Universidad del Rosario
Escuela de Ingeniería, Ciencia y Tecnología
Matemáticas Aplicadas y Ciencias de la Computación
Ingeniería de datos
Claudia Hernández
Sofia Ochoa Gutiérrez, Valentina Pabón León, Mariana Cadena Moreno
Semestre 1, 2022

Tercera Entrega

Link repositorio: https://github.com/Meiiyuu/proyectoinge.git

1. Visualización con dash

Consultas y gráficas

Primero se crean las consultas o queries recursivas en postgresql que relacionan dos columnas para recrear los escenarios posibles a analizar de la base de datos escogida. Combinando las librerías dash, pandas y psycopg2 de Python, se define un método por cada consulta, después se declaran funciones para graficarlas según el gráfico que se quiera realizar, en este caso usamos diagrama de torta, de barras, histograma y de dispersión; la librería dash nos permite visualizarlas y organizarlas en una página html. Para ejecutar el módulo, se debe ubicar la dirección donde está ubicado y correrlo en el símbolo del sistema donde ya se conectará a la base de datos y las respectivas consultas con la siguiente línea de código:

C:\python graficas.py.

Este retorna un link donde está la página con todas las gráficas definidas con anterioridad donde se pueden analizar e interactuar con ellas.

Los archivos para la visualización con dash que se usaron, el archivo **escenarios_SQL.txt** donde están las consultas en postgresql, y **graficas.py** donde se crean las gráficas con la visualización y en el archivo **Conection.py** está la sentencia para establecer la conección entre el módulo de Python y la base de datos en postgresql. Todos los archivos mencionados anteriormente se encuentran en el repositorio en la carpeta **dash.**

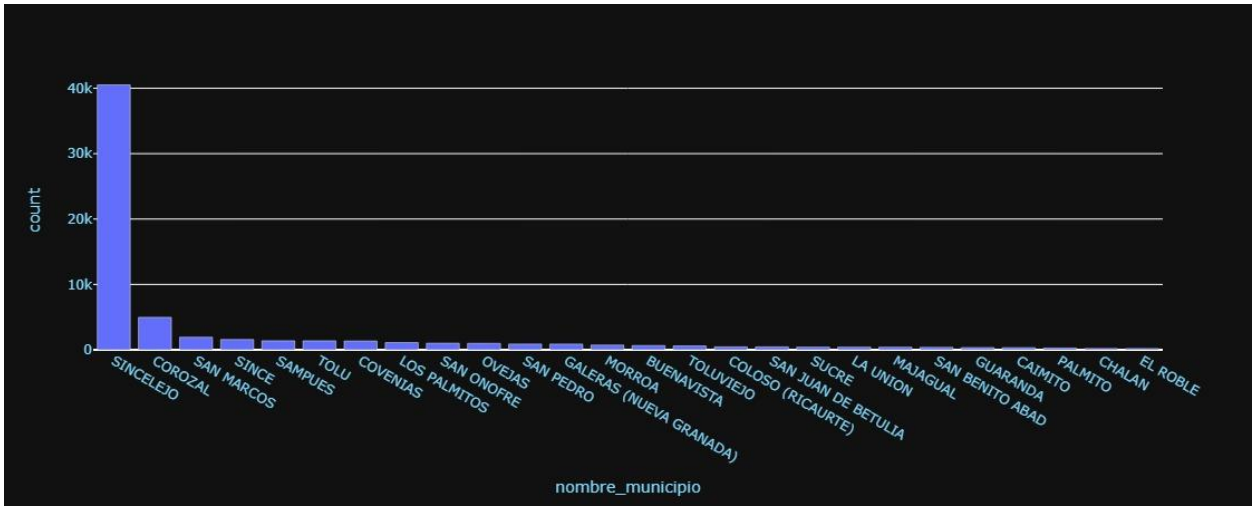

figura 1) Gráfico de barras, casos de recuperados por município.

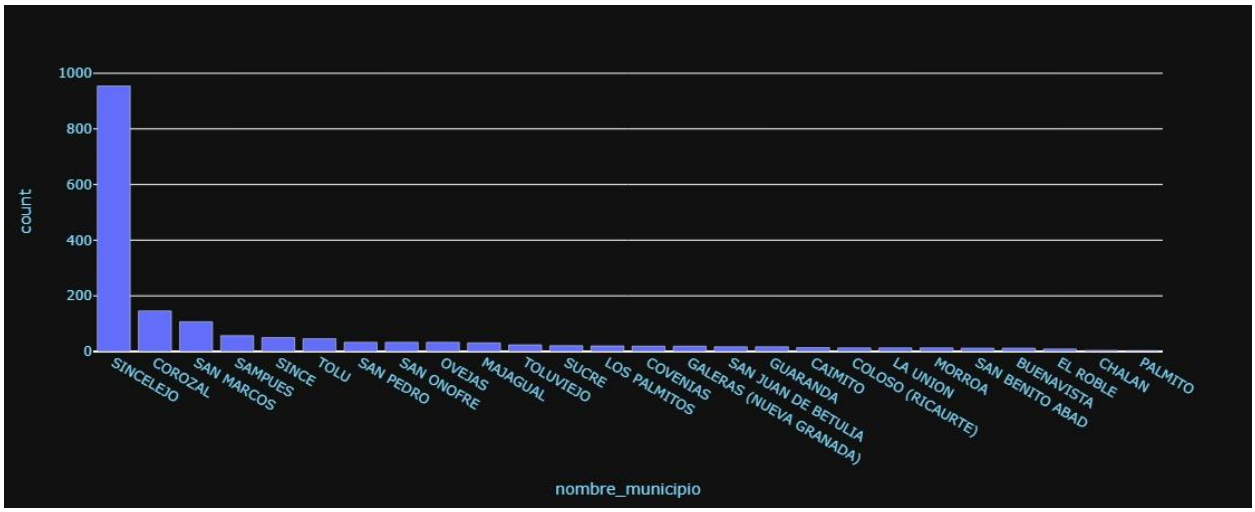

figura 2) Gráfico de barras, casos de fallecimientos por municipio.

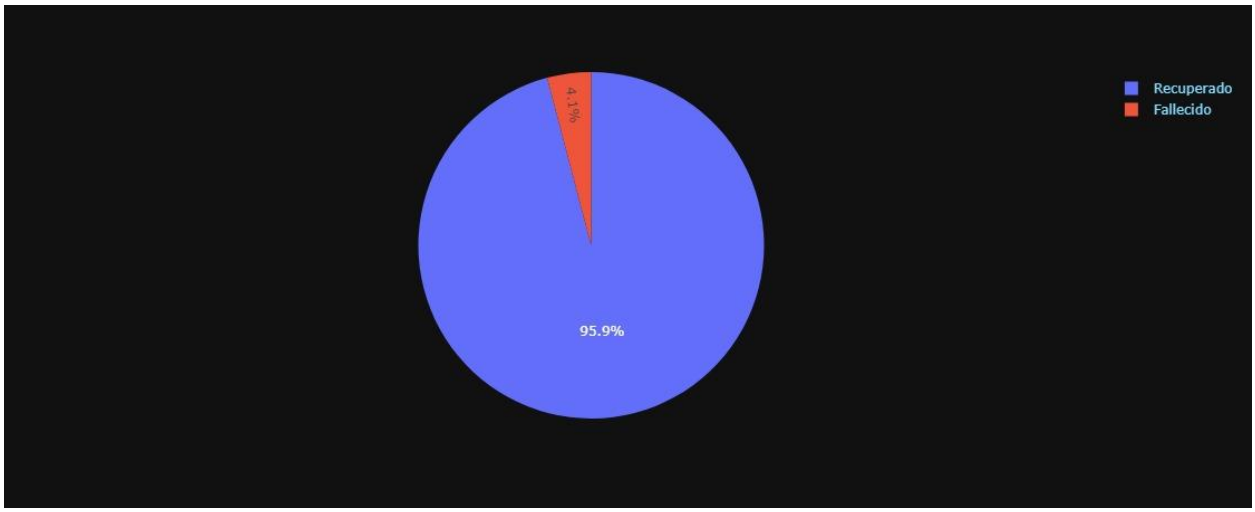

figura 3) Gráfico de torta, estado de los casos en 2020.

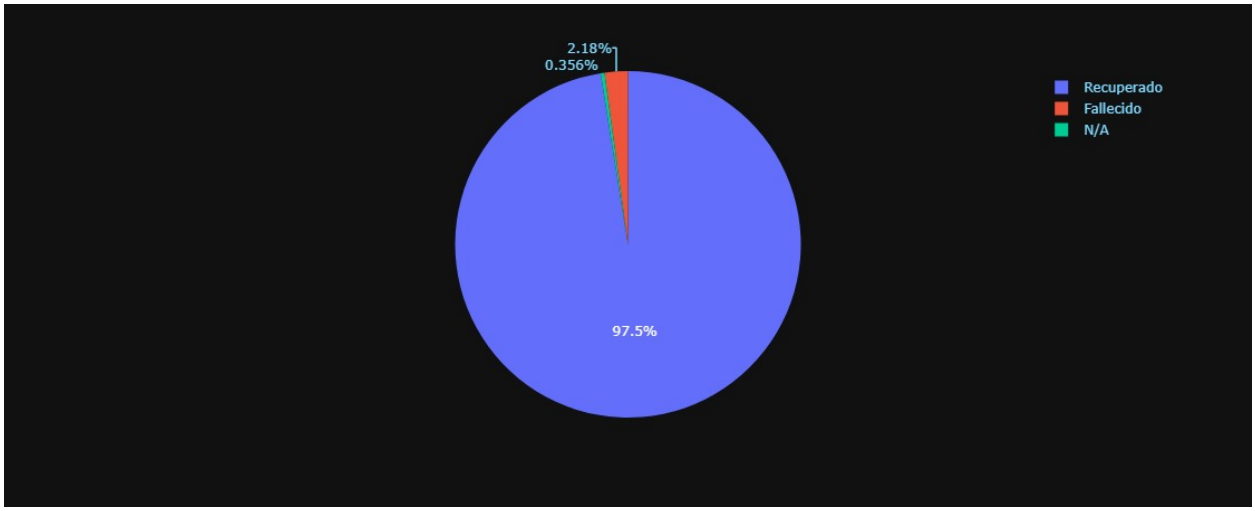

figura 4) Gráfico de torta, estado de casos en el 2021.

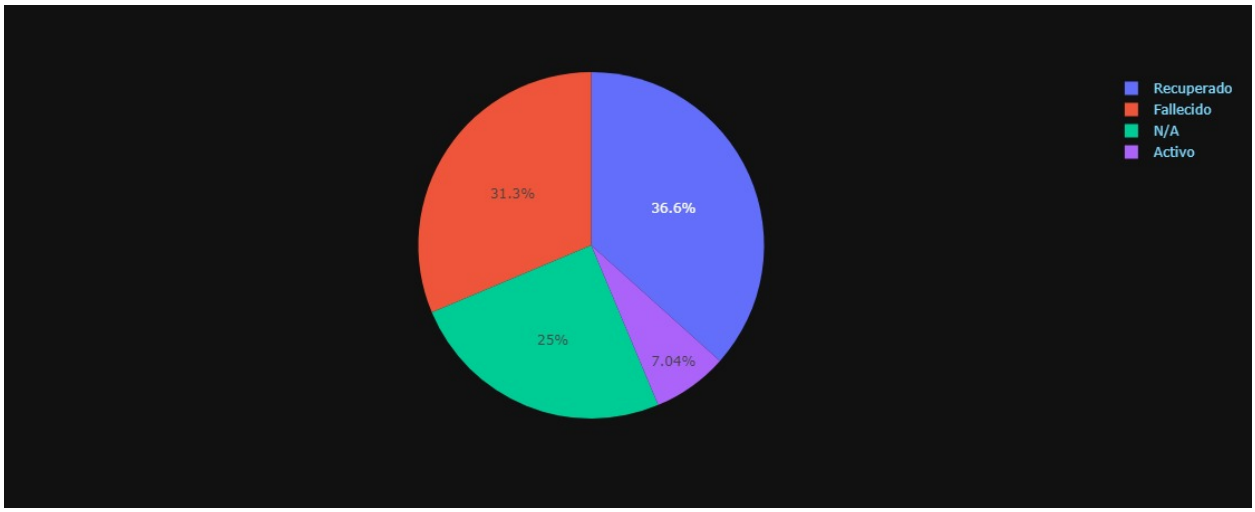

figura 5) Gráfico de torta, estado predominante por edad.

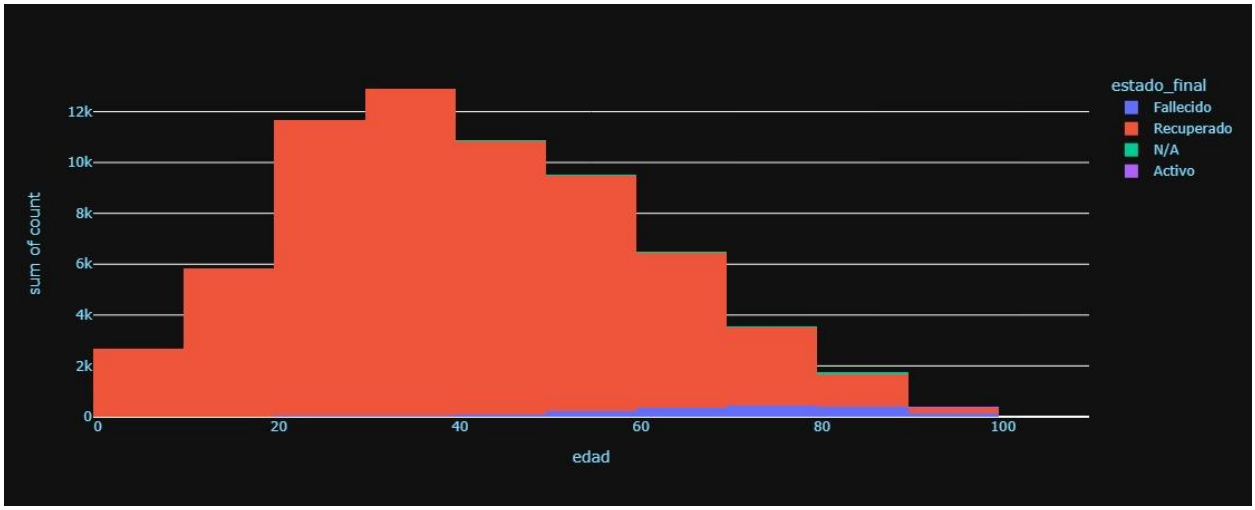

figura 6) Histograma, estado predominante por edad

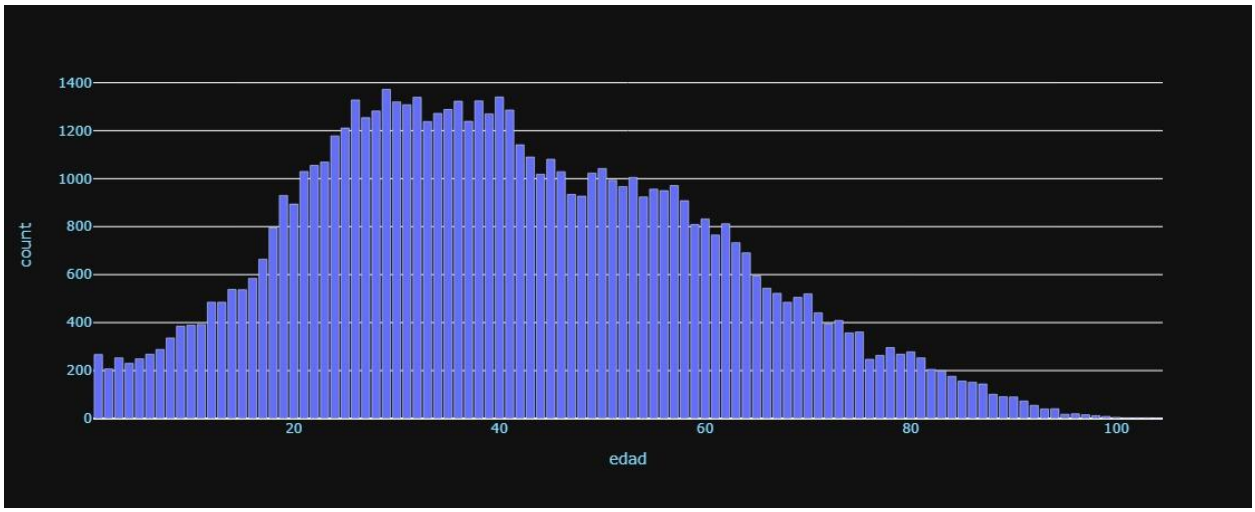

figura 7) Gráfico de barras, casos recuperados por año.

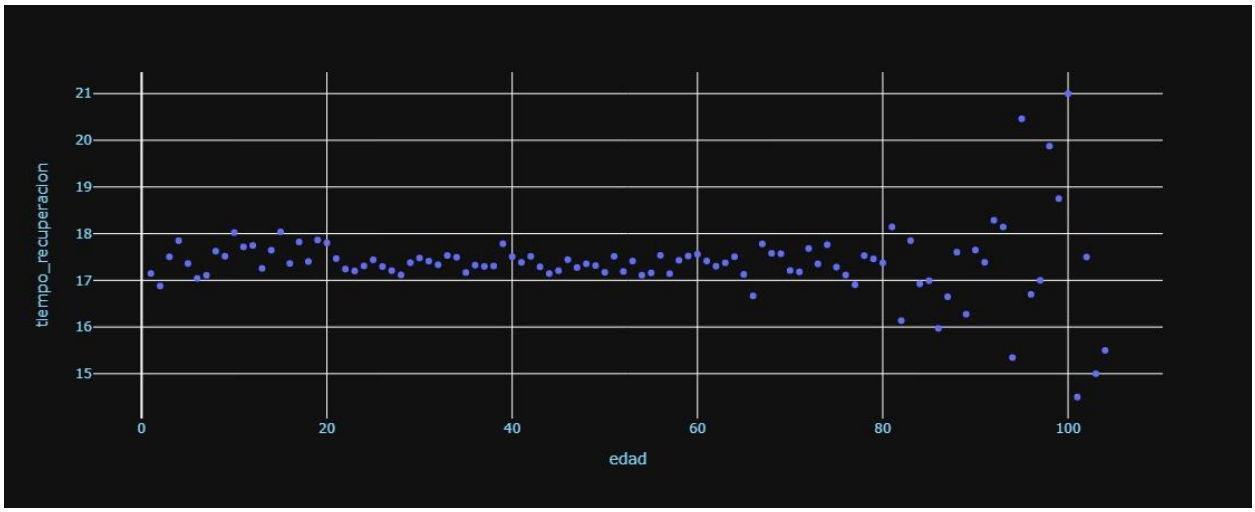

figura 8) Gráfico de puntos, promedio de días para la recuperación por edad.

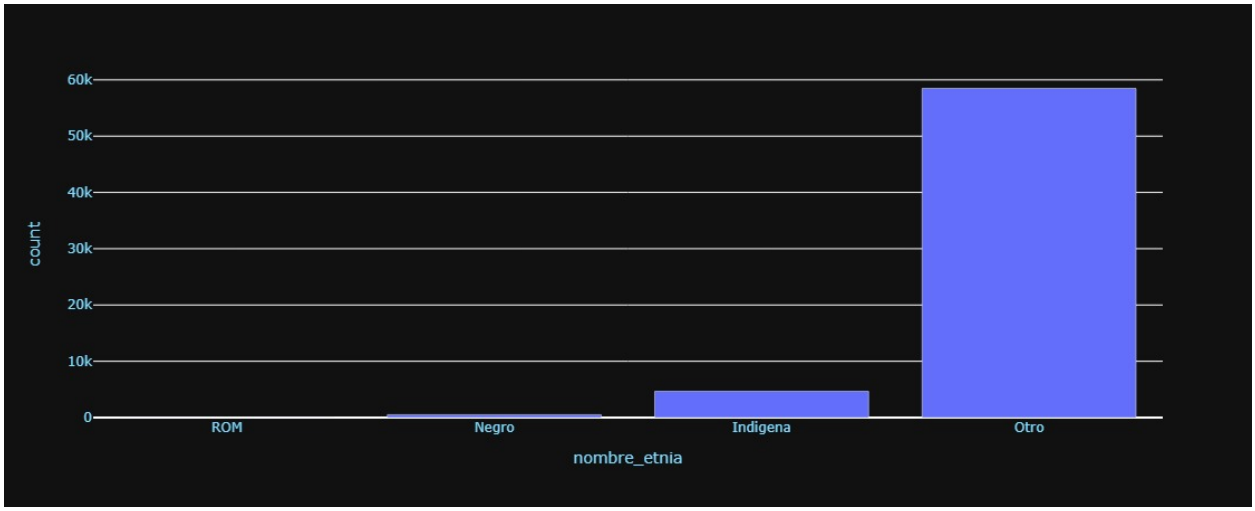

figura 9) Diagrama de barras, casos recuperados por etnia.

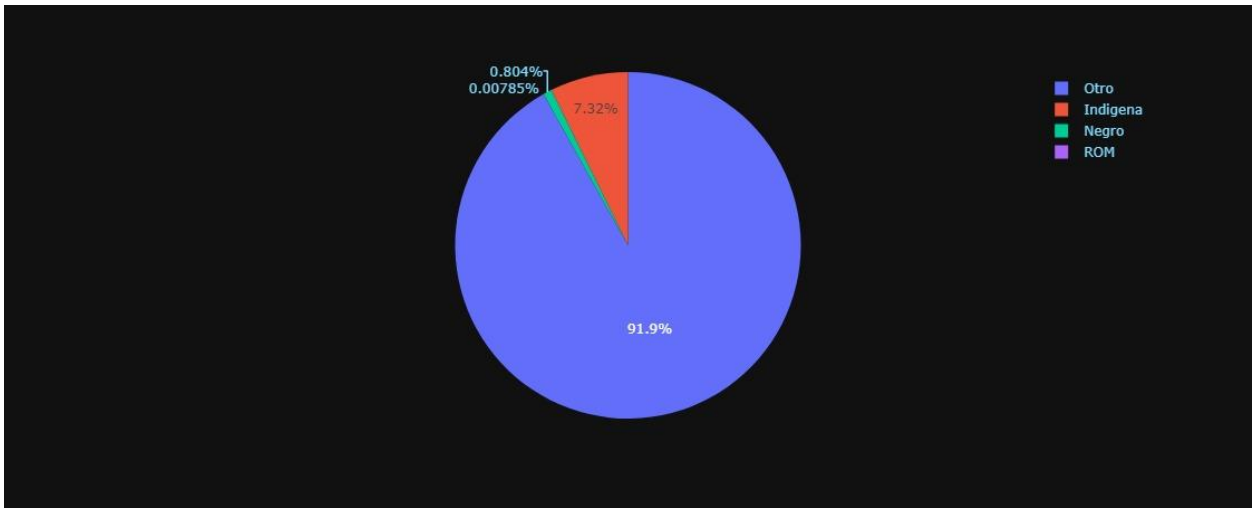

figura 10) Gráfico de torta, casos recuperados por etnia.

2. Sección discusión

Análisis de gráficos

figuras 1 y 2) Contagios por municipio, cantidad recuperados, cantidad fallecidos: Con el primer gráfico podemos advertir en qué municipio del departamento de Sucre se registran más casos de COVID-19 en los cuales el paciente fallece y en el segundo los casos en los cuales el paciente se recupera, quienes se recuperan superan en gran medida a quienes

fallecen, de acuerdo a los resultados, en ambos casos, Sincelejo es quien màs registros presenta y supera en grandes cantidades a los demás municipios y al municipio anterior, Corozal. **Ventajas:** Permite identificar qué municipio en Sucre es el más afectado por el COVID-19 respecto a recuperación y fallecimientos para que autoridades en salud puedan tomar medidas al respecto. **Desventajas:** Puesto que hay una variable con valores mucho mayores que las demás, no se logra percibir bien en esa escala los valores para las variables con menos datos

figuras 3 y 4)Pacientes recuperados y fallecidos en 2020 y 2021: Se realizaron dos gráficos de torta los cuales comparamos el porcentaje de casos en los que el paciente se recuperó y los casos en los que el paciente no se recupera en los años 2020 y 2021, el primer gráfico corresponde a 2020 y el segundo a 2021, conforme a esto logramos deducir que es altamente probable que en el 2020 los pacientes se recuperen del COVID-19 en un 95.9% y un 4.1% fallecen. Mientras que para el 2021 aumenta el porcentaje de personas recuperadas en un 1,6% y las fallecidas corresponden sólo al 2.18%. **Ventajas:** Permite evaluar la evolución del virus y cómo las personas desarrollan anticuerpos para defenderse de la enfermedad y no fallecer. **Desventajas:** Una desventaja de este análisis es que en los datos del 2021 tenemos datos N/A mientras que en 2020 no y esto puede interferir con la exactitud de los análisis de las variables que queremos tener en cuenta. No muestra con exactitud cuántos casos hubo por año, entonces la proporción puede ser diferente, lo que puede afectar el análisis.

figuras 5 y 6) Edades más probables de recuperación: Utilizamos un gráfico de torta y un histograma para analizar el comportamiento de las variables de estado final (Fallecido, Recuperado, N/A, Activo) y analizar en qué edades es más probable que un paciente se recupere del COVID-19. De lo anterior obtenemos que en el 36.6% de los casos, la persona se recupera y el 31.2% no, el 7.04% sigue activo y el resto registran N/A. Basándose en el histograma se puede concluir que las edades más probables de recuperación están entre 20 y 50 años de edad. **Ventajas:** Posibilita el análisis de edades de mayor riesgo frente al contagio y puede dar pie a dar prioridad a estas personas más

vulnerables, por ejemplo con el asunto de la vacunación **Desventajas:** No se toma una muestra de igual tamaño para las edades puesto que hay edades en las que se presentarán más contagios mientras que en otras no.

figuras 7 y 8) Tiempo promedio de recuperación por edad: Para hallar el tiempo promedio de recuperación por año se tuvo en cuenta el comportamiento de la recuperación de los pacientes conforme a su edad. A partir de esto se genera un gráfico de puntos en el cual se evidencia el tiempo de recuperación por días según la edad de cada paciente. Los datos se vuelven más dispersos cuando hay menos registros por ejemplo en edades mayores que 80-85. **Ventajas:** Es útil para reconocer medidas necesarias frente al contagio durante el tiempo en el que es paciente requiere cuidados o dar alertas si el contagio persiste después de este tiempo **Desventajas:** No se puede conocer en qué circunstancias el paciente se recupera, si tuvo o no algún tratamiento frente al contagio lo cual influiría en el análisis.

figuras 9 y 10) Casos de recuperación por etnia: Se realizó un gráfico de barras el cual indica los casos en los cuales los pacientes se recuperaron con respecto a la pertenecia_etnia, de este se puede observar que son pocas las personas que pertenecen a alguna de las pertenencias étnicas enumeradas puesto que la mayoría (casi 60k) se encuentran en alguna otra distinta o simplemente no pertenecen a ninguna y de la misma forma se comportaron los casos de COVID, hay más recuperados en “otros” porque son a quienes más registros corresponden.**Ventajas:** Permite saber cómo afecta la enfermedad a una etnia, y como está preparado el sistema inmune para atacar la enfermedad, que tantos casos de recuperación hay frente a otras etnias y estudiar el comportamiento del COVID-19 en cada una de esas etnias. **Desventajas:** No se reconoce por separado a quienes no pertenecen a ninguna etnia en particular.

3. Sección conclusiones

Respecto al proceso de desarrollo del proyecto, tenemos las siguientes apreciaciones. Para la selección de fuentes de datos, primero habíamos elegido una base de datos con los resultados del ICFES, pero esta base de datos no cumplia con el requerimiento de ser actualizada o creada en el 2022, por lo que cambiamos a una base de datos sobre contagios del covid. En esto perdimos tiempo y tuvimos que hacer de nuevo la primera entrega, pero tenía menos columnas y las relaciones y tablas no eran tan complicadas como en la primera base de datos. Esta se ajustaba a nuestras necesidades, reforzamos nuestros conocimientos de modelo relacional y entidad-relación al tener que repetir estos dos modelos para ambas bases y posteriores correcciones después de normalizar. El diagrama relacional no fue complicado de hacer al ya tener establecidas las relaciones en las reglas de negocio. El diseño de la base de datos no fue complejo, quedamos con pocas tablas y al momento de normalizar se convirtieron en varias tablas pequeñas para evitar redundancia horizontal. La normalización no fue difícil, solamente debíamos crear muchas llaves ficticias para hacer las dependencias funcionales y las relaciones. En la carga de datos se complicaron las cosas porque estas llaves ficticias no eran tan fáciles de hacer en ciertas tablas, por lo que escribimos un pequeño programa en python para concatenar y asignar un id a los datos. Nos tocó separar el excel con todos los datos en las once tablas que creamos y fue un proceso largo. Con la carga masiva fue simple la carga de los datos y no se demoró en cargar a pesar de la gran cantidad de datos. Crear el módulo para conectarse a la base de datos y consultar los datos con python era un proceso tedioso si lo hacíamos como vimos en clase, tabla por tabla, por lo que usamos las librería pandas y psycopg2 que con un query permite visualizar las tablas de manera más eficiente. El desarrollo de las gráficas usando dash fue retador al plantear las consultas en postgresql, y en el módulo de python plantear las funciones correctas para definir cada gráfica según el query, pero la visualización en el símbolo del sistema es sencillo y crea una página interactiva que permite analizar de mejor manera las relaciones planteadas. En conclusión, la base de datos que elegimos era la adecuada porque todo el proceso de diseño de la base de datos, carga de datos, desarrollo de conexiones de bases de datos y desarrollo usando dash fue posible sin muchas complicaciones y pudimos aplicar los

temas vistos en clase en una base de datos real, aprendiendo en el desarrollo del proyecto como crear correctamente una base de datos.