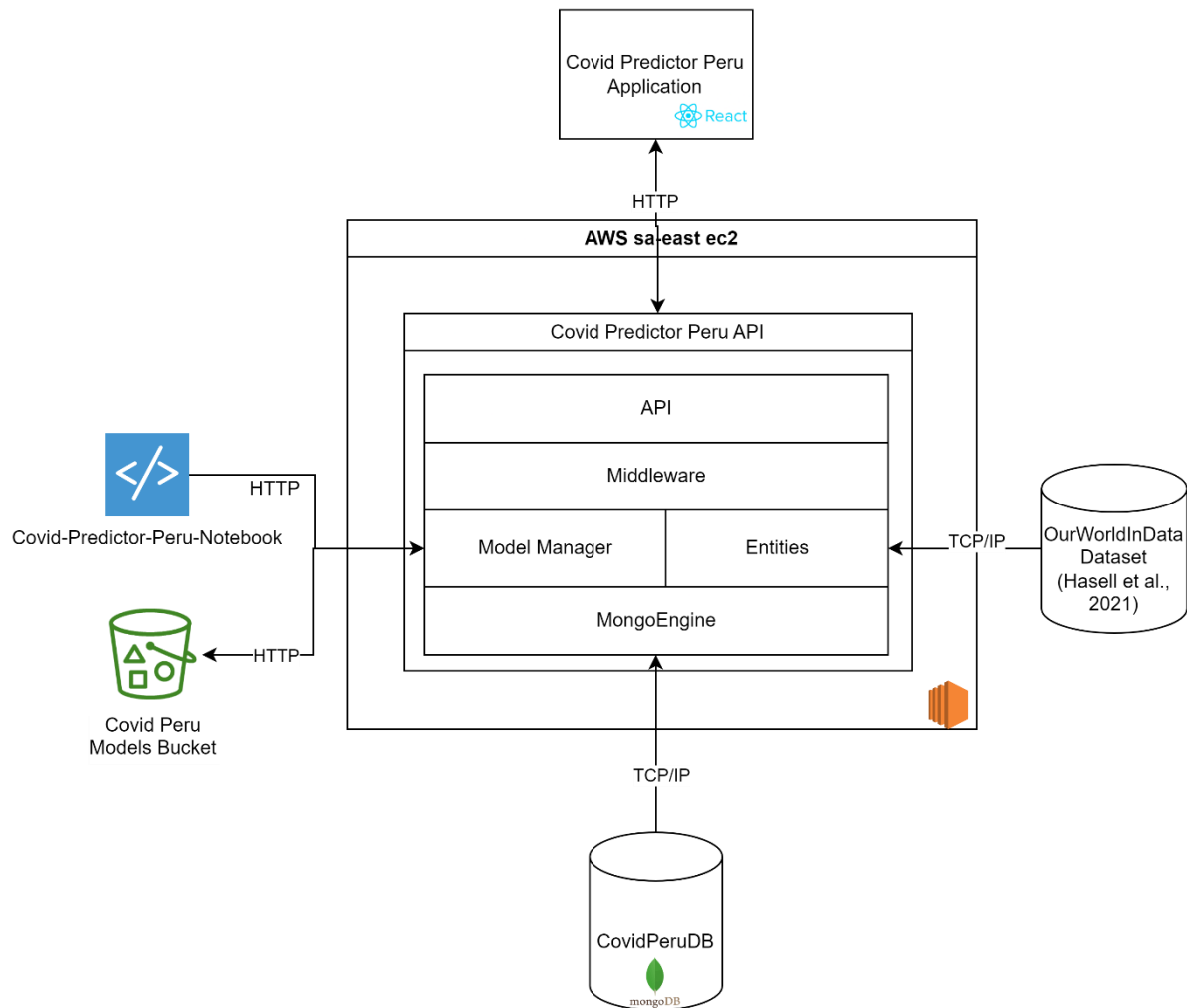
Podemos separar la arquitectura en los siguientes 5 componentes principales:

- Dataset de índice de infección por COVID 19:** Es un conjunto de datos referentes a los índices de infección por COVID 19 a lo largo del mundo. Es un proyecto de webscrapping propuesto por (Hasell et al., 2021) y recolectado por medio del siguiente enlace: <https://covid.ourworldindata.org/data/owid-covid-data.csv>. Finalmente la descripción de este conjunto de datos ya se describió en las tablas 18 y 19.
- Kaggle Environment - COVID Predictor Peru Notebook:** Es un entorno de desarrollo de notebooks, donde se hace uso de recursos computacionales en la nube. De esta manera se desarrolló el siguiente notebook para implementar, entrenar, validar y re-entrenar automáticamente los modelos de predicción de infección por COVID 19 en el Perú. El notebook desarrollado es accesible por medio del siguiente enlace: <https://www.kaggle.com/code/erickp1502/covid-ml-per/notebook>. De igual manera,

como paso final, este notebook enviará empaquetado el nuevo modelo entrenado para su uso en el módulo o API de predicción de Infección por COVID 19 en el Perú.

- **COVID Predictor Peru API:** Para el desarrollo del sistema backend de predicción de infección por COVID 19 en el Perú o API de predicción de COVID 19 en el Perú, se utilizó el framework de desarrollo web de APIs, basado en el lenguaje de programación Python, FastAPI. Framework, que además de poseer un servidor web incluido, proporciona la librería de Swagger para poder documentar y mapear los endpoints desarrollados en este proyecto. Finalmente se establece el uso del protocolo HTTP por medio de envío de datos en estructuras JSON tanto de entrada como de salida del sistema backend. Como se observa en la Figura 14 el backend posee una estructura basada en capas donde las capas inferiores son dependencias de las capas superiores para que cada una de estas cumpla su función asignada. La capa MongoEngine es el ORM encargado de mapear las entidades del sistema a estructuras de base de datos aceptadas por el gestor de base de datos MongoDB. Por encima de esta encontramos a los módulos de Model Manager; encargado de recibir, almacenar y disponer del modelo entrenado de predicción de infección por COVID 19 en el Perú para el resto de módulos el sistema; y al módulo de Entities, donde se define la estructura de las clases entidad utilizadas por el sistema, en este caso los días de infección. A continuación encontramos la capa de Middleware que contiene funciones de tratamiento de datos tanto de entrada como salida del sistema para finalmente ser serializadas y transformadas, conforme dicta el protocolo HTTP, en el formato JSON para ser enviadas o recibidas por parte de la aplicación cliente de consulta.

Figura 14: Arquitectura API



- COVID Peru DB:** La instancia de base de datos utilizada para este trabajo, está montada en el motor de base de datos Mongo DB, y desplegada en el servicio cloud gratuito de Mongo Atlas. Esta instancia es utilizada para almacenar únicamente la entidad “Día de infección” utilizada por el sistema backend de predicción de índice de infección por COVID 19 en el Perú.
- COVID Predictor Peru Application:** La aplicación cliente que mostrará la predicción del índice de infección por COVID 19 en el Perú, está construida en la librería React y es un cliente web construido con componentes de Bootstrap y desplegado en una instancia provista por Vercel.

5.3.2. Sprint 1

El Sprint 1 está compuesto por el HU-001 Predecir la cantidad de Infectados del día siguiente, HU-002 Re-entrenar modelo automáticamente, HU-003 Consultar la predicción de infección del día de mañana del índice de infección por COVID 19 en el Perú y HU-004 Consultar el índice de infección por COVID 19 en el Perú en los últimos 7 días .

5.3.2.1. Desarrollo de HU-001 Predecir la cantidad de infectados del día siguiente

Para desarrollar la primera historia de usuario se consideraron las siguientes acciones:

- **Tarea 1: Extraer y preprocesar los datos de infección por COVID 19 en el Perú**

Para esta tarea se extrajo del conjunto de datos provisto por (Hasell et al, 2020), por medio de la url <https://covid.ourworldindata.org/data/owid-covid-data.csv> y se procedió a filtrar los datos de infección en base al país del Perú. Una vez obtenido el dataset de infección por COVID 19 en el Perú, de igual manera se eliminaron todas las columnas a excepción de “date” y “new_cases_smoothed” (ver Tabla 19). Luego se procedió a reemplazar los valores nulos por los valores más cercanos, preferiblemente posteriores (Forward Filling) a los nulos o por defecto a los anteriores (Backward Filling), posteriormente se estandarizó el valor del campo “new_cases_smoothed” por medio del método MinMax. Finalmente, por cada fila o fecha con datos de infección, se formó una secuencia de datos de infección de 14 días anteriores, la media aritmética y desviación estándar de esta secuencia. Dejándonos la forma mostrada en la Figura 15:

Figura 15: Dimensiones de entrada

```
print(input_reshaped_x.shape)
print(input_reshaped_y.shape)

(990, 16, 1)
(990, 1)
```

Observación: El valor 990 es variable con el tiempo, pues es la cantidad de datos que existen en el dataset que se actualiza diariamente.

Adicionalmente, antes de pasar al entrenamiento se realiza una partición de 70% de datos para entrenamiento y 30% de datos para validación.

Todas estas acciones se realizaron por medio de las librerías de pandas y numpy.

- **Tarea 2: Entrenar el modelo de predicción de infección por COVID 19 en el Perú**

Como segunda tarea se tiene todo lo referente a la construcción del modelo de red neuronal que se encarga de predecir el índice de infección por COVID 19 en el Perú, para la definición de la arquitectura se siguió los lineamientos propuestos por (Luo et al, 2021) utilizando la librería de Keras. Es decir, se definió una clase Sequential a la que se le añadió dos capas LSTM, la primera con una capa adicional de entrada y finalmente una capa de salida de la predicción. De igual manera se definieron los hiper-parámetros y la función de activación de la siguiente manera:

Figura 16: Arquitectura red neuronal

```
import tensorflow as tf
keras=tf.keras
from keras.utils.vis_utils import plot_model

model= keras.models.Sequential()
# Capas LSTM e Input
model.add(keras.layers.LSTM(units=16,activation='tanh',recurrent_activation='sigmoid',return_sequences=True,input_shape=(16,1)))
model.add(keras.layers.LSTM(units=16,activation='tanh',recurrent_activation='sigmoid',return_sequences=False))
# Capa Output
model.add(keras.layers.Dense(n_output,activation='linear'))
model.compile(optimizer='Nadam', loss='mse', metrics=['mape'])
plot_model(model, to_file='model_plot.png', show_shapes=True, show_layer_names=True)
```

Como se puede observar en la Figura 16, los nodos de las capas LSTM son iguales a la cantidad de datos que entran en secuencia según la entrada definida anteriormente y el nodo único de la capa densa sirve como salida de la predicción.

- **Tarea 3: Serializar y guardar el modelo de predicción de infección por COVID 19 en el Perú**

Para guardar el modelo entrenado se utilizó la función save de la clase Model de la librería de Keras que nos permite guardar los datos calculados para cada nodo dentro de la red neuronal y poder reutilizarlos o cargarlos en cualquier otro programa.

- **Tarea 4: Desarrollar API que utiliza el modelo para predecir el índice de infección de infección por COVID 19 en el Perú al día siguiente de la consulta**

Para el desarrollo de la API se hizo uso del framework de desarrollo de servicios Rest, haciendo uso del lenguaje de programación Python, llamado Fast API. Esta API está encargada de recibir las consultas de cualquier aplicación cliente para responder con los datos de la infección hasta el momento y adicionalmente la predicción del día siguiente de la fecha de la consulta. Estos datos son enviados mediante el formato JSON en base al protocolo HTTP. Para esto existen dos endpoints principales que se encargan de recibir el siguiente formato y responder conforme a la siguiente Tabla 24 y Tabla 25:

Tabla 24: Endpoint obtener infección 7 días

Method	GET
Request	/infection/7days
Descripción	retorna un arreglo de objetos de infección diaria, de forma que contiene los índices de infección de la semana pasada hasta la fecha actual
Response	Ejemplo de Respuesta <pre>[{ "date": "2022-12-06", "infection_real": 30215, "infection_smoothed": 11201 }, { "date": "2022-12-07", "infection_real": 30215, "infection_smoothed": 11201 }, ... { "date": "2022-12-12",</pre>

	<pre> "infection_real": 4958, "infection_smoothed": 8414 }] </pre>
--	---

Tabla 25: Endpoint obtener infección 7 días y predicción

Method	GET
Request	/infection/7days/predict
Descripción	retorna un arreglo de objetos de infección diaria aumentando una fecha predecida, de forma que contiene los índices de infección de la semana pasada hasta la fecha actual
Response	Ejemplo de Respuesta <pre> [{ "date": "2022-12-06", "infection_real": 30215, "infection_smoothed": 11201 }, { "date": "2022-12-07", "infection_real": 30215, "infection_smoothed": 11201 }, ... { "date": "2022-12-12", "infection_real": 4958, "infection_smoothed": 8414 }, { "date": "2022-12-13", "infection_real": 0, "infection_smoothed": 8843 }] </pre>

- **Tarea 5: Ejecutar Pruebas**

Para comprobar el correcto funcionamiento de las características de la aplicación se realizaron las siguientes pruebas (ver Tabla 26) :

Tabla 26: Pruebas API de predicción de infección por COVID 19 en el Perú

Caso de Prueba	Efecto deseado	Estado	Observación
Obtener el índice de infección por COVID 19 en el Perú en los últimos 7 días	Se obtiene un arreglo de objetos días de infección con el índice de infección suavizado en los 7 días anteriores	OK	Ninguna
Actualizar la base de datos	La base de datos se actualiza con los datos de infección hasta el día de ejecución de la actualización	OK	Ninguna
Obtener predicción	Se calcula la predicción	OK	Ninguna
Obtener el índice de infección por COVID 19 en el Perú en los últimos 7 días y predicción	Se obtiene un arreglo de objetos días de infección con el índice de infección suavizado en los 7 días anteriores y la predicción agrupada en el día siguiente	OK	Ninguna

- **Tarea 6: Desplegar Backend**

Para realizar el despliegue de la API de predicción de infección por COVID 19 en el Perú, se hizo uso del recurso computacional EC2 Windows provisto por Amazon por medio de su plataforma AWS. En este recurso que nos provee de un servidor Windows como infraestructura se instalaron las dependencias de Python 3.9, Tensor Flow, Fast API, Uvicorn como servidor web y diversas librerías que permiten la ejecución de la aplicación.

5.3.2.2. Desarrollo de HU-002 Re-entrenar modelo automáticamente

- **Tarea 1: Desarrollar el endpoint para recibir el modelo re-entrenado**

Se desarrolló un endpoint que recibe un modelo re-entrenado y lo guarda en una plataforma cloud de almacenamiento de archivos estáticos, en este caso, la plataforma es S3 Buckets de

Amazon. De forma que se tiene un endpoint que sigue el siguiente formato mostrado en la Tabla 27:

Tabla 27: Endpoint actualización de modelo

Method	POST
Request	/model/handle/new
Descripción	recibe el modelo como cuerpo del mensaje form-data que es enviado por el script de entrenamiento y lo almacena en el bucket S3
Body	file:binary
Response	Ejemplo de Respuesta <pre>{ "message": "model uploaded" }</pre>

- **Tarea 2: Ejecutar Pruebas**

Para comprobar el correcto funcionamiento de las características de la aplicación se realizaron las siguientes pruebas (ver Tabla 28) :

Tabla 28: Pruebas Endpoint carga de modelo API de predicción de infección por COVID 19 en el Perú

Caso de Prueba	Efecto deseado	Estado	Observación
Recibir y almacenar nuevo modelo	El modelo es almacenado en el bucket S3	OK	Ninguna
Utilizar modelo remoto	El modelo es descargado y utilizado para realizar predicciones	OK	Ninguna

5.3.2.3. Desarrollo de HU-003 Consultar la predicción de infección del día de mañana del índice de infección por COVID 19 en el Perú

- **Tarea 1: Desarrollar interfaz de consulta de predicción de infección por COVID 19 en el Perú**

Para el desarrollo de la interfaz de usuario de la aplicación cliente de consulta de predicción de infección por COVID 19 en el Perú, se hizo uso de la librería de desarrollo web basado en componentes React. Utilizando, de igual manera, librerías como Axios para el envío de peticiones HTTP al servidor back-end y Google-Charts para la formación de gráficos del índice de infección por COVID 19 en el Perú antes y después de la predicción. En el caso de la aplicación cliente, las funcionalidad que se trabajó en esta tarea es la de presentar un gráfico del índice de infección por COVID 19 en el Perú en la última semana más un día predicho.

- **Tarea 2: Ejecutar Pruebas**

Para comprobar el correcto funcionamiento de las características de la aplicación se realizaron las siguientes pruebas (ver Tabla 29) :

Tabla 29: Pruebas aplicación de consulta de predicción de infección por COVID 19 en el Perú

Caso de Prueba	Efecto deseado	Estado	Observación
Mostrar predicción de infección al día siguiente	La página muestra el gráfico de la infección en los 7 últimos días y un día predicho adicional	OK	Ninguna

- **Tarea 3: Desplegar aplicación cliente**

Para el despliegue de la aplicación cliente se hizo uso de la plataforma Netlify, que permitió enlazar el repositorio de GitHub en el que se trabajaba el proyecto y de esta forma se importó el proyecto del repositorio y se configuró de forma que Netlify construya y despliegue la aplicación en la rama “master” automáticamente.

5.3.2.4. Desarrollo de HU-004 Consultar el índice de infección por COVID 19 en el Perú en los últimos 7 días

- **Tarea 1: Desarrollar interfaz de consulta de índice de infección por COVID 19 en el Perú en los últimos 7 días.**

Para el desarrollo de la interfaz de usuario de la aplicación cliente de índice de infección por COVID 19 en el Perú en los últimos 7 días, se hizo uso de la librería de desarrollo web basado en componentes React. Utilizando, de igual manera, librerías como Axios para el envío de peticiones HTTP al servidor back-end y Google-Charts para la formación de gráficos del índice de infección por COVID 19 en el Perú en los últimos 7 días. En el caso de la aplicación cliente, la funcionalidad que se trabajó en esta tarea es la de presentar un gráfico del índice de infección por COVID 19 en el Perú en la última semana.

- **Tarea 2: Ejecutar Pruebas**

Para comprobar el correcto funcionamiento de las características de la aplicación se realizaron las siguientes pruebas (ver Tabla 30) :

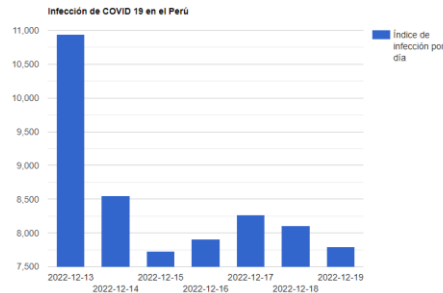
Tabla 30: Pruebas aplicación de consulta de infección por COVID 19 en el Perú

Caso de Prueba	Efecto deseado	Estado	Observación
Mostrar gráfico de infección en la semana	La página muestra el gráfico de la infección en los 7 últimos días	OK	Ninguna

5.4. El sistema de predicción de infección por COVID 19 en el Perú

Al finalizar los sprints tenemos el sistema que predice el índice de infección por COVID 19 en el Perú, como se puede observar en la siguiente figura:

Figura 17: Sistema de Predicción de infección por COVID 19 en el Perú



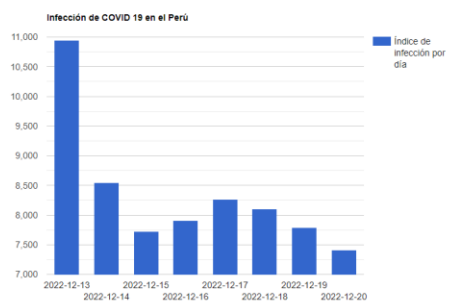
PREDECIR EL DÍA SIGUIENTE

Copyright ©2022. Erick Palomino. All Rights Reserved

Ahora se pasa a explicar el funcionamiento de la aplicación.

1. Primero al ingresar a la página se muestra el índice de infección suavizado en la última semana recolectado del dataset de (Hasell et al.,2020) como se puede observar en la Figura 17.
2. Luego hacemos clic al botón de predecir para realizar el cálculo de la predicción del índice de infección por COVID 19 en el Perú del día siguiente y se mostrará en pantalla la predicción gráficamente (ver Figura 18).

Figura 18: Sistema de Predicción de infección por COVID 19 en el Perú paso 2



PREDECIR EL DÍA SIGUIENTE

Copyright ©2022, Erick Palomino. All Rights Reserved

CAPÍTULO 6: RESULTADOS Y DISCUSIÓN

6.1. Características del conjunto de datos

Este estudio se realizó utilizando el conjunto de datos provisto por (Hasell et al., 2020) mediante la dirección url del csv que es actualizado diariamente: <https://covid.ourworldindata.org/data/owid-covid-data.csv> y posee la estructura explicada en las tablas 18 y 19. Donde se utilizó, únicamente, para el entrenamiento el campo de “new_cases_smoothed” con la respectiva fecha referida al valor. De igual manera para la evaluación se utilizó dicho campo, puesto que se hizo una división de datos de entrenamiento y datos de validación basada en la distribución hecha por (Luo et al., 2022; Torrealba-Rodriguez et al., 2020) quienes aplican la práctica estándar de realizar la partición de datos en referencia a la regla de Pareto tomando como proporciones referenciales la relación 80/20. Esta distribución puede verse de forma más clara en la Tabla 31.

Tabla 31: Distribución de datos de entrenamiento y prueba

Componente	Datos de entrenamiento	Datos de validación
Predicción de Infección por COVID 19 en el Perú	$N \cdot 0.70$	$N \cdot 0.30$

Siendo “N” la cantidad total de filas de datos de infección del Perú hasta el momento del entrenamiento

6.2. Resultados del modelo de predicción por índice de infección por COVID 19 en el Perú

Para medir la capacidad de mantener la precisión y la constante actualización de los modelos, se desarrolló un flujo a partir del modelo de la metodología. Es decir se realizó el procedimiento

de producción de un modelo de predicción del índice de infección por COVID 19 en el Perú. Procedimiento en el que se aplicó el enfoque de reentrenamiento y los resultados de estos fueron recopilados a lo largo de 2 periodos de 8 semanas. De esta manera se podría evidenciar la conservación o mejora de la precisión a lo largo del tiempo con entrenamientos constantes y el aumento de datos. Para medir que tanto se ajustan las predicciones del modelo a los datos reales que se utilizó la métrica del coeficiente de determinación o R^2 , para medir la magnitud promedio del error entre los valores predichos y los reales se utilizó la métrica de la raíz del error cuadrático medio o RMSE, y para calcular el porcentaje de error promedio entre los valores predichos y los valores reales se utilizó la métrica del error porcentual absoluto medio o MAPE.

6.2.1. Primer Periodo (05/10/22 - 23/11/22)

Este modelo tuvo como primer entrenamiento el día 05/10/22, siendo re-entrenado cada semana hasta la fecha del 23/11/22 teniendo como resultados las métricas R^2 , RMSE y MAPE de cada entrenamiento. En promedio se obtuvo un coeficiente de determinación o R^2 de 0.988, una raíz del error cuadrático medio o RMSE de 467.672 y un error porcentual absoluto medio o MAPE de 0.214 al predecir el índice de infección por COVID 19 en el Perú. La tabla 32 nos muestra las métricas de cada entrenamiento.

Tabla 32: Resultados en 2 meses Perú primer periodo

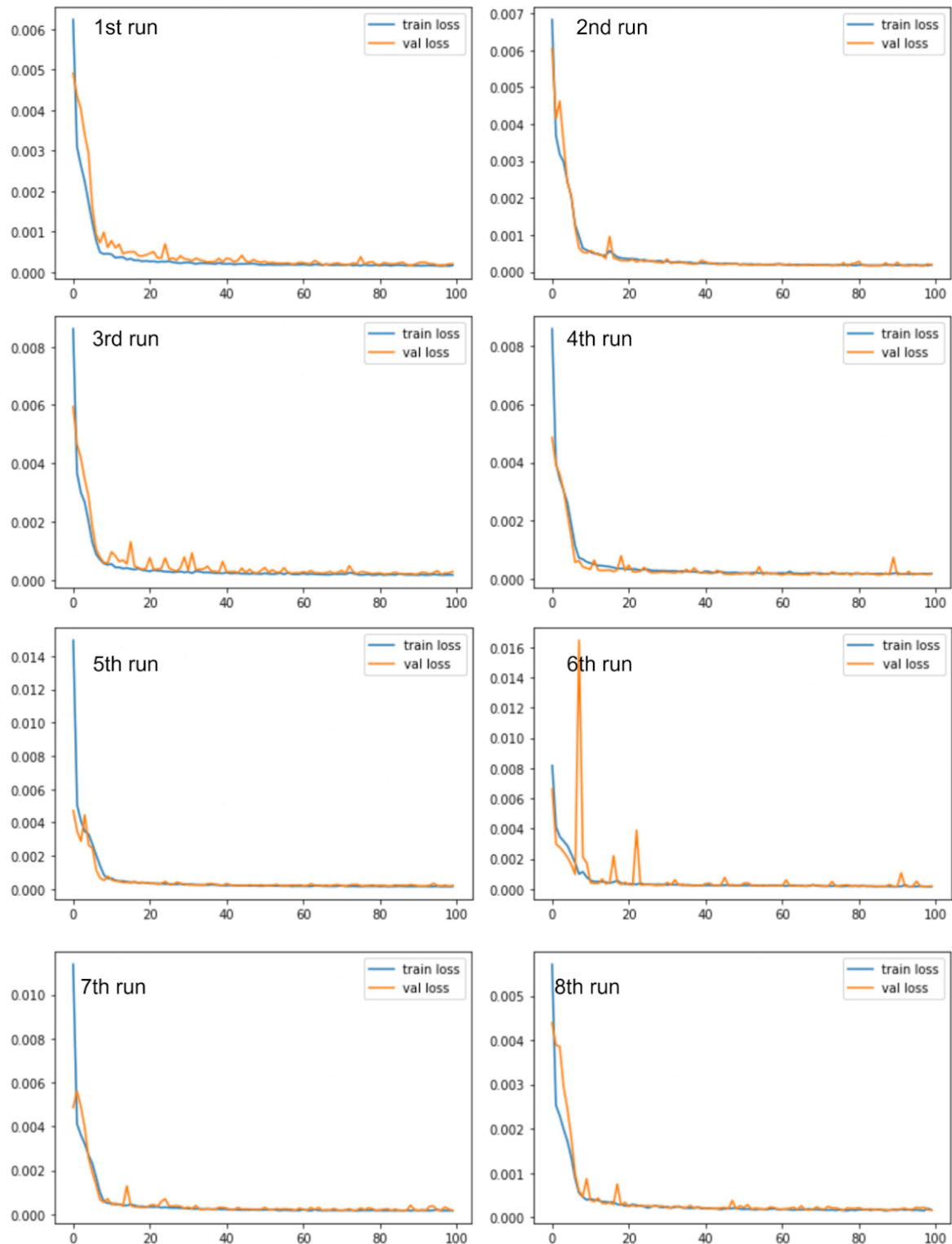
Semana	Nro. de datos	R2	RMSE	MAPE
1	943	0.987	468.143	0.189
2	950	0.992	404.548	0.139
3	957	0.986	606.971	0.314
4	964	0.991	466.169	0.293
5	971	0.985	488.727	0.169

6	978	0.986	498.156	0.280
7	985	0.988	416.817	0.161
8	992	0.991	391.848	0.168
Promedio	-	0.988	467.672	0.214

En la tabla 32, podemos ver como los valores tanto de coeficiente de determinación (R^2) como la raíz del error cuadrático medio (RMSE) y el error porcentual absoluto medio (MAPE) mantienen valores casi constantes y los cambios, tanto mejoras como pérdidas de precisión, pueden ser explicados por la aleatoriedad de los valores utilizados por los nodos para realizar el entrenamiento del modelo. En lo que respecta al periodo, se cumple el objetivo de mantener la precisión durante los re-entrenamientos y el aumento constante de datos.

La figura 19 muestra con más detalle el progreso de la métrica MSE durante el entrenamiento tanto para los valores de entrenamiento como validación en todos los flujos de entrenamiento.

Figura 19: Función de pérdida de entrenamientos primer periodo



Donde podemos observar que en todos los entrenamientos se logra minimizar la función de pérdida al final del entrenamiento. Sin embargo, en el entrenamiento de la 6ta semana se puede observar la mayor subida en la función de pérdida en todos los demás entrenamientos, siendo

este el único caso donde algunos valores aleatorios tomados por los nodos hayan ocasionado esos picos; pero que, finalmente, se acoplaron a los valores mínimos de los demás entrenamientos demostrando la eficacia de la iteración del método.

6.2.2. Segundo Periodo (29/07/23 - 13/09/23)

En este periodo el índice de infección por COVID 19 llegó a valores casi nulos y nulos en algunos casos. En estos casos, el denominador de cálculo del error porcentual absoluto medio o MAPE se aproxima a 0 haciendo que esta métrica genere resultados indefinidos. Por lo que esta métrica MAPE no es considerada como una métrica que refleje de manera precisa el desempeño de los modelos de este periodo.

Este modelo tuvo como primer entrenamiento el día 29/07/23, siendo re-entrenado cada semana hasta la fecha del 13/09/23 teniendo como resultados las métricas R^2 y RMSE de cada entrenamiento. En promedio se obtuvo un coeficiente de determinación o R^2 de 0.994 y una raíz del error cuadrático medio o RMSE de 260.944 al predecir el índice de infección por COVID 19 en el Perú.

La tabla 33 nos muestra las métricas de cada entrenamiento.

Tabla 33: Resultados en 2 meses Perú segundo periodo

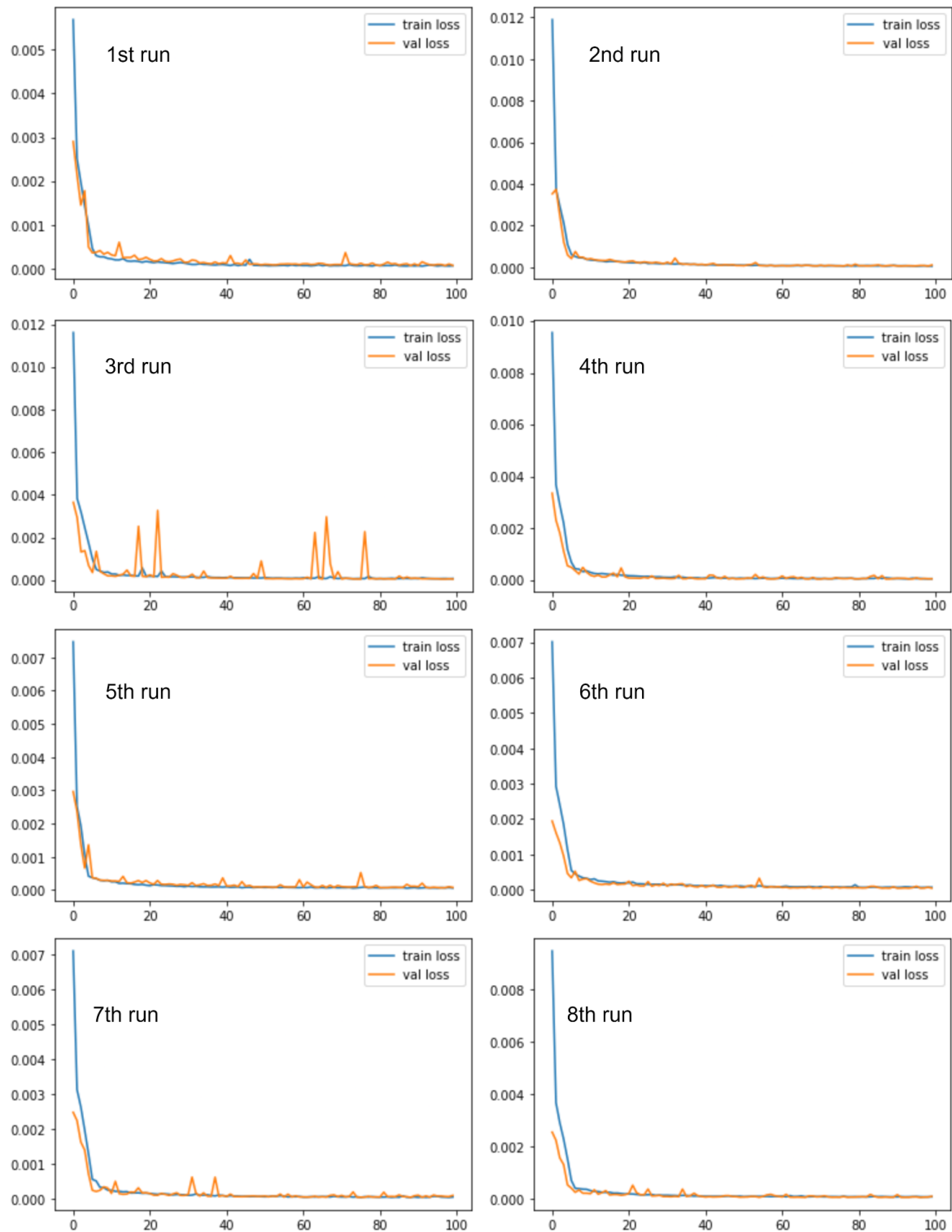
Semana	Nro. de datos	R2	RMSE
1	1304	0.9949	253.758
2	1311	0.9908	370.255
3	1319	0.9957	204.717
4	1324	0.9951	226.319
5	1329	0.9954	230.421
6	1336	0.9958	221.826
7	1346	0.9929	341.552

8	1350	0.9927	238.702
Promedio	-	0.9942	260.944

En la tabla 33, podemos ver como los valores tanto de coeficiente de determinación (R^2) como la raíz del error cuadrático medio (RMSE), al igual que en el primer periodo, mantienen valores casi constantes y los cambios, tanto mejoras como pérdidas de precisión, pueden ser explicados de igual manera que en el primer periodo, por la aleatoriedad de los valores utilizados por los nodos para realizar el entrenamiento del modelo. De igual manera que en el primer periodo, se cumple con el objetivo de mantener la precisión a lo largo del periodo durante los re-entrenamientos y el aumento constante de datos.

La figura 20 muestra con más detalle el progreso de la métrica MSE durante el entrenamiento tanto para los valores de entrenamiento como validación en todos los flujos de entrenamiento.

Figura 20: Función de pérdida de entrenamientos segundo periodo



Donde podemos observar que en todos los entrenamientos se logra minimizar la función de pérdida al final del entrenamiento. Sin embargo, en el entrenamiento de la 3ra semana se puede observar la mayor subida en la función de pérdida explicada por la aleatoriedad de los valores

tomados por los nodos, siendo el mismo caso que en el primer periodo. De igual manera la función de pérdida se acopla a los valores mínimos de los demás entrenamientos demostrando la eficacia de la iteración del método.

6.2.3. Comparación entre periodos

Tabla 34: Comparación entre periodos

	Periodo 1	Periodo 2
Fechas	05/10/22 - 23/11/22	29/07/23 - 13/09/23
Nro. de datos de ultima fecha	992	1350
R^2 promedio	0.988	0.9942
RMSE promedio	467.672	260.944

Observando la Tabla 34 y comparando el segundo periodo con el primer periodo, podemos ver una mejora considerable tanto en el coeficiente de determinación como en la raíz del error cuadrático medio. Esto es debido al aumento de aproximadamente 350 datos nuevos que hubo por el paso del tiempo entre ambos periodos. De esta manera se demuestra que la arquitectura y método propuesto no solo es capaz de mantener la precisión a lo largo del tiempo; ya que es capaz, además, de mejorar la precisión gracias al conocimiento que va a adquiriendo con el paso del tiempo gracias a los nuevos datos que van ingresando a las iteraciones de entrenamientos.

6.3. Discusión

Este trabajo sirve como evidencia de la capacidad del enfoque de reentrenamiento propuesto para mantener la precisión de un modelo de predicción, en este caso, de infección por COVID 19 en el Perú. Sin embargo, para el funcionamiento de un enfoque como este, es necesario de una fuente de datos de fácil acceso y constantemente actualizada como la propuesta de (Hasell

et al., 2020). Aún así, la constante actualización del “conocimiento” de los modelos producidos por este enfoque y la gran precisión obtenida por la arquitectura de la red neuronal hacen que estos modelos puedan ser usados como herramientas de planificación protocolar estatal o particular por un largo periodo de tiempo y no en periodos estáticos como los modelos propuestos por (Luo et al., 2021; Mayta et al., 2021; Nayan et al., 2022; Torrealba-Rodriguez et al., 2020; Shawaqfah & Almomani, 2021; Sun, 2021; Wiczorek, Siłka, & Woźniak, 2020). Los resultados muestran que hay una mejora de la precisión en base al coeficiente de determinación al obtener en el Perú un 0.994. (Sun, 2021) quien, con una arquitectura similar aplicada en Estados Unidos y Wuhan, obtiene un 0.97, y con respecto al RMSE para Perú se obtuvo en promedio 260.944 un valor bastante bajo en comparación de (Luo et al., 2021) que, siendo la base de la arquitectura propuesta en este trabajo, obtuvo valores más altos como 900 en un país con mayor cantidad de habitantes como es Estados Unidos. Aunque, esto último se debe a que la fecha en la que se realizó dicha investigación aún no habían sucedido las mayores olas de infectados en dicho país y por lo tanto el entrenamiento se realizó con valores de infección mucho menores a los considerados en esta investigación.

(Barria-Sandoval et al., 2021) considerando fechas con altos índices de infección pasadas las primeras olas, obtiene como mejor resultado para predecir la infección por COVID 19 en Chile un RMSE de 635 siendo este valor menos preciso que el 260.944 en promedio obtenido por los modelos propuestos en esta investigación. Tomando en cuenta que Chile posee un mejor control de la infección por sus medidas sanitarias, los modelos producidos con el enfoque de esta propuesta resultan más precisos que el de los autores anteriormente mencionados.

(Šegota et al., 2021) con un trabajo bastante similar al presente, desarrollaron una propuesta de predicción de curva epidemiológica, con un enfoque de aprendizaje incremental y reentrenamiento automático, de COVID 19 en cualquier país del mundo. Como ejemplo de su

trabajo presentaron sus resultados en Croacia durante un reentrenamiento diario a lo largo de 1 mes, obteniendo un coeficiente de determinación de 0.999 en un índice de infección acumulado, y en la predicción de curva epidemiológica (que viene a ser los nuevos casos diarios) obtuvieron un coeficiente de determinación de 0.996; siendo estos resultados apenas más precisos al coeficiente de determinación de 0.994 obtenido por esta propuesta. Sin embargo, esto se debe a que el país de Croacia es la décima parte de la población peruana, esto disminuye los rangos de diferencias entre los índices de infección de las olas y los periodos “tranquilos” de la infección (principales causantes de la disminución de la precisión en este tipo de modelos).

6.4. Conclusiones

Se diseñó e implementó un sistema web de predicción del índice de infección por COVID 19 en el Perú, que además mantiene su precisión a lo largo del tiempo a través de un flujo automatizado de reentrenamiento de Machine Learning, el cual predice eficazmente y con una alta precisión los índices de infección por COVID 19 en el Perú de fechas futuras.

Conclusiones Específicas:

- **Determinar el método de machine learning que proporcione los resultados más precisos y duraderos para predecir la cantidad de infectados por COVID 19 en el Perú.**

Se determinó que el método de machine learning que proporciona los resultados más precisos y duraderos para predecir la cantidad de infectados por COVID 19 en el Perú es el de redes neuronales recurrentes LSTM en base a los resultados de (Luo et al., 2021) y (Wieczorek et al., 2020).

- **Determinar la arquitectura de red neuronal que proporcione los resultados más precisos y duraderos de predicción de contagiados por COVID 19 en el Perú:**

Se determinó que la arquitectura de red neuronal que proporciona los resultados más precisos y duraderos de predicción de infección por COVID 19 en el Perú es la

propuesta de (Luo et al, 2021); la cual fue adaptada para ser utilizada en el contexto peruano.

- **Determinar un medio eficaz de validar los modelos de predicción de infección por COVID 19 en el Perú:**

Para validar el modelo de predicción de infección por COVID 19 resultante del flujo de entrenamiento se utilizaron las métricas de coeficiente de determinación o R^2 , que nos sirve para conocer el ajuste de las predicciones del modelo con los datos reales de infección y la raíz cuadrada del error cuadrático medio o RMSE, que nos permite conocer una aproximación al promedio de la diferencia entre las predicciones y los datos reales.

- **Diseñar una arquitectura de aplicación web e inteligente para desarrollar y mantener el entrenamiento constante de sistema web de predicción:**

Se diseñó una arquitectura de aplicación web basada en cliente-servidor. Contando con una aplicación cliente web que consume los servicios de una aplicación backend REST que utiliza como infraestructura recursos en la nube para poder realizar la predicción del índice de infección por COVID 19 en el Perú.

- **Diseñar un flujo de reentrenamiento automatizado para mantener la precisión del modelo de predicción de infección por COVID 19 en base al día seleccionado en el Perú:**

Se diseñó un flujo de reentrenamiento automatizado basado en eventos de tiempo y haciendo uso de recursos constantemente actualizados como es el dataset de (Hasell et al., 2020), fundamental para la alimentación de nuevos datos para el entrenamiento del modelo de predicción de infección por COVID 19 en el Perú; así como también se hace uso del entorno de desarrollo de notebooks de Kaggle que permite el uso de recursos basándose en eventos de tiempo como el paso de los días, semanas o meses.

REFERENCIAS

- Barría-Sandoval, C., Ferreira, G., Benz-Parra, K., & López-Flores, P. (2021). Prediction of confirmed cases of and deaths caused by COVID-19 in Chile through time series techniques: A comparative study. *PLOS ONE*, 16(4), e0245414.
<https://doi.org/10.1371/journal.pone.0245414>
- Chanchí Golondrino, G. E., Campo Muñoz, W. Y., & Sierra Martinez, L. M. (2020). Aplicación de la regresión polinomial para la caracterización de la curva del COVID-19, mediante técnicas de machine learning. *Investigación e Innovación En Ingenierías*, 8(2), 87–105. <https://doi.org/10.17081/invinno.8.2.4103>
- Cortés Martínez, A. E., & Becerra Huertas, C. E. (2022). Caracterización de la tendencia del COVID-19 en Colombia con regresiones polinomiales. *Gerencia y Políticas de Salud*, 20, 1–12. <https://doi.org/10.11144/javeriana.rgps20.ctcc>
- Dhamodharavadhani, S., Rathipriya, R., & Chatterjee, J. M. (2020). COVID-19 mortality rate prediction for India using statistical neural network models. *Frontiers in Public Health*, 8. <https://doi.org/10.3389/fpubh.2020.00441>
- Díaz Pinzón, J. E. (2020). Predicción del COVID-19 a nivel mundial para el año 2021. *Revista Repertorio de Medicina y Cirugía*, 131–137.
<https://doi.org/10.31260/repertmedcir.01217372.1143>
- Dong, E., Du, H., & Gardner, L. (2020). An interactive web-based dashboard to track COVID-19 in real time. *The Lancet Infectious Diseases*, 20(5), 533–534.
[https://doi.org/10.1016/s1473-3099\(20\)30120-1](https://doi.org/10.1016/s1473-3099(20)30120-1)
- El Comercio Perú, & Raffo, F. (2022, January 18). Servicios saturados en la tercera ola: ¿cuál es la situación en las aseguradoras privadas? *El Comercio Perú*.
<https://elcomercio.pe/lima/coronavirus-peru-servicios-saturados-en-la-tercera-ola->

cual-la-situacion-en-las-aseguradoras-privadas-seguros-pacifico-seguros-rimac-eps-covid-19-noticia/

- Hasell, J., Mathieu, E., Beltekian, D., Macdonald, B., Giattino, C., Ortiz-Ospina, E., Roser, M., & Ritchie, H. (2020). A cross-country database of COVID-19 testing. *Scientific Data*, 7(1). <https://doi.org/10.1038/s41597-020-00688-8>
- Haykin, S. S. (2016). *Neural networks and learning machines*.
- Jojoa, M., & Garcia-Zapirain, B. (2020). *Forecasting COVID 19 confirmed cases using machine learning: The case of America*. MDPI AG.
<http://dx.doi.org/10.20944/preprints202009.0228.v1>
- kashif, muhammad. (2021). Estimation of Death and Recovery Rates using Covid-19 Data of Pakistan. *Eurasian Journal of Medicine and Oncology*.
<https://doi.org/10.14744/ejmo.2021.89947>
- Khan, M. A., Khan, R., Algarni, F., Kumar, I., Choudhary, A., & Srivastava, A. (2022). Performance evaluation of regression models for COVID-19: A statistical and predictive perspective. *Ain Shams Engineering Journal*, 13(2), 101574.
<https://doi.org/10.1016/j.asej.2021.08.016>
- Kim, M. H., Kim, J. H., Lee, K., & Gim, G.-Y. (2021). The prediction of COVID-19 using LSTM algorithms. *International Journal of Networked and Distributed Computing*, 9(1), 19. <https://doi.org/10.2991/ijndc.k.201218.003>
- Luo, J., Zhang, Z., Fu, Y., & Rao, F. (2021). Time series prediction of COVID-19 transmission in America using LSTM and XGBoost algorithms. *Results in Physics*, 27, 104462. <https://doi.org/10.1016/j.rinp.2021.104462>
- Mayta, C., Rosales, F., Gines, M., & Valdivia, J. C. (2021). Predicción de mortalidad a causa del Covid 19 en Perú utilizando redes neuronales artificiales. *Innovación y Software*, 2, 14–26.

- Mongia, S., Sharma, S., Natarajan, J., Kumar, M., Arora, V., Stephan, T., Shankar, A., Gupta, P., & Kachhawaha, R. (2021). Prediction of COVID-19 epidemic curve of India using supervised learning approach. *International Journal of Computer Applications in Technology*, 66(3/4), 433. <https://doi.org/10.1504/ijcat.2021.120469>
- Nadella, P., Swaminathan, A., & Subramanian, S. V. (2020). Forecasting efforts from prior epidemics and COVID-19 predictions. *European Journal of Epidemiology*, 35(8), 727–729. <https://doi.org/10.1007/s10654-020-00661-0>
- Nayan, A.-A., Kijisirikul, B., & Iwahori, Y. (2022). Coronavirus disease situation analysis and prediction using machine learning: A study on Bangladeshi population. *International Journal of Electrical and Computer Engineering (IJECE)*, 12(4), 4217. <https://doi.org/10.11591/ijece.v12i4.pp4217-4227>
- Nikhil, Saini, A., Panday, S., & Gupta, N. (2021, August 27). Polynomial based linear regression model to predict COVID-19 cases. *2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*. <http://dx.doi.org/10.1109/rteict52294.2021.9574032>
- Omara, T., & Harby, K. A. (2021). Using mathematical and statistical model to forecast the path of infection by covid-19 in the Kingdom of Saudi Arabia. *IJUM Medical Journal Malaysia*, 20(2). <https://doi.org/10.31436/imjm.v20i2.1683>
- Pérez Abreu C., R., Estrada, S., & de-la-Torre-Gutiérrez, H. (2021). A two-step polynomial and nonlinear growth approach for modeling COVID-19 cases in Mexico. *Mathematics*, 9(18), 2180. <https://doi.org/10.3390/math9182180>
- Refisch, L., Lorenz, F., Riedlinger, T., Taubenböck, H., Fischer, M., Grabenhenrich, L., Wolkewitz, M., Binder, H., & Kreutz, C. (2021). *Data-Driven prediction of COVID-19 cases in Germany for decision making*. Cold Spring Harbor Laboratory. <http://dx.doi.org/10.1101/2021.06.21.21257586>

- Santosh, K. C. (2020). COVID-19 prediction models and unexploited data. *Journal of Medical Systems*, 44(9). <https://doi.org/10.1007/s10916-020-01645-z>
- Saqib, M. (2020). *Forecasting COVID-19 outbreak progression using hybrid polynomial-bayesian ridge regression model*. Research Square Platform LLC.
<http://dx.doi.org/10.21203/rs.3.rs-75292/v1>
- Šegota, S., Lorencin, I., Anđelić, N., Štifanić, D., Musulin, J., Vlahinić, S., Šušteršič, T., Blagojević, A., & Car, Z. (2021). Automated pipeline for continual data gathering and retraining of the machine learning-based COVID-19 spread models. *EAI Endorsed Transactions on Bioengineering and Bioinformatics*, 1(3), 169582.
<https://doi.org/10.4108/eai.4-5-2021.169582>
- Shahrear, P., Rahman, S. M. S., & Nahid, M. M. H. (2021). Prediction and mathematical analysis of the outbreak of coronavirus (COVID-19) in Bangladesh. *Results in Applied Mathematics*, 10, 100145. <https://doi.org/10.1016/j.rinam.2021.100145>
- Shawaqfah, M., & Almomani, F. (2021). Forecast of the outbreak of COVID-19 using artificial neural network: Case study Qatar, Spain, and Italy. *Results in Physics*, 27, 104484. <https://doi.org/10.1016/j.rinp.2021.104484>
- Singh, M., & Dalmia, S. (2020, December 10). Prediction of number of fatalities due to Covid-19 using Machine Learning. *2020 IEEE 17th India Council International Conference (INDICON)*. <http://dx.doi.org/10.1109/indicon49873.2020.9342390>
- Sun, T. (2021). Analysis of COVID-19 based on several machine learning techniques. *Journal of Physics: Conference Series*, 1827(1), 012083.
<https://doi.org/10.1088/1742-6596/1827/1/012083>
- Torrealba-Rodriguez, O., Conde-Gutiérrez, R. A., & Hernández-Javier, A. L. (2020). Modeling and prediction of COVID-19 in Mexico applying mathematical and

computational models. *Chaos, Solitons & Fractals*, 138, 109946.

<https://doi.org/10.1016/j.chaos.2020.109946>

Wieczorek, M., Siłka, J., & Woźniak, M. (2020). Neural network powered COVID-19 spread forecasting model. *Chaos, Solitons & Fractals*, 140, 110203.

<https://doi.org/10.1016/j.chaos.2020.110203>

Yu, X., Lu, L., Shen, J., Li, J., Xiao, W., & Chen, Y. (2021). RLIM: A recursive and latent infection model for the prediction of US COVID-19 infections and turning points.

Nonlinear Dynamics, 106(2), 1397–1410. [https://doi.org/10.1007/s11071-021-06520-](https://doi.org/10.1007/s11071-021-06520-1)