



TRABAJO DE GRADO
Opción Seminario-Diplomado.

**ALGORITMO COMPUTACIONAL PARA EL ANÁLISIS Y TOMA DE DECISIONES
EN DATOS DE PREDICCIÓN DE DEFUNCIONES POR COVID-19, UTILIZANDO
ESTRATEGIAS DE MACHINE LEARNING**

Corporación Universitaria Remington.
Facultad de Ingeniería
Ingeniería de Sistemas

Estudiante:
Devis Kenneth Padilla Vargas.
Tutor: Juan Carlos Briñez de León
Opción de Trabajo de grado Seminario-Diplomado.
2024.

Dedicatoria

A mis padres, quienes han estado en toda la formación académica, apoyándome en todo momento, y por mi compañera sentimental que siempre me apoya con sus palabras de aliento de no padecer en la universidad, sin ellos todo lo que he conseguido sería en vano.

Agradecimientos

A dios, a la universidad, a mis compañeros, a él docente del seminario y demás docentes que me enseñaron, siempre es una virtud ser un aprendiz, además, conocer más personajes en mi vida es un objetivo para la superación de mis capacidades.

Tabla de contenido

Tablas de Ilustraciones	5
Tablas de Tablas	6
Resumen	7
Palabras clave	7
Marco conceptual y contextual	8
Pregunta problema:	9
Acercamiento a los datos:	9
Descripción de variables.....	10
Aproximaciones con gráficos.	16
Objetivos.....	21
Objetivo general.....	21
Objetivos específicos.....	21
Desarrollo e implementación del aprendizaje	22
Procesamiento de los datos	22
Modelo de toma de decisiones.....	26
Implementación en contextos reales	33
Resultados adicionales	34
Conclusiones.....	35
Referencias	36

Tablas de Ilustraciones

Ilustración 1. Distribución de genero.....	17
Ilustración 2. Personas contagiadas	18
Ilustración 3. Personas con Defunción.....	19
Ilustración 4. Gráfico de dispersión edad y Fecha de pacientes fallecidos.	21
Ilustración 5. Cargue de datos del covid en México.....	22
Ilustración 6. Eliminar datos vacíos.....	24
Ilustración 7. Eliminar datos innecesarios.....	24
Ilustración 8. Mostrando datos limpios.....	25
Ilustración 9. Declarando cambios en la tabla.....	25
Ilustración 10. Eliminando columna fecha Defunción.....	26
Ilustración 11. Dividir datos de entrada y de salida.....	26
Ilustración 12. Entrenando y validando datos.....	27
Ilustración 13. Normalización de los datos	27
Ilustración 14. Evaluando los clasificadores.	28
Ilustración 15. Práctica del modelo.....	30
Ilustración 16. Primera prueba.....	31
Ilustración 17. Prueba 2.....	32
Ilustración 18. Prueba propia.....	33

Tablas de Tablas

Tabla 1. Cuestionario Aplicado del Covid	9
Tabla 2. Acercamiento de los datos	10
Tabla 3. Descripción de variables	11
Tabla 4. Clave y descripción de los números del datasets	11

Resumen

En este proyecto de Machine Learning en tiempo de datos, es muy necesario saber que Datasets requiere para el ejercicio, de tal manera fueron los datos de contagios en México específicamente las defunciones de los pacientes allegados a varias zonas del país, de acuerdo con lo anterior, se logró percibir o anexar ciertas dificultades que se tuvieron en ese tiempo. Se manifestaron pacientes de diferente índole niños, mujeres en embarazo, adultos de la tercera edad, incluso personas que superaban los 100 años. Es apropiado que con el análisis de datos que se realizó se pueda identificar a corto plazo el COVID 19, aunque ya allá pasado su pandemia, aun se presentan otras versiones al COVID, de acuerdo con el análisis de posibles contagios los pacientes no sufrían padecimiento acerca de esta enfermedad, mientras que otros sí, por otra enfermedad contribuyente, incurre a que el paciente empeore y no tenga las mismas ganas de seguir que un paciente bien o en excelentes condiciones de vida. Con los datos adquirido se logra manifestar que ya muchos pacientes ya habían fallecido, por eso era necesario su análisis completo, desde el cargue de los datos, hasta el registro final que eran los datos que quedaban, además con ayudan de los clasificadores se tuvo mayor precisión al análisis de los resultados, esta es una competencia que ayuda a comprender los estados de las enfermedades, en vista previa se anexaron los datos a Colaboratory de Google, donde se hizo todo el proceso de cargue, eliminación, conteo y resultados.

Palabras clave

Defunción por Covid-19, Datasets, Análisis de datos, Machine learning, Clasificación, Regresión, Clustering, Cardiovascular, Obesidad, Renal Crónica, Tabaquismo, Resultado.

Marco conceptual y contextual

En las organizaciones, los procesos de transformación digital han permitido la consolidación de datos que registran las decisiones tomadas en su momento, y las variables asociadas a ellas. De acuerdo con la Organización Internacional del trabajo, la pandemia pone de manifiesto un escenario laboral sin precedentes, la pérdida de horas de trabajo a nivel mundial ha provocado su mayor pronosticada en el primer semestre de 2020. Esta situación puso a los países a tomar medidas de contingencia, principalmente México las empresas quebraron. En consecuencia, se abarcaron otras formas de trabajar, y la emergencia sanitaria es la relacionada con la salud mental que trajeron consigo temores, preocupaciones y perturbación en la vida diaria, de acuerdo con lo anterior, el covid-19 puso en marcha la exploración digital en un sin fin de actividades organizacionales para poder salir adelante las empresas las cuales no habían quebrado aún. También la situación primordial por este fenómeno sin precedentes logró crear incertidumbre, comportamientos sociales negativos, percepciones distorsionadas sobre el riesgo, angustia, insomnio, ira, aumento en el consumo de alcohol y tabaco, entre otros vicios. (Referencia 1)

En el campo de la medicina, los coronavirus pertenecen a la familia Coronaviridae: se trata de ARN-virus encapsulados, de sentido positivo y de una sola hebra. Existen cuatro géneros comunes extraídos y afectado principalmente a los seres humanos, murciélagos, y los predominantes a las aves.

La estructura genética está compuesta por una cadena de ARN que contiene 14 sitios de lectura y se dividen en: 16 genes de proteínas no estructurales, 4 genes estructurales que codifican las proteínas spike (S), de envoltura (E), de membrana (M) y de nucleocápside (N) y, finalmente, otros genes accesorios.

Las mutaciones suceden de manera espontánea durante los procesos de la replicación viral. Para su replicación, los virus ARN utilizan una ARN-polimerasa que es intrínsecamente propensa a cometer errores que suceden, incluso en 10^{-6} , 10^{-4} sustituciones por nucleótido por cada célula infectada. En la mayor parte de los casos el destino de estas mutaciones está determinado por la selección natural. (Referencia 2)

El brote de la enfermedad por coronavirus ha infectado a millones de personas, ocasionando una elevada tasa de mortalidad en todo el mundo. Después de la saturación de los hospitales por falta de atención por posibles síntomas de COVID-19, fue necesario contar con un modelo de predicción para clasificar a los pacientes con alto riesgo de deterioro clínico. El objetivo de la investigación fue comparar algoritmos de clasificación basados en máquinas de aprendizaje automático. Se recolectaron 1000 registros de pacientes con sospecha de infección al ingresar por el servicio de emergencia. Después del preprocesamiento de los datos e ingeniería de los atributos, se determinó una muestra de 700 registros. Se diseñaron los modelos y compararon los algoritmos: Regresión Logística, Máquina de soporte vectorial, Vecinos más cercanos, Árbol de decisión, Bosque aleatorio, y Naive Bayes... de los anteriores los de mayor concordancia a los resultados de la realidad

fueron Máquina de soporte vectorial con un 97% y Bosque aleatorio con un 95% de exactitud. (Referencia 3)

Anteriormente el coronavirus afectó una parte de China, demostrando pánico entre la comunidad, demostrando que no era una enfermedad cualquiera, siempre y cuando iban cayendo personas mediante transcurría los primeros meses de la pandemia, esta pandemia quiso demostrarle al mundo que no era un juego su ataque, nos demostró a Colombia que por medio de los viajes de las líneas aeronáuticas, personas infectadas podía ser transmitida de una forma recurrente que era, mediante el saludo, el estrecho de manos y muchas maneras más, su significado indicó el índice más alto de mortalidad en el mundo, el que se salvó fue por seguridad y aislamiento, también su resistencia era la máxima o porque nunca tuvo contacto con personas. (Referencia 4)

Así se vería una variable de cuestionario aplicado.

Componente	Subcomponente	Variable
Emoción Personal por afectación del covid 19	Desespero Encierro total Soledad Enfermedad Hambre Angustia Desempeño Virtualidad Conocimiento de aplicaciones Otras	Adaptación Insatisfacción Pérdida de empleos Aulas virtuales Trabajos virtuales Juegos Familia Tiempo Otros.

Tabla 1. Cuestionario Aplicado del Covid

Pregunta problema:

¿Cómo desarrollar una estrategia computacional para ayudar en la identificación temprana de casos por Covid-19, a partir de algoritmos de Machine Learning?

Acercamiento a los datos:

Descripción	Creadores	Enlace

Este conjunto de datos contiene los resultados de las pruebas de PCR en tiempo real para COVID-19 en México según lo informado.	Dirección General de Epidemiología.	https://www.gob.mx/salud/documentos/datos-abiertos-152127
---	-------------------------------------	---

Tabla 2. Acercamiento de los datos

Descripción de variables.

Total, de datos analizados por Collaboratory en Python: 263.006

#	Column	Non-Null Count	Dtype	Vtype
0	RESULTADO	263001 non-null	int64	0,1
1	SEXO	263001 non-null	int64	1,2
2	TIPO_PACIENTE	263001 non-null	int64	1,2
3	FECHA_DEF	263001 non-null	object	No def.
4	NEUMONIA	263001 non-null	int64	1,2,99
5	EDAD	263001 non-null	int64	No def.
6	EMBARAZO	263001 non-null	int64	1,2,97,98
7	DIABETES	263001 non-null	int64	1,2,98
8	EPOC	263001 non-null	int64	1,2,98
9	ASMA	263001 non-null	int64	1,2,98
10	INMUSUPR	263001 non-null	int64	1,2,98
11	HIPERTENSION	263001 non-null	int64	1,2,98
12	OTRA_COM	263001 non-null	int64	1,2,98
13	CARDIOVASCULAR	263001 non-null	int64	1,2,98
14	OBESIDAD	263001 non-null	int64	1,2,98

15	RENAL_CRONICA	263001	non-null	int64	1,2,98
16	TABAQUISMO	263001	non-null	int64	1,2,98
17	OTRO_CASO	263001	non-null	int64	1,2,99
18	UCI	263001	non-null	int64	1,2,97,99
19	DEFUNCION	263001	non-null	int64	0,1

Tabla 3. Descripción de variables

CLAVE	DESCRIPCIÓN
1	SI
2	NO
97	NO APLICA
98	SE IGNORA
99	NO ESPECIFICADO
1	FEMENINO
2	MASCULINO
0	NEGATIVO
1	POSITIVO

Tabla 4. Clave y descripción de los números del datasets

De acuerdo con lo anterior, hay datos que no se sabe de qué es, por eso se identificarán a continuación para una mayor comprensión:

1. Epoc: es un diagnóstico.
2. Inmusupr: inmunosupresión.

3. Otra_com: diagnóstico de otras enfermedades.
4. Otro caso: Posible contagio de otro individuo que tuvo Covid-19.

Además, se dará a conocer la terminología de los datos o variables presentados en el documento.

Al principio, la neumonía COVID-19 se presenta con las siguientes características:

- Baja elastancia: la compliance casi normal indica que la cantidad de gas en el pulmón es casi normal.
- Baja relación ventilación-perfusión (VA/Q): como el volumen de gas es casi normal, la hipoxemia puede ser mejor explicada por la pérdida de regulación de la perfusión y de la vasoconstricción hipóxica. En consecuencia, en esta etapa, la presión de la arteria pulmonar debe ser casi normal.
- Bajo peso pulmonar: solo las densidades de vidrio esmerilado están presentes en la tomografía computarizada, principalmente ubicadas a nivel subpleural y a lo largo de las fisuras pulmonares. En consecuencia, el peso pulmonar es solo moderadamente aumentado.
- Baja capacidad de reclutamiento pulmonar: la cantidad de tejido no aireado es muy bajo; en consecuencia, la capacidad de reclutamiento es bajo.

(Referencia 5)

La diabetes es una enfermedad crónica que se caracteriza por un aumento de los niveles de azúcar (glucosa) en la sangre. Esto debido a que el páncreas no produce insulina o el cuerpo no la utiliza de forma correcta.

- Tipo 1: representa entre el 5 al 10% de todos los casos de diabetes diagnosticados. Se caracteriza por una producción deficiente de insulina y requiere administración de insulina.
- Tipo 2: representa entre el 80 al 90% de todos los casos. Se caracteriza porque el cuerpo no utiliza correctamente la propia insulina y está muy asociada al sobrepeso y al sedentarismo.
- Gestacional: se caracteriza por una subida de azúcar (hiperglucemia) que aparece durante el embarazo. (Referencia 6)

El asma es una inflamación crónica que afecta a las vías respiratorias, lo que disminuye el flujo de aire que entra y sale de los pulmones. Los síntomas se presentan de manera diferente en cada paciente y varían en frecuencia y severidad.

- ★ Asma alérgica: ocurre en relación con la exposición a sustancias alérgicas que están en el ambiente, como el polen, los ácaros, el pelo de animales, los hongos y muchos más.
- ★ Asma no alérgica: no está relacionada con las alergias y los síntomas pueden aparecer por cambios bruscos de temperatura, infecciones respiratorias virales o bacterianas o exposición a sustancias irritantes.

★ Asma inducida por el ejercicio: los síntomas aparecen durante la realización del ejercicio físico o poco tiempo después de finalizar la actividad.

★ Asma ocupacional: los síntomas aparecen por la exposición a sustancias propias del lugar de trabajo como el polvo de la madera, harinas, metales, resinas plásticas, entre otros. (Referencia 7)

Inmunosupresión: supresión o disminución de las reacciones inmunitarias, puede ser debida a la administración deliberada de fármacos inmunosupresores, empleados en el tratamiento de enfermedades autoinmunes, o en receptores de órganos trasplantados para evitar el rechazo. (Referencia 8)

La hipertensión pulmonar es el incremento anómalo de la presión de la sangre en las arterias pulmonares.

- Hipertensión arterial pulmonar.
- Hipertensión pulmonar asociada a enfermedad cardiaca izquierda.
- Hipertensión pulmonar asociada a enfermedades respiratorias y/o hipoxemia (déficit de oxígeno en la sangre).
- Hipertensión pulmonar tromboembólica crónica.
- Hipertensión pulmonar de causa no aclarada o multifuncional.

(Referencia 9)

Enfermedad cardiovascular es la afección cardíaca que se manifiesta mediante vasos sanguíneos enfermos, problemas estructurales y coágulos sanguíneos. (Referencia 10)

La insuficiencia renal crónica es el deterioro progresivo e irreversible de la función renal. Es decir, los riñones pierden lentamente su capacidad de trabajo. (Referencia 11)

Posibles aplicaciones.

Se está recreando el Dataset para una posible determinación si se tiene Covid-19, por eso, hay que determinar de qué se trata cada componente, eliminando, reemplazando, y analizando su comportamiento.

Ayudará a comprender la angustia de los pacientes, sin necesidad de acercarse a un centro hospitalario, aunque son datos extraídos de México, pero aun así se logra percibir que se puede aplicar a los demás países, porque el Covid-19 en todos los países es lo mismo.

Con ayuda de las herramientas de machine learning, es posible concebir el acuerdo de consolidación de cada paciente e incluso se puede determinar el análisis por los pacientes que han fallecido.

Sin importar los resultados, si te autodiagnostico y crees que tiene los síntomas, es mejor acercarse al centro médico para un posible control y evaluación determinística, además poder aislarse y no contagiar a los seres queridos u otros individuos.

Además, el uso de wearables para la detección temprana del covid-19 en el mundo es de vital importancia ya que se facilita la recopilación de los datos y posterior información importante para la lucha contra esta enfermedad. Poder detectar patrones inusuales en pacientes de su entorno habitual, según lo dicho en este párrafo es una propuesta para el desarrollo de una aplicación de monitoreo de signos vitales a través de pulseras biométricas para la detección temprana de covid-19 y otras afecciones.

(Referencia 12)

Aproximaciones con gráficos.

```
#Cargando librerías
import seaborn as sns
import matplotlib.pyplot as plt

# Crear un gráfico de torta
frecuencias = covid_mexico_limpio['SEXO'].value_counts()
plt.figure(figsize=(7, 7)) # Tamaño del gráfico (opcional)
explode = (0.1,0)
plt.pie(frecuencias, labels=frecuencias.index, explode=explode, autopct='%1.1f%%', startangle=180)
plt.title('Diagrama de Torta de Distribución de género')
plt.show()
```

Diagrama de Torta de Distribución de género

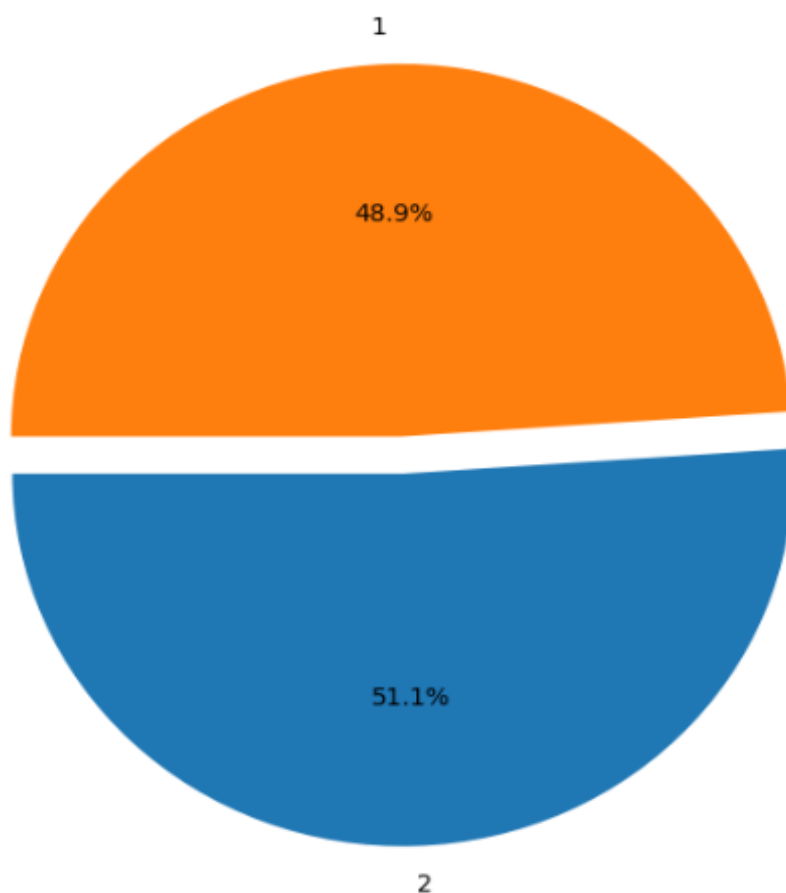


Ilustración 1. Distribución de género

De acuerdo al diagrama de torta de distribución de género, se pretende mostrar que la cantidad de personas que acudieron al hospital o lugar de atención, el cuál logra percibirse 2 géneros ocurrentes que es la parte femenina y la masculina, con ello podría hallar la una similitud muy por debajo que es del 48.9% que es de las mujeres, se representan un total de 128.610 mujeres y en lugar de los hombres con una tasa del 51.1% que sería el restante, daría como resultado 134.397 hombres, además, los hombres como se demuestra tendían a enfermarse más que las mujeres, pero casi hay una igualdad de género.

```
#Cargando librerías
import seaborn as sns
import matplotlib.pyplot as plt

# Crear un gráfico de barras con Seaborn
sns.countplot(data=covid_mexico_limpio, x='RESULTADO')
plt.title('Personas Con o sin Contagio')
plt.xlabel('Resultado')
plt.ylabel('Frecuencia')
plt.show()
```

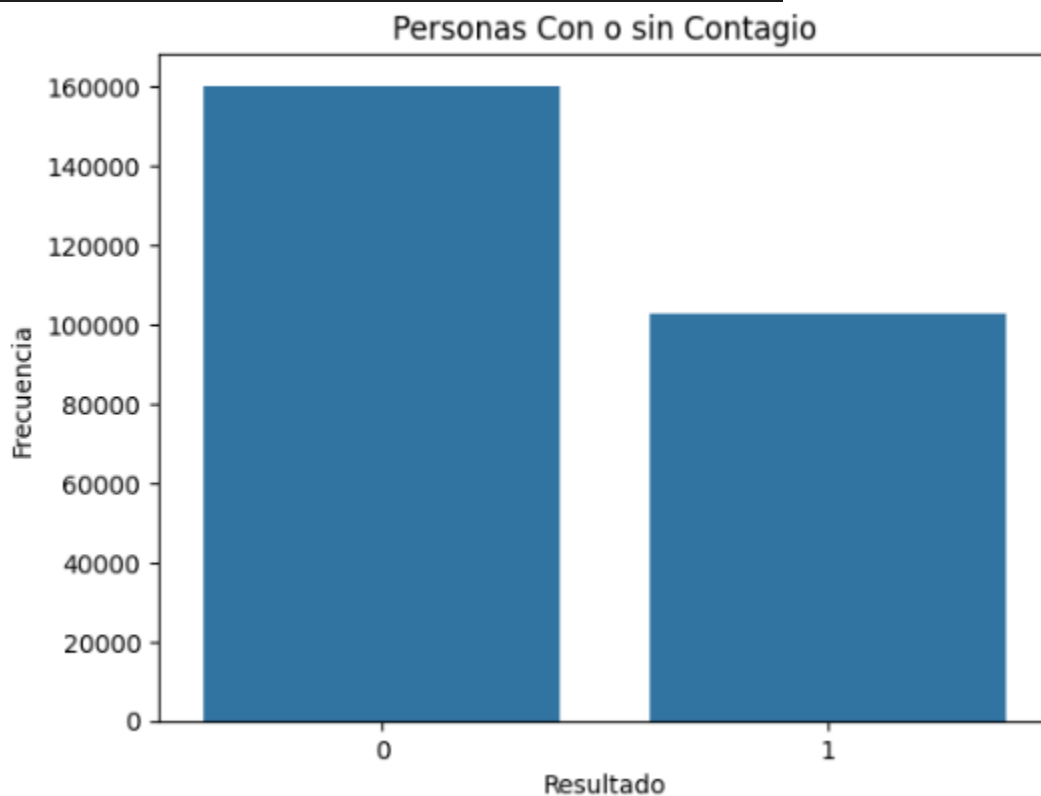


Ilustración 2. Personas contagiadas

Las personas con contagio que son el número 1, son las que más pretendían a tener una muerte segura, aunque los que no tenían contagio, podrían determinarse como no muy limpios porque hay otras enfermedades que podrá atacarse el Covid-19, sin importar la condición en que se encuentre siempre se presenta un contagio mínimo sin importar su estado de salud excelente, aunque en la frecuencia demuestra que las personas que no tienen contagio (0) llega casi a los 160.000 lo que es buen índice de control con esta enfermedad que molestó a muchos en esos momentos.

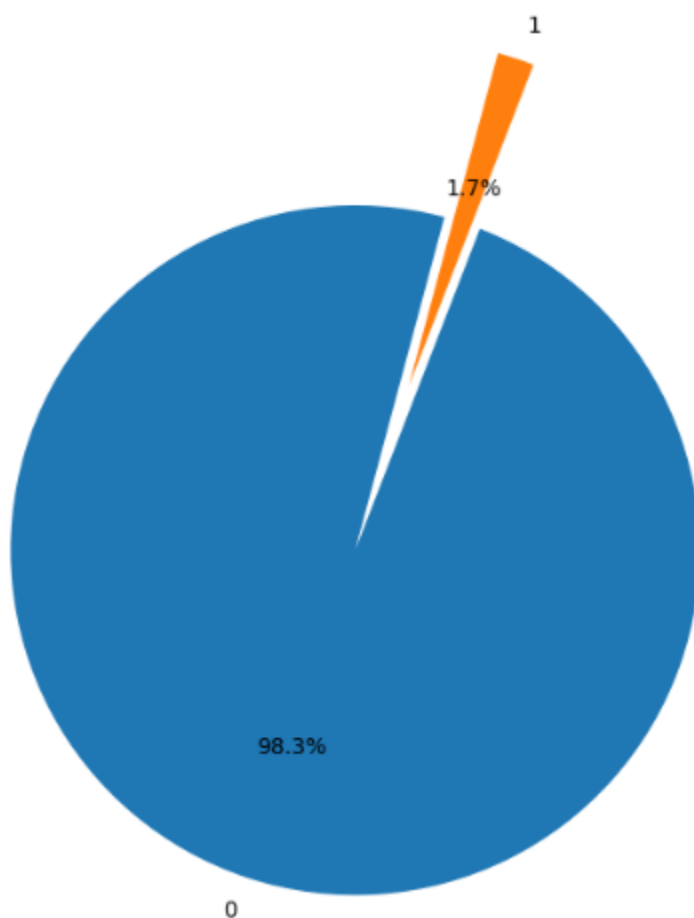
```

#Cargando librerías
import seaborn as sns
import matplotlib.pyplot as plt

# Crear un gráfico de torta
frecuencias = covid_mexico_limpio['DEFUNCION'].value_counts()
plt.figure(figsize=(7, 7)) # Tamaño del gráfico (opcional)
explode = (0.5,0)
plt.pie(frecuencias, labels=frecuencias.index, explode=explode, autopct='%1.1f%%', startangle=75)
plt.title('Diagrama de Torta de Defunción')
plt.show()

```

Diagrama de Torta de Defunción

*Ilustración 3. Personas con Defunción.*

Este un tema muy adyacente a tener una muerte segura con este tipo de enfermedad porque no se tenía una cura rápidamente, lo que acudió a que muchas personas no todas perdieran la vida sin respuesta alguna, por eso, pretendo demostrarles que los resultados son muy bajos acorde al control de esos años, se representó una tasa del 1,7% es decir 4.471 personas perdieron la vida por esta enfermedad porque no se

pudo diagnosticar a corto tiempo y los diagnósticos eran muy bajos porque el covid-19 a los que más atacaba era a personas vulnerables de salud.

```
#Cargando librerías
import seaborn as sns
import matplotlib.pyplot as plt

# Crear un gráfico de dispersión con Seaborn
sns.scatterplot(data = covid_mexico_limpio, x='EDAD', y='FECHA_DEF')
sns.set(rc={'figure.figsize':(17,25)},font_scale=1.1)
plt.title('Gráfico de Dispersión')
plt.xlabel('Edad')
plt.ylabel('Fecha Defunción')
plt.show()
```

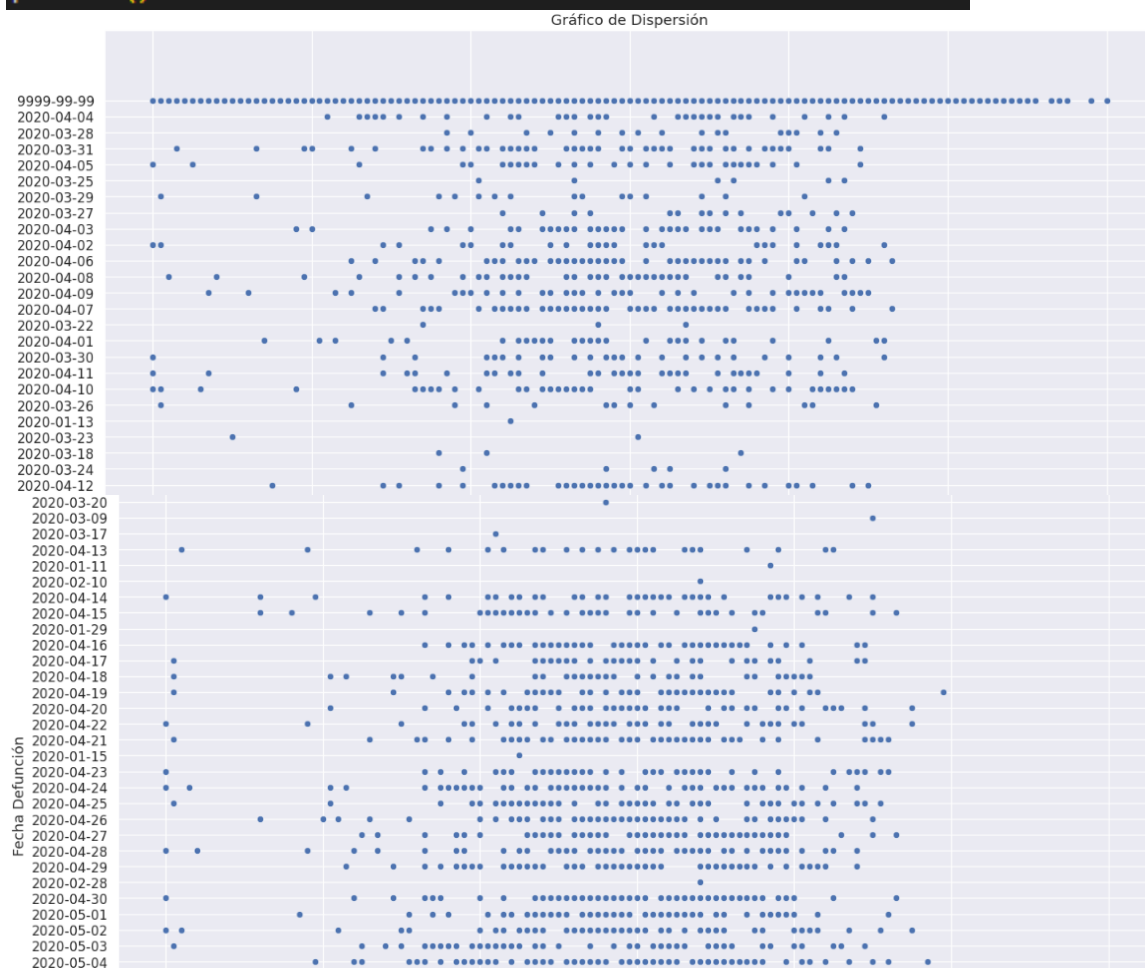




Ilustración 4. Gráfico de dispersión edad y Fecha de pacientes fallecidos.

De acuerdo al gráfico de dispersión se puede concebir que los puntos más críticos de mortalidad fueron las personas que tenían entre 40 y 80 años de edad, es donde se demuestra que a esas personas su nivel de efectividad de esta enfermedades era muy padeciente e incontrolable, aunque hubieron niños que cayeron en esa tragedia, sus defensas eran muy bajas, también se puede observar que hubieron personas muy mayores que pudieron salir de ese peligro, tal vez porque tenían más cuidado o las alejaban más para no contagiarlas, estos pacientes superan los 100 años de edad.

Objetivos:

Objetivo general.

Implementar un algoritmo computacional para el análisis y toma de decisiones a partir de datos de la predicción de defunción por covid 19, utilizando estrategias de machine learning.

Objetivos específicos.

- Caracterizar el procesamiento de los datos de interés, con miras a la toma de decisiones informadas.
- Implementar un algoritmo de Machine learning para la toma de decisiones a partir de los datos de interés.
- Evaluar el análisis en el desempeño de los algoritmos implementados para la toma de decisiones.
- Validar el funcionamiento de toma de decisiones a partir de datos nuevos.

Desarrollo e implementación del aprendizaje

Procesamiento de los datos

1.1. Cargue de datos del Covid-19 de México.

```
#Cargando datos del covid 19 en México
import pandas as pd
from google.colab import files
uploaded = files.upload()
for filename in uploaded.keys():
    Datos_Load = pd.read_csv(filename, sep=',')

Datos_Load.head()
```

Elige archivo: mexico_covid19.csv
 • mexico_covid19.csv(text/csv) - 48056170 bytes, last modified: 27/3/2024 - 100% done
 Saving mexico_covid19.csv to mexico_covid19.csv

id	FECHA_ARCHIVO	ID_REGISTRO	ENTIDAD_UN	ENTIDAD_RES	RESULTADO	DELAY	ENTIDAD_REGISTRO	ENTIDAD	ABR_ENT	...	OTRA_COM	CARDIOVASCULAR	OBESIDAD	RENAL_CRONICA	TABAQUISMO	OTRO_CASO	MIGRANTE	PAIS_NACIONALIDAD	PAIS_ORIGEN	UCI	
0	9269	2020-04-12	00011f	25	25	2	0	25	Sinaloa	SL	...	2	2	1	2	2	2	99	México	97	97
1	33333	2020-04-12	00014e	14	14	2	0	14	Jalisco	JC	...	2	1	2	1	99	99	México	97	2	
2	35483	2020-04-12	000153	8	8	1	0	8	Chihuahua	CH	...	2	2	2	2	99	99	México	97	2	
3	7062	2020-04-12	0001b6	9	15	1	0	9	Ciudad de Mexico	DF	...	2	1	2	2	99	99	México	97	97	
4	23745	2020-04-12	0001c1	9	9	2	0	9	Ciudad de Mexico	DF	...	2	2	2	2	99	99	México	97	97	

5 rows • 41 columns

Ilustración 5. Cargue de datos del covid en México

Como se muestra en la base de datos (mexico_covid19.csv) son muchos los datos que tiene, por eso, se pretende acortar la mayoría de los datos innecesarios y eliminar celdas en blanco como se muestra en lo siguiente:

1.2. Limpieza de datos.

```
# Identificar valores faltantes
print(Datos_Loan.isnull().sum())

# Eliminar filas con valores faltantes
Datos_Loan = Datos_Loan.dropna()
```

```
id                0
FECHA_ARCHIVO     0
ID_REGISTRO       0
ENTIDAD_UM        0
ENTIDAD_RES       0
RESULTADO         0
DELAY            0
ENTIDAD_REGISTRO  0
ENTIDAD           0
ABR_ENT           0
FECHA_ACTUALIZACION 0
ORIGEN           0
SECTOR           0
SEXO             0
ENTIDAD_NAC      0
MUNICIPIO_RES    6
TIPO_PACIENTE    0
FECHA_INGRESO    0
FECHA_SINTOMAS   0
FECHA_DEF        0
INTUBADO         0
NEUMONIA        0
EDAD            0
NACIONALIDAD     0
EMBARAZO        0
HABLA Lengua Indig 0
DIABETES        0
EPOC            0
ASMA            0
INMUSUPR        0
HIPERTENSION     0
OTRA_COM        0
CARDIOVASCULAR  0
OBESIDAD        0
RENAL_CRONICA   0
TABAQUISMO      0
OTRO_CASO       0
MIGRANTE        0
PAIS_NACIONALIDAD 0
PAIS_ORIGEN     0
UCI             0
dtype: int64
```

Ilustración 6. Eliminar datos vacíos

No se identificaron valores nulos en la base de datos, ni se demostró falencias de datos, como se muestra en la ilustración anterior todos los resultados están en blanco, es satisfactorio trabajar con todos los datos de la base de datos.

1.3. Eliminar datos innecesarios del Datasets.

```
columnas_conservar = ['RESULTADO', 'SEXO', 'TIPO_PACIENTE', 'FECHA_DEF', 'NEUMONIA', 'EDAD',
                     'EMBARAZO', 'DIABETES', 'EPOC', 'ASMA', 'INMUSUPR', 'HIPERTENSION', 'OTRA_COM', 'CARDIOVASCULAR',
                     'OBESIDAD', 'RENAL_CRONICA', 'TABAQUISMO', 'OTRO_CASO', 'UCI']
covid_mexico_limpio = Datos_Loan[columnas_conservar]
covid_mexico_limpio.dropna()
covid_mexico_limpio.shape

(263001, 19)
```

Ilustración 7. Eliminar datos innecesarios.

Como se muestra anteriormente, las columnas a conservar son las resaltadas en rojo, se eliminó como el ID, entidad, entre muchas más, las cuales no se pueden analizar, bueno sí, pero ya son para otros temas.

```
#mostrar datos limpios
covid_mexico_limpio.head(20)
```

	RESULTADO	SEXO	TIPO_PACIENTE	FECHA_DEF	NEUMONIA	EDAD	EMBARAZO	DIABETES	EPOC	ASMA	INMUSUPR	HIPERTENSION	OTRA_COM	CARDIOVASCULAR	OBESIDAD	RENAL_CRONICA	TABAQUISMO	OTRO_CASO	UCI
0	2	2	1	9999-99-99	2	74	97	1	2	2	2	1	2	2	1	2	2	2	97
1	2	1	2	9999-99-99	2	71	2	1	1	2	2	1	2	2	1	2	1	99	2
2	1	2	2	9999-99-99	1	50	97	2	2	2	2	2	2	2	2	2	2	99	2
3	1	1	1	9999-99-99	2	25	2	2	2	2	2	2	2	2	1	2	2	99	97
4	2	1	1	9999-99-99	2	28	2	2	2	2	2	2	2	2	2	2	2	99	97
5	2	2	1	9999-99-99	2	67	97	2	2	2	2	1	2	2	1	2	1	99	97
6	2	1	1	9999-99-99	2	44	2	2	2	2	1	2	2	2	1	2	1	99	97
7	2	2	1	9999-99-99	2	62	97	2	2	2	2	2	2	2	1	2	2	1	97
8	2	1	1	9999-99-99	2	30	2	2	2	2	2	2	2	2	2	2	2	99	97
9	2	2	1	9999-99-99	2	30	97	2	2	2	2	2	2	2	2	2	2	2	97
10	2	1	1	9999-99-99	2	32	2	2	2	2	2	2	2	2	2	2	2	99	97
11	2	1	1	9999-99-99	2	53	2	2	2	2	2	1	2	2	2	2	1	2	97
12	2	1	1	9999-99-99	2	39	1	2	2	2	2	2	2	2	2	2	2	99	97
13	2	1	2	9999-99-99	1	42	2	2	2	2	2	2	1	2	2	2	2	99	2
14	2	2	1	9999-99-99	2	45	97	2	2	2	2	2	2	2	2	2	2	99	97
15	2	1	1	9999-99-99	2	23	2	2	2	2	2	2	2	2	2	2	2	1	97
16	2	1	1	9999-99-99	2	36	2	2	2	2	2	2	2	2	2	2	2	99	97
17	2	2	1	9999-99-99	2	8	97	2	2	2	2	2	2	2	2	2	2	99	97
18	1	1	1	9999-99-99	2	29	2	2	2	2	2	2	2	2	2	2	2	99	97
19	2	1	1	9999-99-99	2	34	2	2	2	2	2	1	2	2	2	2	2	2	97

Ilustración 8. Mostrando datos limpios.

Así quedaron los datos limpios para unos análisis gráficos que se encuentran anteriormente, son columnas manejables porque son datos enteros y la interpretación es más rápida y sencilla.

1.3. Declarando datos a la fecha de Defunción, Defunción y Resultado.

```
covid_mexico_limpio.loc[:, 'RESULTADO'] = covid_mexico_limpio['RESULTADO'].replace(2, 0)
covid_mexico_limpio.loc[:, 'DEFUNCION'] = covid_mexico_limpio['FECHA_DEF'] != '9999-99-99'
covid_mexico_limpio.loc[:, 'DEFUNCION'] = covid_mexico_limpio['DEFUNCION'].map({True: 1, False: 0})
covid_mexico_limpio.head(100)
```

	RESULTADO	SEXO	TIPO_PACIENTE	FECHA_DEF	NEUMONIA	EDAD	EMBARAZO	DIABETES	EPOC	ASNA	INMUSUPR	HIPERTENSION	OTRA_COM	CARDIOVASCULAR	OBESIDAD	RENAL_CRONICA	TABAQUISMO	OTRO_CASO	UCI	DEFUNCION	
0	0	2	1	9999-99-99	2	74	97	1	2	2	2	2	1	2	2	1	2	2	2	97	0
1	0	1	2	9999-99-99	2	71	2	1	1	2	2	2	1	2	2	1	2	1	99	2	0
2	1	2	2	9999-99-99	1	50	97	2	2	2	2	2	2	2	2	2	2	2	99	2	0
3	1	1	1	9999-99-99	2	25	2	2	2	2	2	2	2	2	2	1	2	2	99	97	0
4	0	1	1	9999-99-99	2	28	2	2	2	2	2	2	2	2	2	2	2	2	99	97	0
...
95	0	2	1	9999-99-99	2	45	97	2	2	2	2	2	2	2	2	2	2	2	2	97	0
96	1	2	2	2020-03-28	1	53	97	1	2	2	1	1	1	1	1	1	1	2	2	2	1
97	0	1	1	9999-99-99	2	45	2	2	2	2	2	2	2	2	2	2	2	2	2	97	0
98	0	1	2	9999-99-99	2	34	2	2	2	1	2	2	2	2	2	2	2	2	99	2	0
99	0	1	2	9999-99-99	1	43	2	2	2	2	2	2	2	2	2	1	2	2	99	2	0

100 rows x 20 columns

Ilustración 9. Declarando cambios en la tabla.

Como se muestra en la ilustración, aparece una nueva columna que se llama defunción, que se pretende con esto, es que viene de acuerdo a la fecha de defunción la cual muestra fechas donde los pacientes han muerto, y se les tiene un diagnóstico de las enfermedades, de acuerdo con lo anterior se muestra su resultado como 1 para positivo que ya tienen una fecha de fallecimiento y 0 que no tienen riesgo aún de padecer, aunque la modificación del resultado son también ceros y unos, no indica que si esta contagiado va a morir o ya murió, indica que hay que proveer o combatir esta enfermedad.

1.4. Eliminando la columna Fecha Def.

```

columnas_conserva = ['RESULTADO', 'SEXO', 'TIPO_PACIENTE', 'NEUMONIA', 'EDAD',
                    'EMBARAZO', 'DIABETES', 'EPOC', 'ASMA', 'INMUSUPR', 'HIPERTENSION', 'OTRA_COM', 'CARDIOVASCULAR',
                    'OBESIDAD', 'RENAL_CRONICA', 'TABAQUISMO', 'OTRO_CASO', 'UCI', 'DEFUNCION']
covid_19 = covid_mexico_limpio[columnas_conserva]
covid_19.dropna()
covid_19.shape

(263001, 19)

```

```

#mostrando nueva tabla
covid_19.head()

```

	RESULTADO	SEXO	TIPO_PACIENTE	NEUMONIA	EDAD	EMBARAZO	DIABETES	EPOC	ASMA	INMUSUPR	HIPERTENSION	OTRA_COM	CARDIOVASCULAR	OBESIDAD	RENAL_CRONICA	TABAQUISMO	OTRO_CASO	UCI	DEFUNCION
0	0	2	1	2	74	97	1	2	2	2	1	2	2	1	2	2	2	97	0
1	0	1	2	2	71	2	1	1	2	2	1	2	2	1	2	1	99	2	0
2	1	2	2	1	50	97	2	2	2	2	2	2	2	2	2	2	99	2	0
3	1	1	1	2	25	2	2	2	2	2	2	2	2	1	2	2	99	97	0
4	0	1	1	2	28	2	2	2	2	2	2	2	2	2	2	2	99	97	0

Ilustración 10. Eliminando columna fecha Defunción.

Que se pretende con eliminar la columna fecha de defunción, ayuda a entender mejor el mapeo que se va hacer en la toma de decisiones (Clasificación), porque ya podemos reflejar con DEFUNCIÓN que es la columna la cual se va a evaluar.

Modelo de toma de decisiones

1.1. Mapeando los datos de entrada y de salida.

```

#Divide datos en entradas y salidas
import numpy as np
Datos_matriz=np.array(covid_19)
#Datos_matriz[np.isnan(Datos_matriz)] = 0
X = Datos_matriz[:,0:18] #datos de entrada (Todas las variables de los pacientes)
Y = Datos_matriz[:, -1] #Datos de salida (La decisión de la defunción)

```

Ilustración 11. Dividir datos de entrada y de salida.

- Se determina una matriz para los datos de entrada.
- Los datos de entrada son todas las columnas (de la cero a la 18).
- Los datos de salida son representados como la última columna (-1).

1.2. Dividir los datos de entrenamiento.

```
# Divide datos en Entrenamiento y validación
import sklearn
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test= train_test_split(X,Y,test_size=0.1,random_state=20000)
```

Ilustración 12. Entrenando y validando datos.

Se escogen los datos de entrenamiento y validación para que los diferentes programas trabajen con mejor precisión al asertividad de los resultados. En lo cual se le asignaron 20.000 datos de entrenamiento a la matriz.

1.3. Normalizar datos.

```
#Para mejorar la escala de los datos se hace normalización
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

Ilustración 13. Normalización de los datos

Transformar los datos para las pruebas necesarias, donde se pueda interpretar los resultados adyacentes al dataset.

1.4. Evaluación de casos de los clasificadores.

```

# Evaluando casos mediante todos los clasificadores
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.discriminant_analysis import QuadraticDiscriminantAnalysis
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score, precision_score

Modelo_0 = KNeighborsClassifier(5)
Modelo_0.fit(X_train, Y_train)
Y_pred_0 =Modelo_0.predict (X_test)
print("Accuracy KNN",accuracy_score(Y_test, Y_pred_0))

Modelo_1 = GaussianNB()
Modelo_1.fit(X_train, Y_train)
Y_pred =Modelo_1.predict (X_test)
print("Accuracy Bayes",accuracy_score(Y_test, Y_pred))

Modelo_2 = LinearDiscriminantAnalysis()
Modelo_2.fit(X_train, Y_train)
Y_pred_2 =Modelo_2.predict (X_test)
print("Accuracy LDA",accuracy_score(Y_test, Y_pred_2))

Modelo_3 = QuadraticDiscriminantAnalysis()
Modelo_3.fit(X_train, Y_train)
Y_pred_3 =Modelo_3.predict (X_test)
print("Accuracy QDA",accuracy_score(Y_test, Y_pred_3))

Modelo_4 = DecisionTreeClassifier()
Modelo_4.fit(X_train, Y_train)
Y_pred_4 =Modelo_4.predict (X_test)
print("Accuracy Tree",accuracy_score(Y_test, Y_pred_4))

Modelo_5 = SVC()
Modelo_5.fit(X_train, Y_train)
Y_pred_5 =Modelo_5.predict (X_test)
print("Accuracy SVM",accuracy_score(Y_test, Y_pred_5))

Accuracy KNN 0.9825481920839512
Accuracy Bayes 0.9763126877305045
Accuracy LDA 0.98258621345196
Accuracy QDA 0.9765027945705487
Accuracy Tree 0.975362153530284
Accuracy SVM 0.9832705980761188

```

Ilustración 14. Evaluando los clasificadores.

Todos los clasificadores son importantes porque demuestran la validez de los datos, pero, aun así, debemos elegir con mayor o menor posibilidad coherencia en la elección de los resultados.

1.5. Resultados con los mismos datos.

```

#Probando el modelo entrenado sobre un nuevo sujeto
Target=np.zeros((1,18))
#1 SI
#2 NO
#97 NO APLICA
#98 SE IGNORA
#99 NO ESPECIFICADO
#1 FEMENINO
#2 MASCULINO
#0 NEGATIVO
#1 POSITIVO
Target[0,0]=float(input('Ingrese el resultado 0, 1: '))
Target[0,1]=float(input('Ingrese el sexo 1, 2: '))
Target[0,2]=float(input('Ingrese el tipo de paciente 1, 2: '))
Target[0,3]=float(input('Ingrese el estado de la neumonia 1, 2, 99: '))
Target[0,4]=float(input('Ingrese la edad: '))
Target[0,5]=float(input('Ingrese el estado de embarazo 1, 2, 97, 98: '))
Target[0,6]=float(input('Ingrese el estado de la diabetes 1, 2, 98: '))
Target[0,7]=float(input('Ingrese el resultado de la prueba del epoc 1, 2, 98: '))
Target[0,8]=float(input('Ingrese el estado del asma 1, 2, 98: '))
Target[0,9]=float(input('Ingrese la inmunosupresión 1, 2, 98: '))
Target[0,10]=float(input('Ingrese el estado de la hipertension 1, 2, 98: '))
Target[0,11]=float(input('Ingrese el diagnostico de otras enfermedades 1, 2, 98: '))
Target[0,12]=float(input('Ingrese el estado cardiovascular 1, 2, 98: '))
Target[0,13]=float(input('Ingrese el estado de la obesidad 1, 2, 98: '))
Target[0,14]=float(input('Ingrese el estado de la renal cronica 1, 2, 98: '))
Target[0,15]=float(input('Ingrese el estado de tabaquismo 1, 2, 98: '))
Target[0,16]=float(input('Ingrese si tuviste contacto con otro sujeto con covid 1, 2, 99: '))
Target[0,17]=float(input('Ingrese el estado de UCI 1, 2, 97, 99: '))

Target = scaler.transform(Target) #Normalizar los datos

Prediction_0 =Modelo_0.predict (Target)
Prediction_1 =Modelo_1.predict (Target)
Prediction_2 =Modelo_2.predict (Target)
Prediction_3 =Modelo_3.predict (Target)
Prediction_4 =Modelo_4.predict (Target)
Prediction_5 =Modelo_5.predict (Target)

```

```
print(" ")

if Prediction_0==1:
    print("Según KNN, Posible muerte")
else:
    print("Según KNN, Sin preocupacion a defunción")

print(" ")

if Prediction_1==1:
    print("Según Bayes, Posible muerte")
else:
    print("Según Bayes, Sin preocupacion a defunción")

print(" ")

if Prediction_2==1:
    print("Según LDA, Posible muerte")
else:
    print("Según LDA, Sin preocupacion a defunción")

print(" ")

if Prediction_3==1:
    print("Según QDA, Posible muerte")
else:
    print("Según QDA, Sin preocupacion a defunción")

print(" ")

if Prediction_4==1:
    print("Según Tree, Posible muerte")
else:
    print("Según tree, Sin preocupacion a defunción")

print(" ")

if Prediction_5==1:
    print("Según SVM, Posible muerte")
else:
    print("Según SVM, Sin preocupacion a defunción")

print(" ")
```

Ilustración 15. Práctica del modelo.

```
Ingrese el resultado 0, 1: 1
Ingrese el sexo 1, 2: 1
Ingrese el tipo de paciente 1, 2: 2
Ingrese el estado de la neumonia 1, 2, 99: 1
Ingrese la edad: 53
Ingrese el estado de embarazo 1, 2, 97, 98: 2
Ingrese el estado de la diabetes 1, 2, 98: 1
Ingrese el resultado de la prueba del epoc 1, 2, 98: 1
Ingrese el estado del asma 1, 2, 98: 2
Ingrese la inmunosupresión 1, 2, 98: 2
Ingrese el estado de la hipertension 1, 2, 98: 2
Ingrese el diagnostico de otras enfermedades 1, 2, 98: 2
Ingrese el estado cardiovascular 1, 2, 98: 2
Ingrese el estado de la obesidad 1, 2, 98: 1
Ingrese el estado de la renal cronica 1, 2, 98: 2
Ingrese el estado de tabaquismo 1, 2, 98: 2
Ingrese si tuviste contacto con otro sujeto con covid 1, 2, 99: 99
Ingrese el estado de UCI 1, 2, 97, 99: 2

Según KNN, Posible muerte

Según Bayes, Sin preocupacion a defunción

Según LDA, Sin preocupacion a defunción

Según QDA, Sin preocupacion a defunción

Según Tree, Posible muerte

Según SVM, Sin preocupacion a defunción
```

Ilustración 16. Primera prueba.

El paciente consultado de la base de datos si era una persona que ya había defunción, por eso dos de los clasificadores lo identificaron, y su resultado era positivo. La herramienta de machine learning tiene efectividad acerca de los datos asignados.

```
Ingrese el resultado 0, 1: 0
Ingrese el sexo 1, 2: 2
Ingrese el tipo de paciente 1, 2: 1
Ingrese el estado de la neumonia 1, 2, 99: 2
Ingrese la edad: 74
Ingrese el estado de embarazo 1, 2, 97, 98: 97
Ingrese el estado de la diabetes 1, 2, 98: 1
Ingrese el resultado de la prueba del epoc 1, 2, 98: 2
Ingrese el estado del asma 1, 2, 98: 2
Ingrese la inmunosupresión 1, 2, 98: 2
Ingrese el estado de la hipertension 1, 2, 98: 1
Ingrese el diagnostico de otras enfermedades 1, 2, 98: 2
Ingrese el estado cardiovascular 1, 2, 98: 2
Ingrese el estado de la obesidad 1, 2, 98: 1
Ingrese el estado de la renal cronica 1, 2, 98: 2
Ingrese el estado de tabaquismo 1, 2, 98: 2
Ingrese si tuviste contacto con otro sujeto con covid 1, 2, 99: 2
Ingrese el estado de UCI 1, 2, 97, 99: 97

Según KNN, Sin preocupacion a defunción
Según Bayes, Sin preocupacion a defunción
Según LDA, Sin preocupacion a defunción
Según QDA, Sin preocupacion a defunción
Según tree, Sin preocupacion a defunción
Según SVM, Sin preocupacion a defunción
```

Ilustración 17. Prueba 2.

El paciente tomado de la base de datos dio negativo su análisis porque, en sí, no tenía ninguna anomalía defectuosa ante este COVID 19, ni otras enfermedades malignas. Aunque los datos anexados puedan salir bien, pero, aun así, se debe tratar otras enfermedades que se presentan en ese momento, no solo el covid te puede hacer daño.

Implementación en contextos reales

Ingrese el resultado 0, 1: 0
 Ingrese el sexo 1, 2: 2
 Ingrese el tipo de paciente 1, 2: 2
 Ingrese el estado de la neumonia 1, 2, 99: 99
 Ingrese la edad: 25
 Ingrese el estado de embarazo 1, 2, 97, 98: 97
 Ingrese el estado de la diabetes 1, 2, 98: 2
 Ingrese el resultado de la prueba del epoc 1, 2, 98: 98
 Ingrese el estado del asma 1, 2, 98: 2
 Ingrese la inmunosupresión 1, 2, 98: 2
 Ingrese el estado de la hipertension 1, 2, 98: 2
 Ingrese el diagnostico de otras enfermedades 1, 2, 98: 2
 Ingrese el estado cardiovascular 1, 2, 98: 2
 Ingrese el estado de la obesidad 1, 2, 98: 1
 Ingrese el estado de la renal cronica 1, 2, 98: 2
 Ingrese el estado de tabaquismo 1, 2, 98: 2
 Ingrese si tuviste contacto con otro sujeto con covid 1, 2, 99: 2
 Ingrese el estado de UCI 1, 2, 97, 99: 97

Según KNN, Sin preocupacion a defunción

Según Bayes, Sin preocupacion a defunción

Según LDA, Sin preocupacion a defunción

Según QDA, Sin preocupacion a defunción

Según tree, Sin preocupacion a defunción

Según SVM, Sin preocupacion a defunción

Ilustración 18. Prueba propia.

La comprobación de los datos nuevos, fueron personales, es decir, me auto diagnosticué y mis resultados salieron negativos, incluyendo las posibles enfermedades que se manifiestan en el transcurso de los años. Mi familia padece de diabetes, pero hasta el momento no he tenido problemas con esta enfermedad, lo único que si se me resalta es la obesidad, del resto soy una persona sana y sin síntomas malintencionados.

Resultados adicionales

El Covid 19 es la primer causante de una pandemia a nivel mundial, desde hace más de 100 años no se presentaba una causante así, es la sintomatología de esta enfermedad no se desarrolla de la misma forma en todos los pacientes, por lo que en muchas ocasiones no puede ser diagnosticada de manera oportuna. El uso de imágenes radiológicas de tórax se ha presentado como una herramienta alternativa de diagnóstico, ya que estas podrían potencialmente ayudar a detectar la existencia de COVID-19 en los pacientes, además de visualizar la gravedad del daño presente en los pulmones. Avances recientes en las técnicas de machine learning permiten construir modelos basados en imágenes para clasificar o predecir un determinado evento con gran exactitud. Haciendo uso de estos avances se construyó un modelo de red neuronal convolucional, usando el lenguaje de programación Python, con el objetivo de predecir la enfermedad del COVID-19 mediante el análisis de sus radiografías torácicas. Adicionalmente, se elaboró una aplicación web con el objetivo de ofrecer a los profesionales de la salud una forma sencilla de acceder al modelo y sus análisis. Todas las métricas usadas para evaluar el modelo se situaron en valores mayores al 80% y una revisión de los resultados obtenidos mostró que el modelo captura características que efectivamente se encuentran relacionadas con los diagnósticos de los pacientes. Estos resultados muestran la utilidad potencial de aplicar modelos de machine learning en la detección y diagnóstico de enfermedades y abren la puerta para su aplicación en otras áreas de la medicina.

Conclusiones

- Con ayuda del seminario de machine learning en tiempo de datos, es beneficioso para todos los estudiantes que vimos estos temas, ayudan a comprender los datos de muchos sitios Web, que dan a conocer la información, para personas curiosas que desean tomar esos datos y aplicarlos a un ambiente de trabajo le pueda dar un buen uso para crear una excelente herramienta.
- Los datos del covid-19 interpreta un sin fin de cosas, es decir, enfermedades que nuestro cuerpo produce por diferentes situaciones, ambiente o sistema, estas causas respiratorias que conocemos no siempre atacan a los pulmones, sino que empeoran a otras enfermedades presentes en nuestro cuerpo e intentan hundirnos en la incertidumbre y desespero al padecer de COVID 19, nos aislamos para no dar más dificultades a otras personas.
- Con el dataset el cual escogí, se logra normalizar las situaciones por la que pasamos muchos de nosotros como personas, siempre es necesario analizar otras enfermedades porque el covid-19 no es solamente tos o gripa, es más que eso, la mayoría de personas olvidó de lo que sufrían y murieron pensando que era COVID, nuestro cuerpo o nuestro sistema tiene unas fases, que es el cuidado personal, a muchos que sufrieron esta enfermedad fue por causa de acercamiento a otras personas contagiadas, o sus defensas eran bajas y no aguantaba más, el desespero también incurre en estos casos, la soledad y el agotamiento.
- Esta herramienta de machine learning logró percibir resultados favorables, indica que, si puedo padecer de esta enfermedad, o es necesario cuidarse de los síntomas y de las otras enfermedades adyacentes.
- Se pudo hacer comprobaciones con la misma base datos y personal y los resultados eran los previstos, aunque se hicieron esos cambios durante el proceso de muestreo, es importante que tenga un acuerdo con lo que se debe mostrar.

[IAEGLEDGIMBGMcBGK8BwgILEC4YrwEYxwEYgASYAwCSBwkwLjEuOC4yLjSgB8aZAO&scient=gws-wiz](https://www.clinicbarcelona.org/asistencia/enfermedades/insuficiencia-renal-cronica)

Insuficiencia Renal Crónica. (n.d.). Clínic Barcelona. Retrieved April 5, 2024, from <https://www.clinicbarcelona.org/asistencia/enfermedades/insuficiencia-renal-cronica>

Verdugo-Arámbula, G., Munguía-Palma, M. E., Torres-Peralta, R., & Cirett-Galán, F. (2021). Propuesta de Diseño de Arquitectura de Software para el uso de Pulseras Biométricas para la detección temprana de COVID-19. In Avances de Investigación en Ingeniería en el Estado de Sonora (pp. 110-113). Tecnológico Nacional de México.

Franco Monroy, F. A., Intriago Celi, K. G., & Cevallos Valdiviezo, H. (2021). Predicción de estado de salud de pacientes de Covid-19 utilizando métodos de machine learning (Doctoral dissertation, ESPOL. FCNM).