Escuela Superior Politécnica del Litoral

Facultad de Ingeniería en Electricidad y Computación

Identificación de patrones de infecciones respiratorias mediante técnicas de Inteligencia Artificial

TECH-379

Proyecto Integrador

Previo la obtención del Título de:

Ingeniero en Ciencias de la Computación

Presentado por:

Carlos Danilo Gómez Tello

Guillermo Alejandro Veintimilla Altamirano

Guayaquil - Ecuador

Año: 2024 I

Dedicatoria

Carlos Gómez Tello:

El presente proyecto se lo dedico a mis padres, Martha Tello y Félix Gómez dado a su paciencia y amor a lo largo de este trayecto académico. A mis hermanas Luisa Gómez y Paola Gómez por el apoyo, consejos y compañía durante este tiempo. A mis sobrinos Efrén Chávez y Miguel Chávez que con su alegría supieron ayudarme en momentos complicados.

A la persona que considero mi mentor en esta parte académica a mi primo Ernesto Rivera, el cual me mostro el camino inicial para ser politécnico.

A mis amigas Stefany Farias, Antonella López y María Alexandra Ramos, a las cuales estuvieron en momentos difíciles y fueron parte fundamental para no rendirme en este trayecto.

Guillermo Alejandro Veintimilla Altamirano:

Dedico este trabajo principalmente a toda mi familia por el apoyo incondicional y afectuoso que me han brindado a lo largo de mis 5 años de carrera universitaria, sus palabras me han acompañado y se quedarán en mi corazón por toda mi vida. De forma especial quiero agradecer a mí papá, a mí mamá, a mí ñaña y a mí ñaño los cuales han sido el motor de todos mis esfuerzos. Todo este trabajo también quiero dedicárselo a mis abuelitos que ya no se encuentran con vida, sus enseñanzas me han servido de guía en los pasos que he tomado. Por último, quiero agradecer a mis amigos más cercanos que en estos años han estado presentes y me han apoyado con su amistad. *¡Todo lo puedo en Cristo que me fortalece!*

Agradecimientos

Nuestros agradecimientos van dirigidos al Ing. Enrique Peláez por permitirnos trabajar con él en este proyecto, además de todas las herramientas y conocimientos brindados en este tiempo como profesor y tutor del proyecto.

Agradecemos al Dr. Washington Cárdenas y a todo su equipo de trabajo por el material facilitado para la elaboración de este proyecto.

Expresamos nuestro agradecimiento a la Escuela Superior Politécnica del Litoral por los conocimientos y oportunidades otorgadas para nuestro desarrollo académico y profesional.

Declaración Expresa

Nosotros Carlos Danilo Gómez Tello y Guillermo Alejandro Veintimilla Altamirano acordamos y reconocemos que:

La titularidad de los derechos patrimoniales de autor (derechos de autor) del proyecto de graduación corresponderá al autor o autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores.

La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de nuestra innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique los autores que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, 22 de Mayo del 2024.

Carlos Danilo Gómez Tello

Guillermo Alejandro Veintimilla Altamirano

Evaluadores

Firmado electrónicamente por:
BORIS XAVIER
VINTIMILLA BURGOS

Boris Xavier Vintimilla Burgos

Profesor de Materia

Firmado electrónicamente por:
COLON ENRIQUE
PELAEZ JARRIN

Colon Enrique Peláez Jarrin

Tutor de proyecto

Resumen

El proyecto aborda la identificación y agrupación de patrones en datos clínicos de pacientes menores de edad afectados por infecciones respiratorias, causadas por virus como SARS-COV-2, Influenza A o B y VSR. El objetivo principal es mejorar la eficacia de diagnósticos y tratamientos mediante la aplicación de técnicas de Inteligencia Artificial (IA) y Aprendizaje Automático (AA). Para eso, se realizaron procesos como la extracción de datos, limpieza, selección de características y normalización de estos. Luego, se estableció un modelo basado en IA que permitió la creación de agrupaciones con los datos proporcionados, utilizando el algoritmo K-Medoids. Además, se empleó el modelo ARIMA para predecir tendencias relacionadas con síntomas, palabras clave y diagnósticos a lo largo del tiempo. Adicionalmente, se implementó un prototipo de aplicación web que facilita la visualización de las tendencias y patrones dentro de los datos agrupados, obteniendo así un software útil para los médicos. Los resultados indicaron que se logró generar predicciones e identificar patrones en las agrupaciones alcanzando un grado de confianza de 86.96%. En conclusión, el proyecto proporciona una herramienta que facilita el análisis e identificación de patrones en la sintomatología de infecciones respiratorias de pacientes gracias a la ayuda de algoritmos basados en IA.

Palabras Claves: Datos clínicos, Aprendizaje Automático, Predicciones, Agrupaciones.

Abstract

This paper aims for the identification and clustering of patterns in clinic data from minor patients affected by respiratory infections, caused by viruses such as SARS-COV-2, Influenza A or B and VSR. The principal objective is to enhance the efficacy of diagnostics and treatments through the application of Artificial Intelligence techniques and Machine Learning. For that, processes such as data extraction, cleaning, feature selection and normalization were carried out. Then, it was established an AI model which allowed the creation of clusters with the data provided, using the K-Medoids algorithm. Furthermore, the ARIMA model was used to predict tendencies related to symptoms, key words and diagnostics through time. Additionally, there was implemented a web application prototype that facilitates the visualization of tendencies and patterns inside the clustered data, thus obtaining useful software for doctors. The results showed that it was possible to generate predictions and identify patterns in the clusters reaching a confidence level of 86.96%. In conclusion, this paper provides a useful tool which facilitates the analysis and identification of patterns in the respiratory infections' symptomatology from patients with the help of AI algorithms.

Keywords: *Clinic data, Machine Learning, Predictions, Clusters.*

Índice general

Abreviaturas

ARIMA AutoRegressive Integrated Moving Average

AA Aprendizaje Automático

AP Aprendizaje Profundo

ESPOL Escuela Superior Politécnica Del Litoral

RE Expresiones Regulares

HCE Historias Clínicas Electrónicas

IA Inteligencia Artificial

IRA Infecciones Respiratorias Agudas

LIB Laboratorio Para Investigaciones Biomédicas De ESPOL

LSTM Long Short Term Memory

SVM Máquina De Vectores De Soporte

OpenCV Open Source Computer Vision Library

OCR Reconocimiento Óptico De Caracteres

PCA Principal Component Analysis

SARIMAX Seasonal AutoRegressive Integrated Moving Average With eXogenous Regressors

CT Tomografía Computarizada

VSR Virus Sincitial Respiratorio

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

Capítulo 1

1.1 Introducción

Las infecciones respiratorias son consideradas como enfermedades que afectan desde oídos, nariz y garganta hasta los pulmones (Secretaría de Salud, 2015). Generalmente, no se requiere de antibióticos para su cura y comúnmente no suelen durar más de 15 días (Secretaría de Salud, 2015). Este tipo de infecciones pueden ser causadas por virus como el SARS-COV-2, la Influenza de tipo A o B y el Virus Sincitial Respiratorio (VSR), los cuales presentan entre ellos métodos de transmisión similares, pero también diferencias en la severidad de la enfermedad, susceptibilidad poblacional y tiempo de aparición de síntomas (Centros para el Control y la Prevención de Enfermedades, Centro Nacional de Vacunación y Enfermedades Respiratorias (NCIRD), 2024).

Debido a la diversidad de síntomas y signos clínicos similares, resulta necesario buscar herramientas que permitan mejorar la eficacia del manejo y monitoreo de las infecciones respiratorias agudas (IRA).

Es en este punto donde la Inteligencia Artificial (IA) puede ser utilizada como principal herramienta para ayudar a lograr el objetivo de mejorar la eficacia del manejo y monitoreo de infecciones respiratorias agudas. La integración de la IA con historias clínicas electrónicas (HCE) mejora la toma de decisiones clínicas (MedicalHubAssist, 2024). Se ha probado en diferentes estudios anteriormente realizados, como por ejemplo en (Peng, y otros, 2021) y (Masino, y otros, 2019), que los algoritmos de IA pueden detectar patrones y tendencias en los datos de los pacientes, llegando a identificar correlaciones potenciales y prediciendo resultados de enfermedades (MedicalHubAssist, 2024). Además, técnicas basadas en Aprendizaje Automático (AA), un subcampo de la IA, también sirven de apoyo para mejorar la eficiencia y precisión de las HCE en estos estudios (MedicalHubAssist, 2024).

Por ello, este proyecto propone elaborar un prototipo basado en una aplicación web y utilizando técnicas de IA y AA que permitan identificar y agrupar los patrones contenidos en los datos de pacientes menores de edad afectados por infecciones respiratorias causadas por los virus SARS-COV-2, Influenza A o B y VSR.

1.2 Descripción del Problema

Durante la pandemia de COVID-19, se observaron epidemias estacionales como la influenza y brotes de VSR. Estas enfermedades suelen manifestarse inicialmente con síntomas respiratorios superiores y fiebre, seguidos de síntomas respiratorios inferiores (Pochet, 2020). El problema por resolver recae en el hecho de cómo mejorar la prevención, diagnóstico, tratamiento y control de enfermedades respiratorias agudas ante la existencia de la co-infección de los virus causantes de estas. Según la doctora Rosa Wong, es la co-infección de estos virus lo que llega a complicar el cuadro clínico de un paciente (Salud Pública, 2020), provocando que cualquier diagnóstico y tratamiento sea desafiante.

Esto genera una necesidad en los sistemas de salud de monitorear y manejar eficazmente las IRA. Estas enfermedades afectan el aparato respiratorio y son causadas por microorganismos como virus o bacterias (Peng, y otros, 2021). Es fundamental poder comprender los patrones de estas enfermedades para tener un mejor alcance en la prevención, diagnóstico y tratamiento.

La necesidad de un enfoque integral y multidisciplinario que ayude a la integración de una vigilancia epidemiológica, el desarrollo de pruebas de diagnóstico más precisas y accesibles, son desafíos que deben abordarse para mejorar la capacidad de los sistemas de salud, para que puedan enfrentar epidemias estacionales de influenza y los brotes de VSR (Centros para el Control y la Prevención de Enfermedades, Centro Nacional de Vacunación y Enfermedades Respiratorias (NCIRD), 2024). Esto es importante para países de limitados recursos donde ineludiblemente se ven afectados los servicios de salud pública.

Por otro lado, el Laboratorio para Investigaciones Biomédicas de ESPOL (LIB), que forma parte de la Facultad de Ciencias de la Vida, se destaca como un espacio científico en el que se investiga la biología molecular de los virus de ARN (ESPOL, 2020). En el marco de un proyecto de investigación internacional sobre enfermedades respiratorias, se están usando datos clínicos de pacientes enrolados para la investigación, la cual consta de datos sobre los patógenos a estudiar en este prototipo. Sin embargo, los valores suelen ser dispersos y variados, lo que implica que se requerirá un proceso de extracción, limpieza, depuración y normalización de datos antes de su utilización en el análisis para asegurar resultados más precisos.

1.3 Justificación del Problema

Resolver este problema es de alta relevancia por varias razones. En primer lugar, resulta importante buscar formas de mejorar la eficacia de tratamientos y diagnósticos contra las IRA debido al alto impacto que tienen en la salud pública en términos de la morbilidad y mortalidad en todo el mundo especialmente en países en desarrollo (Instituto Nacional De Salud, 2020). Además, la IA ha demostrado ser de ayuda para la identificación de patrones y tendencias en los síntomas de pacientes ya que es actualmente usada en detección y diagnóstico de infecciones respiratorias (Villafuerte, Manzano, Ayala, & García, 2023). Es por lo que, en este proyecto se plantea desarrollar un prototipo web, usando técnicas de IA y AA para analizar los datos de pacientes afectados por las infecciones respiratorias causadas por SARS-COV-2, Influenza A y B, o VSR.

1.4 Objetivos

1.4.1 Objetivo general

Analizar patrones ocultos en los datos clínicos de pacientes menores de edad sobre infecciones respiratorias usando modelos basados en IA para el mejoramiento de la eficacia de diagnósticos y tratamientos brindados.

1.4.2 Objetivos específicos

1. Extraer los datos de las historias y exámenes clínicos de cada paciente con la ayuda de herramientas basadas en IA.
2. Aplicar los procesos de limpieza, selección de características y normalización de los datos usados.
3. Establecer modelos basados en IA que permitan la creación de agrupaciones y predicciones de tendencias con los datos a usar.
4. Implementar un prototipo basado en una aplicación web que permita la visualización de los grupos de datos junto con información pertinente a partir de la historia clínica de los pacientes.

1.5 Marco teórico

1.5.1 Virus que causan infecciones respiratorias

La Influenza se transmite con facilidad entre personas a través de pequeñas partículas expulsadas con la tos o los estornudos y suele propagarse rápidamente en forma de epidemias estacionales (OPS, s.f.). En infantes la infección puede conllevar graves complicaciones de una enfermedad subyacente, provocar neumonía o causar la muerte (OPS, s.f.). Por otro lado, el SARS-COV-2 puede propagarse desde la boca o nariz de una persona infectada en pequeñas partículas cuando tose y no discrimina por parte de su mortalidad ya que, sin importar la edad del infectado, este puede contraer la infección y enfermar gravemente o morir (OPS, s.f.). Con respecto al VSR, esta causa síntomas leves similares a los de un resfriado (OPS, s.f.). Además, los bebés forman parte del grupo que tienen más posibilidades de desarrollar VSR grave y necesitar hospitalización (OPS, s.f.).

1.5.2 Uso de Historias clínicas electrónicas

Como se ha mencionado anteriormente, el presente estudio utilizó datos sobre las historias clínicas de pacientes. Estos, en primera instancia provienen de un sistema de

historias clínicas electrónicas (HCE), en el que se encuentran almacenados en archivos con formato ".pdf"; por ello, fue necesario su conversión a un formato más adecuado para el procesamiento en los modelos basados en IA, como por ejemplo ".csv". Así como mencionan Yadav, Steinbach, Kumar y Simon en "Mining Electronic Health Records(EHRs): A Survey" estas historias clínicas contienen la información de un paciente, conformada por diversos tipos de datos, de tipo demográficos, resultados de pruebas de laboratorio, diagnósticos y medicaciones (Yadav, Steinbach, Kumar, & Simon, 2018). También, indica que profundizar en este tipo de registros puede guiar a un mejoramiento en el manejo de la salud de los pacientes ya que estas historias contienen información detallada relacionada con el pronóstico de enfermedades para grandes poblaciones de pacientes (Yadav, Steinbach, Kumar, & Simon, 2018). Un sistema de HCE ha demostrado ser un sistema de apoyo en la toma de decisiones clínicas, tareas predictivas y reconocimiento de patrones (Guerra, y otros, 2020). Sin embargo, este tipo de sistemas también presentan desafíos, tales como la heterogeneidad, ruido, redundancia o inconsistencia que puede existir en sus datos y por ello un análisis exploratorio de datos siempre resulta esencial realizar primero antes del análisis usando técnicas de AA (Guerra, y otros, 2020).

1.5.3 Protección de los datos

Las historias clínicas usadas para este estudio están protegidas por un documento de confidencialidad firmado por los autores del proyecto. Durante el desarrollo de esta investigación, no se utilizó información personal de los pacientes que aceptaron ser parte de esta, por lo cual se consideran anónimos. Además, el uso de estos datos está autorizado por un convenio entre el LIB y el Hospital Roberto Gilbert.

1.5.4 Pre-procesamiento de datos

En este proyecto, el pre-procesamiento de datos se dividió en 4 etapas. La primera etapa fue la extracción de datos a partir de los archivos ".pdf" facilitados por el LIB, en esta

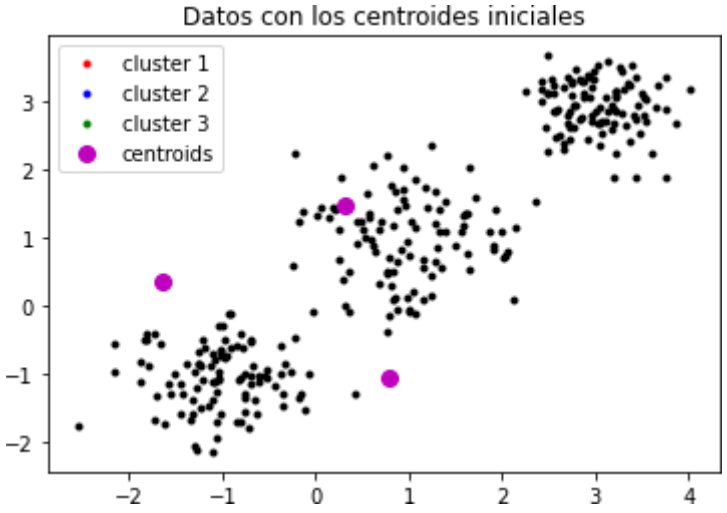
etapa se cambió el formato de los archivos, al formato necesario para procesamiento, tal como .csv. Con la extracción finalizada, se procedió a realizar la limpieza de datos, esta etapa se enfocó en identificar y sustituir aquellos datos o registros considerados como incompletos, inexactos o irrelevantes (DataScientest, 2022). Luego, se realizó el proceso de selección de características donde se eliminaron aquellas variables consideradas como no relevantes para el estudio. Por último, se realizó el proceso de normalización en los datos, una técnica utilizada para ajustar las características de los datos dentro de un rango específico y mejorar su interpretación (Blanco, 2023).

1.5.5 Algoritmos para evaluar

Dado que este proyecto busca identificar los patrones que comparten las agrupaciones de datos que se parecen o son disímiles entre sí, resulta importante mencionar los distintos algoritmos de agrupamiento basados en IA, que ayudaron a conseguir esta meta. Entre ellos están: K-Means, DBSCAN, clustering jerárquico y K-Medoids. En primer lugar, el algoritmo K-Means es un algoritmo de clasificación no supervisada cuyo objetivo es agrupar objetos en k grupos distintos basándose en sus características (Escuela de Ingeniería Informática. Universidad de Oviedo, s.f.). En la siguiente figura se puede apreciar un ejemplo de un conjunto de agrupaciones creadas por el algoritmo.

Figura 1

Ejemplo de agrupaciones creadas por el algoritmo K-Means

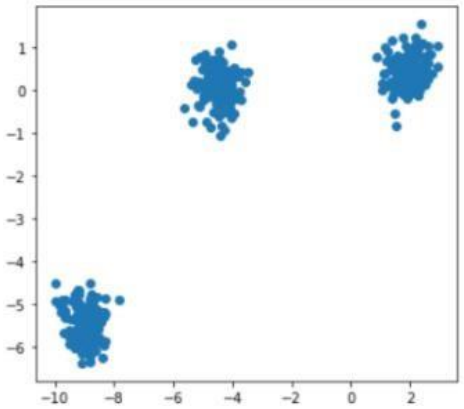


Nota. Obtenido de *(Escuela de Ingeniería Informática. Universidad de Oviedo, s.f.).*

DBSCAN, por su parte, es un algoritmo simple que define las agrupaciones mediante la estimación de la densidad local (DataScientest, s.f.). En la siguiente figura se puede observar un ejemplo de agrupaciones generadas por DBSCAN.

Figura 2

Ejemplo de agrupaciones creadas por el algoritmo DBSCAN



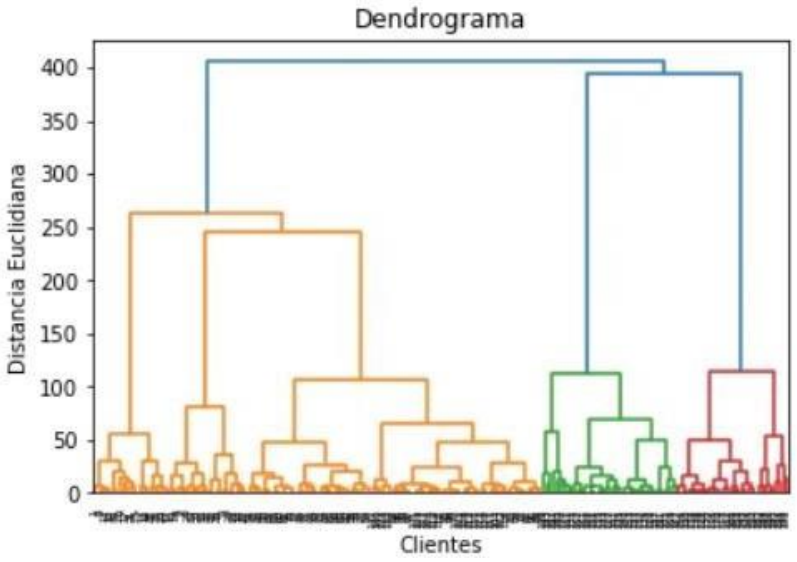
Nota. Obtenido de *(DataScientest, s.f.).*

El algoritmo de agrupamiento jerárquico es un método de AA generalmente empleado para la organización y clasificación de datos (Barrios, Medium, 2023). La jerarquía creada con este algoritmo muestra la información en forma de un Dendrograma, lo cual se compara a

una estructura de datos en árbol (Barrios, Medium, 2023), tal como se muestra en la siguiente figura.

Figura 3

Ejemplo de Dendrograma



Nota. Obtenido de *(Barrios, Medium, 2023).*

Y, el algoritmo K-Medoids es una técnica de agrupamiento utilizada para dividir un conjunto de datos en k grupos. A diferencia de K-Means, en K-Medoids cada clúster está representado por un punto de datos llamado medoide, que es el objeto más central en el clúster (Montoya, 2023). La visualización de las agrupaciones creadas por este algoritmo es similar a la generada con K-Means.

1.5.6 Estudios anteriores

Existen distintos estudios, tales como (Peng, y otros, 2021), (Masino, y otros, 2019) y (Villafuerte, Manzano, Ayala, & García, 2023), los cuales han permitido resolver problemas relacionados a la diferenciación de tumores, la detección temprana de enfermedades y diagnóstico de infecciones respiratorias en el campo de la medicina. El estudio de Peng y otros autores (2021) menciona un modelo basado en aprendizaje profundo (AP), denominado como ThyNet, que fue creado para diferenciar entre tumores malignos y nódulos tiroideos benignos y así ayudar a mejorar la calidad de diagnósticos dados por radiólogos en China. El

algoritmo de IA usado es una estructura combinada de tres tipos de redes: ResNet, ResNeXt y DenseNet que recibe como entradas un conjunto de imágenes de ultrasonido. Al final, los resultados obtenidos indican que se pudo mejorar la precisión en los diagnósticos al diferenciar los nódulos tiroideos, además de que ayuda a decrementar el número de aspiraciones innecesarias con aguja fina, un procedimiento intrusivo común al tratar con nódulos (Peng, y otros, 2021).

También existen otros casos como el presentado por Masino y otros (2019), cuyo enfoque fue desarrollar un modelo basado en AA que permita usar datos clínicos electrónicos para reconocer Sepsis infantil al menos 4 horas antes del reconocimiento clínico. Para este caso, se entrenaron 8 modelos basados en ML entre los cuales se encontraban AdaBoost, Gradient boosting, K-vecinos más cercanos, Regresión logística, Random forest, etc. Los resultados permitieron concluir que efectivamente estos modelos logran identificar Sepsis en infantes antes del reconocimiento clínico e incluso se menciona que, para mejorar aún más el rendimiento de los modelos usados, mayores ejemplos para el entrenamiento pueden ser útiles junto con la adición de nuevas características de entrada.

Un estudio adicional presentado por Villafuerte, Manzano, Ayala, & García (2023) plantea el desarrollo de una aplicación web que junto con algoritmos basados en IA pueda determinar la infección respiratoria de la que sufre el usuario a partir de la sintomatología ingresada a la aplicación. Esta funcionaba con un aparato electrónico integrado y medía signos vitales como el ritmo cardíaco, la saturación de oxígeno en la sangre y la temperatura del cuerpo. Luego, esta información era enviada a un servidor donde se encontraba el algoritmo basado en IA, el cual procedía a determinar la enfermedad respiratoria del paciente. El modelo usado corresponde al algoritmo Máquina de Vectores de Soporte (SVM) que fue escogido luego de evaluar el rendimiento de otros algoritmos basados en AA. Al final, la información de la enfermedad respiratoria era presentada en la interfaz del usuario de la

aplicación. Los resultados indicaron una precisión del 91% lo cual indica que el algoritmo de IA logra proveer una confiable y eficiente forma de diagnosticar infecciones respiratorias (Villafuerte, Manzano, Ayala, & García, 2023).

1.5.7 Estructura de módulos

Los módulos para la implementación de este proyecto están compuestos por:

1. Pre-procesemiento.
2. Análisis exploratorio de datos.
3. Análisis de tendencias y clustering e interpretación de resultados.
4. Implementación de la aplicación web

El pre-procesamiento de datos consiste en la limpieza, normalización, manejo de datos faltantes y reducción de la dimensión de los datos para el análisis. Luego, se realiza los análisis exploratorios de datos, los cuales consisten en un estudio univariable y multivariable dando paso a la visualización de estos valores por medio de pairplots, heatmaps, y gráficos de mosaicos.

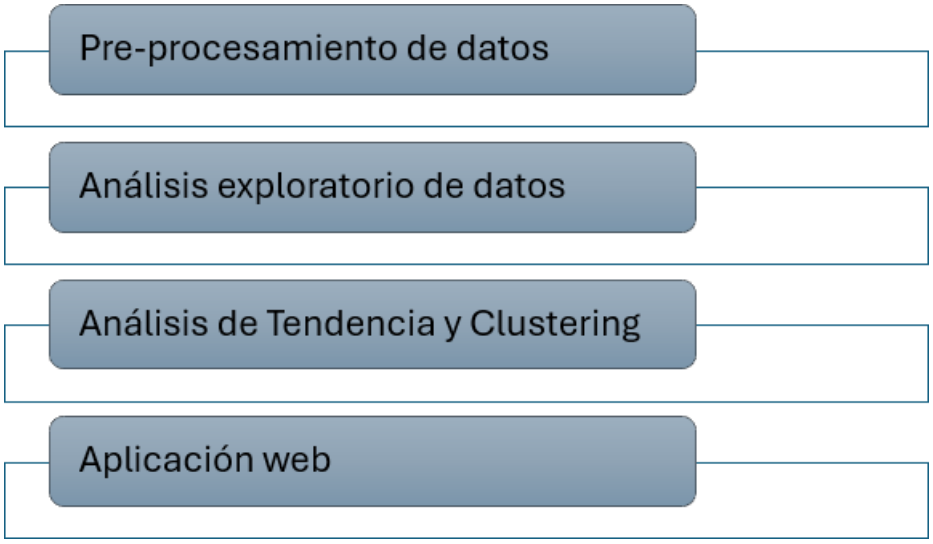
Posteriormente, se procede al análisis de tendencias y clustering para la interpretación de resultados, donde se analizan las agrupaciones generadas para identificar patrones comunes y diferencias entre los grupos. Finalmente, se realiza la implementación de la aplicación web.

Capítulo 2

Este capítulo detalla aquellos procedimientos realizados a lo largo del proyecto y se encuentra dividido en los siguientes módulos.

Figura 4

Módulos del Capítulo 2



Fuente: Autores de este documento.

2.1 Pre-procesamiento de datos

El pre-procesamiento de datos se realizó en varias etapas, para asegurar que los datos utilizados fueran precisos, completos y adecuados para el análisis.

2.1.1 Obtención de datos

Los datos fueron proporcionados por el LIB en archivos denominados Redcap que contenían información detallada sobre enrolamiento, preselección de síntomas, resultados múltiplex, junto con exámenes de muestras y de hemocultivo recolectados de los pacientes.

2.1.2 Extracción de los datos desde el repositorio Redcap

El enfoque para la extracción de estos datos se basó en el uso de ChatPDF el cual es una herramienta en línea que permite leer y extraer texto de documentos “.pdf” (Lichtenberger, 2024). Cabe mencionar que en esta herramienta solo se procesaron los archivos que contenían los resultados clínicos, mientras que los que contenían información personal del paciente no se incluyeron.

Con indicaciones específicas dadas en esta herramienta, se pudo obtener todo el texto. Considerando que en algunos casos los resultados obtenidos fueron palabras incorrectas o incompletas, se tuvo que validar lo extraído manualmente con el fin de verificar que todo este correcto.

Es importante mencionar que durante el proceso de extracción de datos usando ChatPDF surgieron dos problemas.

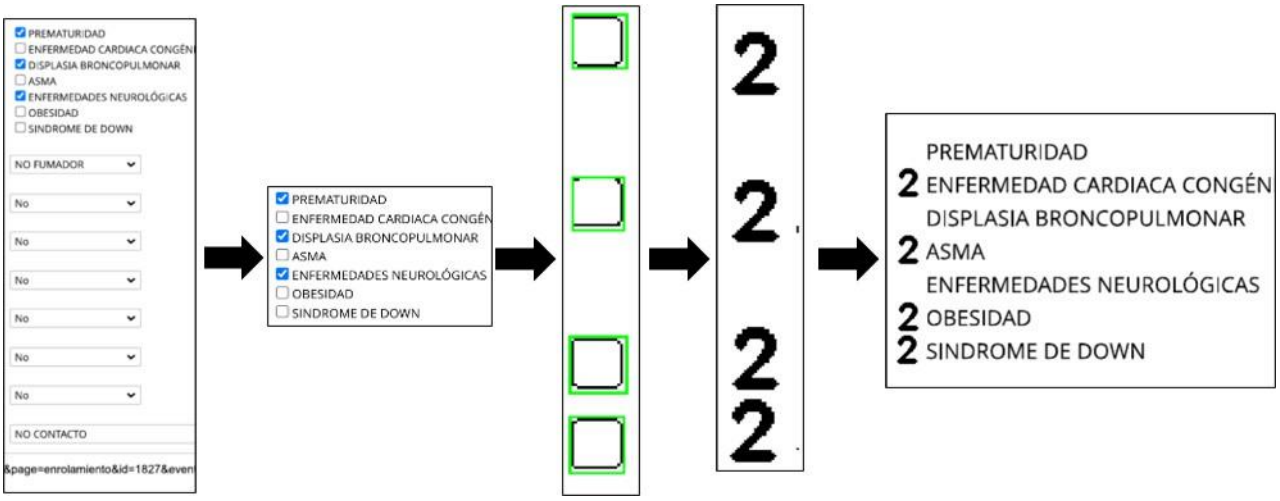
El primero se enfocó en el hecho que ChatPDF no reconocía campos con checkboxes. Para solucionar este inconveniente, se propuso la elaboración de un Pipeline de detección de checkboxes, el cual usó librerías como OpenCV, boxdetect, pdf2image y pytesseract. La librería boxdetect permite detectar fácilmente formas rectangulares como cajas checkbox en formularios escaneados (karolzak, 2023). Este Pipeline consistía en los siguientes pasos:

1. Transformar las páginas de los archivos ".pdf" a imágenes e identificar aquellas imágenes donde se encontraban los campos con checkboxes.
2. Recortar las imágenes para obtener su segunda mitad y así evitar extraer información innecesaria.
3. Identificar la posición de inicio de los pixeles de la sección de checkboxes para proceder con un nuevo recorte de la imagen y así obtener específicamente dicha sección.
4. Aplicar proceso de binarización a imágenes y proceder con la detección de checkboxes junto con inserción de símbolos para conseguir aquellos checkboxes que no se encuentren chequeados con un visto.
5. Extraer los valores correctos de "Sí" o "No" para los campos dependiendo de la existencia o no del símbolo insertado en cada campo.

La siguiente figura representa el flujo del proceso.

Figura 5

Flujo de proceso de detección de checkboxes



Fuente: Autores de este documento.

El segundo problema se enfocó en las dimensiones de los archivos ".pdf" proporcionados. La mayoría de los archivos contienen páginas con las siguientes dimensiones: anchura = 2339 y altura = 1653. Para estos archivos la extracción general de la información de los campos usando ChatPDF funcionaba correctamente, sin embargo, esta herramienta no reconocía la información de los campos para unos pocos archivos cuyas páginas contaban con diferentes dimensiones: anchura = 1654 y altura = 2339. Para solucionar este problema se aplicó el proceso OCR con la ayuda de la librería pytesseract a estos archivos.

Una vez completada la extracción, estos datos se guardaron en un archivo ".JSON" que contenían como clave, los campos de los ".pdf" y como valor los resultados de dichos campos. Luego, se procedió a transformar los archivos ".JSON" a formato .csv, esto se realizó con cada uno de los pacientes para obtener un solo archivo .csv de manera individual. Con los archivos individuales listos se procedió a combinarlos en un archivo .csv general al cual se lo llamó RedCap_Global.csv. En este archivo cada fila correspondía a la información de un paciente específico.

2.1.3 Limpieza de los datos

Para realizar la limpieza de los datos del archivo RedCap_Global.csv se tuvo que identificar los valores duplicados, valores faltantes y se realizó la corrección de los tipos de datos. Los valores de cada variable fueron analizados con el fin de asegurar coherencia entre ellos. Por otro lado, las variables correspondientes a fechas fueron puestas como tipo de dato datetime64, las variables numéricas como int64 y las restantes se mantuvieron como tipo object.

2.1.4 Categorización de los datos en Redcap

Categorización se refiere al proceso de conversión de los datos categóricos a numéricos de forma que cada uno de sus valores esté representado por un número. Un ejemplo de esta conversión es lo realizado a la variable categórica *TOS* cuyos valores eran "Si" o "No", aquí se transformaron los valores "Si" a 1, y los valores "No" a 0. Esta conversión fue realizada debido a que es necesario que los valores de un dataset sean numéricos antes de poder ser usados en un modelo de IA.

Debido a que se contaba con tres variables correspondientes a edades (*EDAD ANIOS*, *EDAD MESES*, *EDAD DIAS*), se tuvo que aplicar escalamiento para contar solo con una de ellas. Para conseguir esto, todas las edades fueron convertidas a su equivalente en días, creando una nueva variable denominada *EDAD ESCALADA DIAS*.

Por otro lado, se tuvo que buscar la forma de convertir las variables *FECHA VAC1*, *FECHA VAC2* y *FECHA VAC3*, variables que indican la fecha de inyección de vacunas contra el COVID-19, a una variable numérica para que así puedan ser utilizadas en el proceso de escalamiento final del dataset. Por esta razón se tomó la decisión de crear las nuevas variables *DIAS ENTRE VAC1-VAC2* y *DIAS ENTRE VAC2-VAC3* con el fin de tener un tipo de dato numérico apto para normalización y que indican la cantidad de días que han transcurrido entre dos fechas de inyecciones de vacunas.

2.2 Análisis exploratorio de datos

2.2.1 Resumen de los datos

Con la limpieza del dataset completo, se obtuvo en total 113 filas, donde cada una pertenece a la información de un paciente específico, y 75 columnas, cada una correspondiente a una característica del paciente.

2.2.2 Análisis univariable

Para realizar el análisis univariable primero se tuvo que identificar las variables continuas o categóricas.

Las variables continuas son aquellas variables numéricas que pueden contener un número infinito de valores entre dos valores cualesquiera (Minitab, s.f.). Las siguientes variables fueron identificadas como continuas.

Tabla 1

Lista de variables continuas

Variables continuas
• *PESO (KG)* • *TALLA (CM)* • *IMC* • *DIAS DESDE EL INICIO DE SINTOMAS* • *VALOR DE CT DE INFLUENZA A* • *VALOR DE CT DE SARS-COV-2* • *VALOR DE CT DE VSR* • *DIAS ENTRE VAC1-VAC2* • *DIAS ENTRE VAC2-VAC3*

Se realizó el proceso de identificación de Outliers con la ayuda de Boxplots para cada una de estas variables. Los Outliers son considerados como valores extremos que se desvían de la tendencia general de un conjunto de datos (Mioti, 2023), también son llamados como datos aberrantes. Existen varias técnicas para tratar con Outliers. En este proyecto, se utilizó la técnica de Winsorización (Zulmuthi, 2022) para tratarlos. Esta técnica en lugar de imputar

valores aberrantes con valores de media, mediana, moda, mínimo o máximo imputa dichos valores con un percentil escogido (Zulmuthi, 2022).

En la variable *DIAS DESDE EL INICIO DE SINTOMAS* no se encontraron valores aberrantes, sin embargo, en la variable *PESO (KG)* sí. Es importante mencionar que la técnica de Winsorización fue usada únicamente para los valores aberrantes de esta variable debido a que los Outliers de las demás variables se consideran relevantes de mantener para la identificación de patrones.

Por otro lado, las variables categóricas son aquellas variables que tienen un conjunto finito y discreto de valores (LinkedIn, s.f.). Las siguientes variables fueron identificadas como categóricas.

Tabla 2

Lista de variables categóricas

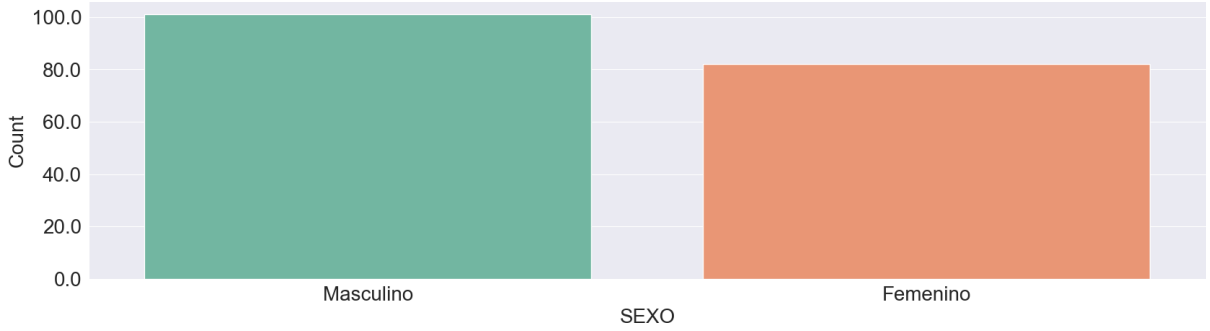
Variables categóricas
• *EDAD ESCALADA DIAS* • *SEXO* • *VACUNA INFLUENZA 2021* • *VACUNA INFLUENZA 2022* • *VACUNA INFLUENZA 2023* • *VACUNA COVID-19* • *USO DE MASCARILLA EN LOS ULTIMOS 7 DIAS* • *DX PREVIO DE COVID-19* • *PREMATURIDAD* • *ENFERMEDAD CARDIACA CONGENITA* • *DISPLASIA BRONCOPULMONAR* • *ASMA-CONDICIONPREEXISTENTE* • *ENFERMEDADES NEUROLOGICAS* • *OBESIDAD* • *ES FUMADOR?* • *CONTACTO O CRIANZA DE AVES?* • *CONTACTO O CRIANZA DE CERDOS?* • *FECHA DE INICIO DE SINTOMAS* • *CONTACTO EN LOS ULTIMOS 14 DIAS CON UNA PERSONA CON DIAGNOSTICO DE COVID-19?* • *ADMINISTRACION ANTIVIRAL SITEMICO DE LA ENFERMEDAD ACTUAL* • *USO DE ANTIBIOTICO DESDE EL INICIO DE SINTOMAS DE LA ENFERMEDAD ACTUAL* • *NEUMONIA*

- *BRONQUIOLITIS AGUDA*
- *ESTADO ASMATICO*
- *SEPSIS*
- *SHOCK SEPTICO*
- *INSUFICIENCIA RESPIRATORIA AGUDA (IRA)*
- *INFECCION DE VIAS RESPIRATORIAS AGUDA/REAGUDIZADA*
- *DISNEA/TOS*
- *SINDROME DE DISTRESS RESPIRATORIO*
- *SINDROME EMETICO/INTOLERANCIA ORAL*
- *DIARREA/NAUSEAS VOMITOS*
- *FIEBRE/ESCALOFRIOS*
- *FIEBRE (MEDIDA O REPORTADA)*
- *ESCALOFRIOS*
- *DOLOR MUSCULAR*
- *DOLOR DE CABEZA*
- *DOLOR DE GARGANTA O TOS*
- *NAUSEAS O VOMITOS*
- *DIARREA*
- *FATIGA*
- *CONGESTION O SECRECION NASAL*
- *TOS*
- *FALTA DE AIRE*
- *DIFICULTAD PARA RESPIRAR*
- *CONFUSION O CAMBIO EN EL ESTADO MENTAL*
- *DOLOR O PRESION PERSISTENTE EN EL PECHO*
- *COLOR DE LA PIEL/ LABIOS O UNIAS PALIDO/ GRIS O AZULADO*
- *DIFICULTAD PARA ESTAR O MANTENERSE DESPIERTO*
- *EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA*
- *SINDROME DE DISTRESS RESPIRATORIO AGUDO (SDRA)*
- *RESULTADO DE SARS-COV-2*
- *RESULTADO DE INFLUENZA A*
- *RESULTADO DE VSR*
- *TIPO DE MUESTRA*
- *MADRE RECIBIO VACUNA COVID-19?*
- *NUMERO DE DOSIS*
- *TIPO VACUNA 1*
- *TIPO VACUNA 2*
- *TIPO VACUNA 3*
- *SE LE HA TOMADO HEMOCULTIVO?*
- *RESULTADOHM*
- *AISLAMIENTO DE*
- *NOMBRE DEL PATOGENO*
- *BENCILPENICILINA RESISTENTE >=0.5*
- *CIPROFLOXACINA*

Para cada una de estas variables se elaboró su respectivo gráfico de barras con el fin de apreciar las frecuencias de cada categoría. La siguiente figura muestra un ejemplo de gráfico de barras elaborado.

Figura 6

Ejemplo de gráfico de barras de las frecuencias de los datos categóricos



Fuente: Autores de este documento.

Los gráficos de barras sirvieron como soporte para identificar aquellas variables que podrían crear un sesgo lo cual afectaría en la experimentación con los algoritmos de clustering. Una vez identificadas, estas fueron eliminadas del conjunto de datos global.

2.2.3 Análisis multivariable

2.2.3.1 Análisis multivariable de variables categóricas.

El análisis multivariable es una técnica estadística utilizada para examinar múltiples variables simultáneamente. Su objetivo es encontrar patrones, relaciones y asociaciones entre las variables (Instituto de Innovación Digital de las Profesiones., 2023).

Se utilizó la prueba de chi-cuadrado para hacer el análisis de las variables categóricas. La prueba de chi-cuadrado es una prueba de hipótesis utilizada para determinar si existe una relación entre dos variables categóricas (datatab, 2024).

Además, se construyó una tabla de contingencia que mostró la frecuencia de cada combinación de categorías entre dos variables y posteriormente se aplicó la prueba de chi-cuadrado para evaluar la dependencia entre ellas, los resultados obtenidos eran el valor de

chi-cuadrado y el valor de p. El valor de p indica la probabilidad observada, y se usa para decidir si la hipótesis es nula, se rechaza o se mantiene (DATATab, 2024).

2.2.3.2 Análisis multivariable de variables continuas.

Se procedió a realizar la matriz de correlación con el método de Pearson, que es una prueba que mide la relación estadística entre dos variables continuas (Ortega, 2024). El coeficiente de correlación puede tomar un rango entre los valores [-1, 1], si el valor es igual a 0 indica que no hay una asociación entre las dos variables y sería una correlación nula, si el valor se inclina hacia un valor positivo indica que, si el valor de una variable aumenta, la otra también lo hará, a esto se denomina correlación positiva. En el caso de que el valor sea menor a cero, indica una asociación negativa es decir que, si una variable disminuye la otra también, conocida también como correlación negativa (Ortega, 2024).

2.2.4 Normalización del Dataset

Para la normalización de todos los valores del dataset se tuvieron dos opciones: StandardScaler y MinMaxScaler. StandardScaler estandariza las características eliminando la media y escala los datos para que su varianza sea igual a 1 (Scikitlearn, s.f.). Por otro lado, MinMaxScaler escala los valores de las características a un rango específico, en este caso en valores entre 0 y 1 (Scikitlearn, s.f.).

La decisión entre cuál de las dos técnicas de normalización usar se basó en la recomendación de usar StandardScaler cuando los datos tienen una distribución normal (Gómez, 2023). En cambio, MinMaxScaler responde correctamente si la distribución de los datos no es normal.

Por ello, primero, se verificó si los valores de cada variable del dataset seguían o no una distribución normal. En este punto entra la prueba de Shapiro-Wilk, la cual es una prueba estadística que evalúa si una muestra de datos se ajusta a una distribución normal (Urrego, 2023). Además, esta prueba proporciona un valor p.

Al realizar la prueba, los valores de p de cada una de las variables presentaban valores mucho menores al nivel de significancia de 0.05 lo cual, según la teoría, indicaba que el dataset no seguía una distribución normal. Por ello, se concluyó que el método de normalización MinMaxScaler era el más adecuado para el conjunto de datos del presente proyecto.

2.3 Análisis de tendencias y clustering

2.3.1 Análisis de series temporales

Una vez los datos fueron categorizados, se procedió a realizar el análisis de series temporales, y poder detectar así las tendencias o ciclos.

Una serie temporal es un conjunto de observaciones que se obtiene midiendo una variable de manera regular en el transcurso del tiempo (IBM Corporation, 2021). Para ver cómo cambian las variables de este proyecto en un intervalo dado, se utilizó las fechas proporcionadas por la variable *FECHA DE INICIO DE SINTOMAS*, se emplearon gráficas de líneas para visualizar las tendencias de manera efectiva. Las variables analizadas fueron aquellas que tenían relación con la sintomatología respiratoria.

2.3.1.1 Descomposición de series temporales.

Con el análisis del proceso anterior, se procedió a separar las series temporales en componentes de tendencia y estacionalidad.

Para entender los patrones subyacentes de los componentes se analiza lo siguiente:

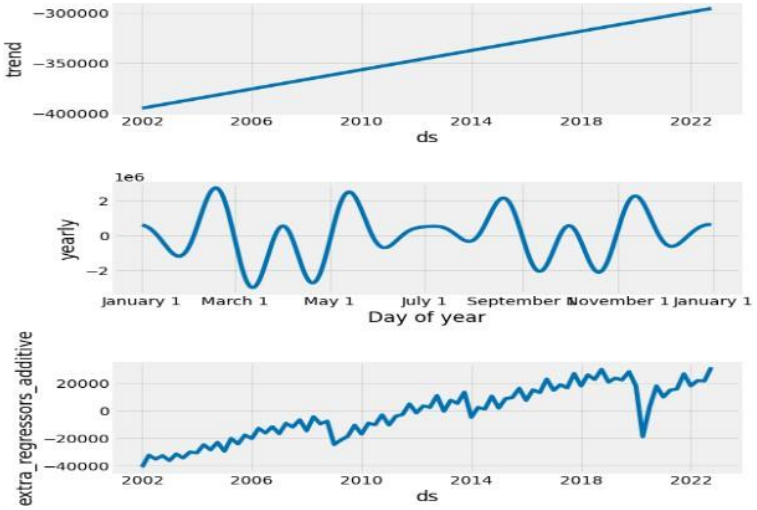
- Tendencia: Es el patrón subyacente que muestra una dirección general en el tiempo, estas pueden ser ascendente, descendente o plana, no siempre sigue una forma lineal (CODEa UNI, 2021).
- Estacionalidad: Cuando una serie muestra patrones recurrentes o ciclos en periodos fijos, como días, meses o trimestre. Este patrón puede estar

influenciado por factores como la temporada de años y/o eventos regulares (CODEa UNI, 2021).

En la Figura 7, se tiene un ejemplo de visualización de una descomposición de series temporales.

Figura 7

Ejemplo de descomposición de series



Nota. Obtenido de (Homar, 2022).

2.3.2 Análisis de estacionalidad

2.3.2.1 Identificación de estacionalidad.

Para el análisis de patrones de estacionalidad, se tuvieron que verificar los resultados obtenidos con las gráficas de descomposición de series temporales. Se usó un medidor para poder identificar los datos estacionales y no estacionales, esta medida se la realizó utilizando la desviación estándar, que sirve para identificar los fortalecimientos de las tendencias, cuando mayor es el valor de medición más fuerte será la tendencia (LiteFinance.org, 2024).

Se analizó todas las columnas del dataset en función del tiempo para comprobar de una manera más efectiva las tendencias simples. Se creó un dataframe en donde la variable *FECHA DE INICIO DE SINTOMAS* se puso como índice para evitar problemas al momento de realizar los análisis respectivos.

2.3.2.2 Pronósticos de series temporales.

Una vez identificados los datos como series temporales y no temporales, se procedió a realizar el pronóstico de tendencias utilizando modelos estadísticos específicos: ARIMA (AutoRegressive integrated Moving Average) y SARIMAX (Seasonal AutoRegressive Integrated Moving Average with eXogenous regressors). Ambos son utilizados para la predicción de series temporales (Escobar Ortiz & Amat Rodrigo, 2024).

En ambos casos, se debe tener en cuenta los parámetros específicos: p, d y q, que representan los componentes autorregresivos, de diferenciación y media móvil. Además, en SARIMAX también se incluyen los parámetros P, D, Q y m, que son componentes adicionales para la parte estacional del modelo (Escobar Ortiz & Amat Rodrigo, 2024).

Para encontrar los parámetros de orden se utilizaron dos estrategias diferentes, con la finalidad de obtener valores más exactos para la evaluación de los modelos. La primera fue con GridSearch (scikit-learn developers, 2024) que emplea un método exhaustivo para ajustar y puntuar, buscando los parámetros óptimos para un estimador especifico. Con esto se puede asegurar tener los valores de p, d y q, pero al evaluar estos resultados en los modelos seleccionados las predicciones de las tendencias no fueron eficientes, por lo cual se buscó otra forma para tener los parámetros más precisos. Como segunda estrategia se usó la librería Mango (Rodrigo, 2024) que se asemeja al funcionamiento de la optimización Bayesiana, la cual consiste en crear un modelo probabilístico a través del cual se encuentre el valor de los parámetros de la función objetivo, reduciendo el número de combinaciones de los híper-parámetros con las que se evalúa el modelo y de esta manera se escoge a los mejores candidatos (Rodrigo, 2024).

Entre las dos estrategias se decidió por la que ofrece la librería Mango dado que sus valores de parámetros eran más eficientes para el modelo ARIMA y SARIMAX, que se aplicaron para la evaluación y análisis de tendencias. Conocidos los datos estacionales, se

procedió a aplicar los modelos correspondientes, para lo cual se dividió los datos en un 70% para entrenamiento y 30% para pruebas.

Se usaron bibliotecas de estos modelos para tener varias predicciones y así poder visualizar las tendencias, estas fueron las bibliotecas de ARIMA, FORESCAST, PMDARIMA, STATSMODELS y SKFORESCAST (Escobar Ortiz & Amat Rodrigo, 2024). Estos modelos se usaron directamente para los datos que contenían estacionalidad.

Statsmodels: Se utilizó para el modelado estadístico en Python y su visualización es parecida a scikit-learn (Escobar Ortiz & Amat Rodrigo, 2024).

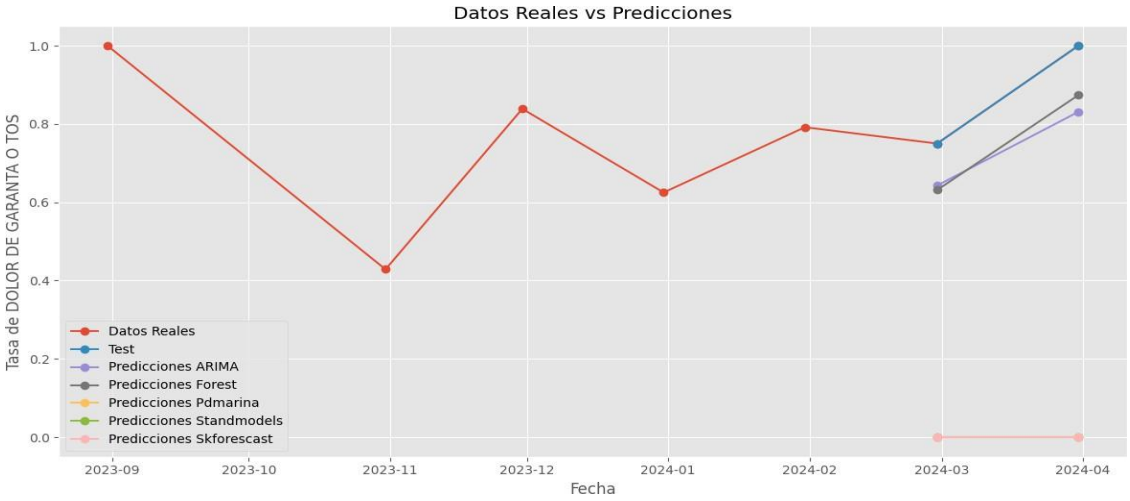
Pmdarima: Adapta SARIMAX de Statsmodels a la API de Scikit-learn, facilitando el modelado de series temporales (Escobar Ortiz & Amat Rodrigo, 2024).

Skforescast: Incluye una versión optimizada de ARIMAX de Statsmodels, ofreciendo mejoras en velocidad (Escobar Ortiz & Amat Rodrigo, 2024).

Se generó un dataframe individual utilizando los datos seleccionados con el objetivo de identificar tendencias. Posteriormente, se realizó una agrupación de los datos por meses y se calculó el promedio mensual, asegurándose de eliminar los valores nulos. Este enfoque fue necesario dado que los modelos mencionados no admiten valores binarios en su procesamiento. Un ejemplo de visualización de los modelos es como se muestra en la siguiente figura.

Figura 8

Ejemplo de tendencias con los modelos



Fuente: Autores de este documento.

2.3.2.3 Modelos basados en aprendizaje automático.

2.3.2.3.1 Modelos LSTM (Long Short Term Memory).

Es una arquitectura de red neuronal recurrente usada en el campo del Aprendizaje Profundo, las cuales tienen conexiones de retroalimentación que les permite recordar dependencias temporales a lo largo de secuencias de datos (Hamad, 2023). Se decidió usar esta arquitectura con el fin de realizar predicciones acerca de las tendencias obtenidas a lo largo del estudio, en las características pertenecientes a diagnósticos y síntomas.

Específicamente se usó el modelo Univariado + Unistep (Sotaquirá, 2023) en el cual se ingresa solo una característica al modelo, como por ejemplo casos de Neumonía, y se espera la predicción de solo un momento específico en el tiempo, por ejemplo, cantidad de casos de Neumonía para un mes próximo.

2.3.3 Análisis de clustering

Este análisis se desarrolló con el fin de poder identificar la existencia de patrones ocultos entre los datos clínicos.

Se tomaron en cuenta 4 algoritmos de clustering: K-Means, K-Medoids, Clustering jerárquico y DBSCAN.

2.3.3.1 K-Means.

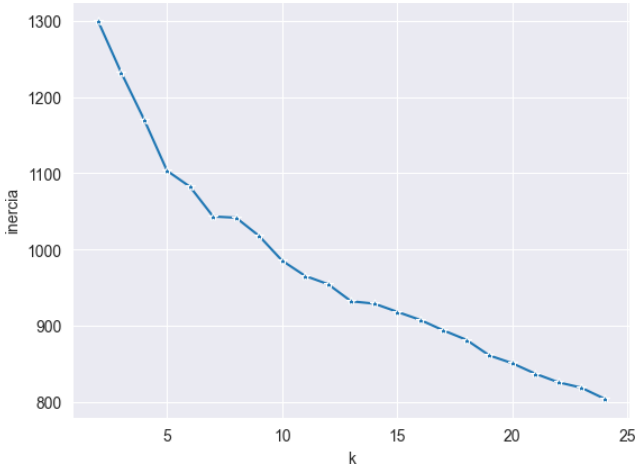
Es un algoritmo utilizado únicamente para variables numéricas (Mujeeb, Educative, s.f.). Como se mencionó antes, este algoritmo usa un número predefinido de agrupaciones a crear, sin embargo, con el fin de no asignar un valor aleatorio, se usan métodos que permitan encontrar un valor recomendable de agrupaciones.

Estos dos métodos son: El método del Codo y el del Coeficiente de Silueta.

El método del Codo consiste en trazar la suma de las distancias al cuadrado entre cada punto de datos y su centroide asignado para diferentes valores de *k*, siendo k el número de agrupaciones creadas (Barrios, Medium, 2023). La siguiente figura presenta un ejemplo del resultado obtenido usando este método.

Figura 9

Ejemplo de gráfico del método del Codo



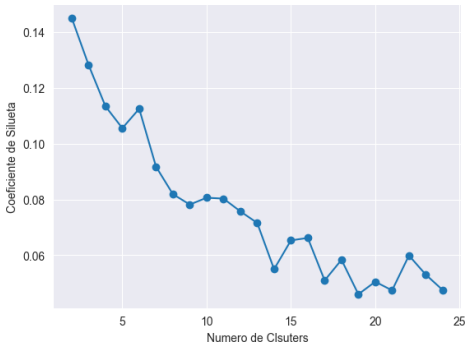
Fuente: Autores de este documento.

En la Figura 9, el codo se identifica donde exista la curvatura de la línea más pronunciada. En este caso, el codo podría estar entre los valores 5, 6 y 7.

El método del Coeficiente de Silueta consiste en calcular el coeficiente de silueta para diferentes valores de k, el cual es una medida que evalúa cuán parecido está un punto de los datos a su clúster asignado en comparación con otros clústeres (Barrios, Medium, 2023). La siguiente figura presenta un ejemplo del resultado obtenido usando este método.

Figura 10

Ejemplo de gráfico del método del Coeficiente de Silueta



Fuente: Autores de este documento.

En esta Figura 10, el número de clústeres óptimo es el punto que presenta el valor más alto de coeficiente de silueta. En este caso, dicho valor es el número 2.

2.3.3.2 K-Medoids.

Es otro algoritmo de clustering que puede ser usado para conjuntos de datos que contengan datos numéricos y categóricos (Mujeeb, Educative, s.f.). Este algoritmo usa medoides, los cuales son puntos del dataset usado cuya suma de distancias hacia otros puntos en una agrupación es mínima (Mujeeb, Educative, s.f.). A diferencia del algoritmo K-Means, este usa la distancia Manhattan y resulta más costosos de implementar (Mujeeb, Educative, s.f.).

Los métodos del Codo y del Coeficiente de Silueta también fueron usados en este algoritmo.

2.3.3.3 Algoritmo de clustering Jerárquico.

Este algoritmo considera cada punto como un clúster individual al inicio, luego empieza a combinar el par de clústeres más cercanos; esto se realiza hasta que todos los clústeres se hayan combinado en un solo clúster que contenga todos los puntos del dataset (Javatpoint , s.f.).

Cuenta con dos enfoques: Aglomerativo y Divisivo. En este caso se utilizó el enfoque Aglomerativo y no el Divisivo ya que éste brinda un rendimiento más lento al ser más costoso desde el punto de vista computacional (Barrios, Medium, 2023).

Con este algoritmo se desarrolla la jerarquía de los clústeres en la forma de un árbol, el cual es llamado Dendrograma (Barrios, Medium, 2023).

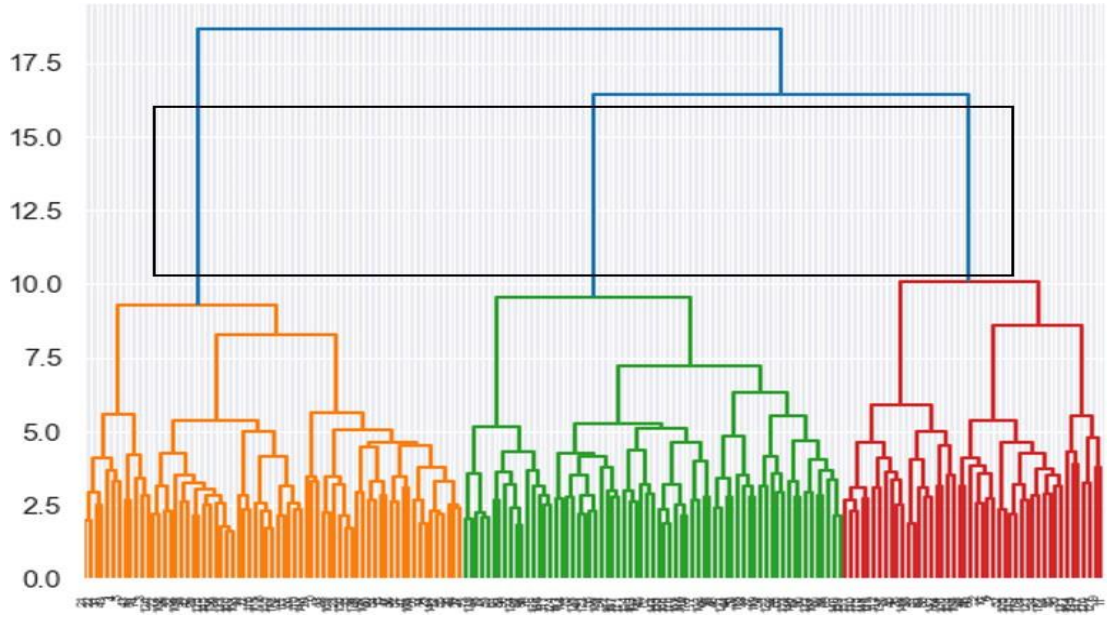
Este algoritmo también incluye los métodos del Codo, Coeficiente de Silueta y el Dendrograma para encontrar el valor óptimo de clústeres a crear.

Para encontrar el número de clústeres óptimo con el Dendrograma, se debe encontrar primero la distancia vertical máxima que no corta ninguna barra horizontal. Una vez encontrada esta distancia máxima, el número de líneas verticales dentro de ella indicará el número óptimo de clústeres.

La siguiente figura presenta un ejemplo de Dendrograma.

Figura 11

Ejemplo de Dendrograma con su distancia vertical máxima



Fuente: Autores de este documento.

Con esta figura se observa que el número de clústeres óptimo a crear es 3 ya que, dentro de la distancia vertical máxima, se encuentran un total de 3 líneas verticales.

2.3.3.4 DBSCAN.

Este algoritmo utiliza una distancia especificada para separar los clústeres densos del ruido más disperso. Su uso es apropiado si se puede utilizar una distancia de búsqueda clara y funciona correctamente con todos los clústeres potenciales. Para ello, es requerido que todos los clústeres significativos presenten densidades similares (ArcGIS Pro 3.3, s.f.).

2.4 Aplicación web

La aplicación web implementada se enfocó principalmente para ser usada por médicos. Esta, tiene la intención de ayudar a tomar decisiones informadas con respecto a diagnósticos por medio del análisis realizado a sintomatología con la ayuda de modelos de IA.

2.4.1 Roles

La aplicación consta de dos roles: Médico, y administrador. El médico debe poder iniciar sesión en la aplicación, una vez que ingresa al sistema se le permitirá cargar archivos con formato ".pdf", los cuales contienen información acerca de la sintomatología de un paciente. Luego, la aplicación extraerá la información de la sintomatología y como resultado presentará distintos gráficos y estadísticas; por ejemplo: se podrán visualizar gráficos de tendencias acerca de la sintomatología del paciente, y también mostrará a que agrupación pertenece el paciente, tomando como referencia las agrupaciones creadas por medio de los algoritmos de IA entrenados. Conociendo la agrupación, el médico podrá tener una idea más clara acerca de distintos patrones de sintomatología que puede presentar el paciente. Es importante mencionar que estos resultados vendrán acompañados por un valor de confianza el cual indicará que tan precisos son los resultados presentados. Luego del análisis, el médico podrá agregar comentarios u observaciones acerca de los resultados y los podrá guardar junto con estos.

El administrador, podrá ingresar a la aplicación para poder revisar el listado de los médicos que estén registrados, podrá observar los resultados que hayan sido guardados por estos, podrá descargarlos y tendrá la opción de registrar más médicos.

2.4.2 Valor de confianza

El valor de confianza que se presentará en la aplicación representa la precisión con la que se asignará el paciente analizado a una agrupación en específico. Este valor es obtenido mediante un modelo de Clasificador de Árbol de Decisión, el cual es un tipo de algoritmo que usa una estructura parecida a un árbol para clasificar instancias en función de los valores de sus características (ScienceDirect, 2019).

2.4.3 Frameworks

Para la sección del Frontend se tomó la decisión de usar la herramienta React.js, la cuál es una librería de JavaScript de código abierto, construida por Facebook que tiene como objetivo simplificar el complejo proceso de creación de interfaces de usuario interactivas (Herbert, 2023).

Por parte del Backend, se tomó la decisión de usar Django, el cuál es un software que se puede utilizar para desarrollar aplicaciones web de forma rápida, eficiente y además es de código abierto (Amazon Web Services, s.f.).

Estas herramientas fueron escogidas debido a los beneficios que brindan gratuitamente los cuales son suficientes para el desarrollo de la aplicación.

Capítulo 3

En este capítulo se presentan los resultados obtenidos de la metodología junto con los análisis pertinentes.

3.1 Resultados del Pre-procesamiento de datos

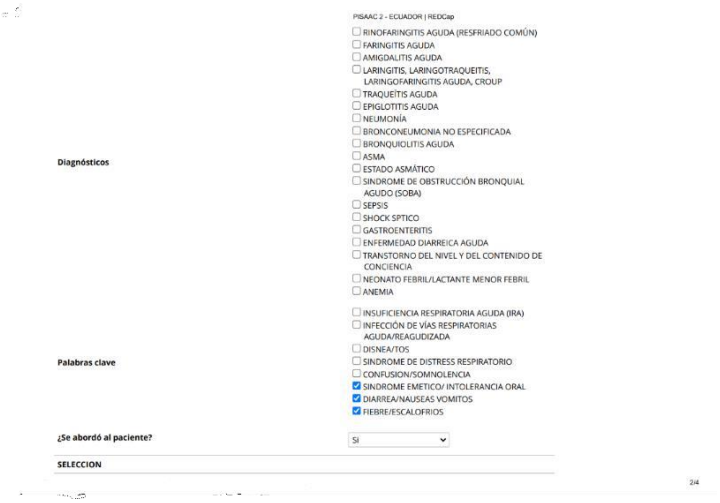
3.1.1 Obtención de datos

Los datos obtenidos y usados en este proyecto son archivos Redcap, los cuales, como se mencionó anteriormente, contienen información sobre la sintomatología de distintos pacientes.

La siguiente figura presenta la estructura del tipo de archivo.

Figura *12*

Estructura de archivo Redcap



Fuente: LIB.

3.1.2 Extracción de los datos desde el repositorio Redcap

Antes de presentar la estructura de los datos obtenidos de los archivos Redcap, es importante mencionar que fueron eliminadas varias variables, las cuales no se consideraron relevantes para alcanzar los objetivos de este proyecto. Entre esas variables se encontraban los detalles de la hospitalización del paciente y campos de confirmación de los exámenes realizados.

Como resultado se obtuvo la siguiente estructura mostrada en la Figura 13 para este tipo de datos.

Figura 13

Estructura de los datos de archivo Redcap

	EDAD ANIOS	SEXO	PESO (KG)	TALLA (CM)	VACUNA INFLUENZA 2021	VACUNA INFLUENZA 2022	VACUNA INFLUENZA 2023	VACUNA COVID-19	USO DE MASCARILLA EN LOS ULTIMOS 7 DIAS
0	3	Masculino	15.5	94	Si	No sabe	No sabe	No	NUNCA
0	4	Masculino	14.5	100	Si	Si	Si	Si	NUNCA
0	0	Masculino	6.2	58	NI	NI	NI	NI	NUNCA
0	11	Masculino	37.3	143	No sabe	No	No	No	CASI SIEMPRE
0	14	Masculino	72.3	162	No	No	No	Si	NUNCA
...	...	...	...	...	...	...	...	...	...
0	0	Masculino	6.0	61	NI	NI	NI	NI	A VECES
0	3	Masculino	5.4	76	No	Si	No	No	NUNCA
0	0	Femenino	2.5	44	NI	NI	NI	NI	NUNCA
0	0	Masculino	3.6	55	NI	NI	NI	NI	NUNCA
0	4	Masculino	22.2	NI	No	Si	No	No	NUNCA

183 rows × 78 columns

Fuente: Autores de este documento.

3.1.3 Limpieza de los datos

En el conjunto de datos analizado no se encontró valores duplicados. Por otro lado, los valores faltantes identificados se mantuvieron debido a que se consideraron como relevantes para el análisis individual de cada variable. Por último, se realizó los debidos cambios de tipos de datos; las variables: *FECHA DE INICO DE SINTOMAS* y *DIAS DESDE EL INICIO DE SINTOMAS* se transformaron a tipo de dato Datetime64, mientras que las variables: *EDAD ANIOS, PESO(KG)* y *TALLA (CM)* se convirtieron en variables de tipo Int64. Por último, el resto de las variables se mantuvieron como tipo de dato Object.

3.1.4 Categorización de los datos en Redcap

Con la conversión de los datos categóricos a numéricos, se obtuvo un conjunto de datos que contenía únicamente datos numéricos. La siguiente tabla muestra lo que representan los nuevos valores de las variables categóricas.

Tabla 3

Representación de los nuevos valores de las variables categóricas

Variables	Conversión y significado
SEXO	Masculino = 0, Femenino = 1
VACUNA INFLUENZA 2021	Si = 1, No = 0, Ni = 2
VACUNA INFLUENZA 2022	Si = 1, No = 0, Ni = 2
VACUNA INFLUENZA 2023	Si = 1, No = 0, Ni = 2
VACUNA COVID-19	Si = 1, No = 0, Ni = 2
USO DE MASCARILLA EN LOS ULTIMOS 7 DIAS	NUNCA = 0, SIEMPRE = 4, A VECES = 2, CASI NUNCA = 1, CASI SIEMPRE = 3
DX PREVIO DE COVID-19	Si = 1, No = 0
PREMATURIDAD	Si = 1, No = 0, Ni = 2
ENFERMEDAD CARDIACA CONGENITA	Si = 1, No = 0, Ni = 2
DISPLASIA BRONCOPULMONAR	Si = 1, No = 0, Ni = 2
ASMA-CONDICIONPREEXISTENTE	Si = 1, No = 0, Ni = 2
ENFERMEDADES NEUROLOGICAS	Si = 1, No = 0, Ni = 2
OBESIDAD	Si = 1, No = 0, Ni = 2
ES FUMADOR?	Si = 1, No = 0, Ni = 2
CONTACTO O CRIANZA DE AVES?	Si = 1, No = 0, Ni = 2
CONTACTO O CRIANZA DE CERDOS?	Si = 1, No = 0, Ni = 2
CONTACTO EN LOS ULTIMOS 14 DIAS CON UNA PERSONA CON DIAGNOSTICO DE COVID-19?	NO CONTACTO = 0, OTRO CONTACTO = 1, CONTACTO DENTRO DE SU LUGAR DE TRABAJO = 2, CONCTACTO DENTRO DE SU CASA = 3
ADMINISTRACION ANTIVIRAL SITEMICO DE LA ENFERMEDAD ACTUAL	Si = 1, No = 0, No sabe = 2
USO DE ANTIBIOTICO DESDE EL INICIO DE SINTOMAS DE LA ENFERMEDAD ACTUAL	Si = 1, No = 0

NEUMONIA	Si = 1, No = 0
BRONQUIOLITIS AGUDA	Si = 1, No = 0
ESTADO ASMATICO	Si = 1, No = 0
SEPSIS	Si = 1, No = 0
SHOCK SEPTICO	Si = 1, No = 0
INSUFICIENCIA RESPIRATORIA AGUDA (IRA)	Si = 1, No = 0
INFECCION DE VIAS RESPIRATORIAS AGUDA/REAGUDIZADA	Si = 1, No = 0
DISNEA/TOS	Si = 1, No = 0
SINDROME DE DISTRESS RESPIRATORIO	Si = 1, No = 0
SINDROME EMETICO/INTOLERANCIA ORAL	Si = 1, No = 0
DIARREA/NAUSEAS VOMITOS	Si = 1, No = 0
FIEBRE/ESCALOFRIOS	Si = 1, No = 0
FIEBRE (MEDIDA O REPORTADA)	Si = 1, No = 0
ESCALOFRIOS	Si = 1, No = 0
DOLOR MUSCULAR	Si = 1, No = 0
DOLOR DE CABEZA	Si = 1, No = 0
DOLOR DE GARGANTA O TOS	Si = 1, No = 0
NAUSEAS O VOMITOS	Si = 1, No = 0
DIARREA	Si = 1, No = 0
FATIGA	Si = 1, No = 0
CONGESTION O SECRECION NASAL	Si = 1, No = 0
TOS	Si = 1, No = 0
FALTA DE AIRE	Si = 1, No = 0
DIFICULTAD PARA RESPIRAR	Si = 1, No = 0
CONFUSION O CAMBIO EN EL ESTADO MENTAL	Si = 1, No = 0
DOLOR O PRESION PERSISTENTE EN EL PECHO	Si = 1, No = 0

COLOR DE LA PIEL/ LABIOS O UNIAS PALIDO/ GRIS O AZULADO	Si = 1, No = 0
DIFICULTAD PARA ESTAR O MANTENERSE DESPIERTO	Si = 1, No = 0
EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA	Si = 1, No = 0
SINDROME DE DISTRESS RESPIRATORIO AGUDO (SDRA)	Si = 1, No = 0
RESULTADO DE SARS-COV-2	Positivo = 1, Negativo = 0
RESULTADO DE INFLUENZA A	Positivo = 1, Negativo = 0
RESULTADO DE VSR	Positivo = 1, Negativo = 0
TIPO DE MUESTRA	HISOPADO SOLO NASAL = 1, HISOPADO SOLO FARINGEO = 2, HISOPADO NASAL Y FARINGEO = 3, MUESTRA DE ASPIRADO ENDOTRAQUEAL = 4
MADRE RECIBIO VACUNA COVID-19?	Si = 1, No = 0, Ni = 2
NUMERO DE DOSIS	Ni = 0, 1 = 1, 2 = 2, 3 = 3, 4 = 4
TIPO VACUNA 1	Ni = 0, Sinovac = 1, Pfizer = 2, AstraZeneca = 3, Cansino = 4
TIPO VACUNA 2	Ni = 0, Sinovac = 1, Pfizer = 2, AstraZeneca = 3, Cansino = 4
TIPO VACUNA 3	Ni = 0, Sinovac = 1, Pfizer = 2, AstraZeneca = 3
SE LE HA TOMADO HEMOCULTIVO?	Ni = 0, Si = 1
RESULTADOHM	Ni = 0, Positivo = 1
AISLAMIENTO DE	Ni = 0, Bacteria = 1
NOMBRE DEL PATOGENO	Ni = 0, Staphylococcus epidermidis = 1, Staphylococcus hominis = 2, Pseudomonas aeruginosa = 3, Staphylococcus aureus = 4

BENCILPENICILINA >=0.5	**RESISTENTE**	Ni = 2, Si = 1
CIPROFLOXACINA		Ni = 2, Si = 1

Antes del proceso de limpieza y categorización del dataset, este contenía un total de 183 observaciones (siendo cada una de ellas la información correspondiente a un paciente específico). Sin embargo, luego de los procesos mencionados la cantidad de observaciones disminuyó manteniendo al final 113 muestras. Esta disminución se debe principalmente a la cantidad de datos faltantes para la variable *TALLA (CM)*. Esto quiere decir que había muchos pacientes los cuales no contaban con información en dicho campo y por ello fueron excluidos del dataset.

3.2 Resultados del análisis exploratorio de datos

3.2.1 Análisis univariable

3.2.1.1 Análisis univariable de variables continuas

Dado que el foco de este análisis es la identificación de Outliers, se analizó cada una de las variables continuas.

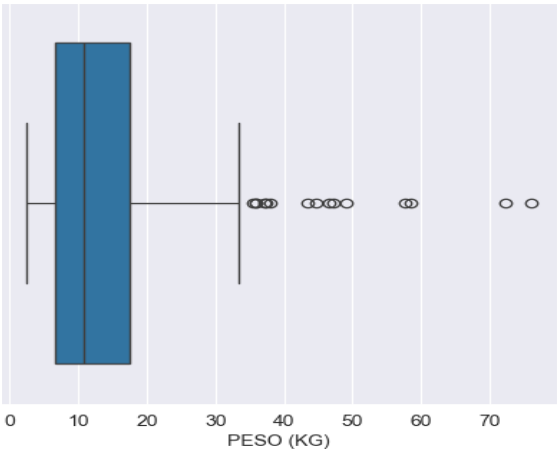
En las variables relacionadas, tanto a los valores de la tomografía computarizada (*CT)* de las infecciones respiratorias, como a las variables *DIAS ENTRE VACUNAS* se encontraron valores aberrantes. Hay que tener en consideración que muchos pacientes no presentan información en estas variables ya que no aplican. Por ejemplo, un paciente que tenga como resultado “Negativo” en el campo vacuna contra la Influenza A, se le colocará un valor numérico 0. Entonces, a pesar de existir valores aberrantes, se tomó la decisión de mantenerlos dado a los pocos casos de pacientes los cuales aplican para presentar valores en estos campos.

Por otro lado, están los campos correspondientes a la información biométrica de los pacientes, los cuales son: El *PESO (KG)*, la *TALLA (CM)* e *IMC*. En las variables de

TALLA(CM) e *IMC* se encontraron muy pocos Outliers los cuales se mantuvieron, sin embargo, en la variable *PESO (KG)* es donde se encontró la mayor cantidad de Outliers. En la Figura 14 se muestran los Outliers encontrados en esta variable.

Figura 14

Outliers de la variable peso



Fuente: Autores de este documento.

Por ello, se aplicó la técnica de Winsorización (Zulmuthi, 2022) para que los valores aberrantes tomen un nuevo valor el cual se encuentre por debajo del límite superior del rango intercuartil.

3.2.1.2 Análisis univariable de variables categóricas

El foco de este análisis se centró en la elaboración de gráficos de barras para cada variable con el fin de observar la frecuencia de sus valores. Entre los gráficos de barras elaborados, se pudo identificar varias variables las cuales presentaban un desbalance considerablemente alto entre sus valores. En Figura 15, presenta un ejemplo de una variable categórica que cumple con dicha característica.

Figura 15

Gráfico de barras de la variable HOSPITALIZACION PREVIA POR COVID-19

Fuente: Autores de este documento.

Esta variable presenta en su gran mayoría observaciones con valor “No”, mientras que solo unas pocas observaciones tienen “Si”. Esto pudiera generar un sesgo al momento de realizar las experimentaciones con los algoritmos de clustering y por ello, las variables que contaban con este comportamiento fueron eliminadas del dataset original.

Con este análisis se redujo el número de características de cada paciente en el dataset pasando de 100 a 75.

3.2.2 Análisis multivariable

3.2.2.1 Análisis multivariable de variables Categóricas

Para el análisis multivariable se procedió a realizar técnicas estadísticas y así verificar la relación entre variables categóricas. Se realizó el análisis de asociación utilizando la prueba de Chi-cuadrado para evaluar relaciones entre las diferentes variables categóricas seleccionadas. En la siguiente figura se visualiza las asociaciones más significativas encontradas:

Figura 16

Dataset de relaciones entre las variables categóricas

	Variable 1	Variable 2	p-value	Chi-Cuadrado
0	SEXO	DISPLASIA BRONCOPULMONAR	0.032611	6.846183
1	VACUNA INFLUENZA 2021	EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA	0.044210	6.237590
2	VACUNA INFLUENZA 2021	TIPO DE MUESTRA	0.034035	10.411712
3	VACUNA INFLUENZA 2022	CONGESTION O SECRECION NASAL	0.046886	6.120066
4	VACUNA INFLUENZA 2023	NEUMONIA	0.044439	6.227291
5	VACUNA INFLUENZA 2023	COLOR DE LA PIEL/ LABIOS O UNIAS PALIDO/ GRIS ...	0.030453	6.983173
6	VACUNA COVID-19	PREMATURIDAD	0.032102	10.550949
7	VACUNA COVID-19	NEUMONIA	0.030972	6.949319
8	VACUNA COVID-19	BRONQUIOLITIS AGUDA	0.030511	6.979311
9	VACUNA COVID-19	ESTADO ASMATICO	0.034634	6.725851
10	VACUNA COVID-19	FIEBRE (MEDIDA O REPORTADA)	0.046818	6.122991
11	USO DE MASCARILLA EN LOS ULTIMOS 7 DIAS	SINDROME DE DISTRESS RESPIRATORIO	0.039215	10.073020
12	DX PREVIO DE COVID-19	ENFERMEDAD CARDIACA CONGENITA	0.032879	6.829869
13	DX PREVIO DE COVID-19	ENFERMEDADES NEUROLOGICAS	0.046226	6.148407
14	DX PREVIO DE COVID-19	ES FUMADOR?	0.033807	6.774185
15	PREMATURIDAD	SINDROME EMETICO/INTOLERANCIA ORAL	0.049945	5.993682
16	PREMATURIDAD	EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA	0.034081	6.758024
17	ENFERMEDAD CARDIACA CONGENITA	SINDROME EMETICO/INTOLERANCIA ORAL	0.030968	6.949584
18	DISPLASIA BRONCOPULMONAR	SINDROME EMETICO/INTOLERANCIA ORAL	0.041791	6.350164
19	DISPLASIA BRONCOPULMONAR	FIEBRE (MEDIDA O REPORTADA)	0.034804	6.716043
20	DISPLASIA BRONCOPULMONAR	TIPO DE MUESTRA	0.049777	9.498565
21	ASMA-CONDICIONPREEXISTENTE	SINDROME EMETICO/INTOLERANCIA ORAL	0.036888	6.599733
22	ASMA-CONDICIONPREEXISTENTE	EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA	0.030793	6.960930
23	ASMA-CONDICIONPREEXISTENTE	TIPO VACUNA 2	0.034105	13.625967
24	ENFERMEDADES NEUROLOGICAS	SINDROME EMETICO/INTOLERANCIA ORAL	0.046741	6.126276
25	ENFERMEDADES NEUROLOGICAS	DIARREA/NAUSEAS VOMITOS	0.044065	6.244173
26	ENFERMEDADES NEUROLOGICAS	TIPO VACUNA 1	0.040091	13.191701
27	OBESIDAD	SINDROME EMETICO/INTOLERANCIA ORAL	0.048724	6.043170
28	OBESIDAD	EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA	0.046554	6.134291
29	ES FUMADOR?	DOLOR MUSCULAR	0.049227	6.022638
30	ES FUMADOR?	DOLOR O PRESION PERSISTENTE EN EL PECHO	0.032257	6.868052
31	CONTACTO O CRIANZA DE AVES?	CONTACTO EN LOS ULTIMOS 14 DIAS CON UNA PERSON...	0.033576	8.698762
32	ADMINISTRACION ANTIVIRAL SITEMICO DE LA ENFERM...	INFECCION DE VIAS RESPIRATORIAS AGUDA/REAGUDIZADA	0.033807	6.774185
33	ADMINISTRACION ANTIVIRAL SITEMICO DE LA ENFERM...	DOLOR MUSCULAR	0.049227	6.022638
34	NEUMONIA	FATIGA	0.040082	4.214427
35	SEPSIS	TIPO VACUNA 1	0.033088	8.731093
36	SHOCK SEPTICO	RESULTADO DE SARS-COV-2	0.047329	3.933637
37	INSUFICIENCIA RESPIRATORIA AGUDA (IRA)	FALTA DE AIRE	0.037591	4.323447
38	INSUFICIENCIA RESPIRATORIA AGUDA (IRA)	RESULTADO DE INFLUENZA A	0.033214	4.534715
39	INSUFICIENCIA RESPIRATORIA AGUDA (IRA)	NOMBRE DEL PATOGENO	0.046394	7.981599
40	INFECCION DE VIAS RESPIRATORIAS AGUDA/REAGUDIZADA	NUMERO DE DOSIS	0.046233	9.677100
41	SINDROME EMETICO/INTOLERANCIA ORAL	FIEBRE/ESCALOFRIOS	0.046895	3.949136
42	SINDROME EMETICO/INTOLERANCIA ORAL	CONGESTION O SECRECION NASAL	0.038131	4.299157
43	FIEBRE/ESCALOFRIOS	EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA	0.031437	4.628949
44	ESCALOFRIOS	DOLOR DE CABEZA	0.043099	4.091528
45	DOLOR DE GARGANTA O TOS	EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA	0.041599	4.151459
46	DOLOR DE GARGANTA O TOS	NUMERO DE DOSIS	0.035251	10.327986
47	NAUSEAS O VOMITOS	DIARREA	0.046575	3.960689
48	DIARREA	TOS	0.049008	3.875090
49	DIARREA	TIPO VACUNA 3	0.034281	8.652818
50	FATIGA	MADRE RECIBIO VACUNA COVID-19?	0.037299	6.577596
51	CONGESTION O SECRECION NASAL	FALTA DE AIRE	0.030606	4.674944
52	DIFICULTAD PARA RESPIRAR	COLOR DE LA PIEL/ LABIOS O UNIAS PALIDO/ GRIS ...	0.046363	3.968361
53	CONFUSION O CAMBIO EN EL ESTADO MENTAL	EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA	0.048794	3.882421
54	EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA	TIPO VACUNA 1	0.032580	8.765253

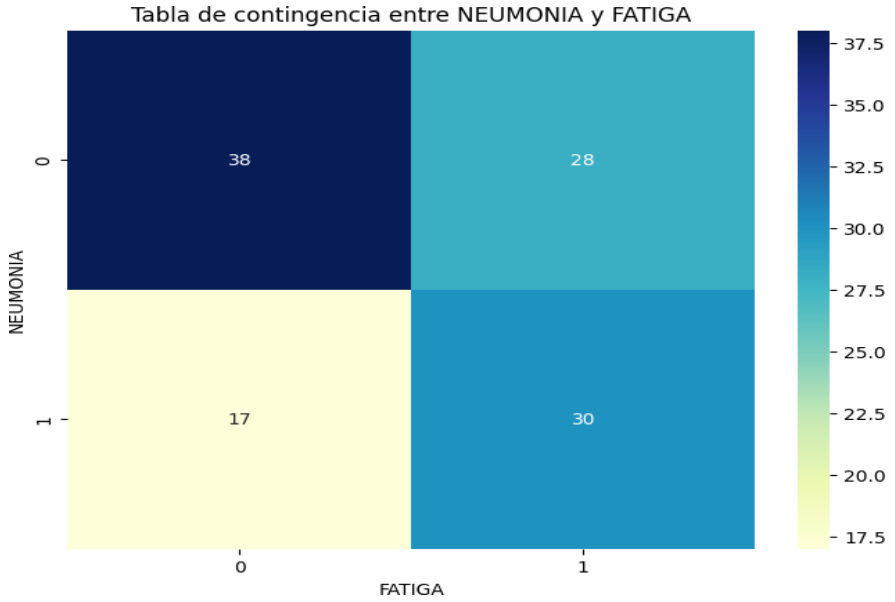
Fuente: Autores de este documento.

3.2.2.2 Tabla de contingencia para mostrar frecuencia

Con los resultados de Chi-cuadrado y el valor de p-value, se procedió a crear la tabla de contingencia entre las variables que mostraron una relación significativa. El rango de p-value considerado fue de 0.03 a 0.05, dado que en este intervalo se obtuvo que las variables tenían una relación más confiable. A continuación, se procede a mostrar la tabla de contingencia entre las variables de *NEUMONIA* y *FATIGA*:

Tabla 4

Tabla de contingencia entre NEUMONIA y FATIGA



Fuente: Autores de este documento.

En la tabla 4 se presenta la distribución conjunta de la variable *NEUMONIA* y *FATIGA*. Los valores de cada celda representan el número de observaciones que corresponde a la combinación de las variables mencionadas.

Se observan las siguientes frecuencias:

- 38 pacientes no tienen ni *NEUMONIA* ni *FATIGA*.
- 28 pacientes no tienen *NEUMONIA*, pero si presentan *FATIGA*.
- 17 pacientes tienen *NEUMONIA*, pero no presentan *FATIGA*.
- 30 pacientes tienen *NEUMONIA* y *FATIGA*.

Esta asociación se interpreta como una evidencia de que la *FATIGA* es más común con los pacientes que presentan *NEUMONIA*.

Con el análisis de las variables categórica se concluye que las variables con mayor asociación entre si son las variables que se encuentran dentro de diagnósticos, síntomas y palabras claves.

3.2.2.4 Análisis multivariable de variables continuas

La matriz de correlación con el método Pearson y el pairplot de las variables continuas otorgó los siguientes resultados:

Figura 17

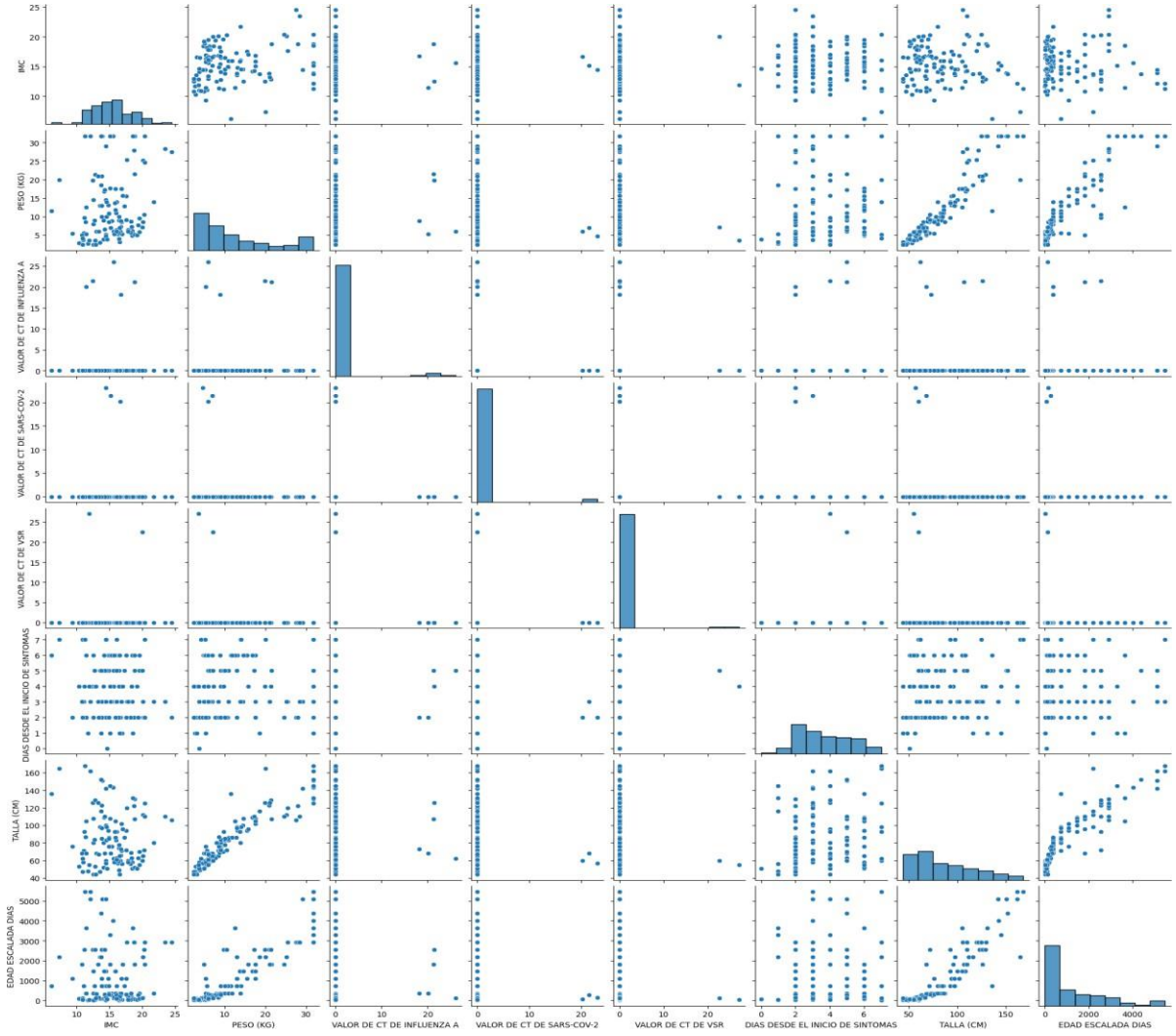
Matriz de correlación método de Pearson

	IMC	PESO (KG)	VALOR DE CT DE INFLUENZA A	VALOR DE CT DE SARS-COV-2	VALOR DE CT DE VSR	DIAS DESDE EL INICIO DE SINTOMAS	TALLA (CM)	EDAD ESCALADA DIAS
IMC	1.000000	0.162044	-0.021403	0.002536	0.010958	-0.126167	-0.140718	-0.061350
PESO (KG)	0.162044	1.000000	-0.008314	-0.123607	-0.109666	0.060254	0.928930	0.900909
VALOR DE CT DE INFLUENZA A	-0.021403	-0.008314	1.000000	-0.03521	-0.028545	0.004370	0.003773	-0.021153
VALOR DE CT DE SARS-COV-2	0.002536	-0.123607	-0.035212	1.000000	-0.022042	-0.129936	-0.129052	-0.119045
VALOR DE CT DE VSR	0.010958	-0.109666	-0.028545	-0.022042	1.000000	0.059245	-0.122926	-0.105167
DIAS DESDE EL INICIO DE SINTOMAS	-0.126167	0.060254	0.004370	-0.129936	0.059245	1.000000	0.169967	0.067508
TALLA (CM)	-0.140718	0.928930	0.003773	-0.129052	-0.122926	0.169967	1.000000	0.892095
EDAD ESCALADA DIAS	-0.061350	0.900909	-0.021153	-0.119045	-0.105167	0.067508	0.892095	1.000000

Fuente: Autores de este documento.

Figura 18

Imagen de pairlot de variables continuas



Fuente: Autores de este documento.

La matriz de correlación y el pairplot mostrado en la Figura 18 muestran las relaciones entre las variables continuas. A continuación, se detallan las observaciones importantes:

***EDAD* vs *PESO (KG)* y *TALLA (CM)*:**

Se observa un patrón lineal ascendente, lo que indica correlaciones positivas fuertes que fueron observadas en la matriz de correlación de la Figura 18. Estas tendencias lineales sugieren que al aumentar *EDAD*, tienden a aumentar *PESO (KG)* y *TALLA (CM)*.

***PESO (KG)* vs *TALLA (CM)*:**

Los puntos entre estas dos variables tienden a formar una línea ascendente, mostrando que a mayor *TALLA (CM)*, mayor *PESO (KG).* Además, la matriz de correlación nos indica un valor de 0.92 que refleja una correlación positiva.

3.2.2.5 *Normalización del Dataset*

Como se mencionó en el anterior capítulo, para decidir que técnica de normalización elegir, se usó la prueba de Shapiro-Wilk. Luego de ser aplicada al dataset, se obtuvieron valores menores al nivel de significancia de 0.05, lo cual indicó que el dataset no seguía una distribución normal y por ello se optó por utilizar MinMaxScaler.

En la Figura 19 se muestra el dataset con MinMaxScaler aplicado.

Figura 19

MinMaxScaler aplicado al dataset

	SEXO	PESO (KG)	TALLA (CM)	VACUNA INFLUENZA 2021	VACUNA INFLUENZA 2022	VACUNA INFLUENZA 2023	VACUNA COVID-19	USO DE MASCARILLA EN LOS ULTIMOS 7 DIAS	DX PREVIO DE COVID-19	PREMATURIDAD	...
count	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	...
mean	0.424779	12.547168	86.185841	0.840708	0.929204	0.876106	0.743363	1.061947	0.061947	1.017699	...
std	0.496511	8.950299	31.580269	0.911889	0.863101	0.867393	0.874017	1.577086	0.242133	0.834334	...
min	0.000000	2.400000	44.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...
25%	0.000000	5.900000	60.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...
50%	0.000000	9.000000	78.000000	0.000000	1.000000	1.000000	0.000000	0.000000	0.000000	1.000000	...
75%	1.000000	17.300000	107.000000	2.000000	2.000000	2.000000	2.000000	2.000000	0.000000	2.000000	...
max	1.000000	31.800000	168.000000	2.000000	2.000000	2.000000	2.000000	4.000000	1.000000	2.000000	...

Fuente: Autores de este documento.

3.3 Análisis de tendencias y clustering

3.3.1 Análisis de series temporales

Durante el análisis de series temporales, se utilizó el dataset sin normalización. Esto se debe a que las fechas no pueden ser normalizadas, y en este tipo de análisis, las fechas son un factor importante por lo que fue esencial mantenerlas en su forma original.

En la siguiente figura se muestra parte el dataset usado:

Figura 20

Sección de dataset

	EDAD ANIOS	SEXO	PESO (KG)	VACUNA INFLUENZA 2021	VACUNA INFLUENZA 2022	VACUNA INFLUENZA 2023	VACUNA COVID-19	USO DE MASCARILLA EN LOS ULTIMOS 7 DIAS	DX PREVIO DE COVID-19	PREMATURIDAD	...	RESULTADOHM	AISLAMIENTO DE	NOMBRE DEL PATOGENO
0	3.000000	0	15.5	1	2	2	0	0	0	0	...	0	0	0
1	4.000000	0	14.5	1	1	1	1	0	0	0	...	0	0	0
2	0.250000	0	6.2	2	2	2	2	0	0	1	...	0	0	0
3	11.000000	0	31.8	2	0	0	0	3	0	1	...	0	0	0
4	14.000000	0	31.8	0	0	0	1	0	1	0	...	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
175	0.250000	0	6.0	2	2	2	2	2	0	2	...	0	0	0
176	3.000000	0	5.4	0	1	0	0	0	0	0	...	0	0	0
177	0.333333	1	2.5	2	2	2	2	0	0	1	...	0	0	0
178	0.083333	0	3.6	2	2	2	2	0	0	2	...	0	0	0
179	4.000000	0	22.2	0	1	0	0	0	0	2	...	0	0	0

180 rows × 73 columns

Fuente: Autores de este documento.

Se utilizaron variables con relaciones significativas, las cuales fueron mencionadas en la sección anterior.

3.3.1.1 Descomposición de series temporales

Se estableció un periodo de 30 días para cada serie temporal, extrayendo así el componente estacional y calculando su desviación estándar para evaluar la variabilidad con respecto a la media.

Con estos resultados, se procedió directamente a realizar la descomposición de series temporales usando las variables que estaban dentro de síntomas, diagnósticos y palabras claves. Esto proporcionó información crucial para las futuras predicciones y análisis.

En el análisis para medir la estacionalidad de las variables se usó la función seasonal_descompose, la cual servirá para la visualización de dichos resultados.

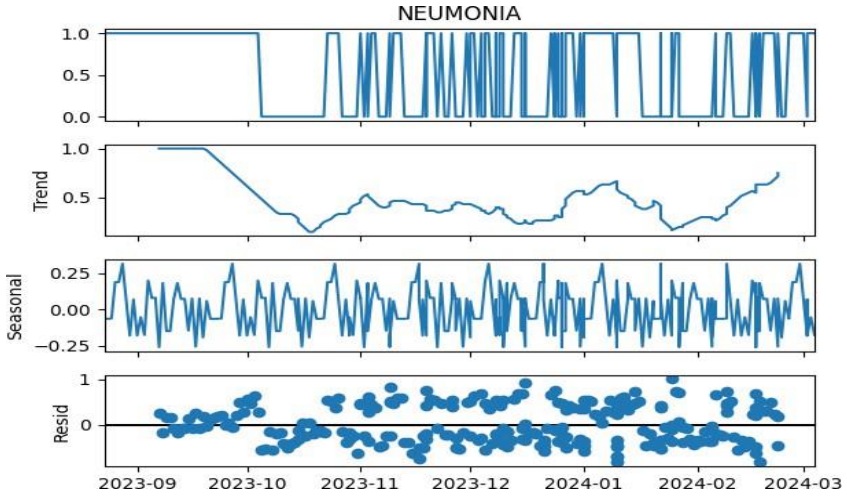
Se creó un dataFrame temporal el cual está compuesto por un rango de fechas completas, es decir tomando la fecha de inicio y la última fecha de la variable *FECHA DE INICIO DE SINTOMAS*. Con esto se creó un rango completo de las fechas y algunos valores de las variables quedaron con valores faltantes. Para solucionar esto y rellenar el dataframe

temporal, se usaron los métodos de forward fill (pandas via NumFOCUS, Inc. Hosted by OVHcloud. , 2024) que llena los valores propagando la última observación valida a la siguiente fila y backward fill el cual llena los valores mediante la siguiente observación válida para llenar el vacío (pandas via NumFOCUS, Inc. Hosted by OVHcloud. , 2024). Con los métodos mencionados se procedió a analizar las variables que se encuentran en la sección de diagnósticos, síntomas y palabras claves. A continuación, se muestra una parte de las variables que pasaron por el proceso de descomposición.

NEUMONIA: Se observó una disminución general de casos a lo largo del tiempo de estudio. Además, Se proyecta una reducción continua con picos identificados en los meses de octubre, noviembre y enero como se observa en la Figura 21.

Figura 21

Descomposición de NEUMONIA



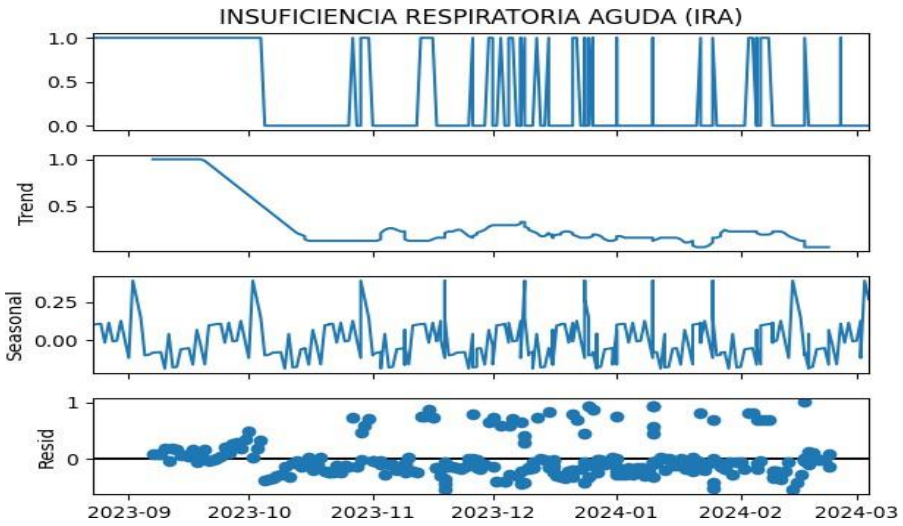
Fuente: Autores de este documento.

INSUFICIENCIA RESPIRATORIA AGUDA(IRA):

En la Figura 22, hay una tendencia inicial que disminuye en el periodo estudiado, seguida de una estabilización con ligeros aumentos estacionales.

Figura 22

Descomposición de INSUFICIENCIA RESPIRATORIA AGUDA(IRA)



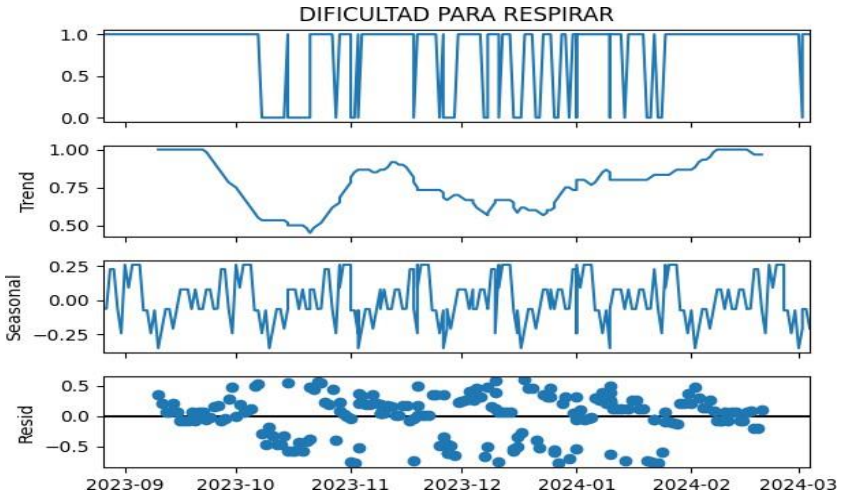
Fuente: Autores de este documento.

DIFICULTAD PARA RESPIRAR:

En la Figura 23, se muestra un decremento inicial hasta noviembre, seguido de un aumento. Además, las predicciones indican un aumento en marzo, seguido de una disminución hasta junio y un incremento posterior.

Figura 23

Descomposición de DIFICULTAD PARA RESPIRAR



Fuente: Autores de este documento.

La mayoría de los síntomas muestran patrones estacionales y fluctuaciones a lo largo del año. En el caso de síntomas como *FIEBRE, ESCALOFRIOS, DOLOR MUSCULAR, DOLOR DE CABEZA, NAUSEAS O VOMITOS, DIARREA, TOS, FALTA DE AIRE, DIFICULTAD PARA RESPIRAR* y síntomas relacionados con problemas respiratorios se tienen variaciones significativas a lo largo de las estaciones.

Generalmente, tienden a aumentar durante los meses más fríos del año y pueden tener picos en invierno.

3.3.1.2 Pronósticos de series temporales

Con el análisis de las descomposiciones de series temporales para verificar cuales variables presentaban una tendencia marcada, se realizó una evaluación exhaustiva de los modelos ARIMA, FORESCAST, PMDARIMA, STATSMODELS y SKFORESCAST. Se observó que las predicciones más precisas fueron brindadas por los modelos ARIMA, y FORESCAST.

Los datos para el análisis se re-muestraron a una frecuencia mensual y se rellenaron los valores faltantes, esto se lo realizó con la función resample(), que es un método para la conversión de frecuencia y muestreos de las series temporales (pandas via NumFOCUS, Inc. Hosted by OVHcloud. , 2024); con esto mantendremos los índices en el formato de fecha, pero en intervalos de meses. Para que las predicciones sean seguras al momento del análisis, se realizó un cálculo de predicción de error con las métricas MAE (Mean Absolute Error), MSE (Mean Square Error) y RMSE (Root Mean Squeare Error). Se usó estos cálculos dado a que MSE y RMSE, penalizan grandes errores de predicción, pero RMSE es preferido por tener las mismas unidades que la variable dependiente a analizar (Chugh, Medium, 2020). Los valores bajos de estos cálculos indican una mayor precisión del modelo.

Los valores de cálculo de error calculados no fueron mayores a 0.3, lo que sugiere una buena precisión en los modelos evaluados.

Se utilizaron los híper-parámetros de los modelos ARIMA(p,d,q) para poder moldear condiciones seleccionadas. Estos híper-parámetros fueron obtenidos con la ayuda de la librería Mango, la cual permite optimizar la configuración de los modelos mediante un proceso iterativo y se va ajustando a los datos posibles.

Los híper-parámetros obtenidos con la librería Mango fueron los mostrados en la siguiente tabla:

Tabla 5

Hiper-parámetos usados para el modelo ARIMA

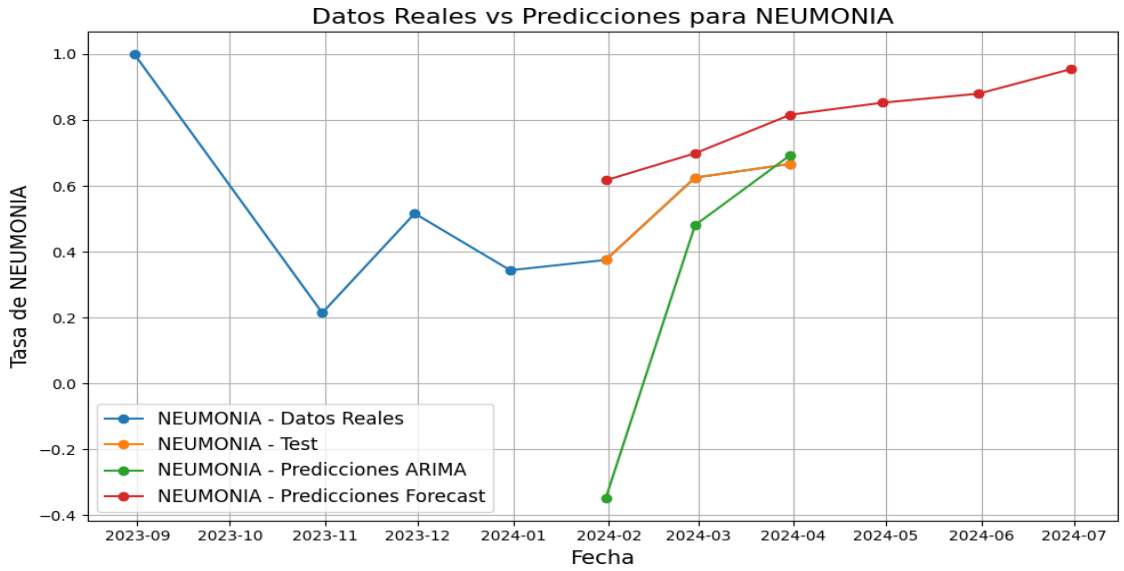
Variables Analizadas	**Híper-parámetros (Números de lags, Número de Repetición, Tamaño de la media del modelo)**
NEUMONIA, DOLOR DE GARGANTA O TOS, DIARREA, TOS, FALTA DE AIRE, SINDROME DE DISTRESS RESPITATORIO AGUDO (SDRA)	2,2,1
SHOCK SEPTICO, INFECCION DE VIAS RESPIRATORIAS AGUDA REGUDIZADA, SINDROME DE DISTRESS RESPIRATORIO	2,0,2
SEPSIS, DIFICULTAD PARA RESPIRAR	2,0,1
ESTADO ASMATICO, INSUFICIENCIA RESPIRATORIA AGUDA (IRA)	1,0,2
BRONQUIOLITIS AGUA	2,0,2

Las predicciones fueron realizadas para mostrarse en un rango de seis meses, dado que fueron las óptimas, por la cantidad de datos que se posee en el dataset. Con estos cálculos previos, se procedió a visualizar las predicciones futuras de cada enfermedad.

Con respeto al caso de *NEUMONIA*, en la Figura 24 muestra una tendencia creciente a lo largo del año. Esta tendencia se relaciona con la estacionalidad marcada que se observa en la Figura 21, donde se identifica patrones repetitivos en la estacionalidad. La relación entre ambas figuras nos sugiere con respecto a los incrementos observados en la Figura 24 pueden estar influenciados por los ciclos estacionales presentados en la Figura 21.

Figura 24

Predicciones de NEUMONIA



Fuente: Autores de este documento.

3.3.2 Modelos basados en aprendizaje automático

3.3.2.1 Análisis del modelo LSTM

Para poder utilizar un modelo LSTM fue necesario estructurar el dataset de una forma diferente en comparación a los análisis anteriormente realizados. En este caso, las muestras del dataset fueron agrupadas de acuerdo con la media de casos de pacientes positivos en campos de diagnósticos y síntomas en periodos de semanas de cada año; esta característica *AÑO_SEMANA* se usó como columna principal para el dataset. De esta forma el dataset

estaba listo para usarse en el modelo LSTM, el cual está en capacidad de identificar dependencias temporales (Muhammad, 2023).

Dado que el dataset original contenía características irrelevantes para realizar predicciones temporales, junto con características que en análisis anteriores demostraron no ser de importancia, estas tuvieron que ser eliminadas. Con esto, el dataset final contó con las siguientes características: *NEUMONIA, FIEBRE (MEDIDA O REPORTADA), DOLOR DE GARGANTA O TOS, NAUSEAS O VOMITOS, FATIGA, CONGESTION O SECRECION NASAL, TOS, FALTA DE AIRE, DIFICULTAD PARA RESPIRAR* y *EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA.*

A continuación, se presenta una tabla con los modelos evaluados.

Tabla 6

Modelos LSTM evaluados

Modelo	Descripción	Métricas
Modelo LSTM 1	Contaba con dos capas LSTM, cada una con 50 neuronas, y dos capas Densas de salida.	Métricas con respecto a datos de entrenamiento: • Media MAE = 0.27 • Media RMSE = 0.33 • Media R2 = 0.004 Métricas con respecto a datos de prueba: • Media MAE = 0.27 • Media RMSE = 0.29 • Media R2 = -1.93
Modelo LSTM 2	Se usó Randomized Search con el fin de encontrar los mejores parámetros para el modelo y así obtener un mejor rendimiento.	Métricas con respecto a datos de entrenamiento: • Media MAE = 0.33 • Media RMSE = 0.39 • Media R2 = -0.91 Métricas con respecto a datos de prueba: • Media MAE = 0.41 • Media RMSE = 0.44 • Media R2 = -4.80
Modelo LSTM 3: Modelo LSTM simple	Se caracteriza por tener una estructura más sencilla en comparación a los modelos anteriores. En este caso se usó como dato de entrada la	Simple LSTM MAE: 0.01 Simple LSTM RMSE: 0.01

	combinación de valores de entrenamiento.

Como se puede observar, los valores indican que el último modelo actúa de manera más precisa con estos datos, en comparación a los otros modelos LSTM. Esto, debido principalmente a la cantidad de datos existentes en el dataset, la cual puede ser la causa del sobreajuste en los otros modelos LSTM.

Los modelos LSTM fueron basados en modelos elaborados por el Doctor Enrique Peláez Jarrín (Peláez, 2024).

3.3.3 Análisis de clustering

Antes de explicar los resultados obtenidos con cada algoritmo de clustering, es importante conocer acerca del Análisis de Componentes Principales (PCA). El análisis PCA es un método de reducción de dimensionalidad y aprendizaje automático usado para simplificar un dataset extenso, mientras se mantienen las tendencias y patrones significantes del dataset original (Jaadi, 2024).

Este método es generalmente usado antes de la implementación de los algoritmos de clasificación y regresión, con el fin de que la exploración de los datos sea más efectiva para los algoritmos de aprendizaje automático; además, permiten visualizar las agrupaciones. Tal como indica Jaadi (2024), la reducción de dimensionalidad viene a expensas de perder precisión. Por ello, se realizó un análisis de cuántos componentes principales usar, con el fin de perder la menor cantidad de información posible.

Hay que considerar que los componentes dados por PCA están ordenados por la varianza que explican (Barreto, 2024), esto quiere decir que cada componente representa cierta cantidad de información con respecto al dataset original. Para seleccionar que cantidad de componentes principales usar, se decidió preservar el 95% de la varianza. Se tomó esta

decisión ya que según las recomendaciones de Barreto (2024), en aplicaciones generales, un buen valor se encuentra entre el 80% y el 95%.

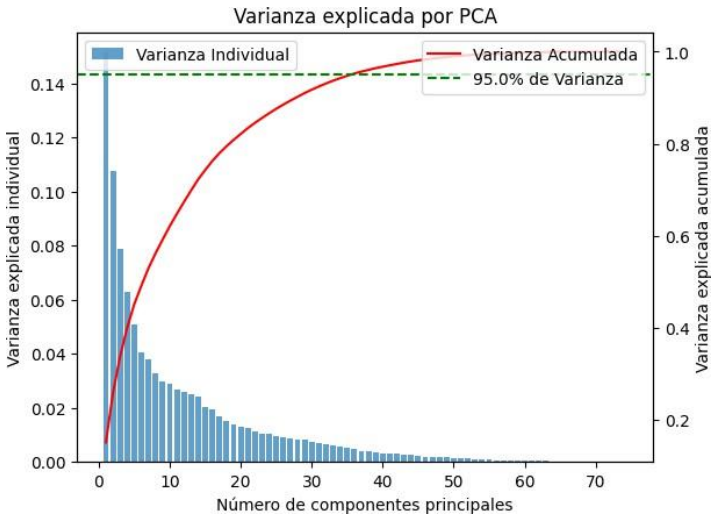
Para conocer las principales diferencias de cada agrupación creada por los distintos algoritmos, se realizó dos tipos de análisis: a) Análisis para variables categóricas; y, b) Análisis para variables continuas. Para las variables categóricas, se identificaron las características predominantes por medio de gráficos de barras, los cuales explican las frecuencias de sus valores. Por otro lado, en las variables continuas, se usaron las medias de las variables pertenecientes a cada clúster, las cuales se compararon con las medias de los demás clústeres.

3.3.3.1 K-Medoids con 95% de varianza.

A través de este análisis la cantidad de componentes principales que mantenían el 95% de la varianza total fue de 36 componentes. La siguiente figura presenta la varianza individual y acumulada del análisis PCA.

Figura 25

Varianza individual y acumulada del análisis PCA (95% de varianza)



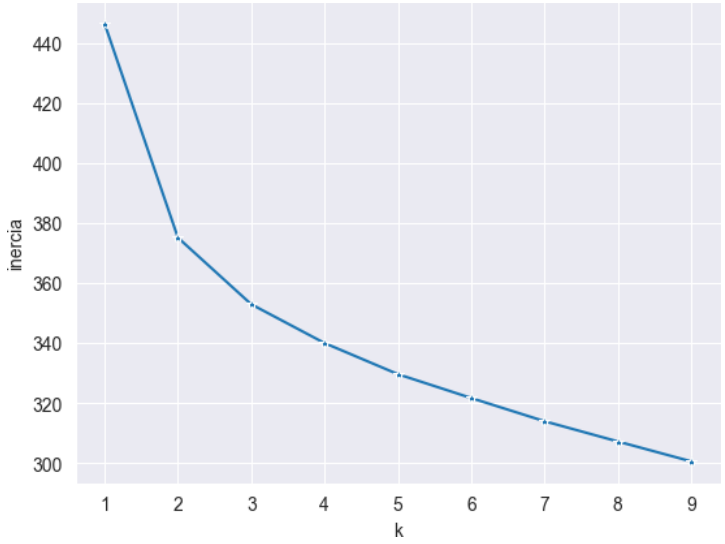
Fuente: Autores de este documento.

Dado que K-Medoids necesita como híper-parámetro el número óptimo de clústeres a crear, se utilizaron las técnicas del Codo y del Coeficiente de Silueta con el fin de encontrar este valor.

Al aplicar el método del Codo, se observó que el número óptimo de clústeres era 2, tal como se puede apreciar en la siguiente figura.

Figura 26

Método del Codo para K-Medoids (95% de varianza)

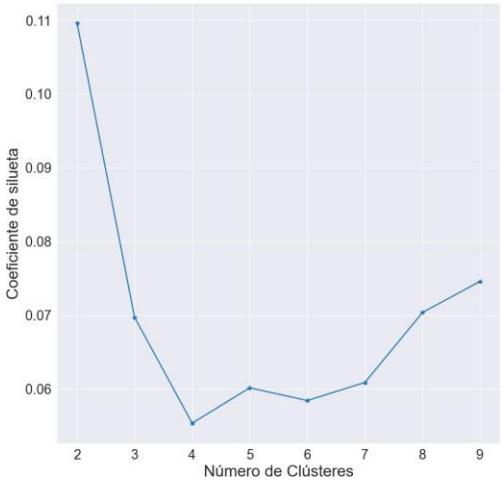


Fuente: Autores de este documento.

Luego, al aplicar el método del Coeficiente de Silueta también se obtuvo que el número óptimo de clústeres era 2, por ser el mayor valor del coeficiente entre todos los números probados en un rango del 2 al 9. Esto se puede apreciar en la siguiente figura.

Figura 27

Método del Coeficiente de Silueta para K-Medoids (95% de varianza)



Fuente: Autores de este documento.

Nota. Valor de coeficiente de silueta más alto: 0.11.

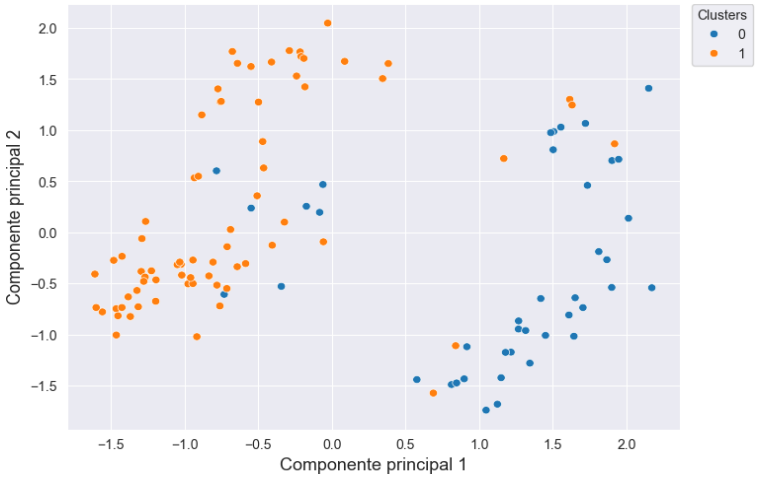
Con el número óptimo de clústeres determinado, se procedió a aplicar el algoritmo K-Medoids. A continuación, se realiza el análisis de características de cada agrupación.

3.3.3.1.1 Análisis de clústeres.

La siguiente figura presenta la distribución de los clústeres obtenidos.

Figura 28

Clústeres obtenidos con 95% de varianza



Fuente: Autores de este documento.

El clúster 0 se caracteriza por contener a pacientes de una edad más alta, los cuáles no tienen un diagnóstico predominate ya que la mayoría de estos tienen este campo como negativo. Además, en su mayoría no se tiene información acerca de sus condiciones pre-existentes. Con respecto a los síntomas, la mayoría de los pacientes presentan *FIEBRE (MEDIDA O REPORTADA)*, *DOLOR DE GARGANTA O TOS*, *NAUSEAS*, *CONGESTION O SECRECION NASAL*, *TOS* y *DIFICULTAD PARA RESPIRAR*.

Por otro lado, el clúster 1 se caracteriza por contener pacientes de edad más baja, y en su mayoría presentan un diagnóstico de *NEUMONIA*. Con respecto a los síntomas, estos presentan *FIEBRE (MEDIDA O REPORTADA)*, *DOLOR DE GARGANTA O TOS*, *FATIGA*, *CONGESTION O SECRECION NASAL*, *TOS*, *FALTA DE AIRE*, *DIFICULTAD PARA RESPIRAR* y *EVIDENCIA CLINICA O RADIOLOGICA DE NEUMONIA*.

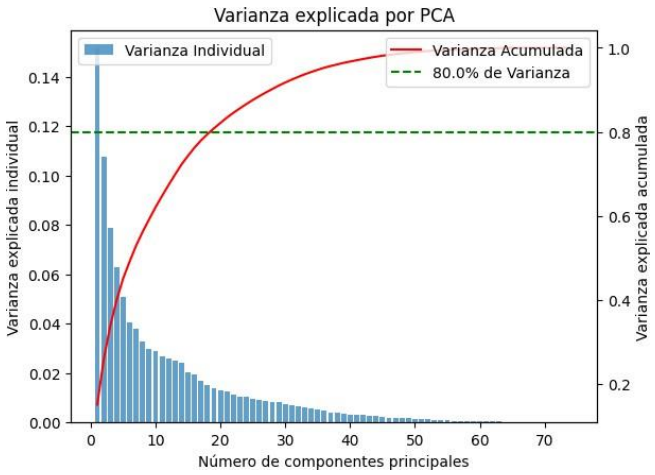
Dado que la única diferencia relevante entre estas dos agrupaciones es la presencia o no de diagnóstico de *NEUMONIA*, se procedió a realizar más experimentos con el fin de encontrar clústeres con mayor cantidad de diferencias.

3.3.3.2 K-Medoids con 80% de varianza.

Para este experimento se optó por usar un 80% de varianza acumulada. Esta configuración dio como resultado 19 componentes principales. La siguiente figura presenta la varianza individual y acumulada del análisis de PCA.

Figura 29

Varianza individual y acumulada del análisis PCA (80% de varianza)

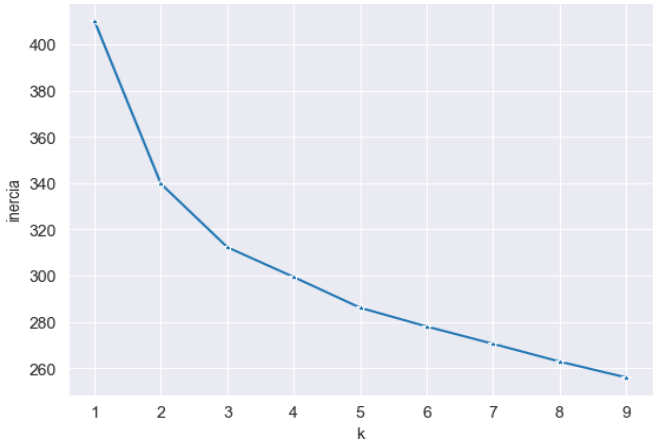


Fuente: Autores de este documento.

Tal como en la anterior prueba, también se aplicaron los métodos del Codo y del Coeficiente de Silueta para determinar el número óptimo de clústeres a crear. Ambos métodos determinaron que 3 era la cantidad de clústeres óptimo; esto se puede apreciar en las siguientes dos figuras.

Figura 30

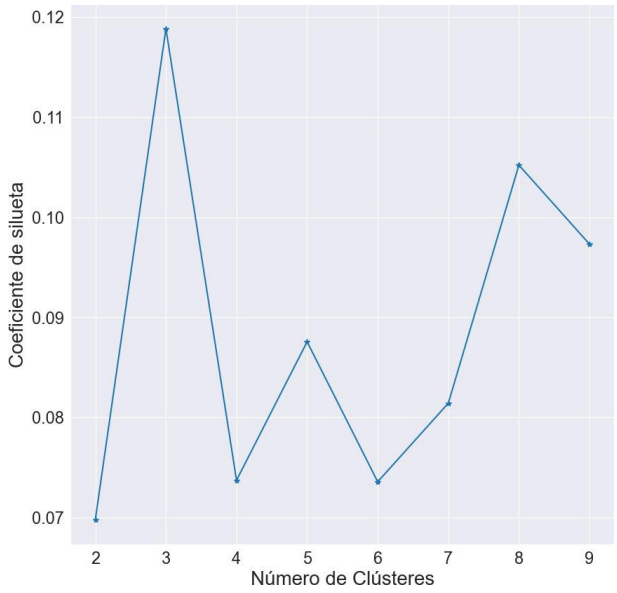
Método del Codo para K-Medoids (80% de varianza)



Fuente: Autores de este documento.

Figura 31

Método del Coeficiente de Silueta para K-Medoids (80% de varianza)



Fuente: Autores de este documento.

Nota. Valor de coeficiente de silueta más alto: 0.12.

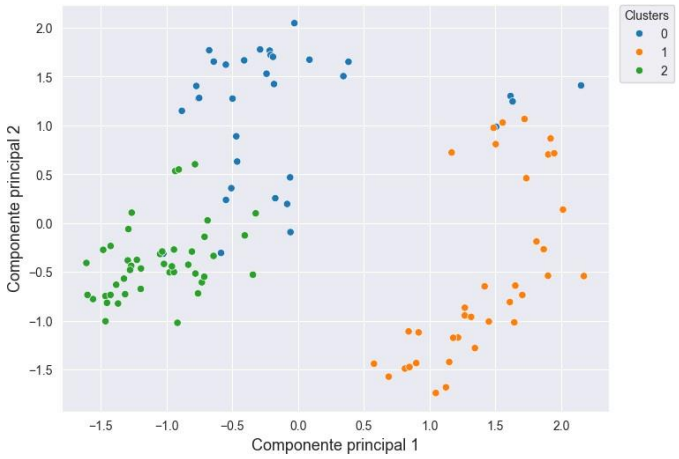
Con este híper-parámetro se procedió a realizar el análisis de clústeres, con el fin de identificar las diferencias existentes.

3.3.3.2.1 Análisis de clústeres

La siguiente figura presenta la distribución de los clústeres obtenidos.

Figura 32

Clústeres obtenidos con 80% de varianza



Fuente: Autores de este documento.

El clúster 0 se caracteriza por contener pacientes que presentan pocos síntomas, sin un diagnóstico que predomine. Además, en su mayoría no se conoce si recibieron vacunas. Todos en este grupo dieron negativo para la prueba VSR y fue el grupo en el que los pacientes acudieron al hospital en la menor cantidad de días. En este clúster la mayoría de paciente cuentan con *PREMATURIDAD* como condición pre-existente, lo cual invita a la reflexión debido a que, según la Academia Americana de Pediatría, los bebés prematuros tienen mayor incidencia en discapacidades y enfrentan tasas más bajas de supervivencia (American Academy of Pediatrics, 2024).

En el clúster 1, a pesar de que la mayoría de los pacientes no recibieron las vacunas contra la Influenza A, ni contra el SARS-COV-2, no se presenta un diagnóstico predominante. Por otro lado, esta agrupación se caracteriza por presentar una mayor cantidad de síntomas en comparación al clúster 0. Además, comparado con el anterior clúster, aquí se presentan pocos casos positivos de VSR.

Por último, el clúster 2 es la única agrupación donde la mayoría de los pacientes presentan un diagnóstico predominante positivo de *NEUMONIA* lo cual viene acompañado por una mayor cantidad de síntomas en comparación con las otras agrupaciones. Muchos de los pacientes en este grupo sí recibieron algún antibiótico desde el inicio de síntomas de la enfermedad. En esta agrupación también, en su mayoría, los pacientes no recibieron ningún tipo de vacuna. Todos los pacientes resultaron negativos ante la prueba VSR y en su mayoría a SARS-COV-2 e Influenza A. También, esta agrupación contiene pacientes los cuales demoraron en acudir al centro de salud en comparación a los otros grupos.

Debido a las diferencias existentes en estos clústeres, se tomó la decisión de experimentar con los algoritmos faltantes usando el 80% de la varianza acumulada.

3.3.3.3 Otros algoritmos

La siguiente tabla muestra los resultados obtenidos con los algoritmos faltantes.

Tabla 7

Descripción de experimentos con otros algoritmos de agrupación

Algoritmo	**Métrica (Coeficiente de Silueta)**	**Descripción de agrupaciones**
K-Means	0.18	Las agrupaciones obtenidas con el algoritmo K-Means presentan patrones muy parecidos a los obtenidos por el algoritmo K-Medoids con 95% de varianza donde su diferencia más clara es el diagnóstico predominante de NEUMONIA.
Algoritmo de clustering Jerárquico Aglomerativo	0.17	Los clústeres creados son similares a las generados anteriormente por otros algoritmos, lo cual indica que no existen varias características relevantes que los diferencien.
Algoritmo DBSCAN	No aplica	Los clústeres presentan como diferencias significativas el diagnóstico del Estado Asmático y la cantidad de síntomas, sin embargo, no presentan entre ellos más características relevantes.

3.3.3.4 Algoritmo escogido para Aplicación web

Luego del análisis de los algoritmos, se tomó la decisión de usar el algoritmo K-Medoids para el desarrollo de clústeres en la aplicación web debido a que los clústeres creados por éste presentaban características claras y relevantes que permitían establecer diferencias entre ellos. A pesar de que la métrica de coeficiente de silueta para este algoritmo en específico presentó un valor de 0.12 indicando que las observaciones están muy cerca de la frontera entre las agrupaciones (Ayala, s.f.), esto no indica que la calidad de los clústeres es baja. Para entender esto, es importante mencionar que el coeficiente de silueta supone que la distribución de los datos es adecuada para agrupar. Sin embargo, dada la forma irregular que presentan las agrupaciones al usar dos componentes principales, el valor del coeficiente de silueta se ve afectado y no termina reflejando la calidad de los clústeres (Mastery, 2024). Además, resulta importante mencionar que un valor de coeficiente de silueta inferior a 0

indicaría que los agrupamientos son deficientes (Nair, 2023), sin embargo, al presentarse un valor positivo, las agrupaciones resultan ser aceptables.

3.4 Resultados de la aplicación web

3.4.1 Funcionalidad

El médico puede ingresar al aplicativo web por medio de credenciales que serán brindadas por el administrador. Una vez dentro, puede cargar de 4 a 5 archivos ".pdf". Estos archivos deben tener los nombres de "Enrolamiento", "Preselección", "Resultado Multiplex", "Toma de muestra" o "Colecta de muestra" y "Hemocultivo, caso contrario el sistema no le permitirá proceder con el procesamiento y análisis de los archivos. A su vez, el contenido de estos archivos debe tratar sobre la sintomatología de un paciente o no se podrá continuar con los siguientes pasos. Si los archivos son adecuados, el sistema procede a extraer la información. Una vez extraída se elaboran predicciones de tendencias y se asigna el paciente a un clúster específico junto con un valor de confianza, el cual es 86.96%. Estos resultados se muestran por pantalla y el médico tiene la opción de escribir observaciones que considere pertinentes. Si el médico ha terminado con su análisis, este puede seleccionar la opción de "Guardar". Esta acción guarda localmente un archivo ".pdf" con los resultados obtenidos y también lo guarda en una base de datos.

El administrador, por su lado, puede ingresar con credenciales predeterminadas al aplicativo web. Dentro del sistema este puede registrar nuevos médicos, puede revisar un listado de todos los médicos ya registrados anteriormente y, por último, puede revisar un listado de los archivos de resultados creados por cada médico.

3.4.2 Recursos

Uno de los principales recursos usados para el desarrollo de la aplicación fue ChatPDF. Esta herramienta, mediante una cuenta de Google, brinda una llave y permite utilizar su API para enviar hasta 500 mensajes y 5,000 páginas en formato ".pdf" por mes

(Chatpdf, s.f.). El límite de mensajes y páginas puede aumentarse al suscribirse a un plan de pago.

3.4.3 Costos

A continuación, se presentan los costos estimados que tendría la aplicación web realizada. En la sección de "Costos de desarrollo" se presenta el número de desarrolladores que trabajaron en el aplicativo, el sueldo por mes basado en el sueldo de un programador junior en la ciudad de Guayaquil y el número total de meses trabajado en el proyecto completo. En la sección de "Gastos" se presentan viáticos y movilización lo cual indica gastos realizados al movilizarse a ESPOL para reuniones presenciales. La sección "Costos adicionales" presenta la cuenta plus de ChatPDF que fue comprada y usada para la primera extracción de datos de los archivos RedCap. Por último, la suma total de estos costos y gastos indica un total de 7,028 dólares.

Tabla 8

Representación de los costos de la aplicación

Costos de desarrollo	
Número de desarrolladores	2
Sueldo por mes	$ 700
Números de meses trabajados	5
Gastos	
Viáticos/Movilización	$ 20
Costos adicionales	
Cuenta Plus de ChatPDF	$ 8
Costo total	$ 7028

Capítulo 4

4.1 Conclusiones y recomendaciones

En esta sección se presentan las conclusiones obtenidas y varias recomendaciones.

4.1.1 Conclusiones

Luego de los análisis realizados y el desarrollo de la aplicación web donde se presenta para el usuario, en este caso un médico, un conjunto de resultados obtenidos con la ayuda de algoritmos de IA resulta correcto mencionar que este proyecto investigativo logró llegar a las siguientes conclusiones:

1. En la extracción de datos de los archivos Redcap, el desarrollo de un pipeline de detección de checkboxes para identificar los valores de los campos con estos recuadros resultó ser de utilidad en comparación a la API de ChatPDF. La API de ChatPDF demostró ser útil para automatizar la extracción de información netamente textual de archivos en formato ".pdf", pero no logró identificar el valor de campos con checkboxes.
2. Durante el proceso de selección de características, se tuvo que realizar un análisis para identificar aquellas características no relevantes para el proyecto investigativo. Específicamente, con el análisis de variables categóricas se determinaron aquellas características cuyas frecuencias se encontraban desbalanceadas para así poder eliminarlas del dataset final ya que estas podrían crear sesgos al trabajar con los algoritmos de IA usados.
3. Durante el análisis multivariable, el uso de p-value y la prueba Chi-cuadrado fueron fundamentales para evaluar las relaciones entre las variables categóricas. Estos métodos permitieron identificar relaciones significativas, considerando un rango p-value de 0.03 a 0.05, donde se observaron las relaciones más robustas. Esta selección rigurosa de variables garantizó la inclusión de aquellas más relevantes para el análisis de tendencias y predicciones.

4. Las métricas de MAE, MSE y RMSE obtenidas para cada variable analizada muestran valores inferiores a 0.3, lo que sugiere una alta precisión en las predicciones del modelo ARIMA. Esto indica que el modelo tiene un buen desempeño y sus predicciones están muy cerca de los valores reales.
5. Se ha podido determinar que el valor correcto de varianza a preservar al usar la técnica de PCA depende del contexto de la investigación realizada. Esto debido a que al preservar el 90% de la varianza para el algoritmo K-Medoids, las dos agrupaciones creadas presentaron características muy similares. Pero al preservar el 80% de la varianza, el algoritmo generó tres agrupaciones las cuales presentaban características que permitían diferenciarlas entre ellas.
6. Debido a la naturaleza de las muestras brindadas donde el gráfico de las agrupaciones presenta formas irregulares, las métricas para evaluar el rendimiento de los distintos algoritmos de clustering dieron valores cercanos a 0. Sin embargo, por el contexto del proyecto de investigación el rendimiento de estos algoritmos fue evaluado de acuerdo con la cantidad de diferencias presentadas en cada agrupación creada. Por ello, el algoritmo K-Medoids, a pesar de presentar un valor cercano a 0 en la métrica del coeficiente de silueta, generó agrupaciones aceptables las cuales presentan características que permiten diferenciarlas y que además permiten identificar patrones entre la sintomatología de las muestras usadas.
7. La contribución fue la integración de un algoritmo que permite realizar todos los procesos que componen la metodología del proyecto investigativo. Es decir, este algoritmo permite extraer información de sintomatología de archivos ".pdf", realiza la limpieza de estos datos, usa estos nuevos datos para alimentar el modelo ARIMA, predice a que agrupación creada por el algoritmo K-Medoids pertenece el nuevo

paciente y presenta los resultados mediante una interfaz en el aplicativo web para que el médico los utilice como información adicional en la elaboración de diagnósticos.

1.1.2 Recomendaciones

A pesar de la obtención de conclusiones relevantes del proyecto realizado, es importante mencionar los límites que se tuvieron, las mejoras que pueden realizarse y distintas recomendaciones para trabajos futuros. Estos puntos se mencionan a continuación:

- Sería productivo aumentar la cantidad de muestras sobre sintomatología para alimentar con más datos los modelos basados en IA para así tener la posibilidad de identificar nuevos patrones ocultos e incluso aumentar el valor de confianza obtenido.
- La herramienta ChatPDF presenta un límite de uso mensual lo cual, a futuro, puede causar incomodidad a los usuarios al no brindarles la facilidad de obtener resultados en cualquier momento. Para ello, la creación de un pipeline de procesos OCR dedicado a los archivos con dimensiones de anchura = 2339 y altura = 1653 sería útil.
- En términos de escalabilidad para el aplicativo web, se podría hacer uso de distintas tecnologías para optimizar el almacenamiento de los archivos ".pdf" de resultados creados como por ejemplo Firebase o Amazon Simple Storage Service.
- En el prototipo actual, se han utilizado variables centradas en el diagnóstico para infecciones respiratorias. Sin embargo, se pueden explorar otras variables que podrían ofrecer resultados valiosos para mejorar la atención médica. Por ejemplo, incluir la variable *NUMERO DE CAMA* la cual podría ser analizada en relación con el número de pacientes mensuales y esto podría ayudar a identificar patrones en el entorno hospitalario para permitir el desarrollo de planes de contingencia más efectivo para el manejo de brotes de síntomas estacionarios.

Referencias

Instituto de Innovación Digital de las Profesiones. (21 de Febreri de 2023). *INESDI*. Recuperado el 7 de Junio de 2024, de https://www.inesdi.com/blog/analisis-multivariante-que-es-ejemplos/

All, M. (Mayo de 2024). *datacamp*. Recuperado el 15 de junio de 2024, de https://www.datacamp.com/es/tutorial/seaborn-python-tutorial

Amazon Web Services. (s.f.). *Amazon Web Services*. (Amazon Web Services) Recuperado el 26 de Junio de 2024, de https://aws.amazon.com/es/what-is/django/

American Academy of Pediatrics. (18 de Enero de 2024). *HealthyChildren*. Obtenido de HealthyChildren: https://www.healthychildren.org/Spanish/ages-stages/baby/preemie/Paginas/health-issues-of-premature-babies.aspx#:~:text=Los%20beb%C3%A9s%20prematuros%20a%20menudo,tasas%20m%C3%A1s%20bajas%20de%20supervivencia.

ArcGIS Pro 3.3. (s.f.). *ArcGIS Pro*. Obtenido de ArcGIS Pro: https://pro.arcgis.com/es/pro-app/latest/tool-reference/spatial-statistics/how-density-based-clustering-works.htm#:~:text=El%20algoritmo%20DBSCAN%20es%20el,cl%C3%BAsteres%20significativos%20presenten%20densidades%20similares.

Ayala, J. (s.f.). *RPubs*. Obtenido de RPubs: https://rpubs.com/JairoAyala/MP#:~:text=Un%20valor%20del%20coeficiente%20de,est%C3%A9n%20asignadas%20al%20cl%C3%BAster%20err%C3%B3neo.

Barreto, S. (28 de Marzo de 2024). *Baeldung*. (Baeldung) Recuperado el 4 de Julio de 2024, de https://www.baeldung.com/cs/pca-proportion-variance-components-number

Barrios, A. (7 de Febrero de 2023). *Medium*. (Medium) Recuperado el 30 de Mayo de 2024, de https://medium.com/latinxinai/tutorial-del-algoritmo-de-agrupamiento-jer%C3%A1rquico-en-python-b3472c0dd829

Barrios, A. (7 de Febrero de 2023). *Medium*. (Medium) Recuperado el 30 de Mayo de 2024, de https://medium.com/latinxinai/tutorial-del-algoritmo-de-agrupamiento-jer%C3%A1rquico-en-python-b3472c0dd829

Barrios, A. (8 de Agosto de 2023). *Medium*. (Medium) Recuperado el 13 de Junio de 2024, de https://medium.com/latinxinai/tutorial-del-algoritmo-k-means-en-python-d8055751e2f3

Bernzweig, M. (27 de Noviembre de 2023). *Software Oasis*. (Software Oasis) Recuperado el 23 de Junio de 2024, de https://softwareoasis.com/review-of-chatpdf/#:~:text=ChatPDF%20is%20an%20innovative%20AI,language%20conversations%20with%20PDF%20documents.

Blanco, J. (28 de Abril de 2023). *Medium*. (Medium) Recuperado el 29 de Mayo de 2024, de https://jorgeiblanco.medium.com/por-qu%C3%A9-la-normalizaci%C3%B3n-es-clave-e-importante-en-machine-learning-y-ciencia-de-datos-4595f15d5be0

Brownlee, J. (10 de Diciembre de 2020). *Machine Learning Mastery*. Obtenido de https://machinelearningmastery.com/decompose-time-series-data-trend-seasonality/

Centros para el Control y la Prevención de Enfermedades, Centro Nacional de Vacunación y Enfermedades Respiratorias (NCIRD). (20 de Marzo de 2024). *CDC*. (CDC) Recuperado el 15 de Mayo de 2024, de https://espanol.cdc.gov/flu/symptoms/flu-vs-covid19.htm

Chatpdf. (s.f.). *Chatpdf*. Obtenido de Chatpdf: https://www.chatpdf.com/

Chugh, A. (8 de Diciembre de 2020). *Medium*. Obtenido de Medium: https://medium.com/analytics-vidhya/mae-mse-rmse-coefficient-of-determination-adjusted-r-squared-which-metric-is-better-cd0326a5697e

Chugh, A. (8 de Diciembre de 2020). *Medium*. Obtenido de https://medium.com/analytics-vidhya/mae-mse-rmse-coefficient-of-determination-adjusted-r-squared-which-metric-is-better-cd0326a5697e

CODEa UNI. (2021). *CODEa UNI*. Recuperado el 15 de Junio de 2024, de https://www.codeauni.com/comunidad/blog/123/

Coder. (13 de Mayo de 2024). *Código Help*. (Código Help) Recuperado el 15 de Junio de 2024, de https://codigo.help/python/python-para-el-analisis-y-prediccion-de-series-temporales/

Collegelib. (8 de Agosto de 2023). *Collegelib*. (Collegelib) Recuperado el 26 de Junio de 2024, de https://www.collegelib.com/what-is-pythonanywhere/

Cubed. (19 de Marzo de 2024). *Cubed*. Obtenido de Cubed: https://cubed.run/blog/analyzing-the-impact-of-lagged-features-in-time-series-forecasting-a-linear-regression-approach-730aaa99dfd6

DataScientest. (7 de Abril de 2022). *DataScientest*. (DataScientest) Recuperado el 29 de Mayo de 2024, de https://datascientest.com/es/datacleaning-limpieza-de-datos-definicion-tecnicas-importancia-en-data-science#:~:text=La%20limpieza%20de%20datos%20es,)%2C%20para%20poder%20explotarlos%20despu%C3%A9s.

DataScientest. (s.f.). *DataScientest*. (DataScientest) Recuperado el 30 de Mayo de 2024, de https://datascientest.com/es/machine-learning-clustering-dbscan#:~:text=El%20DBSCAN%20es%20un%20algoritmo%20sencillo%20que%20define%20los%20cl%C3%BAsteres,%CE%B5%2Dvecindad%20de%20la%20observaci%C3%B3n.

datatab. (2024). *datatab*. Recuperado el 7 de Junio de 2024, de https://datatab.es/tutorial/chi-square-test

DATATab. (2024). *DATATab*. Recuperado el 7 de Junio de 2024, de https://datatab.es/tutorial/p-value

desarrolladores de scikit-learn. (2024). *Scikit-learn*. Recuperado el 26 de Junio de 2024, de https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.PolynomialFeatures.html

Docs Apryse. (s.f.). *Apryse Docs*. Recuperado el 5 de Junio de 2024, de https://docs.apryse.com/documentation/cli/guides/pdf2image/

Escobar Ortiz, J., & Amat Rodrigo, J. (2024). *Ciencia de Datos*. (Attribution-NonCommercial-ShareAlike 4.0 International.) Recuperado el 23 de Junio de 2024, de https://cienciadedatos.net/documentos/py51-modelos-arima-sarimax-python

Escuela de Ingeniería Informática. Universidad de Oviedo. (s.f.). *Unioviedo*. (Unioviedo) Recuperado el 30 de Mayo de 2024, de https://www.unioviedo.es/compnum/laboratorios_py/new/kmeans.html

ESPOL. (7 de Mayo de 2020). *ESPOL*. (ESPOL) Recuperado el 16 de Mayi de 2024, de https://www.espol.edu.ec/es/noticias/laboratorio-para-investigaciones-biomedicas-de-espol-tomara-pruebas-de-diagnostico-de

ESTRATEGIAS de INVERSION. (2024). *estrategiasdeinversion*. Recuperado el 23 de Junio de 2024, de https://www.estrategiasdeinversion.com/herramientas/diccionario/fondos/r2-r-cuadrado-o-coeficiente-de-determinacion-t-1163

Foundation Python Software. (5 de Junio de 2024). *python*. (Python Software Foundation) Recuperado el 6 de Junio de 2024, de https://docs.python.org/es/3/library/re.html

Gómez, T. (23 de Abril de 2023). *Medium*. (Medium) Recuperado el 7 de Junio de 2024, de https://tutegomeze.medium.com/todo-sobre-el-feature-scaling-bba836d6c212

Guerra, C., Aguiar, V., Súarez, V., Docampo, F., López, J. M., & Pereira, J. (2020). *Electronic Health Records Exploitation Using Artificial Intelligence Techniques.* Coruña: MDPI.

Hamad, R. (2 de Diciembre de 2023). *Medium*. (Medium) Recuperado el 27 de Junio de 2024, de https://medium.com/@rebeen.jaff/what-is-lstm-introduction-to-long-short-term-memory-66bd3855b9ce

Herbert, D. (13 de Noviembre de 2023). *HubSpot*. (HubSpot) Recuperado el 23 de Junio de 2024, de https://blog.hubspot.com/website/react-js

Homar, G. (22 de Septiembre de 2022). *Medium*. Recuperado el 15 de Junio de 2024, de https://medium.com/tacosdedatos/herramientas-para-pron%C3%B3sticos-de-series-de-tiempo-en-python-parte-3-prophet-aad43d86cf0d

IBM Corporation. (2021). *IBM*. Recuperado el 15 de Junio de 2024, de https://www.ibm.com/docs/es/spss-statistics/saas?topic=forecasting-introduction-time-series

Instituto Nacional De Salud. (2020). *Infección Respiratoria Aguda.* Instituto Nacional De Salud de Colombia.

Jaadi, Z. (23 de Febrero de 2024). *Builtin*. (Builtin) Recuperado el 4 de Julio de 2024, de https://builtin.com/data-science/step-step-explanation-principal-component-analysis

Javatpoint . (s.f.). *Javatpoint*. (Javatpoint) Recuperado el 16 de Junio de 2024, de https://www.javatpoint.com/hierarchical-clustering-in-machine-learning

karolzak. (18 de Enero de 2023). *pypi*. Obtenido de pypi: https://pypi.org/project/boxdetect/

Landa, N. (20 de Diciembre de 2021). *Medium*. Obtenido de Medium: https://medium.com/@nicolasarrioja/m%C3%A9tricas-en-regresi%C3%B3n-5e5d4259430b

Lichtenberger, M. (2024). Recuperado el 5 de Junio de 2024, de https://www.chatpdf.com/

LinkedIn. (s.f.). *LinkedIn*. (LinkedIn) Recuperado el 7 de Junio de 2024, de https://es.linkedin.com/advice/1/how-do-you-handle-categorical-numerical-variables?lang=es#:~:text=Las%20variables%20categ%C3%B3ricas%20son%20aquellas,dividir%20en%20ordinales%20y%20nominales.

LiteFinance.org. (2024). *LiteFinance*. Recuperado el 23 de Junio de 2024, de https://www.litefinance.org/es/blog/for-beginners/mejores-indicadores-forex/desviacion-estandar/

Masino, A., Harris, M., Forsyth, D., Ostapenko, S., Srinivasan, L., Bonafide, C., . . . Grundameier, R. (2019). *Machine Learning models for early sepsis recognition in the neonatal intensive care unit using readily available electronic health record data.* Pennsylvania: PLOS ONE.

Mastery, T. (2024). *DotComMagazine*. Obtenido de DotComMagazine: https://dotcommagazine.com/2024/04/silhouette-analysis-a-comprehensive-guide/

MedicalHubAssist. (16 de Enero de 2024). *MedicalHubAssist*. (MedicalHubAssist) Recuperado el 15 de Mayo de 2024, de https://www.medicalhubassist.ai/es/blog/historia-clinica-electronica-con-inteligencia-artificial

Melillanca, E. (8 de Junio de 2018). *Linkedin*. Obtenido de Linkedin: https://es.linkedin.com/pulse/noci%C3%B3n-de-r-cuadrado-o-coeficiente-determinaci%C3%B3n-en-eric#:~:text=Valores%20negativos%20de%20R2%20son,ser%C3%ADa%20recomendable%20interpretarlo%20como%20cero.

Minitab. (s.f.). *Soporte de Minitab*. Recuperado el 5 de Junio de 2024, de https://support.minitab.com/es-mx/minitab/help-and-how-to/statistical-modeling/regression/supporting-topics/basics/what-are-categorical-discrete-and-continuous-variables/

Mioti. (26 de Julio de 2023). *Mioti*. (Mioti) Recuperado el 5 de Junio de 2024, de https://mioti.es/es/que-son-los-

outliers/#:~:text=En%20t%C3%A9rminos%20sencillos%2C%20los%20outliers,la%20mayor%C3%ADa%20de%20las%20observaciones.

Montoya, M. (12 de Mayo de 2023). *RPubs*. Obtenido de RPubs: https://rpubs.com/maribiua/1042001

Muhammad, I. (14 de Noviembre de 2023). *Medium*. Obtenido de Medium: https://medium.com/@iqra1804/time-series-forecasting-using-lstm-an-introduction-with-code-explanations-c5c2e8ca137d

Mujeeb, I. (s.f.). *Educative*. (Educative) Recuperado el 13 de Junio de 2024, de https://www.educative.io/answers/what-is-the-difference-between-k-means-and-k-medoids-algorithms

Mujeeb, I. (s.f.). *Educative*. (Educative) Recuperado el 13 de Junio de 2024, de https://www.educative.io/answers/what-is-the-difference-between-k-means-and-k-medoids-algorithms

Mullin, T. (9 de Julio de 2020). *Medium*. Obtenido de Medium: https://medium.com/@tarammullin/dbscan-parameter-estimation-ff8330e3a3bd

Nair, R. (Enero de 2023). *ResearchGate*. Obtenido de ResearchGate: https://www.researchgate.net/post/Threshold_silhouette_score_for_cluster_analysis#:~:text=Typically%2C%20a%20silhouette%20score%20of,assigned%20to%20the%20wrong%20cluster.

NextGen Invent. (s.f.). *NextGen Invent*. (NextGen Invent) Recuperado el 6 de Junio de 2024, de NextGen Invent: https://nextgeninvent.com/blogs/7-steps-of-image-pre-processing-to-improve-ocr-using-python-2/

OPS. (s.f.). *Pan American Health Organization*. (Pan American Health Organization) Recuperado el 21 de Mayo de 2024, de https://www.paho.org/es/temas/influenza-sars-cov-2-vsr-otros-virus-respiratorios#:~:text=SARS%2DCoV%2D2%20es%20el,sin%20requerir%20un%20tratamiento%20especial.

Ortega, C. (2024). *QuestionPro*. Recuperado el 7 de Junio de 2024, de https://www.questionpro.com/blog/es/coeficiente-de-correlacion-de-pearson/

pandas via NumFOCUS, Inc. Hosted by OVHcloud. . (2024). Obtenido de https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.resample.html

pandas via NumFOCUS, Inc. Hosted by OVHcloud. . (2024). *pandas*. Obtenido de https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.fillna.html

Parra, R. (14 de Marzo de 2023). *Linkedin*. Recuperado el 26 de Junio de 2024, de https://es.linkedin.com/pulse/triple-suavido-exponencial-o-metodo-de-holt-winters-raul

Peláez, E. (25 de Julio de 2024). LSTM para temporal análisis de los virus. Guayaquil, Guayas, Ecuador.

Peng, S., Liu, Y., Lv, L., Zhou, Q., Yang, H., Ren, J., . . . Xiao, H. (2021). *Deep learning-based artificial intelligence model to assist thyroid nodule diagnosis and management: a multicentre diagnostic study.* Guangzhou: Elsevier Ltd.

Pochet, M. S. (7 de Septiembre de 2020). *Revista ciencia y salud*. Recuperado el 15 de Mayo de 2024, de https://www.revistacienciaysalud.ac.cr/ojs/index.php/cienciaysalud/article/view/176/292

Priya, B. (3 de Noviembre de 2023). *KDnuggets*. Obtenido de KDnuggets: https://www.kdnuggets.com/hyperparameter-tuning-gridsearchcv-and-randomizedsearchcv-explained

Pyimagesearch. (15 de Noviembre de 2021). *Pyimagesearch* . Recuperado el 6 de Junio de 2024, de https://pyimagesearch.com/2021/11/15/tesseract-page-segmentation-modes-psms-explained-how-to-improve-your-ocr-accuracy/

PythonAnyWhere. (s.f.). *PythonAnyWhere*. Obtenido de PythonAnyWhere: https://www.pythonanywhere.com/pricing/

Reddy, Y. (10 de Abril de 2023). *Medium*. (Medium) Recuperado el 13 de Junio de 2024, de https://medium.com/@reddyyashu20/k-means-kmodes-and-k-prototype-76537d84a669

Rodrigo, J. A. (2024). *CienciasdeDatos.net*. (Attribution-NonCommercial-ShareAlike 4.0 International) Recuperado el 23 de Junio de 2024, de https://cienciadedatos.net/documentos/62_optimizacion_bayesiana_hiperparametros

Salud Pública. (16 de Octubre de 2020). *Gaceta*. (Gaceta) Recuperado el 26 de Mayo de 2024, de https://gaceta.facmed.unam.mx/index.php/2020/10/16/la-temporada-de-la-coexistencia-viral-sars-cov-2-e-influenza/

ScienceDirect. (2019). *ScienceDirect*. Obtenido de ScienceDirect: https://www.sciencedirect.com/topics/computer-science/decision-tree-

classifier#:~:text=A%20Decision%20Tree%20Classifier%20is,nodes%20indicate%20the%20class%20label.

scikit-learn developers. (2024). *scikit-learn.org*. Recuperado el 23 de Junio de 2024, de https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

Scikitlearn. (s.f.). *scikir-learn*. (scikir-learn) Recuperado el 7 de Junio de 2024, de https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html

Scikitlearn. (s.f.). *scikit-learn*. (scikit-learn) Recuperado el 7 de Junio de 2024, de https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

Secretaría de Salud. (17 de Septiembre de 2015). *Gobierno de México*. (Gobierno de México) Recuperado el 15 de Mayo de 2024, de https://www.gob.mx/salud/articulos/infecciones-respiratorias-agudas-iras.%20Published%202009#:~:text=%C2%BFQu%C3%A9%20son%20las%20infecciones%20respiratorias,durar%20m%C3%A1s%20de%2015%20d%C3%ADas.

Sharma, P. (4 de Junio de 2024). *Analytics Vidhya*. (Analytics Vidhya) Recuperado el 13 de Junio de 2024, de https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/

Software Foundation Python. (2024). *pypi*. (Python Software Foundation) Recuperado el 5 de Junio de 2024, de https://pypi.org/project/PyPDF2/

Software Foundation Python. (s.f.). *pypi*. (Foundation Python Software) Recuperado el 5 de Junio de 2024, de https://pypi.org/project/pytesseract/

Sotaquirá, M. (16 de Octubre de 2023). *CodificandoBits*. Recuperado el 2 de Junio de 2024, de https://www.codificandobits.com/tutorial/forecasting-redes-lstm-6-codificacion-variable-temporal/

team OpenCV. (s.f.). *opencv.org*. (OpenCV team) Recuperado el 6 de Junio de 2024, de https://opencv.org/about/

The Data Schools. (2024). *The Data Schools*. Recuperado el 26 de Junio de 2024, de https://thedataschools.com/python/regresion-polinomial/

Urrego, N. (14 de Julio de 2023). *Medium*. (Medium) Recuperado el 7 de Junio de 2024, de https://nicolasurrego.medium.com/la-distribuci%C3%B3n-normal-un-pilar-fundamental-en-el-an%C3%A1lisis-de-datos-6d6ea8c7ee71#:~:text=En%20una%20distribuci%C3%B3n%20normal%2C%20los,la%20otra%20mitad%20por%20debajo.

Villafuerte, N., Manzano, S., Ayala, P., & García, M. (2023). *Artificial Intelligence in Virtual Telemedicine Triage: A Respiratory Infection Diagnosis Tool with Electronic Measuring Device.* Ambato: Universidad Tecnica de Ambato.

Yadav, P., Steinbach, M., Kumar, V., & Simon, G. (2018). *Mining Electronic Health Records.* ACM Computing Surveys.

Zulmuthi, H. (12 de Mayo de 2022). *Medium*. (Medium) Recuperado el 5 de Junio de 2024, de https://medium.com/@haniszulaikha/out-with-the-outliers-fc39c2bcacd7#:~:text=Winsorization%20is%20essentially%20similar%20to,bottom%205%25%20of%20the%20data.

Anexos

Anexo 1: Manual de instalación

Requisitos previos

Backend

Python: Es necesario tener instalado Python 3.10.2 o una versión superior.

Base de datos: Debe estar instalada una versión de MySQL superior a 8.0.0.

Framework: Este proyecto utiliza Django debido a su eficiencia en el desarrollo de aplicaciones web.

Entorno virtual: Se usa pipenv como entorno virtual para desarrollar, modificar y eliminar estructura en el código, tanto en el backend como en el frontend.

Tesseract-OCR: Es necesario instalar tesseract versión 5.3.3.20231005

Pdftoppm: Se debe tener instalada la versión 24.02.0 o la versión 24.07.0

Frontend

React: Se usó React como Framework para el frontend.

Librerías: la instalación de las librerías se realiza con npm install “nombre de la librería”.

Pasos para la instalación del backend

En la aplicación realizada para acceder al entorno virtual se realiza los siguientes pasos:

1. pip install pipenv
2. npm install
3. pipenv install --dev
4. pipenv shell

Crear los modelos

En la siguiente Figura se muestra un ejemplo de cómo realizar los modelos respecto a la base de datos.

Figura 33

Imagen de modelos

djangoAppBackend > api > models.py > ...

```python
from django.db import models

class Usuario(models.Model):
    id = models.AutoField(primary_key=True)
    nombre_usuario = models.CharField(max_length=100)
    apellido = models.CharField(max_length=100)
    email = models.EmailField(unique=True)
    contrasena = models.CharField(max_length=100)
    rol = models.CharField(max_length=50)

class Administrador(models.Model):
    id_usuario = models.OneToOneField(Usuario, on_delete=models.CASCADE, primary_key=True)

class Medico(models.Model):
    id_usuario = models.OneToOneField(Usuario, on_delete=models.CASCADE, primary_key=True)
    id_administrador = models.ForeignKey(Administrador, on_delete=models.SET_NULL, null=True)
    estado = models.CharField(max_length=50)
    especialidad = models.CharField(max_length=100)

class Diagnostico(models.Model):
    diagnostico_id = models.AutoField(primary_key=True)
    medico = models.ForeignKey(Medico, on_delete=models.CASCADE)
    fecha = models.DateField(auto_now_add=True)
    archivo_pdf = models.CharField(max_length=255)
```

Fuente: Autores de este documento.

Configuración del archivo settings.py

Es importante mencionar que Django por default en la variable “ENGINE”, de la sección de DATABASES, viene con un valor de 'django.db.backends.postgreSQL'.en la parte de PostgreSQL, cambiar por “mysql”, dado a que ese es el motor de base de datos utilizado para la aplicación en la cual se está trabajando.

Figura 34

Estructura de Databases en settings.py

```python
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'sistema_diagnostico',
        'USER': '',
        'PASSWORD': '',
        'HOST': 'localhost',
        'PORT': '3306',
    }
}
```

Fuente: Autores de este documento.

Configuración para las migraciones a la base de datos

Con respecto a las variables USER y PASSWORD, en la parte de desarrollo se deben colocar las credenciales de MySQL que tenga el usuario en la computadora instalada.

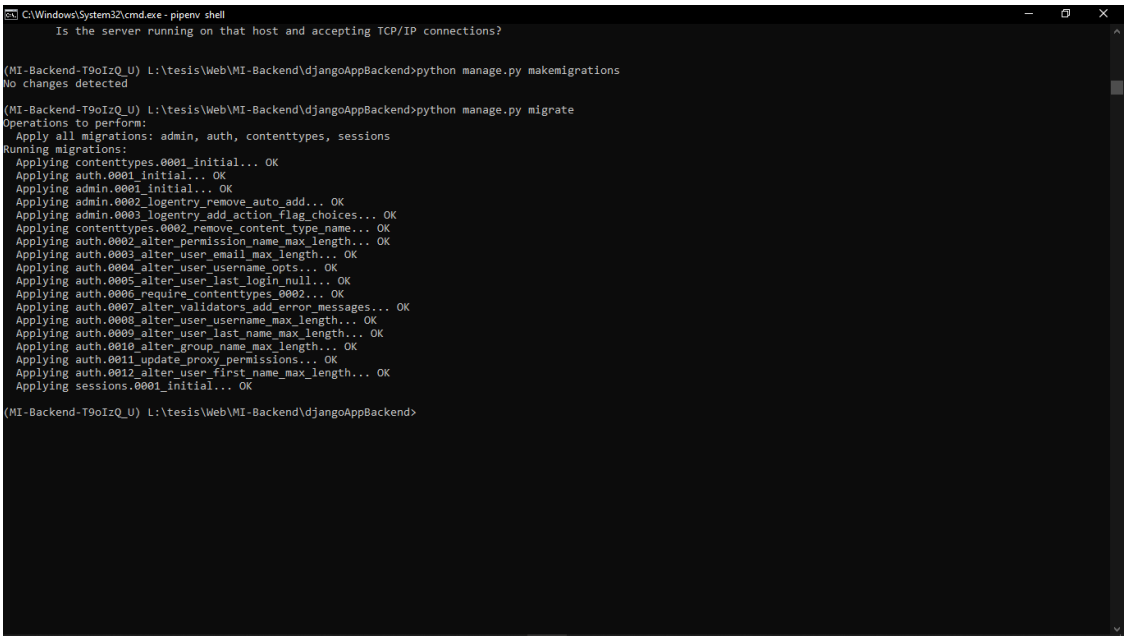
Realizadas las configuraciones mencionadas, se procede a crear aplicar las migraciones ejecutando los siguientes comandos:

1. python manage.py makemigrations
2. python manage.py migrate

En la siguiente figura se podrá observar la ejecución de los comandos mencionados.

Figura 35

Migraciones de los modelos a la base



Fuente: Autores de este documento.

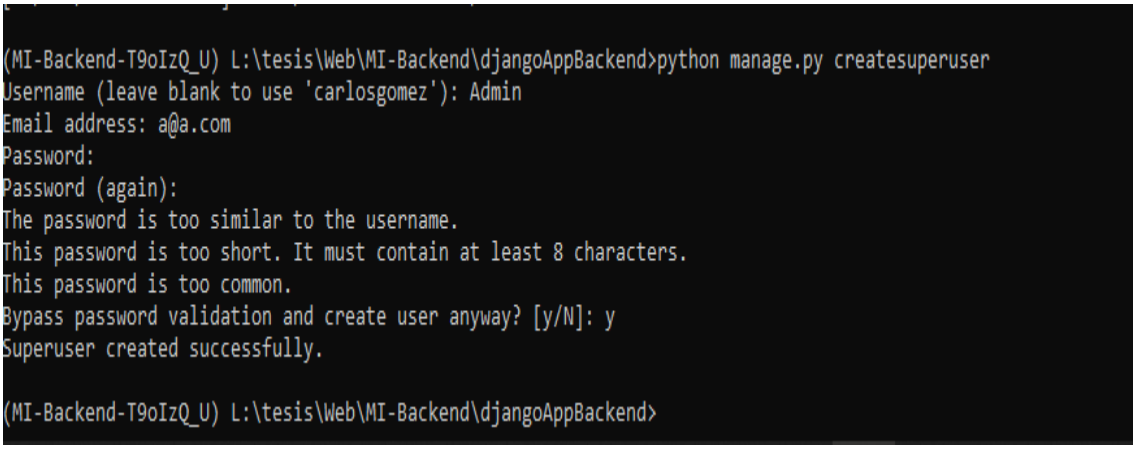
Con la base de datos creada, se debe crear el Superusuario de Django el cual es el que tiene el control de todo el sitio, es decir el usuario con privilegios de usuario que le permiten acceder y administrar la interfaz de administración de Django, además de ser usuario con el cual se puede iniciar sesión.

La creación del Superusuario se realiza con el siguiente comando:

python manage.py createsuperuser

Figura 36

Creación del Superusuario



Fuente: Autores de este documento.

Realizado eso se procede a levantar el servidor para entrar a la interfaz de Django y entrar con las credenciales de Superusuario. Esta acción se realiza con el siguiente comando.

python manage.py runserver

Con el servidor en ejecución y trabajando de manera local, se debe ingresar a la siguiente URL: http://127.0.0.1:8000/admin

Librerías utilizadas en el backend

Las librerías instaladas para el desarrollo de la aplicación son:

- django = "*"
- djangorestframework = "*"
- django-cors-headers = "*"
- mysqlclient = "*"
- djangorestframework-simplejwt = "*"
- pyjwt = "*"
- pytz = "*"
- sqlparse = "*"
- requests = "*"
- pandas = "*"
- notebook = "*"
- PDFs2image = "*"
- pyPDFs2 = "*"
- opencv-python = "*"
- boxdetect = "*"
- pytesseract = "*"
- matplotlib = "*"
- scikit-learn = "==1.5.1"

Cabe mencionar, que en el entorno virtual de pipenv, al usar el siguiente comando pipenv install –dev, este instala todas las librerías instaladas con anterioridad.

Endpoints del proyecto

En la siguiente tabla se procede a mostrar los Endpoint utilizado para este proyecto. Los endpoints utilizados empiezan con la siguiente **url:** http://127.0.0.1:8000/

Tabla 9

Tabla de endpoints

Endpoint	Detalles
url/admin	Endpoint donde puede ingresar el Superusuario.
api/PDFs-data	Endpoint donde se almacena la información de los PDFs en formato JSON.
export_combined_PDFs_data/<int:id_medico>/<int:id_JSONdata>/	Endpoint en el cual se combina la información del JSON con el ID médico.
api/create_diagnostico/<int:id_medico>/<int:id_diagnosticos>/	Endpoint para poder crear el diagnóstico que ingrese el medico desde el front.
api/token	Endpoint donde se administra el inicio de sesión.
api/token/refresh	Endpoint usado para manejar el tiempo de uso del usuario en la aplicación web.
api/carga_PDFs	Endpoint que valida que los PDFs sean correctos.
api/identificacion_PDFs_paginas_grandes/	Endpoint que identifica las páginas de los PDFs con dimensiones distintas.
api/extraccion_validacion_contenido_archivosPDFSS/	Endpoint que realiza la validación y extracción del contenido de las páginas de PDFs con dimensiones diferentes.
api/combinar_diccionarios/	Endpoint que sirve para combinar todos los diccionarios obtenidos desde el front, con la finalidad de

	almacenar en un diccionario general.
api/resultado_cluster/<int:id_JSONdata>/<int:id_medico>/	Endpoint destinado a almacenar los resultados que indican a qué clúster pertenece el paciente diagnosticado.
api/guardar_resultados_PDFs	Endpoint donde se guardará los resultados de los PDFs.
api/lista_diagnostico	Endpoint usado para poder visualizar los diagnostico desde la vista del administrador.

Librerías usadas en el frontend

Las librerías usadas para el proyecto fueron las siguientes:

- @emotion/react@11.11.4
- @emotion/styled@11.11.5
- @types/react-dom@18.3.0
- @types/react@18.3.3
- @vitejs/plugin-react@4.3.1
- axios@1.7.2
- eslint-plugin-react-hooks@4.6.2
- eslint-plugin-react-refresh@0.4.7
- eslint-plugin-react@7.34.3
- eslint@8.57.0
- file-saver@2.0.5
- framer-motion@11.2.14
- jsPDFs@2.5.1
- jwt-decode@4.0.0
- PDFsjs-dist@4.4.168
- react-dom@18.3.1
- react-icons@5.2.1
- react-PDFs@9.1.0
- react-router-dom@6.24.1
- react-spinners@0.14.1
- react-toastify@10.0.5
- react@18.3.1
- styled-components@6.1.11
- universal-cookie@4.0.4
- vite@4.5.3

nota: El entorno virtual para el front es similar al usado en el backend.

Anexo 2: Manual de usuario

Rol de médico

La principal funcionalidad dirigida para el médico es poder subir archivos “.pdf” acerca de la sintomatología de un paciente los cuales se procederán a procesar usando técnicas de Inteligencia Artificial.

La primera pantalla mostrada será el inicio de sesión. En esta pantalla deberá ingresar las credenciales proporcionadas por el administrador.

Figura 37

Pantalla inicio sesión

Fuente: Autores de este documento.

Una vez iniciada la sesión, se presentará la pantalla principal. En esta pantalla el médico podrá seleccionar entre 4 y 5 archivos “.pdf” relacionados a sintomatología de un paciente. Los títulos de estos archivos deben contener alguno de los siguientes nombres: *Enrolamiento, Preselección, Resultado Multiplex, Resultados Multiplex, Toma de muestra, Colecta de muestra, Toma muestra, Hemocultivo.* Los nombres Resultado Multiplex y Resultados Multiplex son nombres diferentes de un mismo archivo. Este caso también aplica para los títulos Toma de muestra, Colecta de muestra y Toma muestra.

Es importante mencionar que al momento de seleccionar los archivos estos deben ser seleccionados en conjunto para poder ser cargados. Los conjuntos de archivos permitidos para ser cargados son:

1. Enrolamiento, Preselección, Resultado Multiplex (diferentes títulos), Toma de muestra (diferentes títulos)
2. Enrolamiento, Preselección, Resultado Multiplex (diferentes títulos), Toma de muestra (diferentes títulos), Hemocultivo

Figura 38

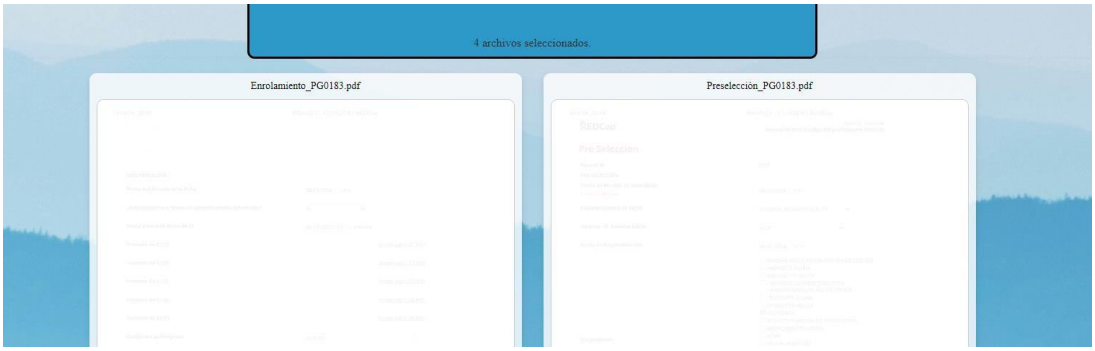
Pantalla principal

Fuente: Autores de este documento.

Al momento de seleccionar los archivos, estos se cargarán en pantalla para su visualización. Por motivos de protección de datos, las siguientes cargas de archivos se presentan difuminadas.

Figura 39

Sección carga de archivos ".pdf"



Fuente: Autores de este documento.

El médico puede recorrer las páginas de los archivos “.pdf” en caso de que el archivo tenga más de una página.

Figura 40

Opción para pasar de página

Fuente: Autores de este documento.

Si el médico ha seleccionado correctamente los archivos que quiere cargar, este puede presionar el botón “Procesar archivos” para proceder a su procesamiento.

Figura 41

Botón para procesar archivos

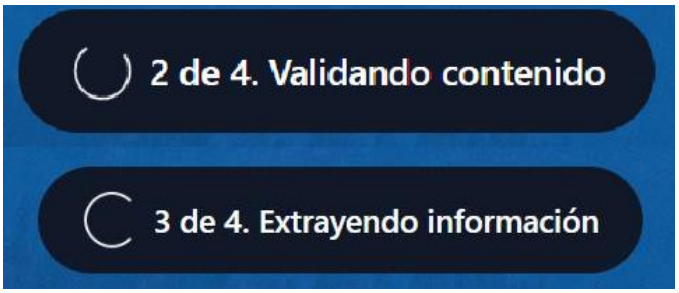
Fuente: Autores de este documento.

Al presionar el botón, comenzará el procesamiento de los archivos “.pdf” y se deshabilitará la opción de “Elegir archivos” para que el procesamiento no se detenga.

Mientras se realiza el procesamiento, diferentes mensajes de retroalimentación se mostrarán en el botón “Procesar archivos” para que el médico conozca los diferentes procesos que ocurren.

Figura 42

Mensajes de retroalimentación de procesos

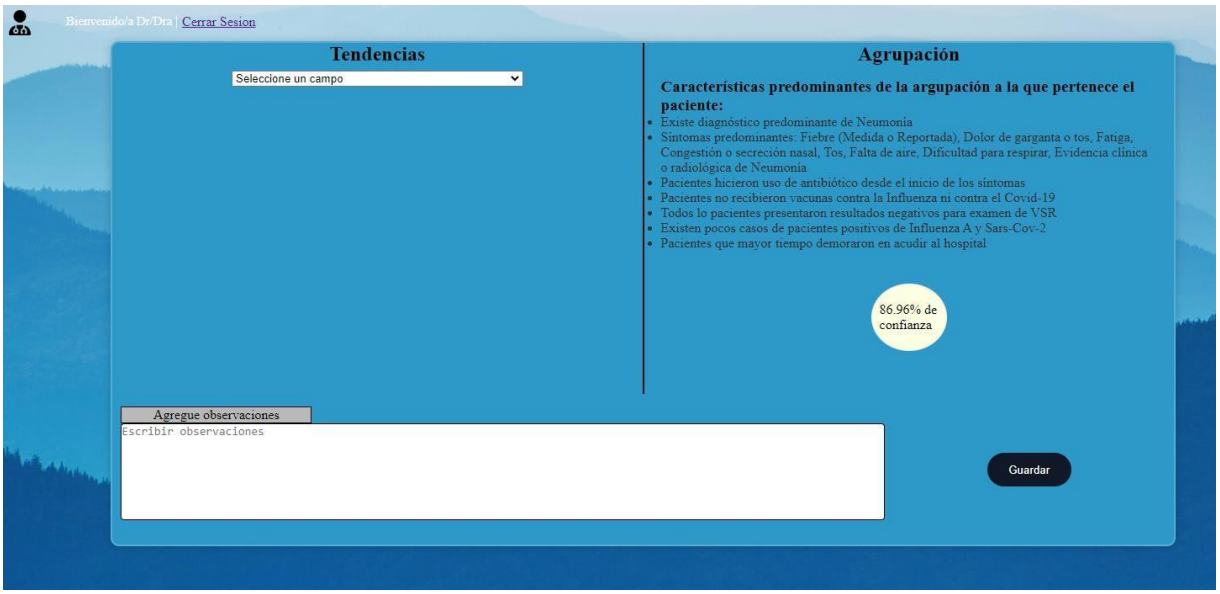


Fuente: Autores de este documento.

Cuando el procesamiento termine, se presentará la pantalla resultados donde se mostrarán los resultados obtenidos acerca de predicción de tendencias y predicción de agrupación a la cual pertenece el paciente.

Figura 43

Pantalla de resultados

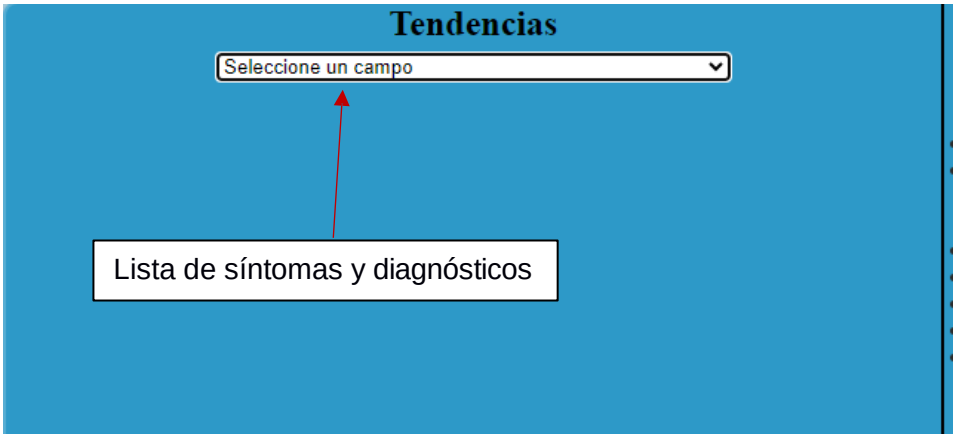


Fuente: Autores de este documento.

En la sección de **Tendencias**, el médico podrá seleccionar los gráficos de predicciones de tendencias con respecto a síntomas y diagnósticos de una lista debajo del título de la sección. Una vez seleccionado un campo, se presentará el correspondiente gráfico de tendencias. Para poder visualizar el gráfico con mayor tamaño, el médico puede presionar el gráfico y se abrirá una nueva pestaña que lo presentará con dimensiones más grandes.

Figura 44

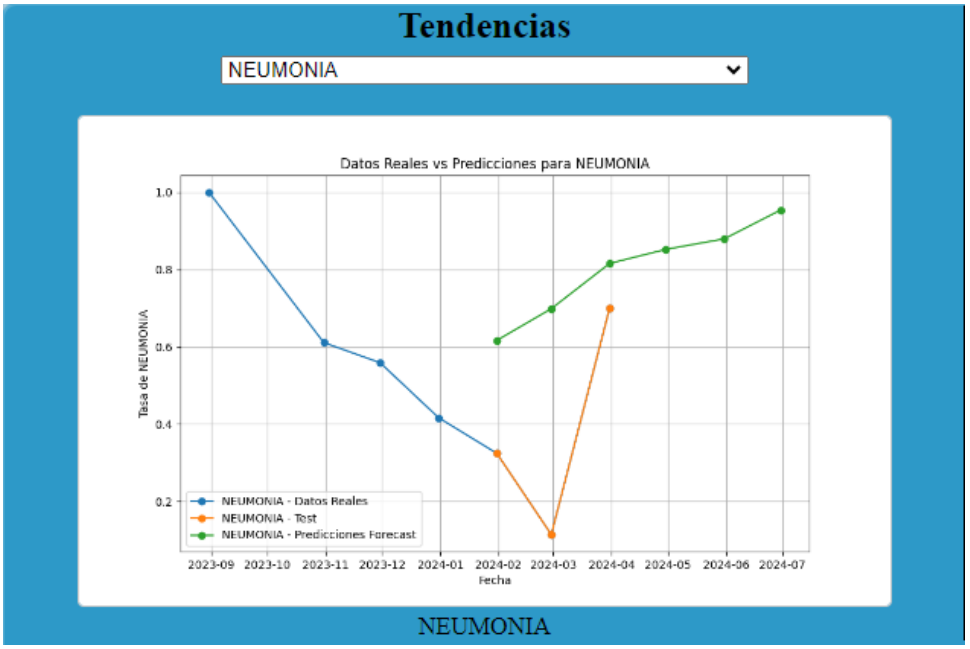
Sección de tendencias



Fuente: Autores de este documento.

Figura 45

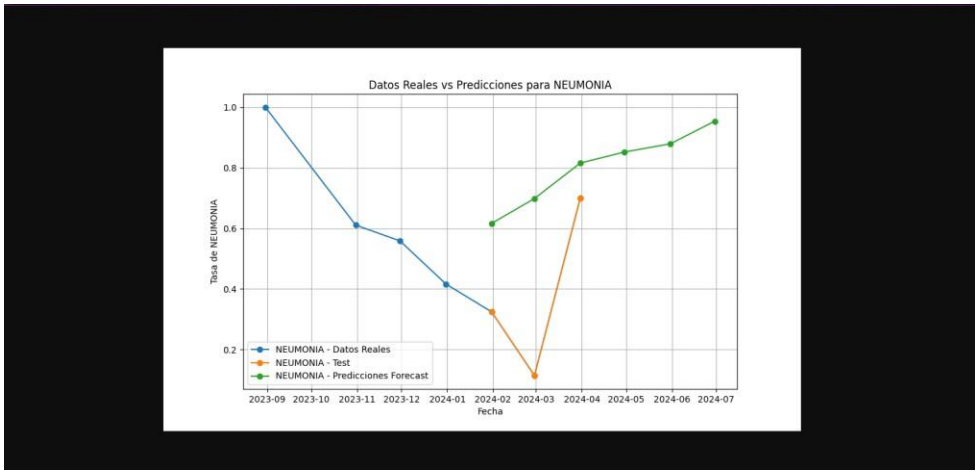
Sección de tendencias con gráfico de tendencias



Fuente: Autores de este documento.

Figura 46

Gráfico de tendencias con tamaño más grande

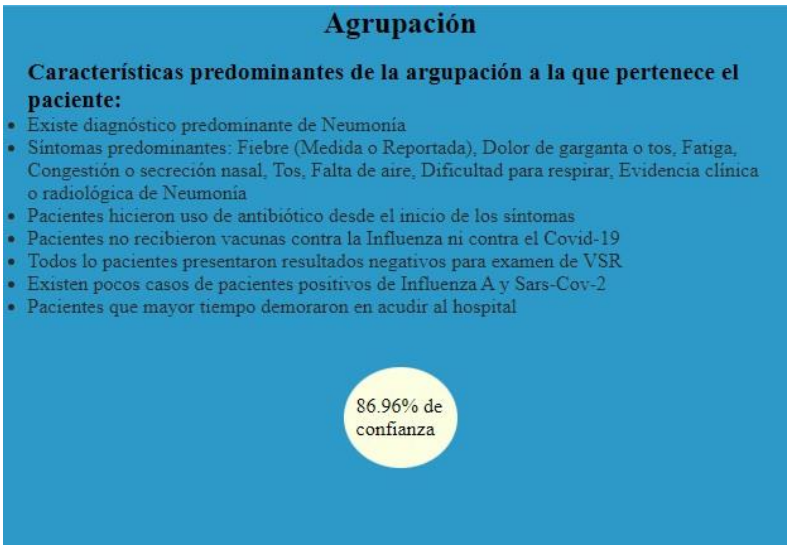


Fuente: Autores de este documento.

En la sección de **Agrupación**, el médico podrá visualizar las características predominantes de la agrupación a la que pertenece el nuevo paciente junto con el valor de confianza de esta predicción realizada.

Figura 47

Sección de agrupación

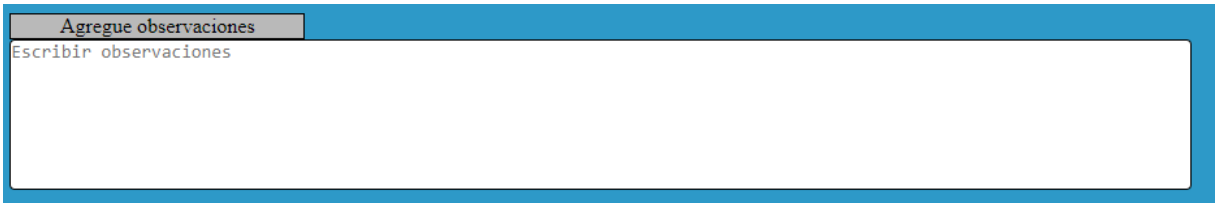


Fuente: Autores de este documento.

En la sección de **Agregue observaciones**, el médico tiene la opción de escribir comentarios que considere relevantes acerca del análisis que haya hecho a los resultados obtenidos.

Figura 48

Sección de observaciones

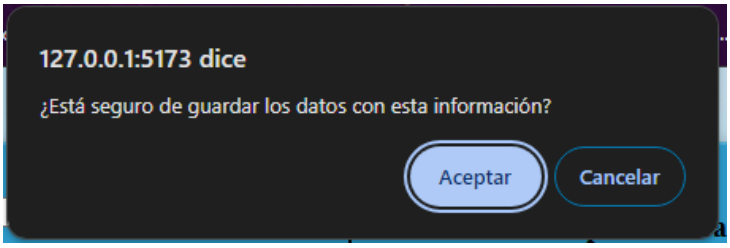


Fuente: Autores de este documento.

Una vez el médico haya realizado su análisis, este puede presionar el botón “Guardar” el cual generará un archivo “.pdf” que contenga los gráficos de tendencias, características de la agrupación y comentarios realizados. Antes de la generación de este archivo, aparecerá un mensaje de retroalimentación preguntando al médico si está seguro de generar este archivo con los datos presentados.

Figura 49

Mensaje de retroalimentación para guardar resultados



Fuente: Autores de este documento.

Este archivo se procederá a guardar localmente y en una base de datos. Por pantalla se presentará un mensaje de retroalimentación indicando que el archivo se ha guardado exitosamente y se mostrará una opción “Volver al inicio” para que el médico pueda volver a la pantalla principal para cargar nuevos archivos.

Figura 50

Pantalla con opción de volver al inicio

Fuente: Autores de este documento.

Por último, el médico tiene la opción de “Cerrar Sesión” en la parte superior izquierda en todas las pantallas.

Figura 51

Pantalla principal con señalización a la opción de cerrar sesión

Fuente: Autores de este documento.

Rol de administrador

Las funciones del rol del administrador son las siguientes:

- Registrar médicos
- Visualizar lista de médicos registrados
- Visualizar lista de archivos de resultados generados por médicos

Tal como el rol médico, la primera pantalla mostrada será el inicio de sesión. Aquí el médico debe ingresar sus credenciales creadas de manera predeterminada.

Figura 52

Login de aplicación

Fuente: Autores de este documento.

A continuación, se presentará la pantalla principal. Aquí el administrador tendrá tres opciones:

1. Ver médicos
2. Ver resultados guardados
3. Registrar médicos

Figura 53

Opciones de Administrador

Fuente: Autores de este documento.

La opción **Ver médicos** presentará una nueva pantalla con la lista de médicos registrados junto con su información. En la parte superior izquierda estará el botón “Volver” el cual al presionarlo permitirá al administrador volver a la pantalla principal.

Figura 54

Listado de médicos

Fuente: Autores de este documento.

Los íconos en la columna **“Archivos”** permitirán al administrador acceder a una nueva pantalla donde se mostrará una lista de los archivos de resultados generados por un médico en específico. Por motivos de protección de datos, no se muestran los números de historias clínicas pertenecientes a los títulos de los archivos ni las urls que muestran la dirección en la que se encuentran guardados estos archivos. En la lista de archivos del doctor, al presionar un ícono de archivo, este procederá a abrirse en una pestaña nueva para que su contenido pueda ser visualizado. Además, se puede apreciar la url donde está guardado cada archivo “.pdf”.

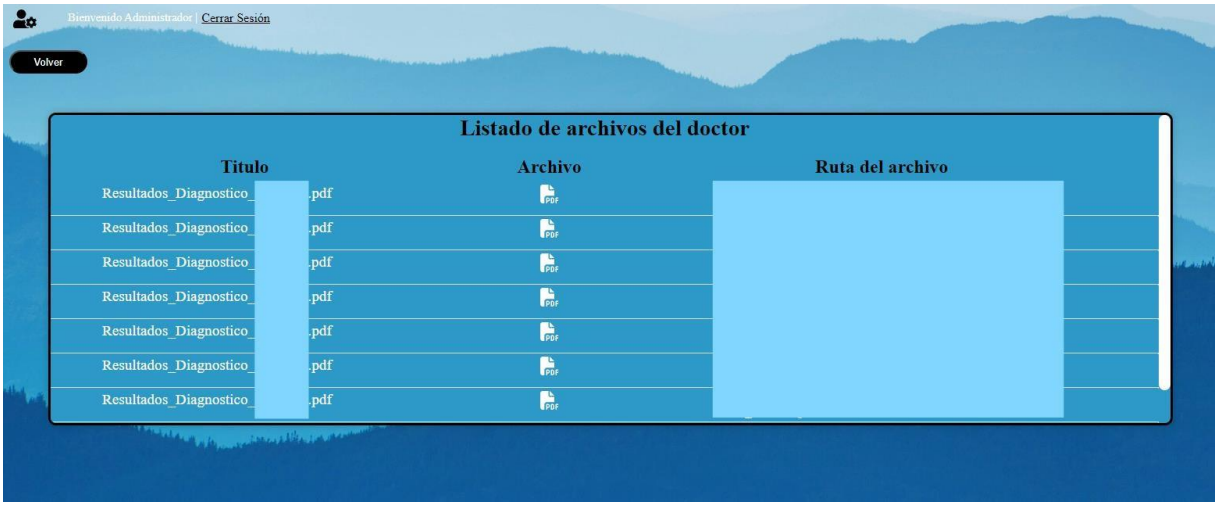
Figura 55

Listado de médicos, visualización de archivos ".pdfs"

Fuente: Autores de este documento.

Figura 56

Listado de archivos de médicos

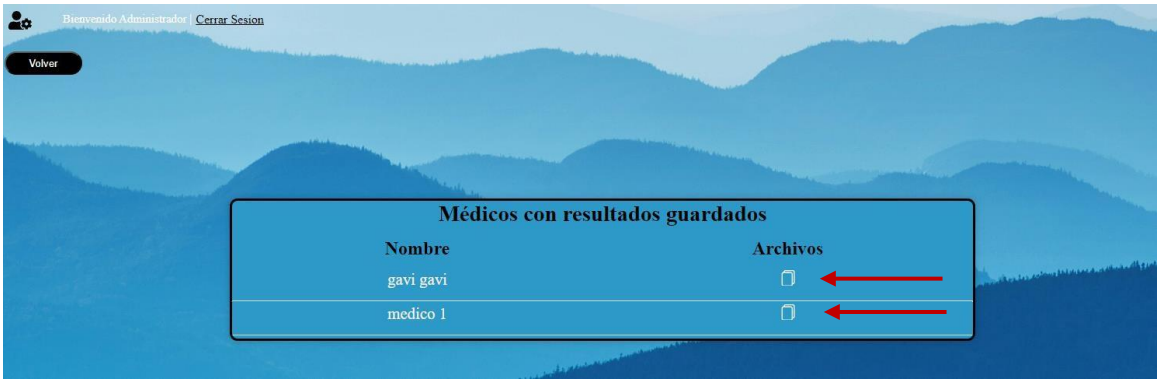


Fuente: Autores de este documento.

La opción **Ver resultados guardados** presentará por pantalla una lista de médicos junto con un ícono de archivos. Al acceder a uno de estos íconos, se presentará una pantalla con una lista de archivos de resultados generados por un médico en específico como se muestra en la Figura 65.

Figura 57

Listado de resultados guardados

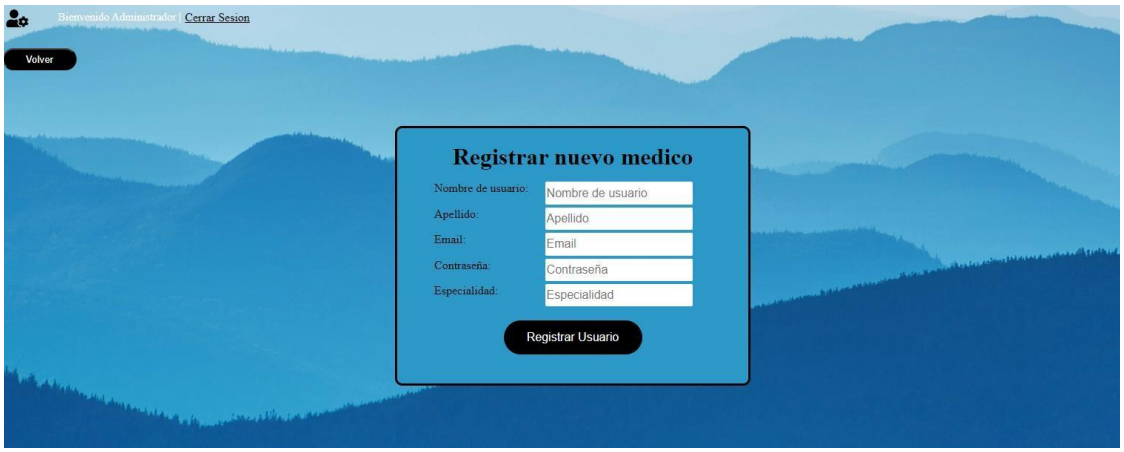


Fuente: Autores de este documento.

La opción **Registrar médicos** presentará por pantalla un formulario para registrar nuevos médicos. Aquí el administrador deberá llenar todos los campos solicitados para realizar un registro exitoso. Al presionar el botón “Registrar Usuario”, se mostrará un mensaje de retroalimentación preguntando si el administrador está seguro de realizar el registro con los datos escritos. Si acepta, se mostrará otro mensaje de retroalimentación indicando que el registro ha sido exitoso.

Figura 58

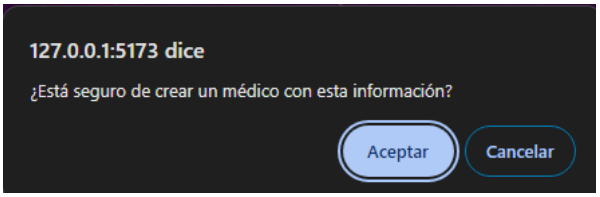
Registro de médicos



Fuente: Autores de este documento.

Figura 59

Confirmación de registro de médico



Fuente: Autores de este documento.

Figura 60

Registro exitoso

Fuente: Autores de este documento.

Por último, en todas las pantallas el administrador tiene la opción de “Cerrar Sesión” en la parte superior izquierda.

Anexo 3: Experimentaciones

Experimentaciones con modelos LSTM

Primer modelo basado en una red LSTM.

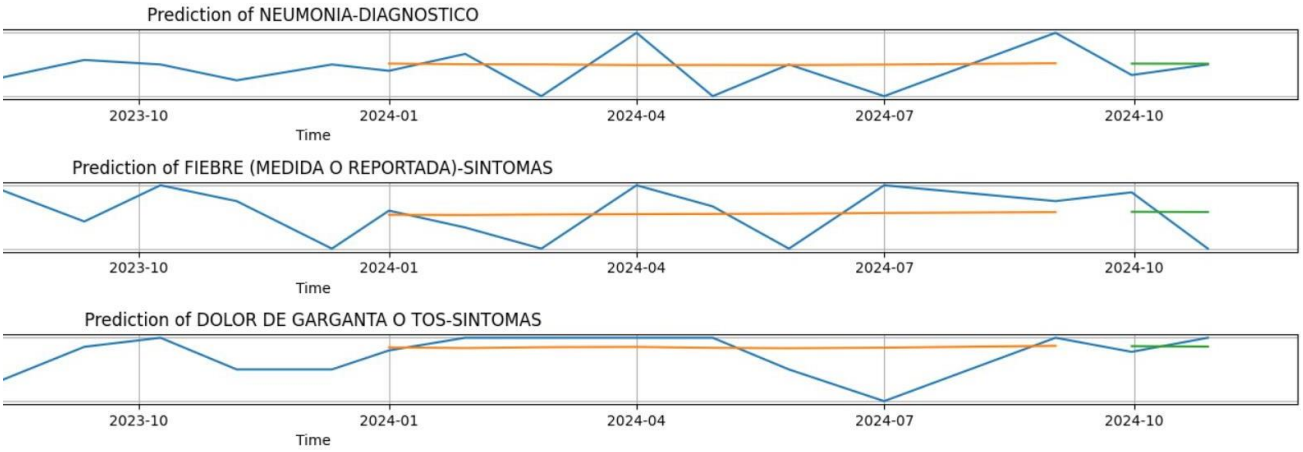
El primer modelo LSTM con el que se experimentó contaba con dos capas LSTM, cada una con 50 neuronas, y dos capas Densas de salida.

El dato de entrada para el modelo contenía un conjunto de 12 periodos, cada uno con las medias de los pacientes positivos para cada característica de síntomas y diagnósticos. Por otro lado, la predicción tenía como objetivo obtener la media de los valores en el siguiente periodo.

Luego de entrenar el modelo, se realizaron las distintas predicciones. A continuación, se muestran algunas predicciones obtenidas.

Figura 61

Predicciones obtenidas para tres características



Fuente: Autores de este documento.

Para estas predicciones la línea azul representa los datos reales, la línea naranja representa las predicciones de entrenamiento y la línea verde las predicciones de prueba. En esta figura se observan los datos reales, de entrenamiento y de predicción para los casos de *NEUMONIA*, *FIEBRE (MEDIDA O REPORTADA)* y *DOLOR DE GARGANTA O TOS*. Como se puede apreciar en el primer y último caso la predicción tiene un comportamiento similar al esperado; sin embargo, en el segundo caso ésta es diferente a los valores correspondientes a los datos reales.

Con estos resultados se procedió a evaluar el rendimiento del modelo con respecto a todas las características del dataset usando las métricas: MAE, RMSE y R2.

- MAE: Representa el promedio de la diferencia absoluta entre los valores reales y predichos de un dataset (Chugh, Medium, 2020).
- MSE: Representa el promedio de la diferencia al cuadrado entre los valores originales y predichos de un dataset (Chugh, Medium, 2020).

- R2: Representa la proporción de la varianza en la variable dependiente (Chugh, Medium, 2020).

En la siguiente figura se pueden observar las medias de las tres métricas obtenidas con respecto a los datos de entrenamiento.

Figura 62

Medias de las tres métricas con respecto a los datos de entrenamiento

```
Medias de entrenamiento
Media MAE: 0.2722871193289757
Media RMSE: 0.3275152944985313
Media R2: 0.004957925060950408
```

Fuente: Autores de este documento.

A pesar de que valores pequeños de MAE y RMSE indiquen una mayor precisión en el modelo de regresión (Landa, 2021), estos valores no llegan a ser muy cercanos a 0, y por ello no se puede afirmar que el modelo en cuestión tenga una gran precisión. Por otro lado, (Landa, 2021), un valor más grande de R2 es mejor; sin embargo, en este caso se tiene un valor negativo lo cual indica que el modelo está menos ajustado que el promedio (Melillanca, 2018).

En la siguiente figura se observan las medias de las tres métricas obtenidas con respecto a los datos de prueba.

Figura 63

Medias de las tres métricas con respecto a los datos de prueba

```
Medias de prueba
Media MAE: 0.26515441801812917
Media RMSE: 0.2948968018869132
Media R2: -1.9263471614526757
```

Fuente: Autores de este documento.

Al observar estos valores, son similares a los obtenidos de las medias de entrenamiento. Por esta razón se decidió experimentar con otro modelo con el fin de comparar la precisión.

Segundo modelo LSTM con ajuste de híper-parámetros aleatorio.

Para este segundo modelo se usó Randomized Search con el fin de encontrar los mejores parámetros para el modelo y así obtener un mejor rendimiento. Randomized search es una técnica de ajuste de híper-parámetros que explora combinaciones aleatorias de híper-parámetros dentro de rangos específicos (Priya, 2023).

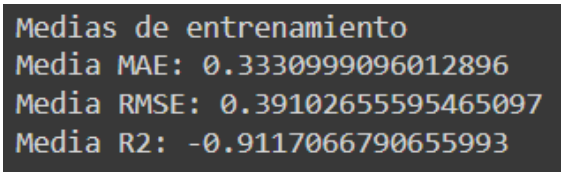
A su vez se agregaron características rezagadas al dataset o conexiones tipo skip, estas características son esencialmente valores pasados del dataset que sirven como espejo de los patrones y tendencias históricas que probablemente influirán en resultados futuros (Cubed, 2024).

Una vez obtenido el modelo junto con el dataset depurado, se procedió a entrenarlo y se obtuvo los datos necesarios para calcular las métricas antes mencionadas: MAE, RMSE y R2.

Las siguientes dos figuras representan las medias de las métricas mencionadas por parte del conjunto de datos de entrenamiento y de prueba.

Figura 64

Medias de las tres métricas con respecto a los datos de entrenamiento



Fuente: Autores de este documento.

Figura 65

Medias de las tres métricas con respecto a los datos de prueba

```
Medias de prueba
Media MAE: 0.40515178718737194
Media RMSE: 0.4369701500416521
Media R2: -4.798350934830906
```

Fuente: Autores de este documento.

Como se puede apreciar, estos valores no mejoran a los obtenidos con el primer modelo. Estos resultados podrían indicar que posiblemente un modelo LSTM no se ajusta de manera precisa a los datos, debido a la poca cantidad. Por esta razón, se re-diseñó el modelo LSTM con una arquitectura menos compleja.

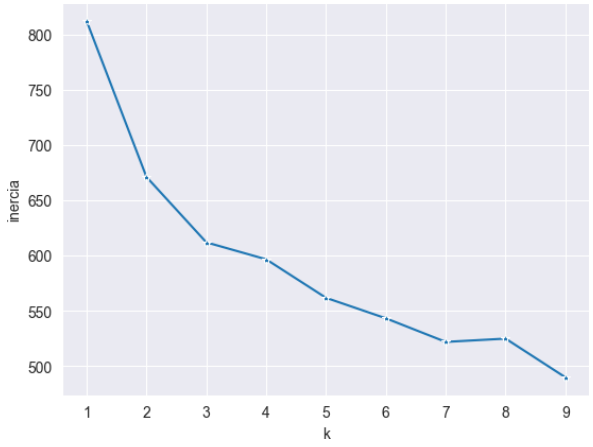
Experimentaciones de algoritmos de clustering

K-Means.

Para este algoritmo, al aplicar el método del Codo y del Coeficiente de Silueta, se determinó que 2 es el número de clústeres óptimo. Las siguientes dos figuras presentan los resultados de estos métodos.

Figura 66

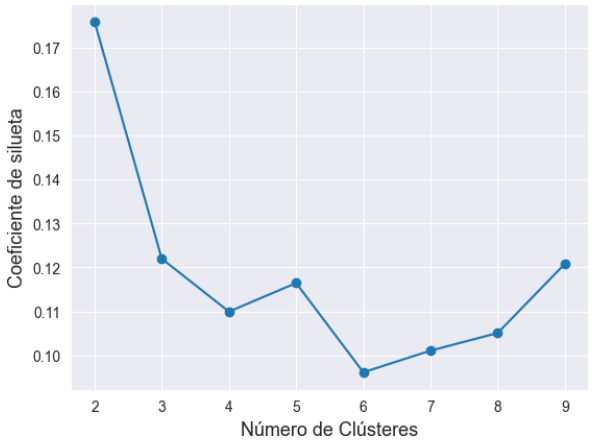
Método del Codo para K-Means



Fuente: Autores de este documento.

Figura 67

Método del Coeficiente de Silueta para K-Means



Fuente: Autores de este documento.

Nota. Valor de coeficiente de silueta más alto: 0.18.

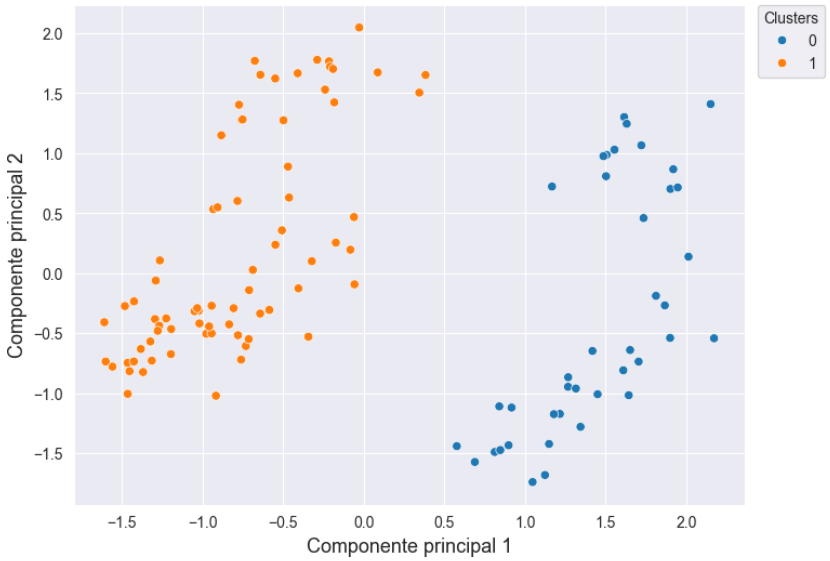
Una vez obtenido el número óptimo de clústeres, se aplicó el algoritmo K-Means y se procedió a identificar las diferencias entre los dos clústeres creados.

Análisis de clústeres.

La siguiente figura presenta la distribución de los clústeres obtenidos.

Figura 68

Clústeres obtenidos con K-Means



Fuente: Autores de este documento.

El clúster 0 se caracteriza por contener pacientes que, en su mayoría, no recibieron vacunas contra la Influenza en 2021 ni 2023, y no se tiene información si recibieron dicha vacuna en 2022. Por otro lado, no presentan algún diagnóstico predominante; sin embargo, presentan síntomas como *FIEBRE (MEDIDA O REPORTADA)*, *TOS*, *NAUSEAS O VOMITOS* y *DIFICULTAD PARA RESPIRAR*. Esta agrupación también se caracteriza por tener la media de edad más alta, el valor de talla más alto y también el mayor valor de peso.

El clúster 1 se caracteriza por contener pacientes los cuales, en su mayoría, no recibieron vacunas contra la Influenza ni contra el COVID-19. Estos pacientes tienen como diagnóstico predominante la *NEUMONIA* y presentan varios síntomas muy parecidos al clúster anterior como *FIEBRE (MEDIDA O REPORTADA)*, *TOS*, *DOLOR DE GARGANTA O TOS* y *CONGESTION O SECRECION NASAL*.

Las agrupaciones obtenidas con el algoritmo K-Means presentan patrones muy parecidos a los obtenidos por el algoritmo K-Medoids con 95% de varianza donde su diferencia más clara es el diagnóstico predominante de *NEUMONIA*.

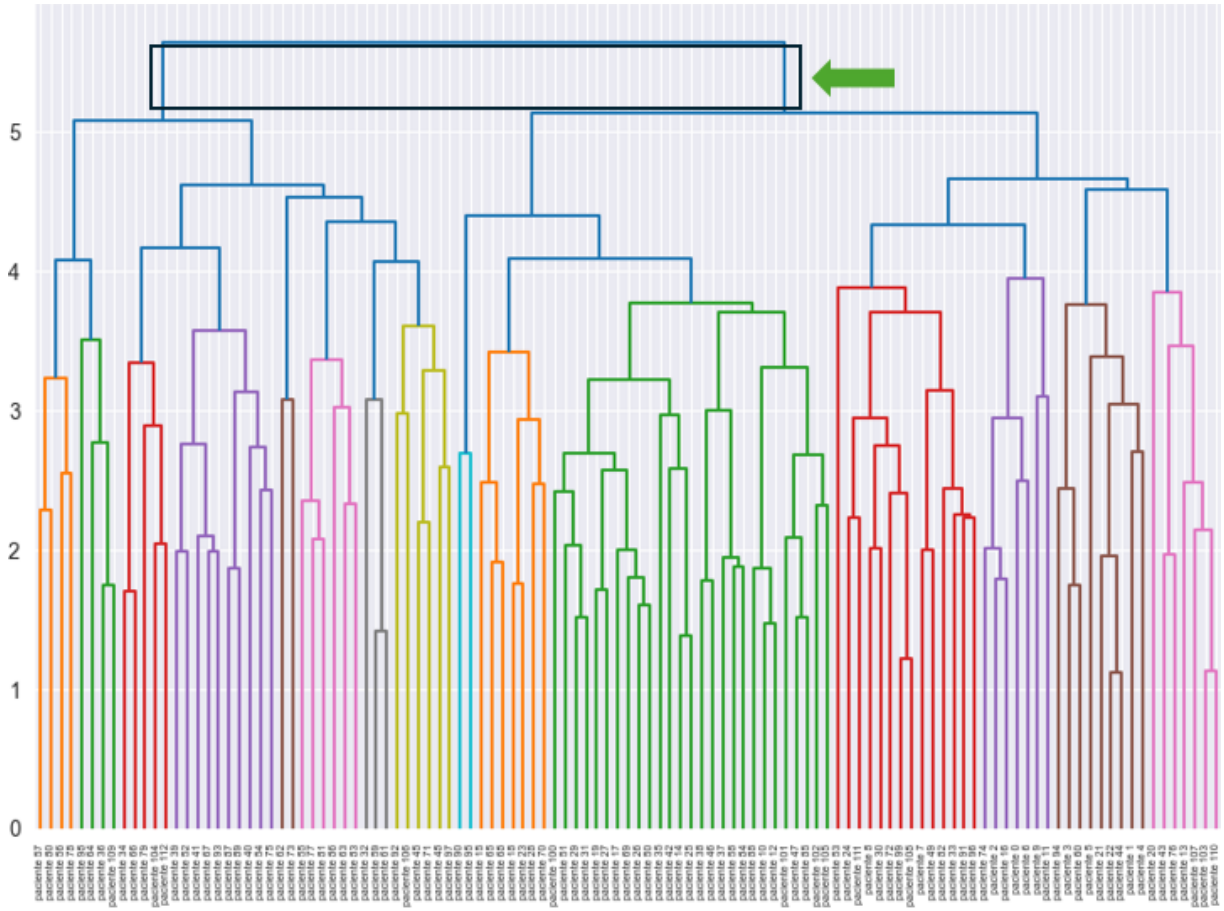
Algoritmo de clustering Jerárquico Aglomerativo.

Antes de la implementación de este algoritmo, se aplicó el Dendrograma y el método del Coeficiente de Silueta para obtener el número óptimo de clústeres.

Con el Dendrograma se determinó que el número óptimo es 2 clústeres, debido a que la distancia vertical máxima contenía dos líneas verticales. Esto se puede apreciar en la siguiente figura.

Figura 69

Dendrograma obtenido con el algoritmo de clustering Jerárquico Aglomerativo



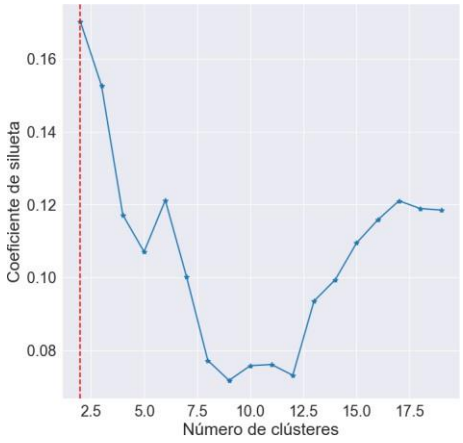
Fuente: Autores de este documento.

Nota. Cada valor del eje de las x pertenece a la información de un paciente en específico. Por ejemplo: *paciente 1*.

Al usar el método del Coeficiente de Silueta también se obtuvo al número 2 como valor óptimo, debido a que este contaba con el valor de coeficiente más alto. Esto se puede observar en la siguiente figura.

Figura 70

Método del Coeficiente de Silueta para el algoritmo de clustering Jerárquico Aglomerativo



Fuente: Autores de este documento.

Nota. Valor de coeficiente de silueta más alto: 0.17.

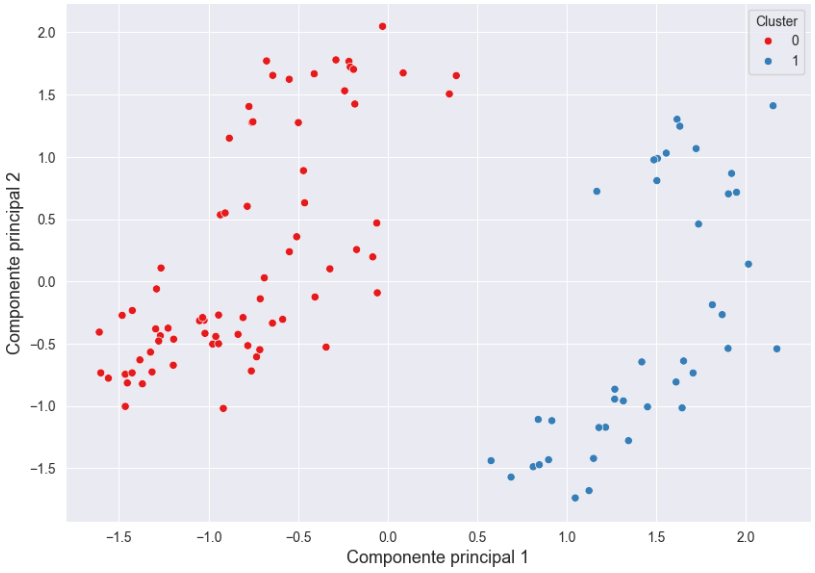
Con el número óptimo de clústeres obtenido se procedió a aplicar el algoritmo y luego a identificar las características más relevantes de las 2 agrupaciones creadas.

Análisis de clústeres.

La siguiente figura presenta la distribución de los clústeres obtenidos.

Figura 71

Clústeres obtenidos con algoritmo de clustering Jerárquico Aglomerativo



Fuente: Autores de este documento.

El clúster 0 se caracteriza por contener pacientes que, en su mayoría, no recibieron vacunas ni contra la Influenza ni contra el COVID-19. Entre ellos se presenta la *NEUMONIA* como diagnóstico predominante junto con síntomas como la *FIEBRE (MEDIDA O REPORTADA), DOLOR DE GARGANTA O TOS*, *FATIGA* y *FALTA DE AIRE*. Es importante mencionar que existen pacientes que presentan positivo a resultado de SARS-COV-2 e Influenza A, sin embargo todos presentaron negativo al resultado de VSR.

Por el lado del clúster 1, para la mayoría de los pacientes se conoce que no recibieron vacuna contra el COVID-19 y no cuentan con un diagnóstico predominante, sin embargo presentan síntomas parecidos a los mencionados en el anterior clúster.

Con este análisis es correcto decir que ambas agrupaciones son similares a las creadas anteriormente por otros algoritmos, lo cual indica que no existen varias características relevantes que las diferencien.

Algoritmo DBSCAN.

A diferencia de los algoritmos mencionados, DBSCAN no usa un número predeterminado de clústeres a crear en el momento de su aplicación; lo que necesita es un valor óptimo para su parámetro *epsilon*, el cual indica la distancia máxima en la que dos puntos pueden estar entre sí mientras pertenecen al mismo clúster (Mullin, 2020). Este valor se lo encontró usando un gráfico, el cual se mostrará luego, creado mediante los resultados del algoritmo K-Vecinos más cercanos. Con este algoritmo se encontró la distancia de cada punto a su 3 tercer vecino más cercano.

Este número se utilizó debido a que inicialmente se necesitaba el parámetro *min_samples*, el cual indica el mínimo número de puntos requeridos para formar un clúster al usar DBSCAN (Mullin, 2020). Este número es recomendable inicializarlo con el doble de la dimensionalidad del dataset. Para este caso se utilizó la dimensionalidad de dos debido a que

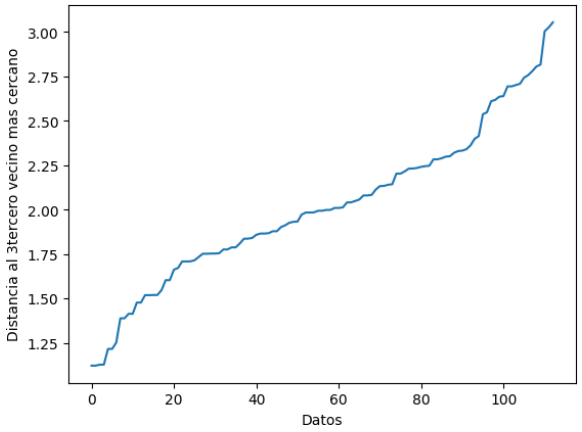
una dimensionalidad mucho más alta provocaba que el algoritmo no forme clústeres. Entonces, siendo la dimensionalidad 2, *min_samples* se igualó a 4.

Luego, el parámetro *n_neighbors* para el algoritmo de K-vecinos más cercanos terminó siendo 3 ya que este es igual a *min_samples* – 1.

A continuación, se presenta el gráfico con el que se encontró el valor de *epsilon*. En el se puede identificar el valor óptimo de dicho parámetro al encontrar una curva pronunciada a lo largo de una línea. En la siguiente figura se puede apreciar que es alrededor del valor 2.45 donde se encuentra dicha curva.

Figura 72

Gráfico que permite identificar el valor de epsilon



Fuente: Autores de este documento.

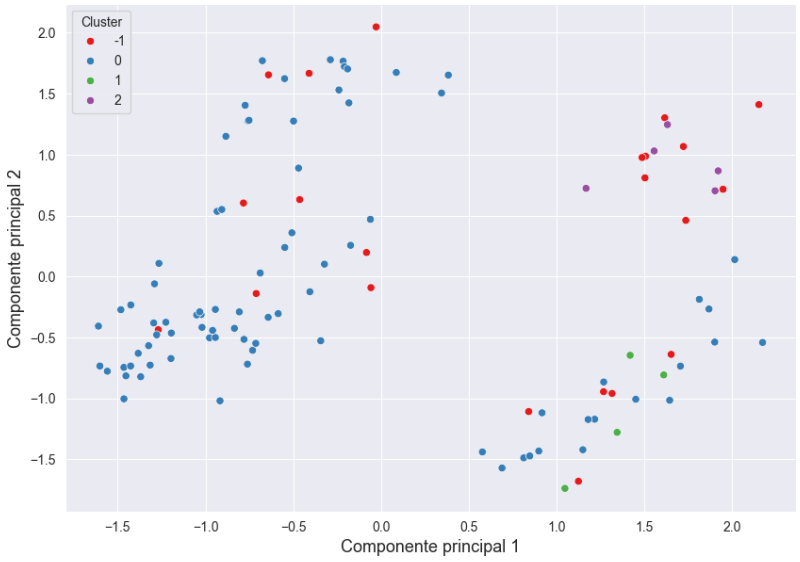
Obtenido este valor, se procedió a aplicar el algoritmo DBSCAN con los 19 componentes principales y se obtuvo en total 3 clústeres, los cuales fueron analizados para encontrar patrones que los distingan. Es importante mencionar que para este algoritmo se utilizaron solo dos componentes principales de la técnica PCA debido a que al usar más componentes principales, el algoritmo procedía a identificar todos los puntos como ruido.

Análisis de clústeres.

La siguiente figura presenta la distribución de los clústeres obtenidos

Figura 73

Clústeres obtenidos con algoritmo DBSCAN



Fuente: Autores de este documento.

Se debe aclarar que los puntos rojos fueron considerados como ruido por el algoritmo.

El clúster 0 se caracteriza por contener pacientes que cuentan con *PREMATURIDAD* como condición pre-existente, estos no cuentan con un diagnóstico predominante y presentan síntomas como *FIEBRE (MEDIDA O REPORTADA)*, *DOLOR DE GARGANTA O TOS*, etc. Todos los pacientes dieron negativo para resultado VSR y existen pacientes positivos para Influenza A y SARS-COV-2.

El clúster 1 se caracteriza por contener pacientes a los que se desconoce si cuentan con alguna condición pre-existente. Por otro lado, en esta agrupación se presenta el *ESTADO ASMATICO* como diagnóstico predominante junto con síntomas parecidos al anterior clúster. Resulta importante aclarar este punto ya que es el primer clúster que cuenta con un diagnóstico predominante diferente a *NEUMONIA*.

Por último, el clúster 2 se caracteriza por no tener un diagnóstico predominante pero sí cuenta con la mayor cantidad de síntomas predominantes de entre todas las agrupaciones.

Como se pudo notar, estos clústeres presentan como diferencias significativas el diagnóstico del Estado Asmático y la cantidad de síntomas.