

Práctica Profesional

Exploración y Análisis de Datos (EDA)



Proyecto: **DataDynasty**

- 94404602 | **Giovanny Aguilar Rojas**

- 38455889 | **Barozzi Eugenia**

- 32930970 | **Julieta Battauz**

- 34311109 | **Federico Gurrea**

- 27077572 | **Juan Marcelo Molina**

- 37739346 | **Dante Javier Pagano**

• Introducción

- Contexto del Dataset:** El conjunto de datos utilizado en este análisis corresponde a mediciones clínicas de pacientes, que incluye información sobre presiones arteriales, edad, género, y otros factores de salud. Estos datos fueron recolectados como parte de un estudio longitudinal con el objetivo de identificar factores de riesgo en la hipertensión
- Características del Dataset :** "El dataset contiene 4854 registros de pacientes, con un total de 32 variables. Estas variables incluyen mediciones numéricas como presión sistólica, presión diastólica y edad, además de variables categóricas como género y condición médica (por ejemplo, presencia de hipertensión)
- Objetivo del Análisis:** "El objetivo de este análisis exploratorio es comprender la distribución de las presiones arteriales en la población estudiada, identificar posibles outliers que puedan indicar errores en la recolección de datos o casos extremos, y analizar la relación entre las variables clínicas.

- **Carga y Revisión Inicial del DataSet** La carga del dataset se realizó utilizando la biblioteca pandas, que permite manejar datos en forma de DataFrame de manera eficiente

```
[268]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

[27]: test = pd.read_csv("Tests.csv", sep=';')
completos = pd.read_csv("CompletosTest.csv", sep=';')
```

Una vez cargado el dataset, se revisó su estructura utilizando el método `.info()` de pandas. El dataset cuenta con 3.148 registros y 9 columnas. A continuación, se muestran los nombres de las columnas y los tipos de datos asociados a cada una. A su vez se presenta una vista preliminar de los primeros 5 registros del conjunto de datos utilizando el método `.head()`. Se observan las primeras mediciones de presión arterial y las características asociadas.

```
[272]: test.info()A
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3148 entries, 0 to 3147
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   TestId      3148 non-null   object
1   RawDataId   3148 non-null   object
2   Date        3148 non-null   object
3   Time        3148 non-null   object
4   Systolic    3148 non-null   int64
5   Diastolic    3148 non-null   int64
6   MAP         3148 non-null   int64
7   HR          3148 non-null   int64
8   PP          3148 non-null   int64
dtypes: int64(5), object(4)
memory usage: 221.5+ KB

[19]: test.head()
[19]:
```

	TestId	RawDataId	Date	Time	Systolic	Diastolic	MAP	HR	PP
0	35C77615-22AE-4A58-A5F0-07C761F9A787	B3E82F30-394E-4103-AAA9-0007A9EB4616	07/05/2024	13:05:00	144	79	99	64	65
1	35C77615-22AE-4A58-A5F0-07C761F9A787	11E05A98-BA56-4E2B-AE21-0AFC4F2CABE3	07/05/2024	14:48:00	151	79	100	81	72
2	35C77615-22AE-4A58-A5F0-07C761F9A787	EFCC5840-98F3-4FB4-8EA5-0E6020218B46	07/05/2024	12:45:00	139	89	106	67	50
3	35C77615-22AE-4A58-A5F0-07C761F9A787	09472283-CAC3-4E11-B1C0-108E63E20F39	07/05/2024	11:05:00	135	76	97	63	59
4	35C77615-22AE-4A58-A5F0-07C761F9A787	D7E8F986-9F8B-484A-A271-10D998F5C0D5	07/05/2024	12:25:00	145	82	106	67	63

```

[158]: completos.head()
[158]:
```

	TestId	PatientId	Interpretation	HookupStartTime	HookupEndTime	SystolicMax	SystolicMin	DiastolicMax	DiastolicMin	MAPMax
0	83E62D41-079A-4CEF-808D-00283FC49150	A0D6DF20-52D5-4B3A-B748-116F142D54AF	Presento hipertension sistolica en horas de su...	2023-06-12 10:33:00.000	2023-06-13 09:53:00.000	240	70	150	40	200
1	6063F63A-693F-4D42-A415-0033EF7D0133	45910F80-CC24-4C9A-9610-DBC19A593FBA	Presento hipertension sistolica de moderada a ...	2017-06-14 10:43:00.000	2017-06-15 06:52:00.000	240	70	150	40	200
2	188028F6-0302-4A2E-90A5-0036A792BE35	B15844CB-88EE-4E16-AE51-D3C4BE82C9AF	VALORES PROMEDIOS DE TA DENTRO DE LIMITES	2019-09-09 10:31:00.000	2019-09-10 10:10:00.000	240	70	150	40	200

Se realizó la unión de los conjuntos de datos "test y completos" mediante la columna "TestId", que está presente en ambos datasets, con 4854 registros y 32 campos. El método de unión utilizado es "right join", por lo que se mantienen todas las filas del dataset completos, incluso si no tienen un valor coincidente en test, luego se selecciono los campos a utilizar guardando dentro de un Dataframe.

```
[29]: df_limpio = df_unido[['TestId', 'RawDataId', 'Date', 'Time', 'Systolic', 'Diastolic', 'MAP',
                           'HR', 'PP', 'PatientId', 'Interpretation', 'HookupStartTime',
                           'HookupEndTime', 'Duration', 'SuccessfulReading', 'PercentSuccessfulReading',
                           'SysDipping', 'DiaDipping', 'MapDipping', 'Age', 'GenderId',
                           'BirthDate']]

[28]: df_limpio.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4845 entries, 0 to 4844
Data columns (total 22 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   TestId      4845 non-null   object
1   RawDataId   3148 non-null   object
2   Date        3148 non-null   object
3   Time        3148 non-null   object
4   Systolic    3148 non-null   float64
5   Diastolic    3148 non-null   float64
6   MAP         3148 non-null   float64
7   HR          3148 non-null   float64
8   PP          3148 non-null   float64
9   PatientId   4845 non-null   object
10  Interpretation  4798 non-null   object
11  HookupStartTime  4845 non-null   object
12  HookupEndTime   4845 non-null   object
13  Duration        4845 non-null   object
14  SuccessfulReading  4845 non-null   int64
15  PercentSuccessfulReading  4845 non-null   object
16  SysDipping      4845 non-null   object
17  DiaDipping      4845 non-null   object
18  MapDipping      4845 non-null   object
19  Age             4845 non-null   int64
20  GenderId        4845 non-null   object
21  BirthDate       4845 non-null   object
dtypes: float64(5), int64(2), object(15)
memory usage: 832.9+ KB

[19]: df_unido = pd.merge(test, completos, on='TestId', how='right')

[21]: df_unido.shape

[21]: (4845, 32)
```

El conjunto de datos contiene tanto variables numéricas como categóricas. Entre las variables numéricas se encuentran la "presión sistólica y Diastólicas" y la "edad", mientras que entre las variables categóricas están "género" y "condición médica". Esta distinción es importante ya que las técnicas de análisis varían según el tipo de variable.

• Limpieza de datos :

Se lleva a cabo la limpieza de datos en archivos CSV para facilitar el análisis, asegurando la integridad y precisión de la información. Este informe detalla el proceso de limpieza realizado en los archivos de datos obtenidos, como "Completos test.csv" y "Test.csv". se ha

decidido realizar una revisión adicional utilizando bibliotecas de Python para garantizar la integridad y precisión de los datos.

Se mejoro la calidad del conjunto de datos al abordar los siguientes problemas, facilitando decisiones basadas en información confiable:

- Valores Faltantes:** Identificar y manejar datos ausentes.
- Duplicados:** Eliminar registros duplicados que distorsionan el análisis.
- Errores de Formato:** Corregir tipos de datos y formateo inadecuado.
- Valores Erróneos:** Detectar y corregir valores fuera de rango.
- Inconsistencias:** Asegurar coherencia entre columnas (ej. edad y fecha de nacimiento).
- Datos Irrelevantes:** Eliminar columnas o registros que no aportan valor.
- Normalización:** Ajustar datos a un rango común.
- Transformaciones:** Modificar variables para mejorar su utilidad.

a continuación se muestra algunos procedimientos realizados para la limpieza de datos y exportado “df_limpio.csv”.

```
[252]: # filtrando valores para tener una dataframe limpio
# Ver duplicados
duplicados = df_limpio[df_limpio.duplicated()]

# Eliminar columnas irrelevantes
df_limpio = df_limpio.drop_duplicates()

# Corregir valores fuera de rango
df_limpio = df_limpio[(df_limpio['Age'] >= 0) & (df_limpio['Age'] <= 120)]

# Otro ejemplo: presión arterial sistólica (90-180)
df_limpio = df_limpio[(df_limpio['Systolic'] >= 90) & (df_limpio['Systolic'] <= 180)]

# Eliminar columnas innecesarias
df_limpio = df_limpio.drop(['RawDataId', 'SysDipping', 'DiaDipping', 'MapDipping'], axis=1)

# Normalizar los datos
df_limpio = df_limpio.apply(lambda x: x.astype(int) if x.dtype == 'float' else x)

# Crear una columna categórica de grupos de edad
df_limpio['grupo_edad'] = pd.cut(df_limpio['Age'], bins=[0, 18, 35, 50, 65, 100],
                                labels=['Infantil', 'Joven', 'Adulto joven', 'Adulto', 'Senior'])

# Eliminar filas con cualquier valor nulo en cualquier columna
df_limpio = df_limpio.dropna()
# esportamos el archivo a un csv.
df_limpio.to_csv("df_limpio.csv", index=False, sep=';')
# verificamos cantida de Registros y campos
df_limpio.shape
```

```
df_limpio.info()

<class 'pandas.core.frame.DataFrame'>
Index: 3079 entries, 63 to 4839
Data columns (total 19 columns):
#   Column                Non-Null Count  Dtype
---  -
0   TestId                 3079 non-null  object
1   Date                   3079 non-null  object
2   Time                   3079 non-null  object
3   Systolic               3079 non-null  int32
4   Diastolic              3079 non-null  int32
5   MAP                    3079 non-null  int32
6   HR                     3079 non-null  int32
7   PP                     3079 non-null  int32
8   PatientId              3079 non-null  object
9   Interpretation          3079 non-null  object
10  HookupStartTime         3079 non-null  object
11  HookupEndTime           3079 non-null  object
12  Duration                3079 non-null  object
13  SuccessfullReading      3079 non-null  int64
14  PercentSuccessfullReading 3079 non-null  object
15  Age                     3079 non-null  int64
16  GenderId                3079 non-null  object
17  BirthDate               3079 non-null  object
18  grupo_edad              3079 non-null  category
dtypes: category(1), int32(5), int64(2), object(11)
memory usage: 400.1+ KB
```

```
[252]: (3079, 19)
```

A continuación presentamos una tabla que contiene **valores estándar** de presiones sistólica y diastólica por edad y sexo, usados como referencia para detectar **variaciones** en los datos que estamos analizando. Con estos valores normales, puedes comparar las presiones de los pacientes

VALORES NORMALES DE LA TENSIÓN ARTERIAL SEGÚN LA EDAI				
EDAD	PRESIÓN SISTÓLICA		PRESIÓN DISTÓLICA	
	HOMBRE	MUJER	HOMBRE	MUJER
16 a 18	105 - 135	100 - 130	60 - 86	60 - 85
19 a 24	105 - 139	100 - 130	62 - 88	60 - 85
25 a 29	108 - 139	102 - 135	65 - 89	60 - 86
30 a 39	110 - 145	105 - 139	68 - 92	65 - 89
40 a 49	110 - 150	105 - 150	70 - 96	65 - 96
50 a 59	115 - 155	110 - 155	70 - 98	70 - 98
60 o más	115 - 160	115 - 160	70 - 100	70 - 100

• El Análisis Univariado

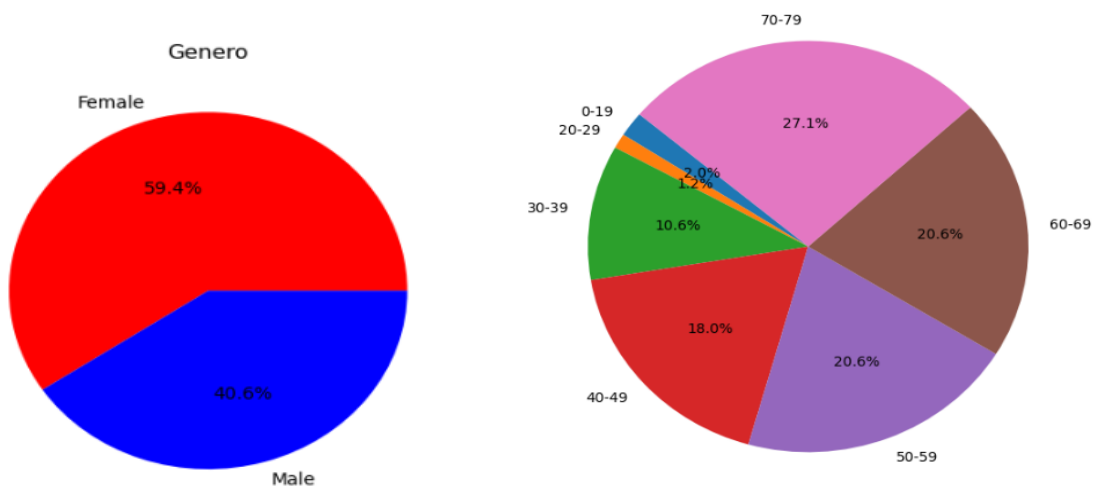
Se analiza cada variable por separado para entender su distribución, características y posibles anomalías. Proporciona un resumen estadístico de las columnas numéricas de los datos obtenidos. Este resumen incluye varias estadísticas descriptivas univariadas, tales como:

- **Count:** El número de valores no nulos.
- **Mean:** La media aritmética de los valores.
- **Std:** La desviación estándar, que mide la dispersión de los valores respecto a la media.
- **Min:** El valor mínimo.
- **25%:** El primer cuartil, que es el valor por debajo del cual se encuentra el 25% de los datos.
- **50%:** La mediana o segundo cuartil, que es el valor central.
- **75%:** El tercer cuartil, que es el valor por debajo del cual se encuentra el 75% de los datos.
- **Max:** El valor máximo.

```
df_limpio.describe()
```

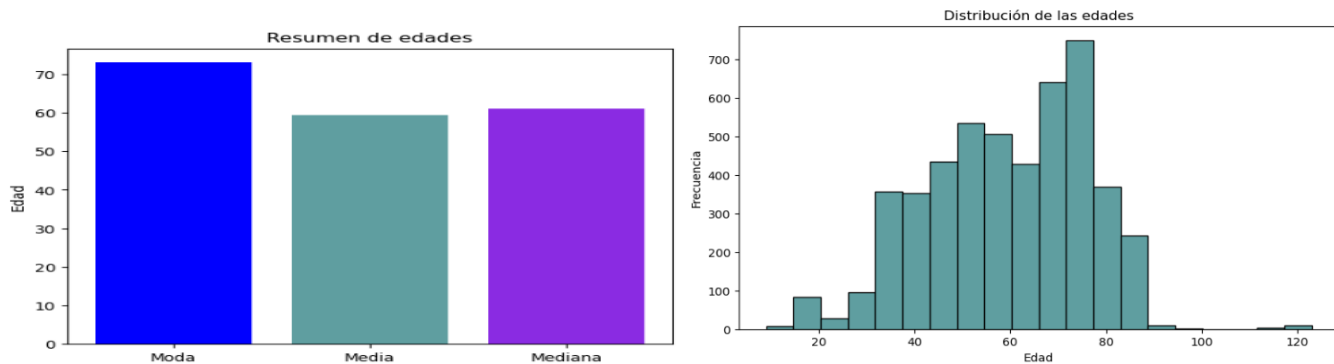
	Systolic	Diastolic	MAP	HR	PP	SuccessfulReading	Age
count	3148.000000	3148.000000	3148.000000	3148.000000	3148.000000	4845.000000	4845.000000
mean	129.380241	78.152795	95.597205	72.770330	51.227446	58.155624	59.297007
std	19.632757	14.632187	15.551684	14.130255	11.899259	7.861107	16.777024
min	75.000000	38.000000	54.000000	33.000000	17.000000	1.000000	9.000000
25%	116.000000	68.000000	85.000000	63.000000	43.000000	54.000000	47.000000
50%	128.000000	77.000000	95.000000	71.000000	50.000000	60.000000	61.000000
75%	141.000000	88.000000	105.250000	81.000000	59.000000	64.000000	73.000000
max	206.000000	134.000000	155.000000	186.000000	102.000000	74.000000	123.000000

A continuación se puede visualizar la distribución de género del conjunto de datos. De la misma manera visualizamos la distribución porcentual de los rangos etarios de los pacientes.



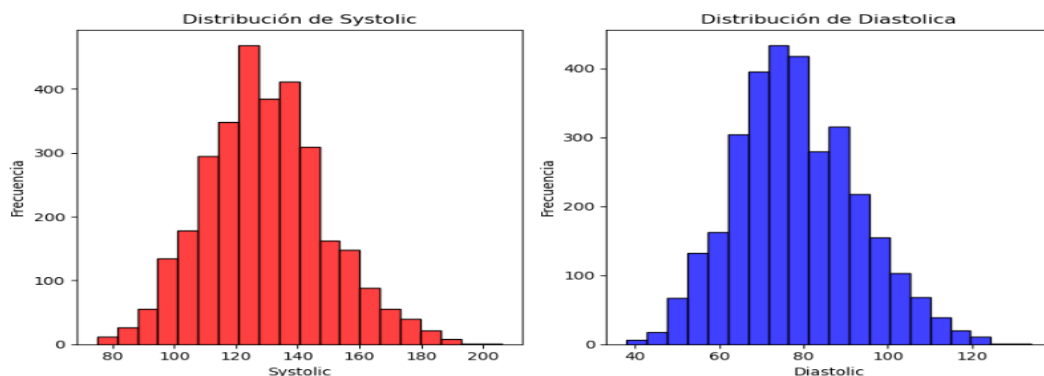
En Este Gráfico Utilizando **matplotlib** y buscamos mostrar la **Moda**, **Media** y **Mediana** de los datos de **Edades**.

- La **moda** es el valor que aparece con mayor frecuencia en un conjunto de estos datos.
- La **media** muestra la suma de todos los valores en un conjunto de datos edades dividida por el número de valores en ese conjunto.
- La **mediana** es el valor medio de un conjunto de estos datos. EL proximo Gráfico muestra lo mismo pero en un gráfico de Histograma.



Histograma: Distribución de las variables Systolic y Diastolic

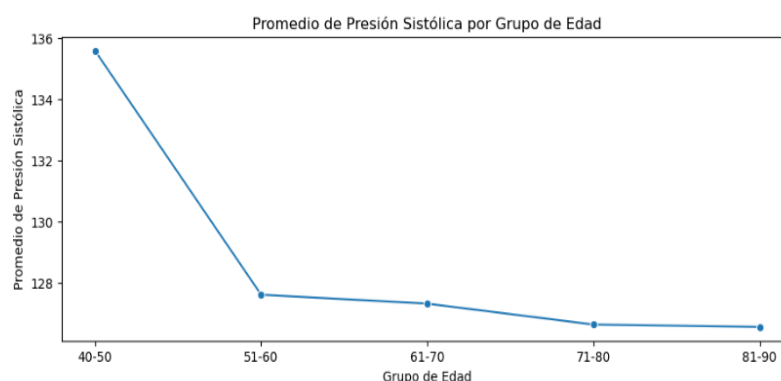
un gráfico con dos histogramas en subplots, uno para la presión sistólica y otro para la diastólica. Ambos histogramas muestran la distribución de los valores de presión en intervalos definidos. El histograma de presión sistólica (izquierda) permite observar la frecuencia de los valores, posibles picos y asimetrías. El de presión diastólica (derecha) muestra la distribución de esos valores, ayudando a identificar sesgos o anomalías en los datos.



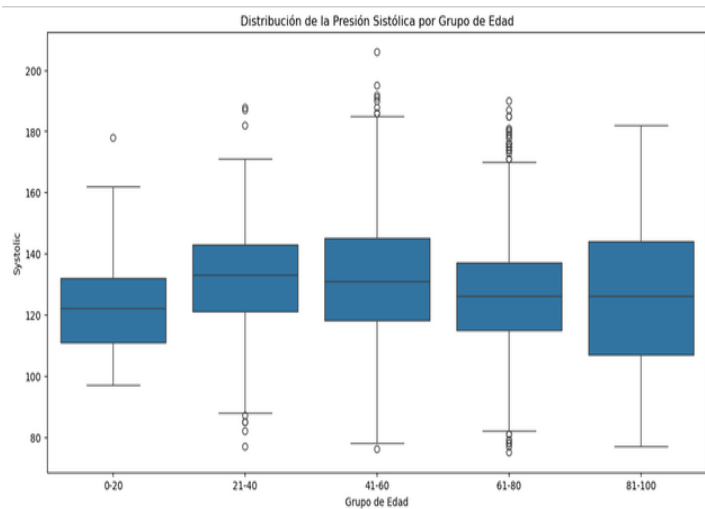
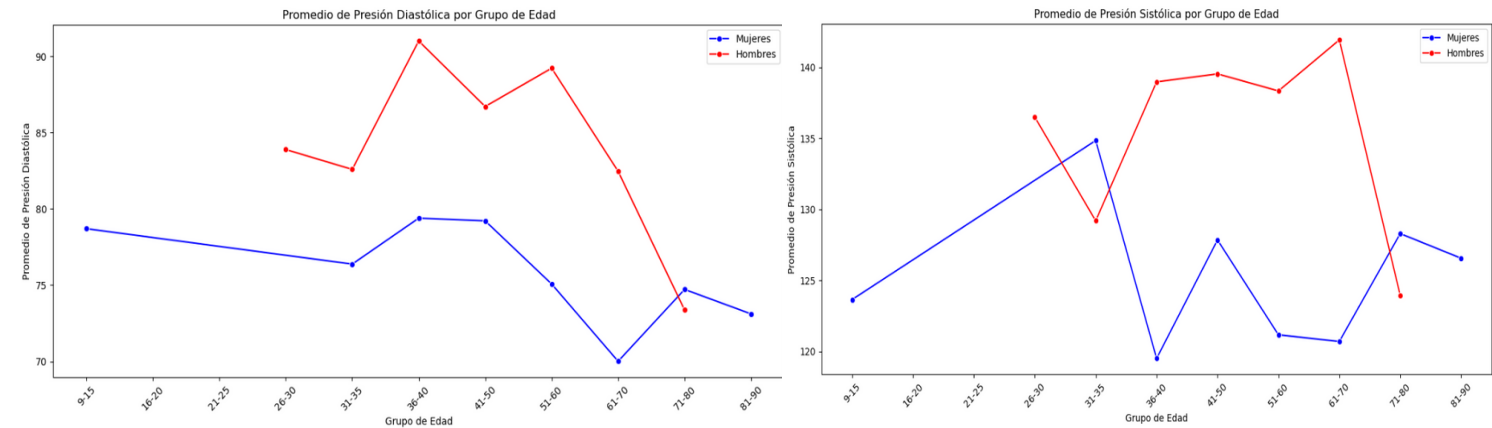
• Análisis Bivariado

Es una técnica estadística que se utiliza para investigar la relación entre dos variables. A diferencia del análisis univariado, que se centra en una sola variable, el análisis bivariado examina cómo dos variables interactúan entre sí.

Gráfico de líneas: explica cómo se examina la relación entre la edad y la presión sistólica agrupando las edades en rangos y calculando el promedio de la presión sistólica en cada grupo. La línea en el gráfico conecta estos puntos para mostrar la tendencia de la presión sistólica según la edad. Si la línea sube, significa que la presión aumenta con la edad; si baja, indica lo contrario. En resumen, el gráfico visualiza cómo varía la presión sistólica en función de la edad, destacando la tendencia promedio.



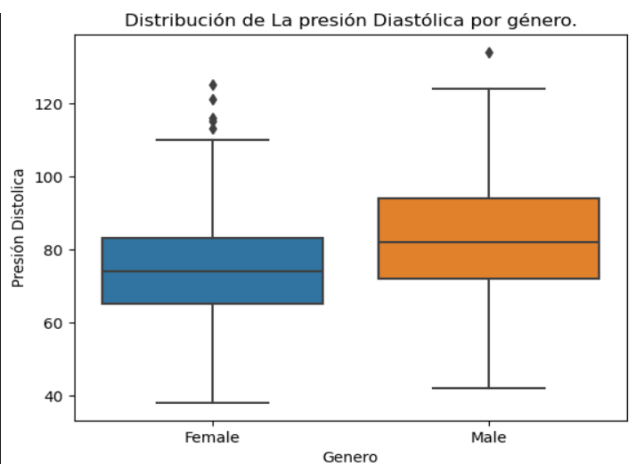
Este gráfico lineal compara el promedio de las presiones sistólica y diastólica por grupo de edad entre hombres y mujeres. La línea azul representa a las mujeres y la roja a los hombres, permitiendo observar tendencias y diferencias significativas en la presión arterial según la edad y el sexo. El gráfico muestra cómo las presiones cambian con la edad para ambos géneros.



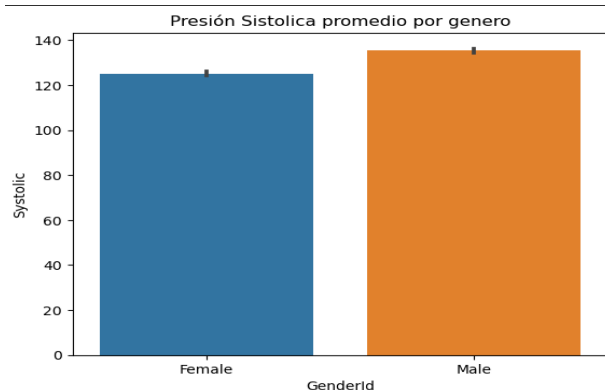
Este boxplot muestra cómo varía la presión sistólica en diferentes grupos de edad (0-20, 21-40, 41-60, etc.). Presenta la mediana (línea dentro de la caja), el rango intercuartílico (la caja de Q1 a Q3) y los valores atípicos (puntos fuera de los bigotes). Los "bigotes" representan la dispersión de los datos sin incluir los outliers. El gráfico permite observar si la presión sistólica tiende a aumentar con la edad y si hay mucha variabilidad dentro de cada grupo.

Boxplots: Comparación de Género y Presión Diastólica

El gráfico de boxplots que compara la distribución de la presión diastólica entre géneros. Cada boxplot muestra la mediana (línea en la caja) y el rango intercuartílico (la caja entre los cuartiles Q1 y Q3) para la presión diastólica en cada género. También se destacan posibles adwdawd atípicos, que se muestran como puntos fuera de la caja, indicando outliers o una mayor dispersión en la distribución

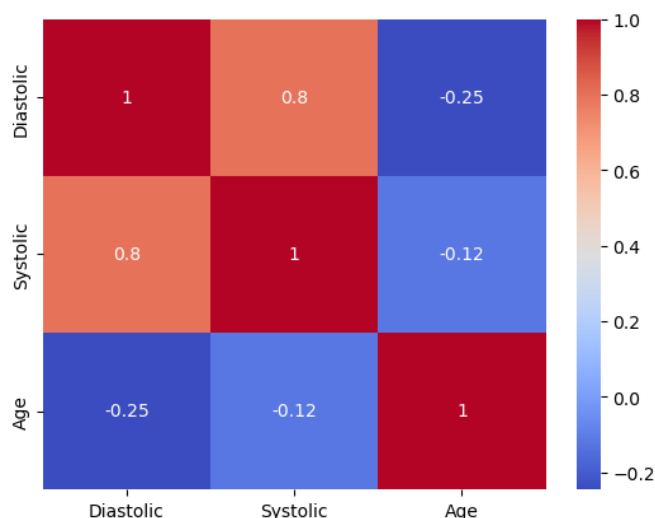


Presión Sistólica y Diastólica Promedio por Género

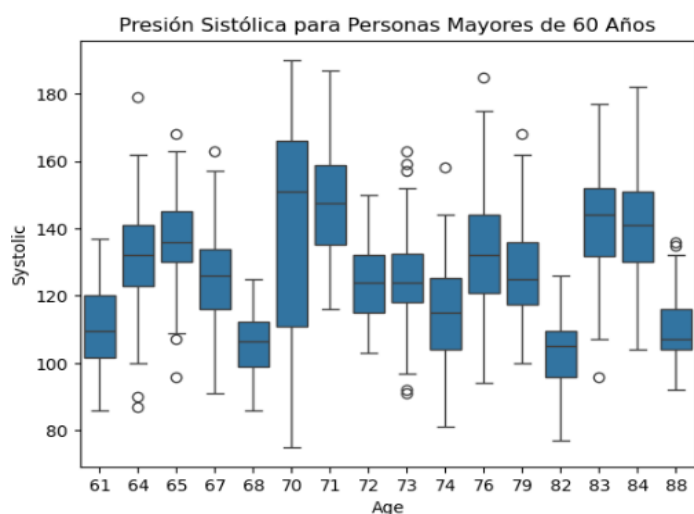


El gráfico te permite visualizar de forma rápida si existe alguna diferencia en los promedios de la presión sistólica entre los géneros. Si una barra es visiblemente más alta que la otra, indicará que uno de los géneros tiene una presión sistólica promedio mayor que el otro. Este gráfico es útil para comparar de manera sencilla los promedios de presión sistólica entre hombres y mujeres, y puede revelar posibles diferencias significativas entre ambos grupos.

La matriz de correlación muestra cómo las variables se relacionan mediante coeficientes que van de -1 a 1. Una fuerte correlación positiva entre la presión sistólica y diastólica indica que ambas aumentan juntas. Esto sugiere que están estrechamente relacionadas y tienen patrones similares. Sin embargo, la correlación entre la edad y las presiones sistólica y diastólica es débil, lo que significa que la edad no tiene una influencia significativa sobre la presión arterial en los datos analizados.

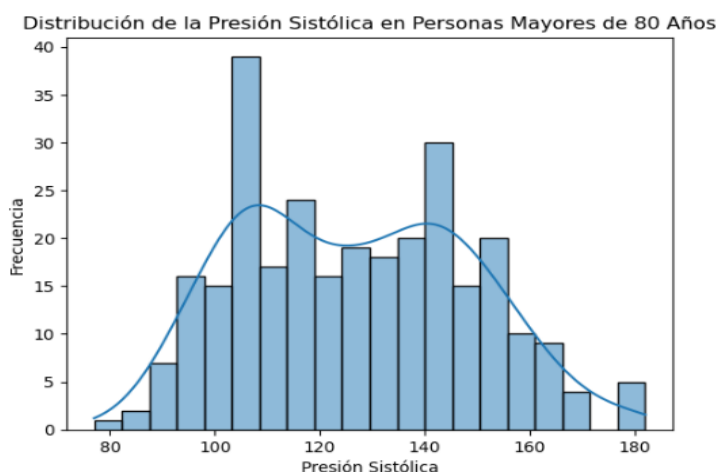


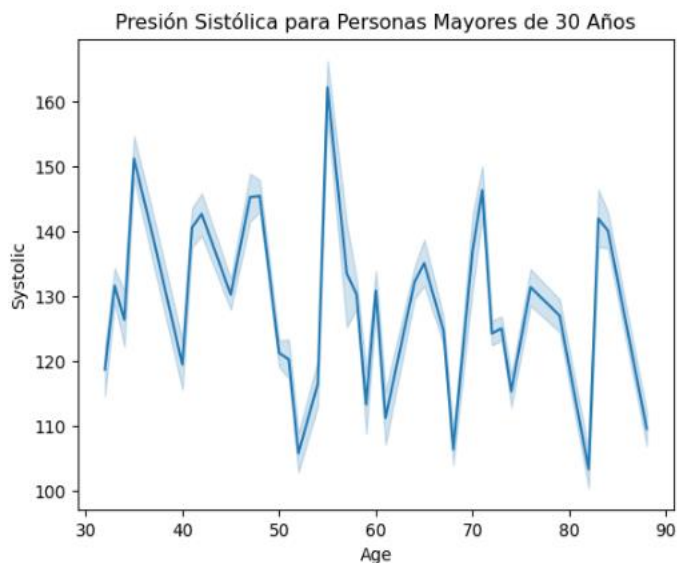
- Usamos gráficos de Boxplot, Histplot y Lineplot



Este análisis examina la relación entre la edad (grupos mayores de 30, 60 y 80 años) y la presión sistólica, ayudando a entender cómo una variable puede influir en la otra. A medida que se recopilen más datos, estos gráficos se volverán más específicos, permitiendo identificar patrones y tendencias detalladas para mejorar diagnósticos y tratamientos personalizados.

El gráfico muestra la distribución de la presión Sistólica para personas mayores de 80 años. El histograma, junto con la curva de densidad, ilustra cómo se distribuyen los valores de presión sistólica en este grupo de edad, permitiendo identificar patrones y concentraciones de datos en distintos rangos de presión.





El gráfico de líneas muestra la variación de la presión sistólica en función de la edad para las personas mayores de 30 años. La línea permite observar tendencias y patrones en cómo cambia la presión sistólica a medida que aumenta la edad en el grupo filtrado.

Los siguientes gráficos nos permite observar cómo varía la presión arterial sistólica y diastólica a lo largo de las 24 horas del día por paciente, se mostrara 3 gráficos relevantes. Las líneas de presión proporcionan una visión clara de los cambios a lo largo del tiempo, mientras que las líneas horizontales punteadas sirven como referencia para los límites recomendados de presión arterial. de cada paciente

