

# Documentación del Dataset

Este dataset se obtuvo directamente de la base de datos del sistema utilizado para realizar estudios de Presurometría en un consultorio médico.

## Limpieza de datos

Se lleva a cabo la limpieza de datos en archivos CSV para para facilitar el análisis asegurar la integridad y precisión de la información. Este informe detalla el proceso de limpieza realizado en los archivos de datos obtenidos, específicamente en el archivo **Completos test.csv** y **Test CSV**. Aunque se ha llevado a cabo una limpieza inicial con la herramienta Excel, se ha decidido realizar una revisión adicional utilizando bibliotecas de Python para garantizar la integridad y precisión de los datos.

### Mejora de la Calidad del Conjunto de Datos

Al abordar los siguientes problemas, se mejorará la calidad del conjunto de datos, 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.

Se han importado las siguientes bibliotecas de Python para facilitar la manipulación y análisis de los datos:

```
Se importa las Librerías para montar el dataset y verificar los registros y manipular el archivo en
gral

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

Montaje del dataset

df = pd.read_csv("Tests.csv", sep=';')
```

Posterior a la carga del archivo de datos, se procede a verificar las variables presentes en el DataFrame, clasificándolas en variables categóricas y variables numéricas. Este paso es fundamental para asegurar que los datos estén en el formato correcto y sean aptos para el análisis posterior.

```
Cotejamos las variables categoricas y numericas

df.info()

<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
```

```
cantidad de registros y columnas
```

```
[20] df.shape
```

```
(3148, 9)
```

A continuación, se procede a identificar y eliminar las filas duplicadas. Este proceso asegura que cada registro en el DataFrame sea único, lo cual es crucial para un análisis preciso. Este proceso garantiza que el conjunto de datos esté limpio y listo para el análisis posterior

Se realiza el conteo de filas duplicados cuando encuentre un registro duplicado que lo elimine.

+ Código + Texto

```
[37] # Contar duplicados
total_duplicates = df.duplicated().sum()
print(f'Total de filas duplicadas: {total_duplicates}')

# Eliminar duplicados
df_cleaned = df.drop_duplicates()
print(f'Total de filas después de eliminar duplicados: {df_cleaned.shape[0]}')
```

```
Total de filas duplicadas: 0
Total de filas después de eliminar duplicados: 3148
```

Este proceso permite detectar y corregir inconsistencias, mejorando así la calidad de los datos.

Se revisa la columna de edad para asegurarse de que los valores sean razonables y estén dentro de un rango esperado. Se buscarán inconsistencias como:

Edades negativas.

Edades excesivamente altas (por ejemplo, mayores a 100 años).

Se verifica las fechas de edades calculando el cumpleaños con la fecha actual ,para encontrar inconsistencias.

```
# Verificar inconsistencias entre edad y fecha de nacimiento
df['Calculated_Age'] = (pd.to_datetime('today') - pd.to_datetime(df['BirthDate'])).d
inconsistent_ages = df[df['Age'] != df['Calculated_Age']]
print("Inconsistencias en la edad:")
print(inconsistent_ages)
```

372	72
373	124
374	53
375	43
376	76
377	56
378	59
379	124
380	79

El objetivo de esta etapa es mejorar la calidad de los datos eliminando registros que indiquen una edad imposible o poco razonable, específicamente aquellos que superan los 100 años.

Se filtra los registros erroneos mayores a 100 años y se eliminan

```
[49] # Mostrar el número de filas antes de eliminar
print(f"Número de filas antes de eliminar: {df.shape[0]}")

# Filtrar el DataFrame para eliminar filas donde la edad es mayor a 100
df = df[df['Age'] <= 100]

# Mostrar el número de filas después de eliminar
print(f"Número de filas después de eliminar: {df.shape[0]}")
```

Número de filas antes de eliminar: 1752  
Número de filas después de eliminar: 1739

## **Archivos CSV “Completo test” y “Test”**

### **Descripción de Columnas DE N°1 “Completo test CSV”**

**Registros:3148**

**Columnas:24**

#### **Columna:** TestId

- Tipo de dato: Carácter
- Valores posibles: Cadenas alfanuméricas únicas.
- Ejemplos: "83E62D41-079A-4CEF-808D-00283FC49150"
- Descripción: Identificador único de cada prueba.

#### **Columna:** PatientId

- Tipo de dato: Carácter
- Valores posibles: Cadenas alfanuméricas únicas.
- Ejemplos: "A0D6DF20-52D5-4B3A-B74B-116F142D54AF"
- Descripción: Identificador único del paciente.

#### **Columna:** Interpretation

- Tipo de dato: Carácter
- Valores posibles: Cadenas de texto descriptivas.
- Ejemplos: "Presento hipertensión sistólica en horas de sueño. Valor promedio diurno 125/70."
- Descripción: Interpretación médica de los resultados.

#### **Columna:** HookupStartTime

- Tipo de dato: Fecha y Hora
- Valores posibles:
- Formato: AAAA-MM-DD HH:MM:SS.mmm
- Ejemplos: "2023-06-12 10:33:00.000"
- Descripción: Fecha y hora de inicio de la monitorización.

#### **Columna:** HookupEndTime

- Tipo de dato: Fecha y Hora
- Valores posibles:
- Formato: AAAA-MM-DD HH:MM:SS.mmm
- Ejemplos: "2023-06-13 09:53:00.000"
- Descripción: Fecha y hora de finalización de la monitorización.

#### **Columna:** SystolicMax

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 240
- Descripción: Presión sistólica máxima registrada

**Columna:** SystolicMin

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 70
- Descripción: Presión sistólica mínima registrada.

**Columna:** DiastolicMax

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 150
- Descripción: Presión diastólica máxima registrada.

**Columna:** DiastolicMin

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 40
- Descripción: Presión diastólica mínima registrada.

**Columna:** MAPMax

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 200
- Descripción: Presión arterial media máxima registrada.

**Columna:** MAPMin

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 40
- Descripción: Presión arterial media mínima registrada.

**Columna:** PPMMax

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 150
- Descripción: Presión del pulso máxima registrada.

**Columna:** PPMin

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 20
- Descripción: Presión del pulso mínima registrada.

**Columna:** HRMax

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 200
- Descripción: Frecuencia cardíaca máxima registrada.

**Columna:** HRMin

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 20
- Descripción: Frecuencia cardíaca mínima registrada.

**Columna:** Duration

- Tipo de dato: Carácter
- Valores posibles: Cadenas de texto en formato HH: MM.
- Ejemplos: "23:20"
- Descripción: Duración de la monitorización.

**Columna:** SuccessfullReading

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros.
- Ejemplos: 64
- Descripción: Número de lecturas exitosas.

**Columna:** PercentSuccessfullReading

- Tipo de dato: Carácter
- Valores posibles: Cadenas de texto que representan porcentajes.
- Ejemplos: "100%"
- Descripción: Porcentaje de lecturas exitosas.

**Columna:** SysDipping

- Tipo de dato: Numérico (Decimal)
- Valores posibles: Valores decimales.
- Ejemplos: 1.650.000
- Descripción: Valor de reducción de la presión sistólica.

**Columna:** DiaDipping

- Tipo de dato: Numérico (Decimal)
- Valores posibles: Valores decimales.
- Ejemplos: 0.050000
- Descripción: Valor de reducción de la presión diastólica.

### **Columna:** MapDipping

- Tipo de dato: Numérico (Decimal)
- Valores posibles: Valores decimales.
- Ejemplos: 0.050000
- Descripción: Valor de reducción de la presión arterial media.

### **Columna:** Age

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros que representan la edad en años.
- Ejemplos: 67, 71, 78
- Descripción: Edad del paciente.

### **Columna:** GenderId

- Tipo de dato: Numérico (Entero)
- Valores posibles: Valores enteros que representan el género.
- Ejemplos: 2
- Descripción: Identificador del género del paciente.

### **Columna:** BirthDate

- Tipo de dato: Fecha y Hora
- Valores posibles:
- Formato: AAAA-MM-DD HH:MM:SS.mmm
- Ejemplos: "1956-05-15 00:00:00.000"
- Descripción: Fecha de nacimiento del paciente.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	TestId	PatientId	Interpretatic	HookupStart	HookupEndT	SystolicMax	SystolicMin	DiastolicMax	DiastolicMin	MAPMax	MAPMin	PPMax	PPMin	HRMax	HRMin	Duration	Succes
2	83E62D41-07	A0D6DF20-5	Presento	2023-06-12 1	2023-06-13 0	240	70	150	40	200	40	150	20	200	20	23:20	
3	6063F63A-69	45910F80-CC	Presento hic	2017-06-14 1	2017-06-15 0	240	70	150	40	200	40	150	20	200	20	20:09	
4	188028F6-03	B15844CB-88	VALORES PRI	2019-09-09 1	2019-09-10 1	240	70	150	40	200	40	150	20	200	20	23:39	
5	3A8ED937-52	759BC417-A5	Presento hic	2019-04-09 1	2019-04-10 0	240	70	150	40	200	40	150	20	200	20	23:32	
6	3C97CD99-84	3EC79126-F1	Presento hic	2016-07-14 1	2016-07-15 0	240	70	150	40	200	40	150	20	200	20	23:39	
7	C57C4658-1F	CA8A2848-7E	Valores	2022-08-08 1	2022-08-09 0	240	70	150	40	200	40	150	20	200	20	23:18	
8	04B7A458-7E	E3A9A18D-9	Presento hic	2016-06-27 1	2016-06-28 0	240	70	150	40	200	40	150	20	200	20	21:54	
9	E8C4B592-48	51DC08C5-B	Presento	2020-07-28 1	2020-07-29 1	240	70	150	40	200	40	150	20	200	20	23:36	
10	3A4C9983-2E	E34970A2-93	Presento hic	2016-07-18 0	2016-07-19 1	240	70	150	40	200	40	150	20	200	20	25:19:00	
11	5892EB63-87	7B7BC8B5-0	Valores pron	2016-11-14 1	2016-11-15 0	240	70	150	40	200	40	150	20	200	20	22:51	
12	31E63F52-7F	5418F018-65	Presento hic	2018-10-03 1	2018-10-04 0	240	70	150	40	200	40	150	20	200	20	22:37	
13	E084684E-3F	60332D8B-D	Presento	2020-06-16 1	2020-06-17 0	240	70	150	40	200	40	150	20	200	20	23:16	
14	0A3E0459-08	C8AADAB0-3	Presento hic	2017-06-15 1	2017-06-16 0	240	70	150	40	200	40	150	20	200	20	23:34	
15	A1D7B7C6D-3	17AFFE6E-3B	Valores	2023-05-24 1	2023-05-25 0	240	70	150	40	200	40	150	20	200	20	22:58	
16	5CF414A7-04	F7AB3480-AC	Valores	2022-08-18 1	2022-08-19 1	240	70	150	40	200	40	150	20	200	20	23:01	
17	2DAE95DB-3	A5262D2E-A	Valores	2019-10-08 1	2019-10-09 0	240	70	150	40	200	40	150	20	200	20	21:54	
18	9AE4583A-D	20DE6770-AE	Presento	2023-06-08 1	2023-06-09 0	240	70	150	40	200	40	150	20	200	20	23:19	
19	A4089C84-BE	162A2DFF-37	Presento hic	2017-06-09 1	2017-06-10 0	240	70	150	40	200	40	150	20	200	20	23:10	
20	71B5AD57-71	45453206-E7	Presento hic	2018-03-12 1	2018-03-13 0	240	70	150	40	200	40	150	20	200	20	23:19	
21	38C8F0E7-05	3055B51B-75	Valores	2023-01-19 1	2023-01-20 0	240	70	150	40	200	40	150	20	200	20	23:18	
22	699C4356-0F	68A4EA51-84	Presento	2023-03-08 1	2023-03-09 0	240	70	150	40	200	40	150	20	200	20	23:19	
23	38C4CECD-8	4FAF160F-D9	Presento	2019-06-13 1	2019-06-14 0	240	70	150	40	200	40	150	20	200	20	23:19	



## **Descripción de Columnas DE N°2**

### **"Test CSV"**

**Registros:314**

**Columnas:9**

#### **Columna:** TestId

- Tipo de dato:
- Valores posibles: Cadenas alfanuméricas únicas..
- Ejemplos: "83EA-GH4566GFDNJ87"
- Descripción: Identificador único de cada prueba

#### **Columna:** RawDataId

- Tipo de dato: Carácter
- Valores posibles: Cadenas alfanuméricas únicas.
- Ejemplos:
- Descripción: Identificador único de cada registro de datos en bruto.

#### **Columna:** Date

- Tipo de dato: Fecha
- Valores posibles: Fechas en formato YYYY-MM-DD.
- Ejemplos: "2023-09-15"
- Descripción: Fecha en la que se registraron los datos.

#### **Columna:** Time

- Tipo de dato: Tiempo
- Valores posibles: Tiempos en formato HH:MM:SS.
- Ejemplos: "14:30:00"
- Descripción: Hora exacta en la que se registraron los datos.

#### **Columna:** Systolic

- Tipo de dato: Numérico
- Valores posibles: Números enteros o decimales.
- Ejemplos: 120, 135.5
- Descripción: Valor de presión arterial sistólica, medido en mmHg.

#### **Columna:** Diastolic

- Tipo de dato: Numérico
- Valores posibles: Números enteros o decimales.
- Ejemplos: 80, 90.5
- Descripción: Valor de presión arterial diastólica, medido en mmHg.

#### **Columna:** MAP

- Tipo de dato: Numérico
- Valores posibles: Números enteros o decimales.
- Ejemplos: 93.4, 88
- Descripción: Presión arterial media, un indicador del flujo sanguíneo, medido en mmHg.

### **Columna: HR**

- Tipo de dato: Numérico
- Valores posibles: Números enteros.
- Ejemplos: 70, 85
- Descripción: Frecuencia cardíaca, medida en latidos por minuto.

### **Columna: PP**

- Tipo de dato: Numérico
- Valores posibles: Números enteros o decimales.
- Ejemplos: 40, 50.2
- Descripción: Presión de pulso, calculada como la diferencia entre la presión sistólica y diastólica, medida en mmHg

1	TestId	RawDataId	Date	Time	Systolic	Diastolic	MAP	HR	PP	
2	35C77615-22AE-4A58-A5F0-07C761F9A787	83E82F30-394E-4103-AAA9-0007A9EB4616	7/5/2024	13:05:00	144	79	99	64	65	
3	35C77615-22AE-4A58-A5F0-07C761F9A787	11E05A98-BA56-4E2B-AE21-0AFC4F2CA8E3	7/5/2024	14:48:00	151	79	100	81	72	
4	35C77615-22AE-4A58-A5F0-07C761F9A787	EFCC5840-98F3-4FB4-8EA5-0E602021B846	7/5/2024	12:45:00	139	89	106	67	50	
5	35C77615-22AE-4A58-A5F0-07C761F9A787	09472283-CAC3-4E11-B1C0-108E63E20F39	7/5/2024	11:05:00	135	76	97	63	59	
6	35C77615-22AE-4A58-A5F0-07C761F9A787	D7E8F986-9FBB-4B4A-A271-10D998F5C0D5	7/5/2024	12:25:00	145	82	106	67	63	
7	35C77615-22AE-4A58-A5F0-07C761F9A787	799E706D-C3C4-4F5F-898F-1438CE93FBCD	7/5/2024	21:25:00	156	84	104	80	72	
8	35C77615-22AE-4A58-A5F0-07C761F9A787	4346C340-40D4-484B-8459-16C6898B05CA	7/5/2024	14:25:00	145	86	103	71	59	
9	35C77615-22AE-4A58-A5F0-07C761F9A787	25B9EC8D-F4A5-4C8D-8BA6-182B92C849CC	7/5/2024	21:45:00	151	82	109	85	69	
10	35C77615-22AE-4A58-A5F0-07C761F9A787	32BEDDD9-6C4A-4295-8522-1DC7B40D2A02	7/5/2024	20:05:00	140	90	106	74	50	
11	35C77615-22AE-4A58-A5F0-07C761F9A787	09E5F571-B32D-42B7-A3E5-2533B7B21988	8/5/2024	04:35:00	127	79	95	62	48	
12	35C77615-22AE-4A58-A5F0-07C761F9A787	26DDF24E-A2FD-49D1-AA38-296F3AEFFFOA	7/5/2024	19:45:00	138	83	101	75	55	
13	35C77615-22AE-4A58-A5F0-07C761F9A787	47422977-778C-4510-8E15-2C8699C9B00B	8/5/2024	01:35:00	128	77	97	63	51	
14	35C77615-22AE-4A58-A5F0-07C761F9A787	144FF207-315C-4160-9EEF-42AFCDD87993	7/5/2024	20:25:00	151	101	105	84	50	
15	35C77615-22AE-4A58-A5F0-07C761F9A787	ADF3F3EF-090D-4EB5-8E1F-437C093442D1	7/5/2024	18:05:00	148	82	102	81	66	
16	35C77615-22AE-4A58-A5F0-07C761F9A787	6476EF82-FCFA-4D11-8FE2-4405E6F9AF47	8/5/2024	07:08:00	135	81	100	71	54	
17	35C77615-22AE-4A58-A5F0-07C761F9A787	6075F981-0C13-46A5-AA16-4414FEFA8DF5	7/5/2024	15:25:00	135	76	93	82	59	
18	35C77615-22AE-4A58-A5F0-07C761F9A787	E6E4B447-3435-44B9-8B05-468C85F28C94	8/5/2024	05:35:00	130	65	87	60	65	
19	35C77615-22AE-4A58-A5F0-07C761F9A787	034AE3C7-6D59-471E-8CD1-4804C19FB6A5	8/5/2024	09:25:00	139	76	98	76	63	
20	35C77615-22AE-4A58-A5F0-07C761F9A787	456CA1EC-B3CE-48D5-BFB5-4C31C500E3D8	8/5/2024	02:05:00	137	81	101	58	56	
21	35C77615-22AE-4A58-A5F0-07C761F9A787	9BAB3F81-B93C-4B52-A032-4E344997DA63	7/5/2024	17:05:00	154	99	119	78	55	
22	35C77615-22AE-4A58-A5F0-07C761F9A787	F8E2EAB0-A126-4DB7-9C45-5209C9D02E42	7/5/2024	17:45:00	158	88	111	82	70	
23	35C77615-22AE-4A58-A5F0-07C761F9A787	45F3B342-4CF6-4A1F-99A2-59597E3D7491	7/5/2024	13:27:00	140	87	108	72	53	

### **Conclusión**

Una vez realizados los procedimientos de limpieza y verificación de datos, se puede proceder a manipular los archivos de manera efectiva. Este proceso incluye diversas operaciones que permiten trabajar con los datos de forma más precisa y útil para el análisis Filtrado, análisis y visualización