DW

June 29, 2025

1 Data Wrangling Inicial - Preparación del Dataset para Random Forest

Este notebook contiene el proceso de limpieza, exploración y preparación del dataset médico para el entrenamiento de un modelo Random Forest que predice el riesgo de infarto agudo del miocardio.

1.1 Objetivos

- Explorar la estructura y calidad de los datos
- Identificar y manejar valores faltantes/outliers
- Estandarizar nombres de variables
- Preparar datos para entrenamiento del modelo (Random Forest)

[4]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

[5]:
```
df = pd.read_csv("../data/raw/Medicaldataset.csv")


print("Filas, columnas:", df.shape)
print("Variables:", df.columns.tolist())
df.head(3)
```

```
Filas, columnas: (1319, 9)
Variables: ['Age', 'Gender', 'Heart rate', 'Systolic blood pressure', 'Diastolic
blood pressure', 'Blood sugar', 'CK-MB', 'Troponin', 'Result']
```

[5]:

	Age	Gender	Heart rate	Systolic blood pressure	Diastolic blood pressure	Blood sugar	CK-MB	Troponin	Result
0	64	1	66	160	83	160.0	1.80	0.012	negative
1	21	1	94	98	46	296.0	6.75	1.060	positive
2	55	1	64	160	77	270.0	1.99	0.003	negative

[6]:
```
sns.set_theme(style="whitegrid", palette="Blues", font="serif", font_scale=1)
```

2 Referencias de nombres de variables del dataset

Nombre original	Nombre estandarizado
Age	age
Gender	gender
Heart rate	hr
Systolic blood pressure	sbp
Diastolic blood pressure	dbp
Blood sugar	bs
CK-MB	ckmb
Troponin	trop
Result	res

2.1 Frecuencia cardíaca (Heart Rate)

2.1.1 Valores de Referencia (Adultos)

Parámetro	Unidad	Normal	Bradicardia	Taquicardia
Heart Rate (reposo)	bpm	60–100	<60 bpm	>100 bpm

- **Bradicardia**: frecuencia <60 bpm. No siempre patológica (común en atletas, sueño); puede causar síntomas si es significativa.
- **Taquicardia**: frecuencia >100 bpm en reposo. Puede ser fisiológica (ejercicio) o patológica, con riesgo de síncope y eventos trombóticos.

Fuente de validación científica:

- Medical News Today - Tachycardia vs. bradycardia https://www.medicalnewstoday.com/articles/tachycardia-vs-bradycardia
- Cleveland Clinic - Bradycardia: Symptoms, Causes & Treatment https://my.clevelandclinic.org/health/diseases/17841-bradycardia

---

2.2 Presión Arterial Sistólica (Systolic Blood Pressure)

2.2.1 Valores de Referencia (Adultos)

Parámetro	Unidad	Normal	Elevado	Etapa 1	Etapa 2	Crisis hipertensiva
SBP	mm Hg	<120	120–129	130–139	140	180

- **Elevado** (120–129 mm Hg): riesgo aumentado, requiere cambios en estilo de vida.
- **Hipertensión etapa 1** (130–139): se inicia tratamiento si hay alto riesgo cardiovascular.
- **Hipertensión etapa 2** (140): indicador definitivo, se recomienda medicación.
- **Crisis hipertensiva** (180): emergencia médica inmediata si hay síntomas (dolor torácico, dificultad respiratoria…).

Fuente de validación científica:

- American Heart Association - Understanding Blood Pressure Readings https://www.heart.org/en/health-topics/high-blood-pressure/understanding-blood-pressure-readings
- Harvard Health - A look at diastolic blood pressure https://www.health.harvard.edu/heart-health/a-look-at-diastolic-blood-pressure

---

2.3 Presión Arterial Diastólica (Diastolic Blood Pressure)

2.3.1 Valores de Referencia (Adultos)

Parámetro	Unidad	Normal	Hipotensión	Etapa 1	Etapa 2	Crisis hipertensiva
DBP	mm Hg	< 80	< 60	80–89	90	120

- **Normal**: diastólica menor de 80 mm Hg :contentReferenceoaicite:3.
- **Hipotensión (presión baja)**: diastólica < 60 mm Hg. Puede causar mareo, síncope y aumentar riesgo en órganos.
- **Etapa 1 de hipertensión**: 80–89 mm Hg.
- **Etapa 2 de hipertensión**: 90 mm Hg.
- **Crisis hipertensiva**: diastólica 120 mm Hg, junto con sistólica 180, es emergencia médica..

2.3.2 Contexto Clínico y Significado

- La presión diastólica mide la presión en las arterias entre latidos, cuando el corazón está en reposo.
- Una diastólica **muy baja (< 60 mm Hg)**, especialmente en ancianos, puede provocar cansancio, mareos y aumento de caídas.
- Aunque la sistólica suele recibir más atención en mayores de 50 años, la diastólica también importa: valores elevados reflejan riesgo cardiovascular.

Fuente de validación científica:

- American Heart Association - Understanding Blood Pressure Readings https://www.heart.org/en/health-topics/high-blood-pressure/understanding-blood-pressure-readings
- Harvard Health - A look at diastolic blood pressure https://www.health.harvard.edu/heart-health/a-look-at-diastolic-blood-pressure

---

2.4 Glucemia (Blood Sugar)

2.4.1 Valores de Referencia (Ayuno)

Parámetro	Unidad	Normal	Hipoglucemia	Prediabetes	Diabetes
Blood Sugar	mg/dL	70–99	< 70 (síntomas < 55)	100–125	126

- **Hipoglucemia**: <70 mg/dL; síntomas comunes como sudoración, temblor, confusión; <55 suele ser sintomática.
- **Hiperglucemia**: ayuno >125 mg/dL, o >180 mg/dL posprandial; causa fatiga, sed, visión borrosa.

Fuente de validación científica:

- NCBI StatPearls - Blood Glucose https://www.ncbi.nlm.nih.gov/books/NBK279364/
- Mayo Clinic - Blood Glucose https://www.mayoclinic.org/diseases-conditions/diabetes/diagnosis-treatment/drc-20371451

2.5 CK-MB (Isoenzima MB de Creatina Quinasa)

(Unidad: ng/mL)

2.5.1 Valores de Referencia (Adultos)

Parámetro	Unidad	Normal	Elevado	Crítico
CK-MB	ng/mL	< 5	5–10	> 10
CK-MB%	%	< 1.0	1.0–2.0	>2.0

2.5.2 Criterios Diagnósticos para IAM

1. CK-MB 5 ng/mL + CK-MB% >2%
2. Aumento >50 % en 2 muestras consecutivas (3 h de separación)

2.5.3 Contexto Clínico

- Pico a las 12–24 h tras el inicio de síntomas
- Retorno a valores basales en 48–72 h
- Valores >10 ng/mL sugieren infarto extenso

Fuentes de validación científica:

- University of Michigan MLabs – CK-MB Isoenzyme https://mlabs.umich.edu/tests/creatine-kinase-total-and-mb-isoenzyme
- PathologyOutlines – Cardiac CK https://www.pathologyoutlines.com/topic/chemistrycardiacck.html

---

2.6 Troponina T (solo)

(Unidad: ng/mL; 1 ng/mL = 1,000 ng/L)

2.6.1 Valores de Referencia (Adultos)

Parámetro	Unidad	Normal	Elevado	Crítico
Troponin T	ng/mL	<0.01	0.01–0.13	0.14
hs-Troponin T	ng/mL	<0.014	0.014–0.052	0.053

- **Troponina T convencional**: <0.01 ng/mL normal; >0.14 sugiere infarto :contentReferenceoaicite:9.
- **hs-Troponin T**: $>99.^{o}$ percentil o 0.053 ng/mL es crítico y altamente sugestivo de IAM :contentReferenceoaicite:10.

Fuentes de validación científica:

- PathologyOutlines – Cardiac Troponins
  https://www.pathologyoutlines.com/topic/chemistrycardiactroponins.html
- University of Michigan MLabs – High-Sensitivity Troponin T
  https://mlabs.umich.edu/tests/troponin-t-high-sensitivity

2.7 Resumen General

Variable	Unidad	Normal	Elevado / Bajo	Crítico / IAM / Crisis
HR (reposo)	bpm	60–100	—	<60 (bradicardia), >100 (taquicardia)
SBP	mm Hg	<120	120–129 (elevado)	130 etapa 1–2; 180 crisis
DBP	mm Hg	<80	<60 hipotensión; 80–89 etapa 1	90 etapa 2; 120 crisis
Glucemia ayuno	mg/dL	70–99	<70 hipoglucemia; 100–125 prediabetes	126 diabetes
CK-MB	ng/mL	<5	5–10 elevado	>10 infarto extenso
CK-MB %	%	<1.0	1.0–2.0 elevado	>2.0
Troponina T	ng/mL	<0.01	0.01–0.13 elevado	0.14
hs-Troponin T	ng/mL	<0.014	0.014–0.052 elevado	0.053

---

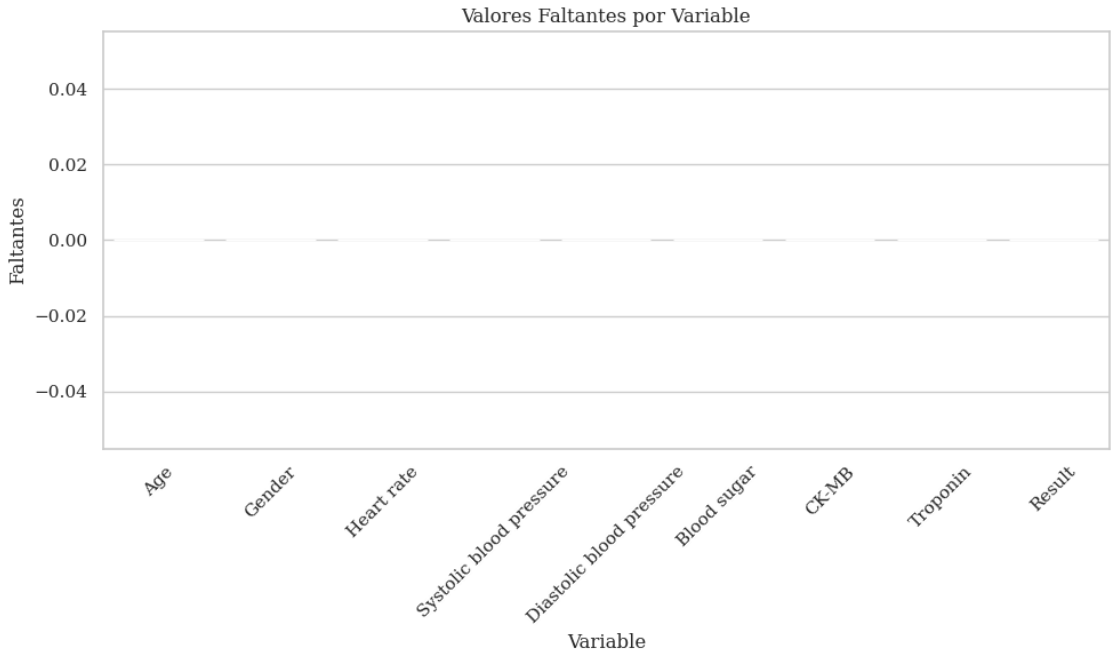
3 Análisis de Valores Faltantes

Se explora el dataset buscando valores faltantes, se observa que no hay valores faltantes.

[7]:
```
print("Valores faltantes por columna:")
print(df.isnull().sum())

missing = df.isnull().sum().reset_index()
missing.columns = ['Variable', 'Faltantes']
```

```
# Gráfico interactivo
plt.figure(figsize=(10, 6))
sns.barplot(data=missing, x='Variable', y='Faltantes')
plt.title('Valores Faltantes por Variable')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

```
Valores faltantes por columna:
Age                          0
Gender                       0
Heart rate                   0
Systolic blood pressure      0
Diastolic blood pressure     0
Blood sugar                  0
CK-MB                        0
Troponin                     0
Result                       0
dtype: int64
```



3.1 Renombrado de Variables

Se decide cambiar el nombre de las variables del dataset para facilitar el trabajo posterior y mantener consistencia en el código. Los cambios incluyen: - Nombres más cortos y manejables - Eliminación de espacios y caracteres especiales - Nomenclatura en minúsculas para mejor compatibilidad

Cambios realizados: - Age → age - Gender → gender - Heart rate → hr - Systolic blood pressure → sbp - Diastolic blood pressure → dbp - Blood sugar → bs - CK-MB → ckmb - Troponin → trop - Result → res

[8]:
```
df.rename(columns={'Age': 'age',
                   'Gender': 'gender',
                   'Heart rate': 'hr',
                   'Systolic blood pressure': 'sbp',
                   'Diastolic blood pressure': 'dbp',
                   'Blood sugar': 'bs',
                   'CK-MB': 'ckmb',
                   'Troponin': 'trop',
                   'Result': 'res',},
                    inplace=True)
df
```

[8]:

	age	gender	hr	sbp	dbp	bs	ckmb	trop	res
0	64	1	66	160	83	160.0	1.80	0.012	negative
1	21	1	94	98	46	296.0	6.75	1.060	positive
2	55	1	64	160	77	270.0	1.99	0.003	negative
3	64	1	70	120	55	270.0	13.87	0.122	positive
4	55	1	64	112	65	300.0	1.08	0.003	negative
…	…	…	..	…	…	…	…	…	…
1314	44	1	94	122	67	204.0	1.63	0.006	negative
1315	66	1	84	125	55	149.0	1.33	0.172	positive
1316	45	1	85	168	104	96.0	1.24	4.250	positive
1317	54	1	58	117	68	443.0	5.80	0.359	positive
1318	51	1	94	157	79	134.0	50.89	1.770	positive

[1319 rows x 9 columns]

3.2 Exploración Estadística del Dataset

Se realiza un análisis estadístico descriptivo del dataset para comprender la distribución y características de las variables numéricas.

[9]:
```
print('Dimesniones: Filas - Columnas')
print(df.shape)
print('                    ')
print('Información General:')
print(df.info())
```

```
Dimesniones: Filas - Columnas
(1319, 9)

Información General:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1319 entries, 0 to 1318
Data columns (total 9 columns):
```

```
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   age     1319 non-null   int64
 1   gender  1319 non-null   int64
 2   hr      1319 non-null   int64
 3   sbp     1319 non-null   int64
 4   dbp     1319 non-null   int64
 5   bs      1319 non-null   float64
 6   ckmb    1319 non-null   float64
 7   trop    1319 non-null   float64
 8   res     1319 non-null   object
dtypes: float64(3), int64(5), object(1)
memory usage: 92.9+ KB
None
```

[10]:
```
df.describe()
```

[10]:

	age	gender	hr	sbp	dbp	bs	ckmb	trop
count	1319.000000	1319.000000	1319.000000	1319.000000	1319.000000	1319.000000	1319.000000	1319.000000
mean	56.191812	0.659591	78.336619	127.170584	72.269143	146.634344	15.274306	0.360942
std	13.647315	0.474027	51.630270	26.122720	14.033924	74.923045	46.327083	1.154568
min	14.000000	0.000000	20.000000	42.000000	38.000000	35.000000	0.321000	0.001000
25%	47.000000	0.000000	64.000000	110.000000	62.000000	98.000000	1.655000	0.006000
50%	58.000000	1.000000	74.000000	124.000000	72.000000	116.000000	2.850000	0.014000
75%	65.000000	1.000000	85.000000	143.000000	81.000000	169.500000	5.805000	0.085500
max	103.000000	1.000000	1111.000000	223.000000	154.000000	541.000000	300.000000	10.300000

[11]:
```
df['res'].value_counts()
```

[11]:
```
res
positive    810
negative    509
Name: count, dtype: int64
```

[12]:
```
df.isnull()
```

[12]:

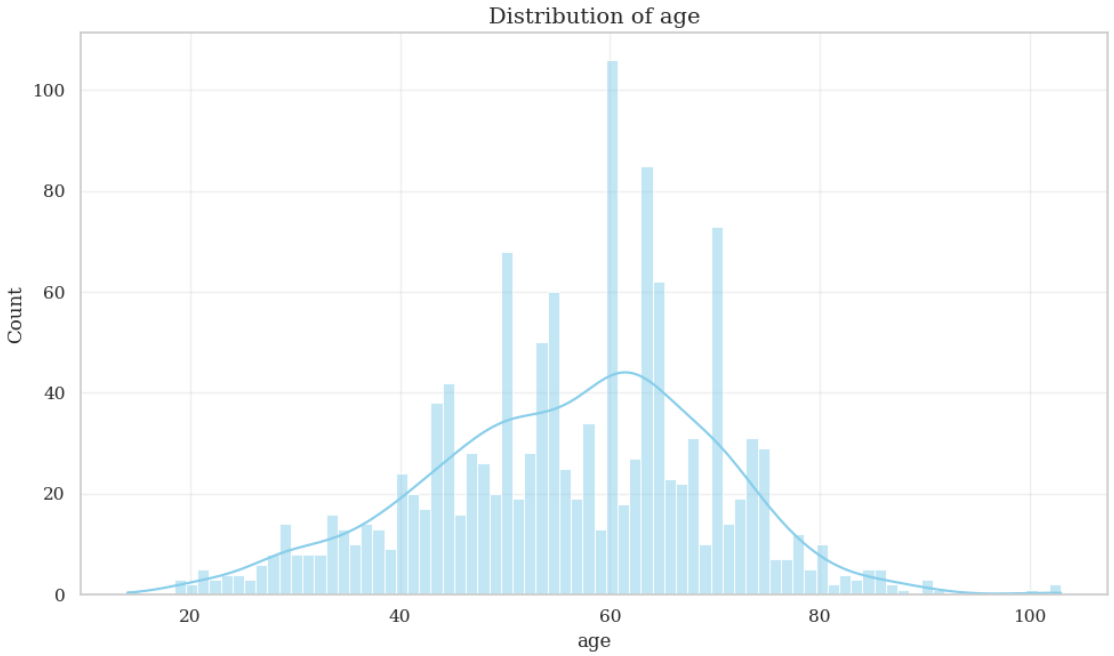
	age	gender	hr	sbp	dbp	bs	ckmb	trop	res
0	False	False	False	False	False	False	False	False	False

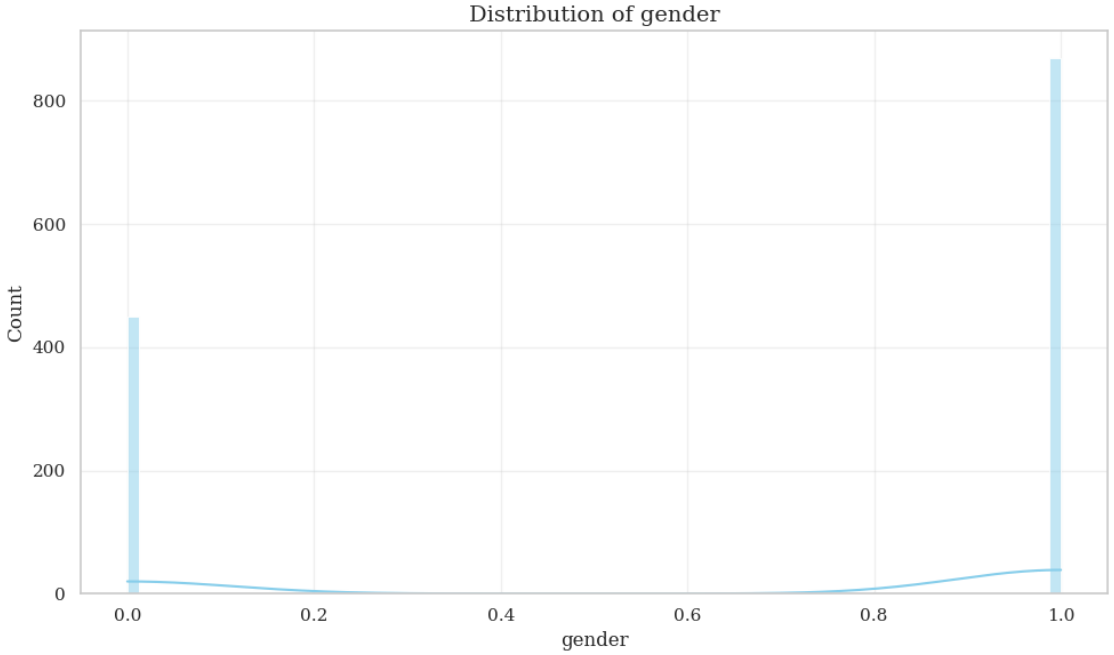
```
1     False   False  False  False  False  False  False  False  False
2     False   False  False  False  False  False  False  False  False
3     False   False  False  False  False  False  False  False  False
4     False   False  False  False  False  False  False  False  False
…       …       …      …      …      …      …      …      …
1314  False   False  False  False  False  False  False  False  False
1315  False   False  False  False  False  False  False  False  False
1316  False   False  False  False  False  False  False  False  False
1317  False   False  False  False  False  False  False  False  False
1318  False   False  False  False  False  False  False  False  False

[1319 rows x 9 columns]
```

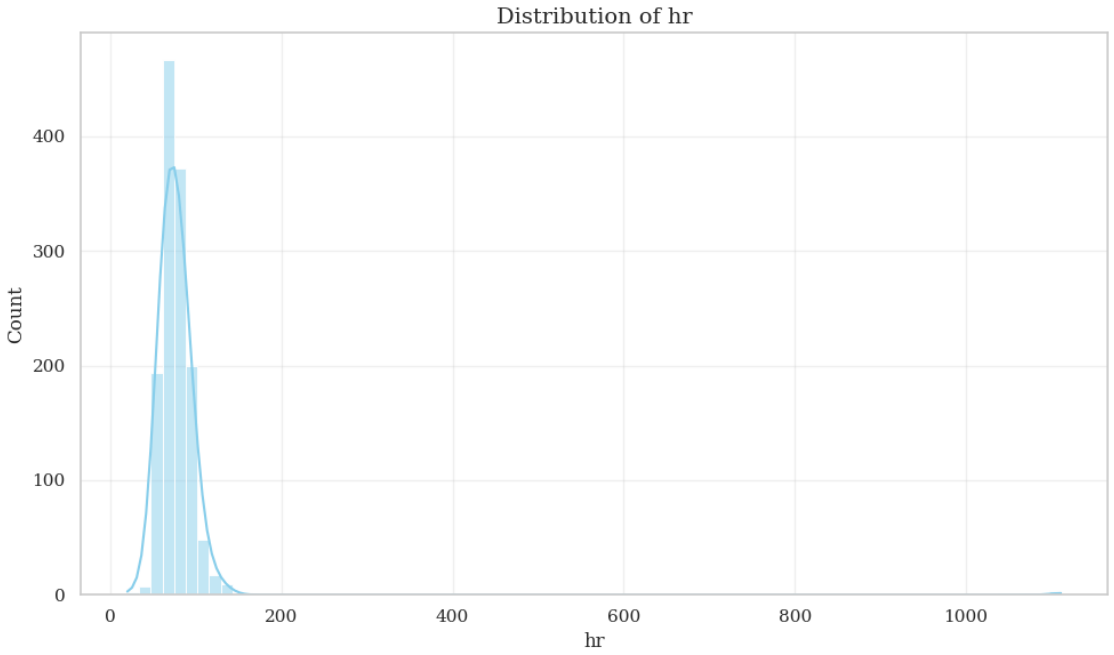
3.3 Análisis de Distribución de Variables Numéricas

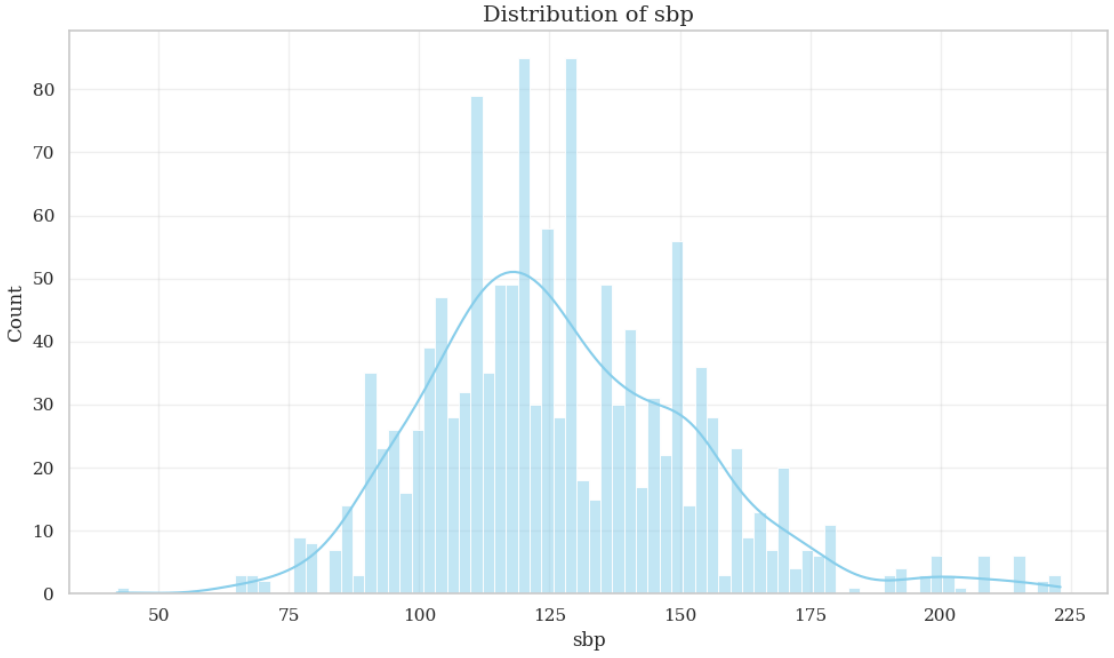
Se realiza un análisis visual exhaustivo de las variables numéricas del dataset mediante histogramas, diagramas de caja (boxplots) y gráficos de violín para comprender mejor la distribución, presencia de valores atípicos y características estadísticas de cada variable.

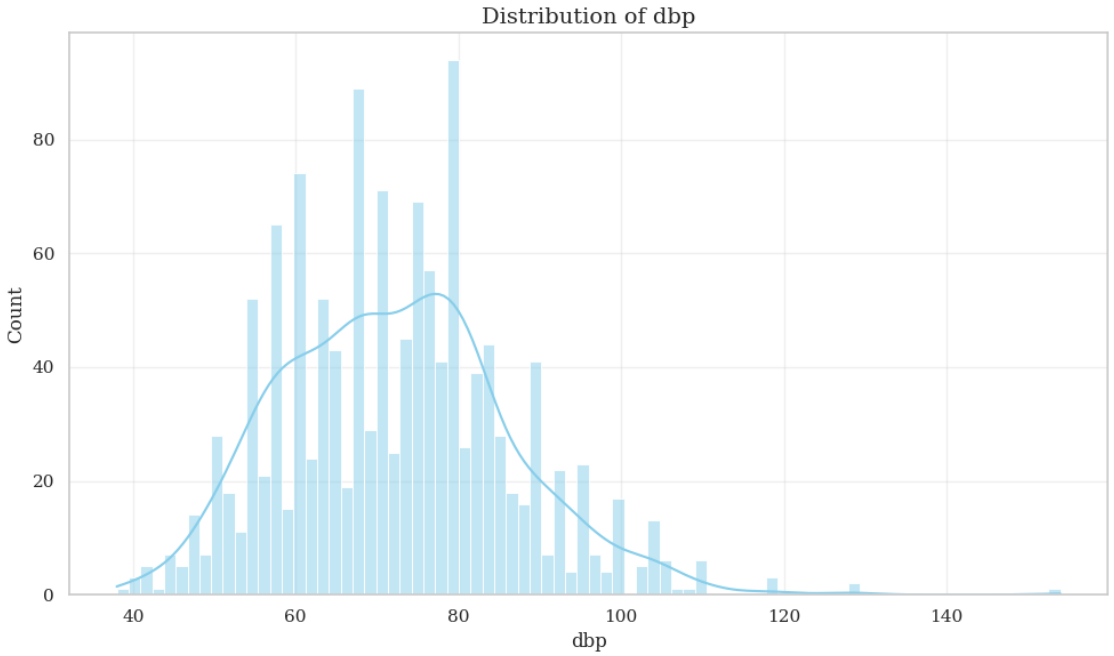
[13]:
```python
numeric_cols = df.select_dtypes(include='number').columns

for i, col in enumerate(numeric_cols):
    plt.figure(figsize=(10, 6))
    sns.histplot(
        data=df,
        x=col,
        bins=80,
        kde=True,
        color='skyblue',
    )
    plt.title(f'Distribution of {col}', fontsize=14)
    plt.xlabel(col, fontsize=12)
    plt.ylabel('Count', fontsize=12)
    plt.grid(True, alpha=0.3)
    plt.tight_layout()
    plt.show()
```

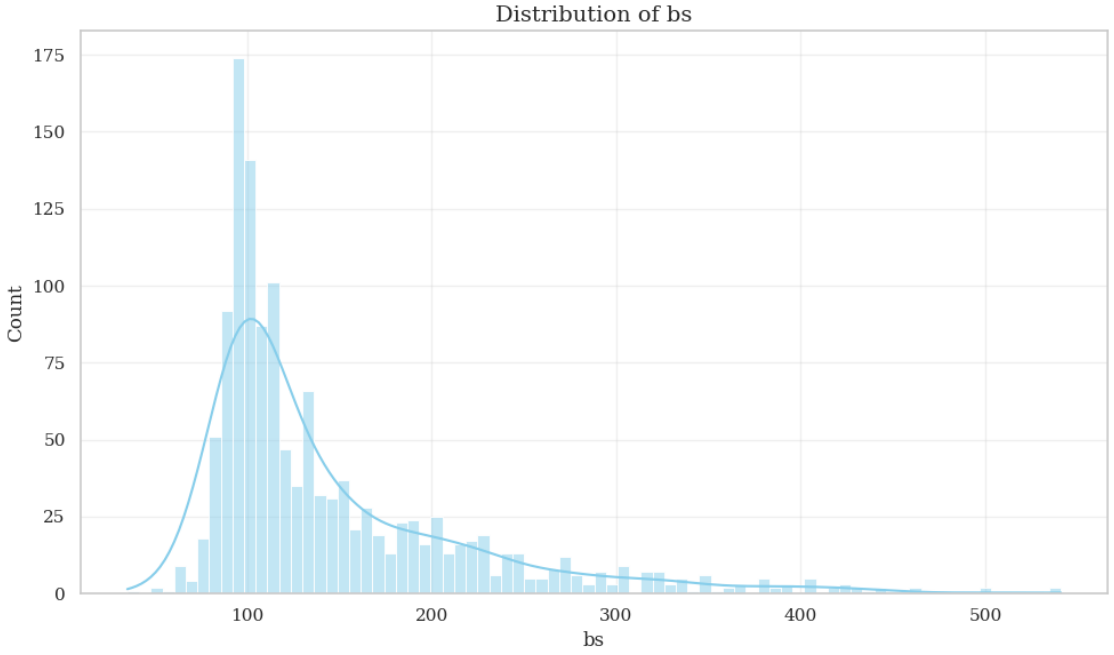

Distribution of age
Count
age
0
20
40
60
80
100
20
40
60
80
100

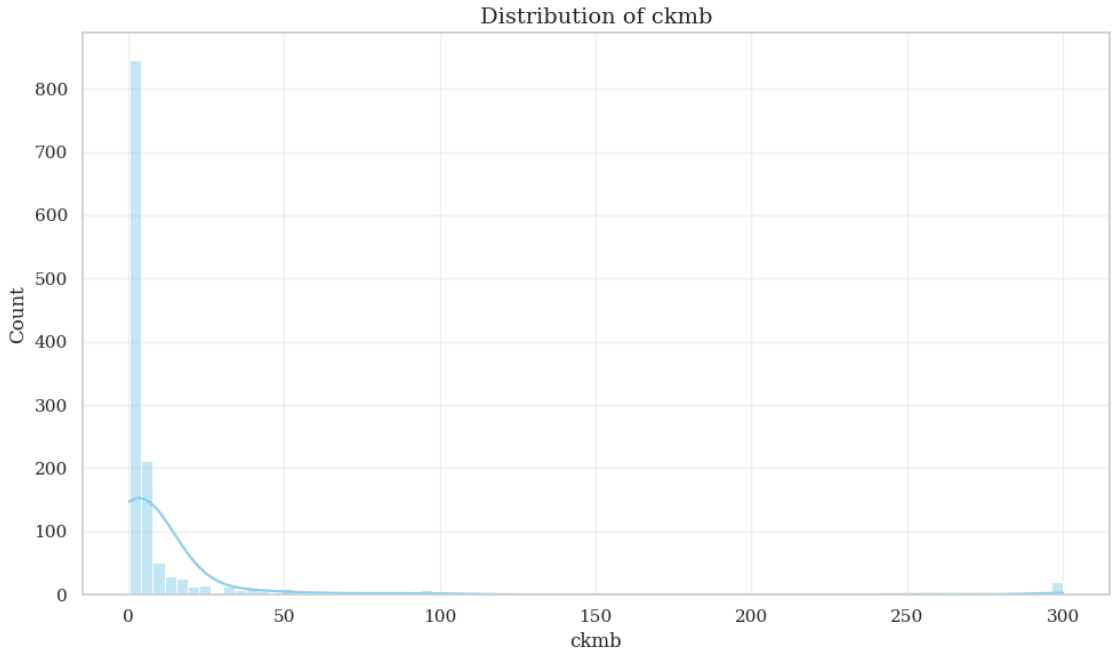

Distribution of gender
Count
gender
0
200
400
600
800
0.0
0.2
0.4
0.6
0.8
1.0

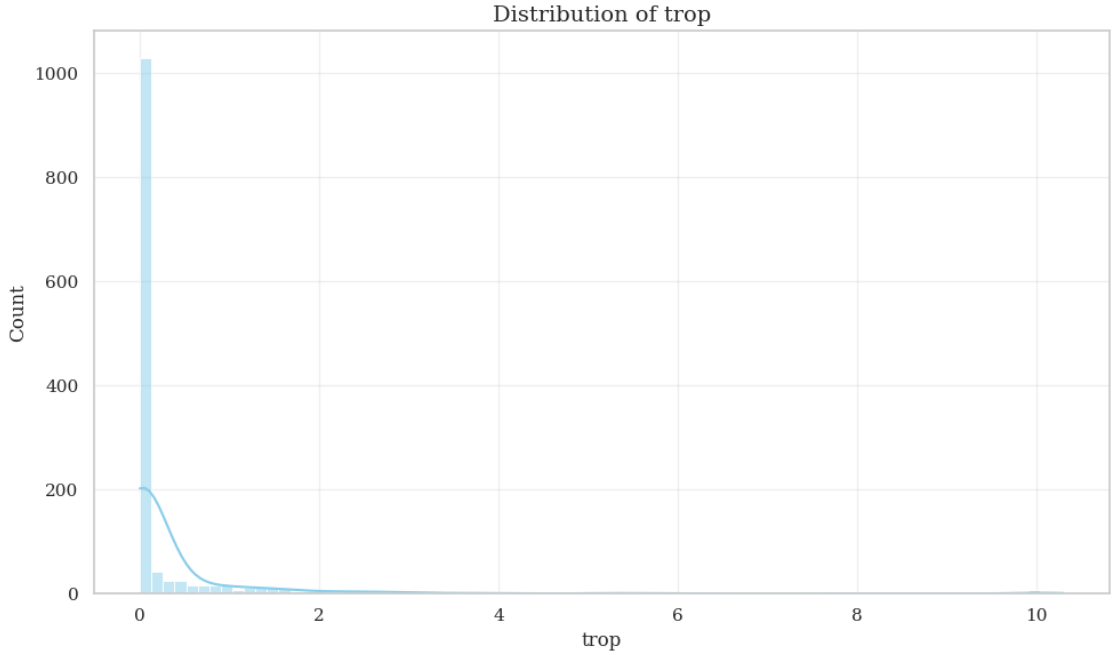

Distribution of hr
Count
hr
0
100
200
300
400
0
200
400
600
800
1000


Distribution of sbp
Count
sbp
0
10
20
30
40
50
60
70
80
50
75
100
125
150
175
200
225


Distribution of dbp
Count
dbp
0
20
40
60
80
40
60
80
100
120
140

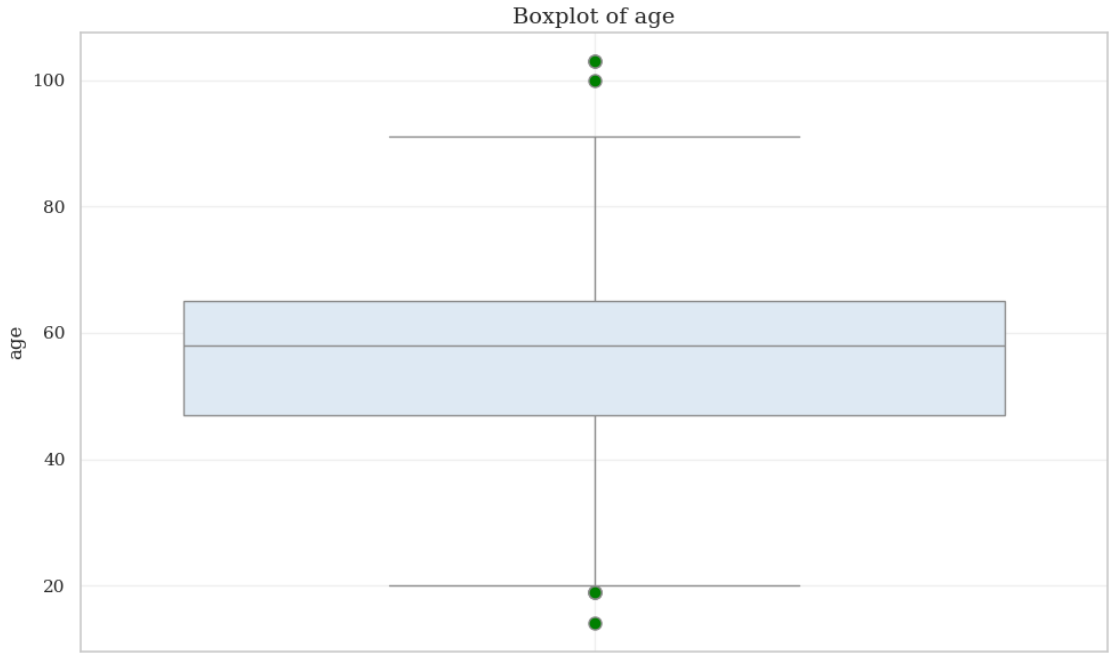

Distribution of bs
Count
bs
0
25
50
75
100
125
150
175
100
200
300
400
500



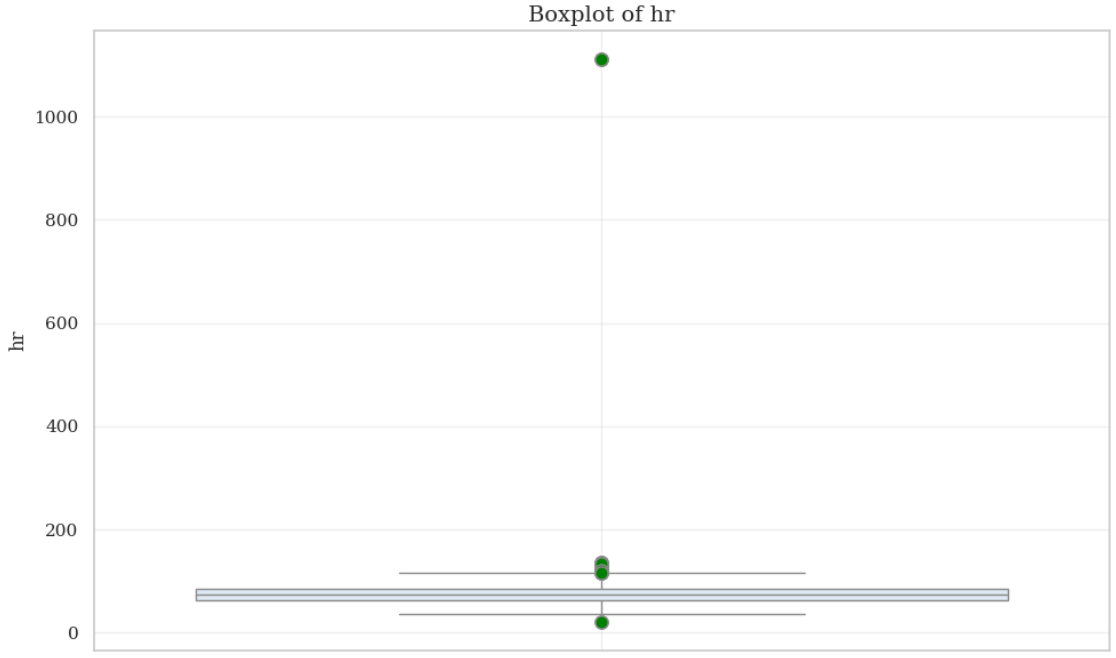


[14]:
```
numeric_cols = df.select_dtypes(include='number').columns

for col in numeric_cols:
    plt.figure(figsize=(10, 6))
```

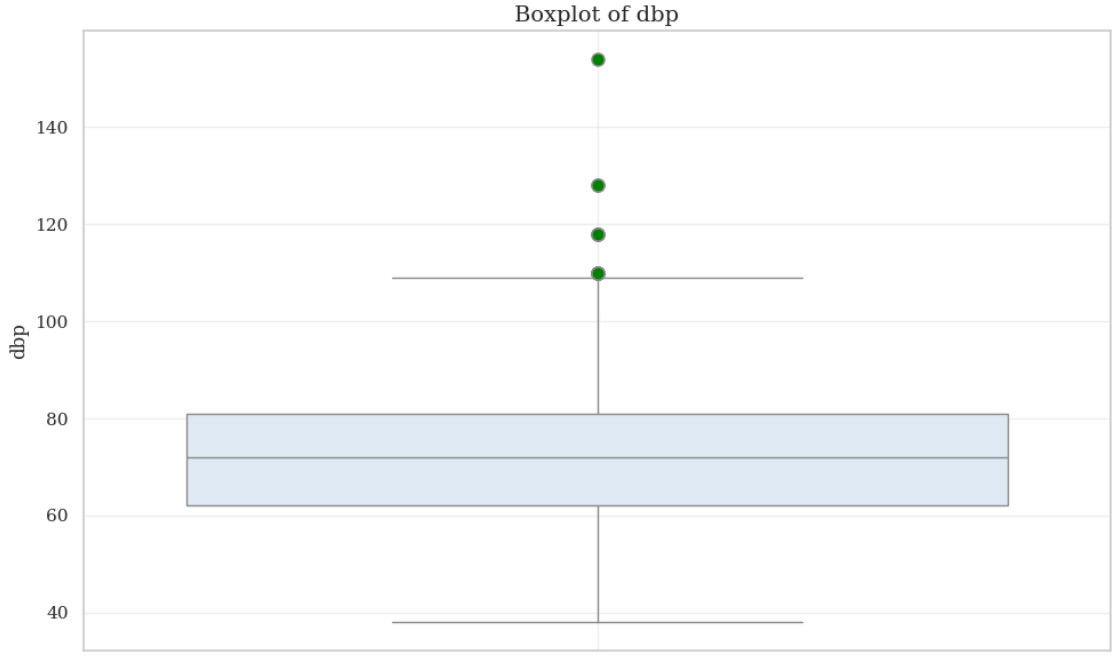
```
    sns.boxplot(
        data=df,
        y=col,
        flierprops={'marker': 'o', 'markerfacecolor': 'green', 'markersize': 8,␣
↪'linestyle': 'none'}
    )
    plt.title(f'Boxplot of {col}', fontsize=14, )
    plt.ylabel(col, fontsize=12)
    plt.grid(True, alpha=0.3)
    plt.tight_layout()
    plt.show()
```

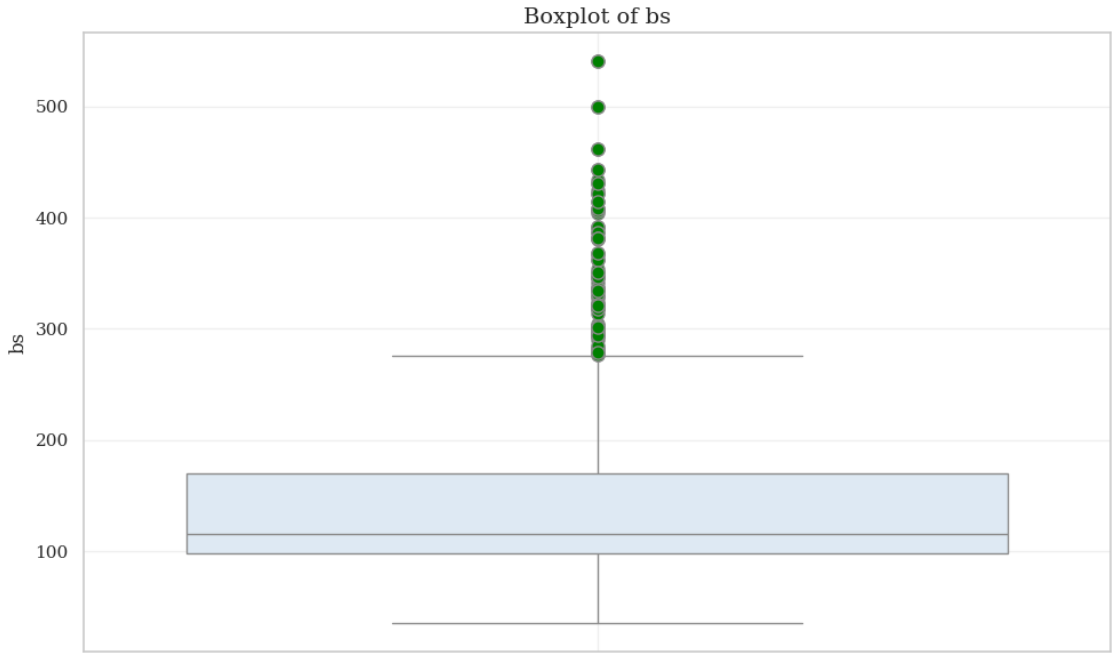


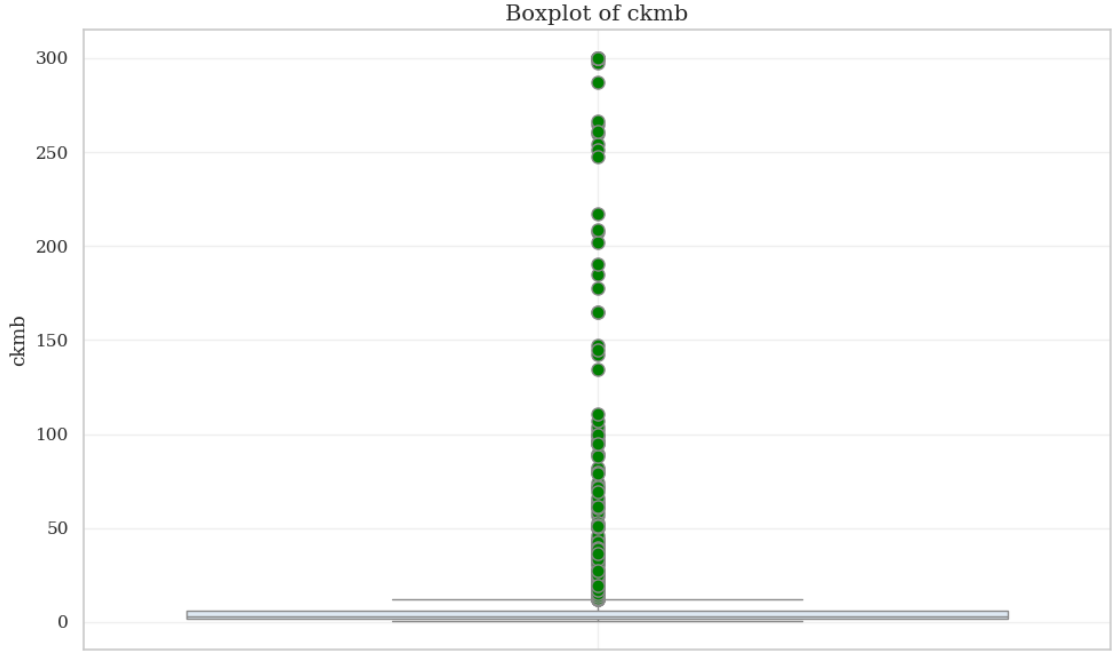

Boxplot of gender
gender
1.0
0.8
0.6
0.4
0.2
0.0

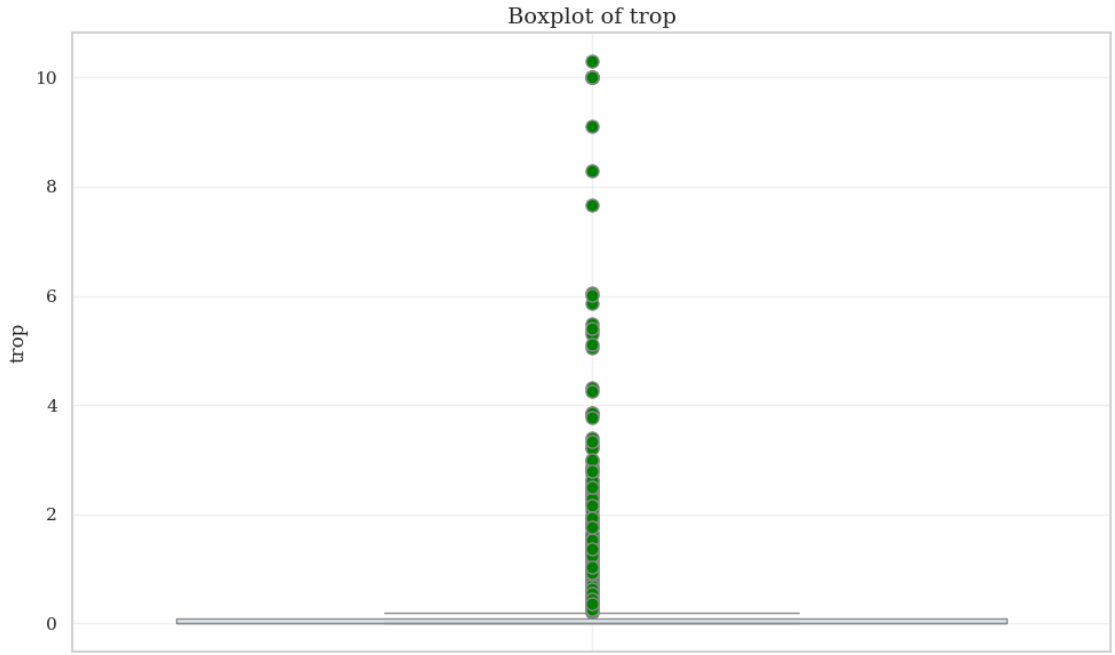

Boxplot of hr
hr
1000
800
600
400
200
0


Boxplot of sbp
sbp
225
200
175
150
125
100
75
50


Boxplot of dbp
dbp
140
120
100
80
60
40


Boxplot of bs
bs
500
400
300
200
100


Boxplot of ckmb
ckmb
300
250
200
150
100
50
0



3.4 Eliminando un registro de hr

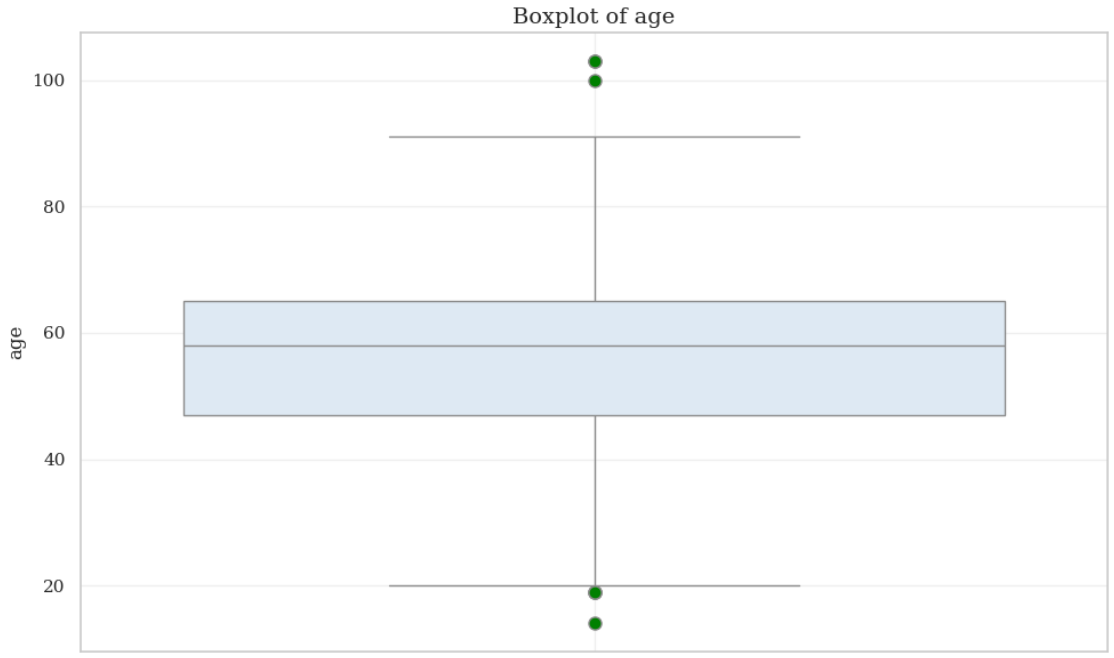
Se decide eliminar un registro de Heart Rate = 1111 debido a que era solo uno, es un valor prácticamete alejado de la realidad clínica y más cercano a un error al introducirlo, lo cual pordría provocar ruido a la hora de entrenar el modelo.

[15]:
```
df = df[df['hr'] <= 500]
df.hr
```

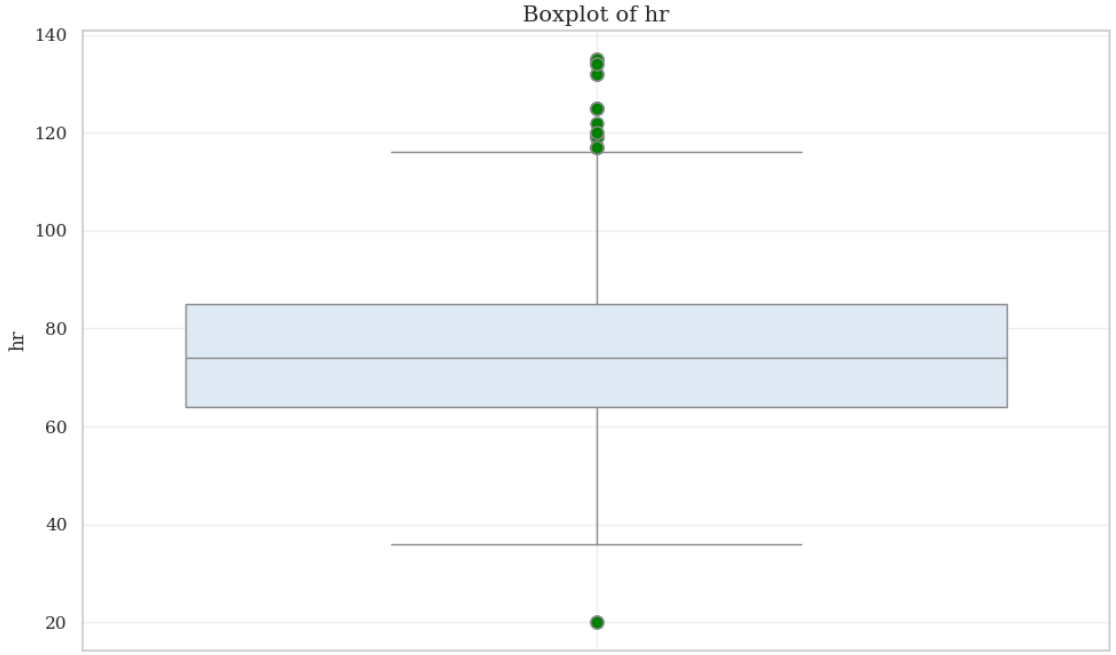
[15]:
```
0       66
1       94
2       64
3       70
4       64
        ..
1314    94
1315    84
1316    85
1317    58
1318    94
Name: hr, Length: 1316, dtype: int64
```

[16]:
```
numeric_cols = df.select_dtypes(include='number').columns

for col in numeric_cols:
    plt.figure(figsize=(10, 6))
```

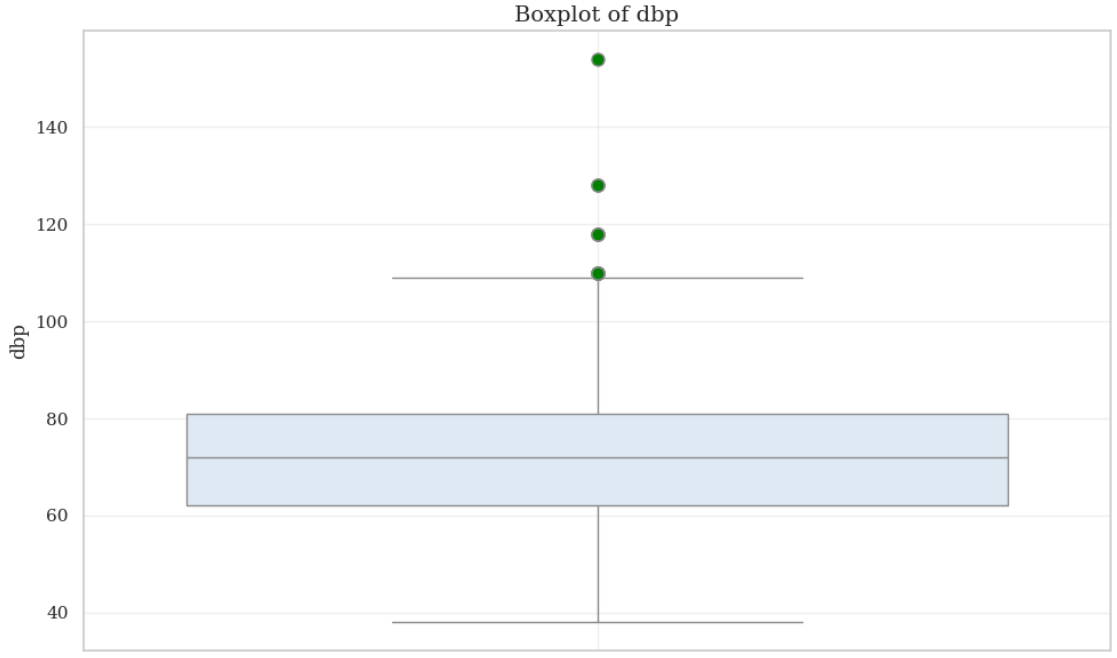
```
    sns.boxplot(
        data=df,
        y=col,
        flierprops={'marker': 'o', 'markerfacecolor': 'green', 'markersize': 8,␣
↪'linestyle': 'none'}
    )
    plt.title(f'Boxplot of {col}', fontsize=14, )
    plt.ylabel(col, fontsize=12)
    plt.grid(True, alpha=0.3)
    plt.tight_layout()
    plt.show()
```

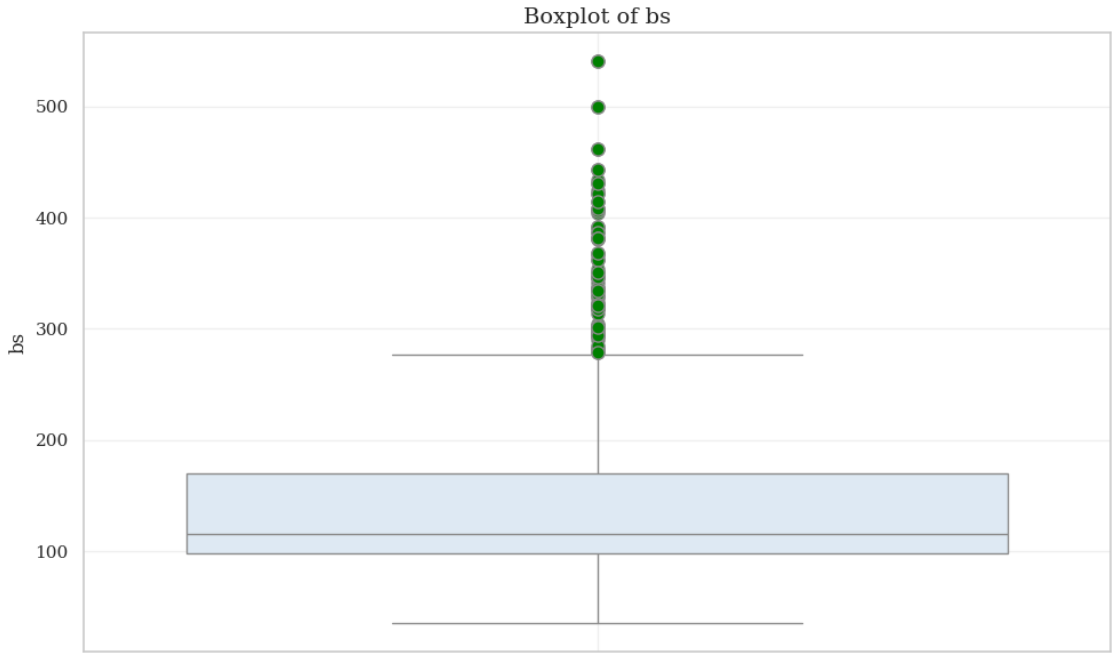



Boxplot of gender
gender
1.0
0.8
0.6
0.4
0.2
0.0

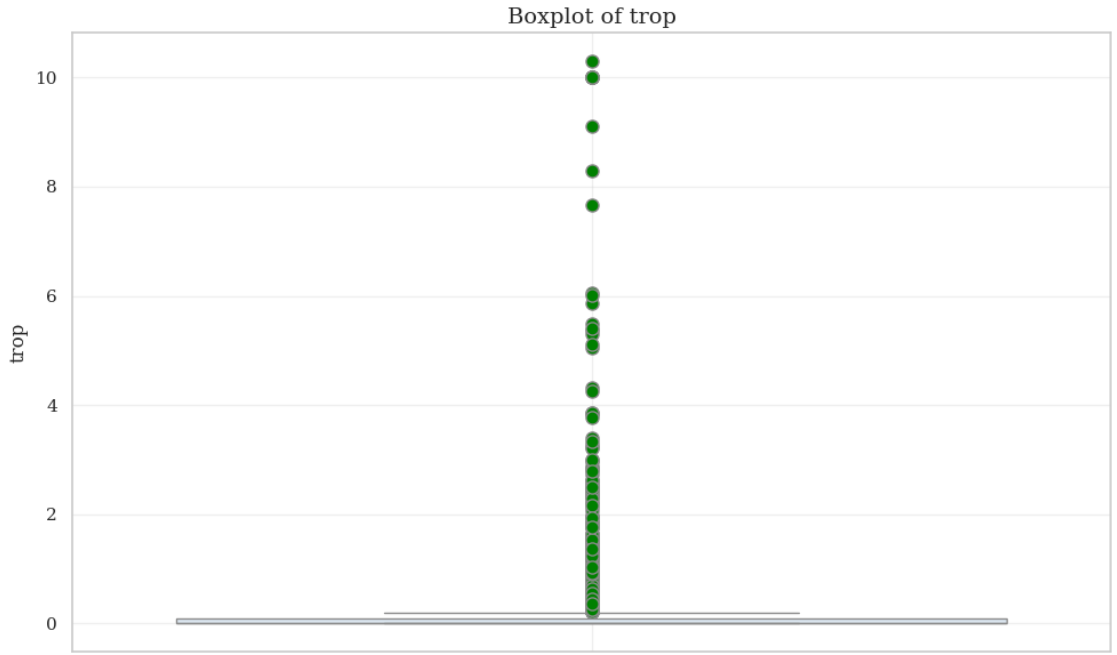

Boxplot of hr
hr
140
120
100
80
60
40
20


Boxplot of sbp
sbp
225
200
175
150
125
100
75
50

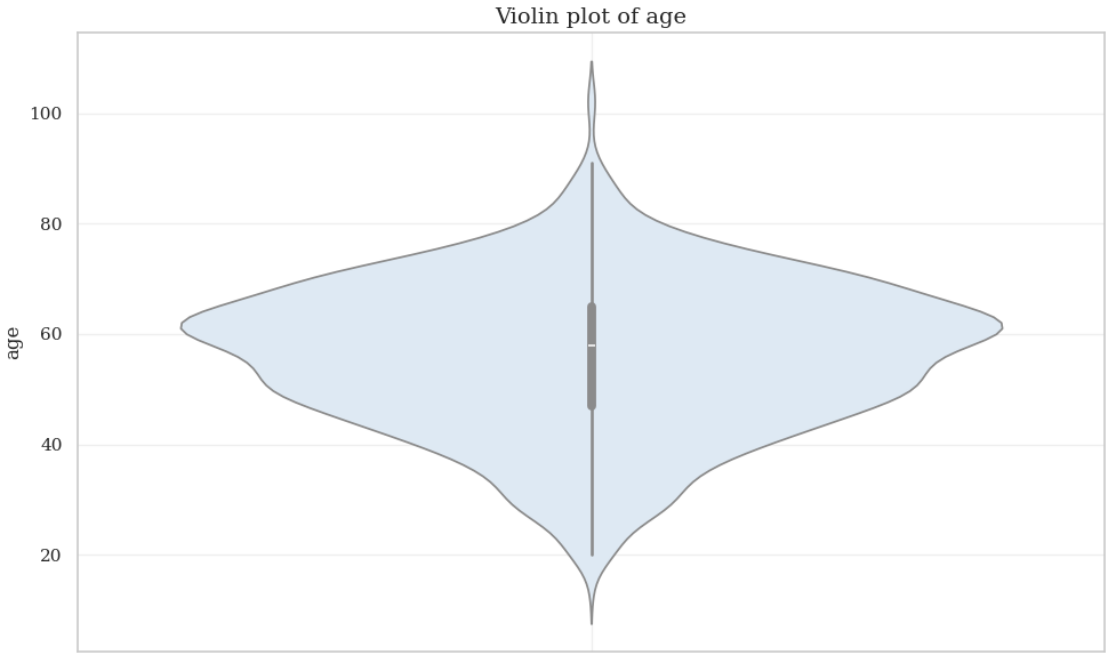

Boxplot of dbp
dbp
140
120
100
80
60
40

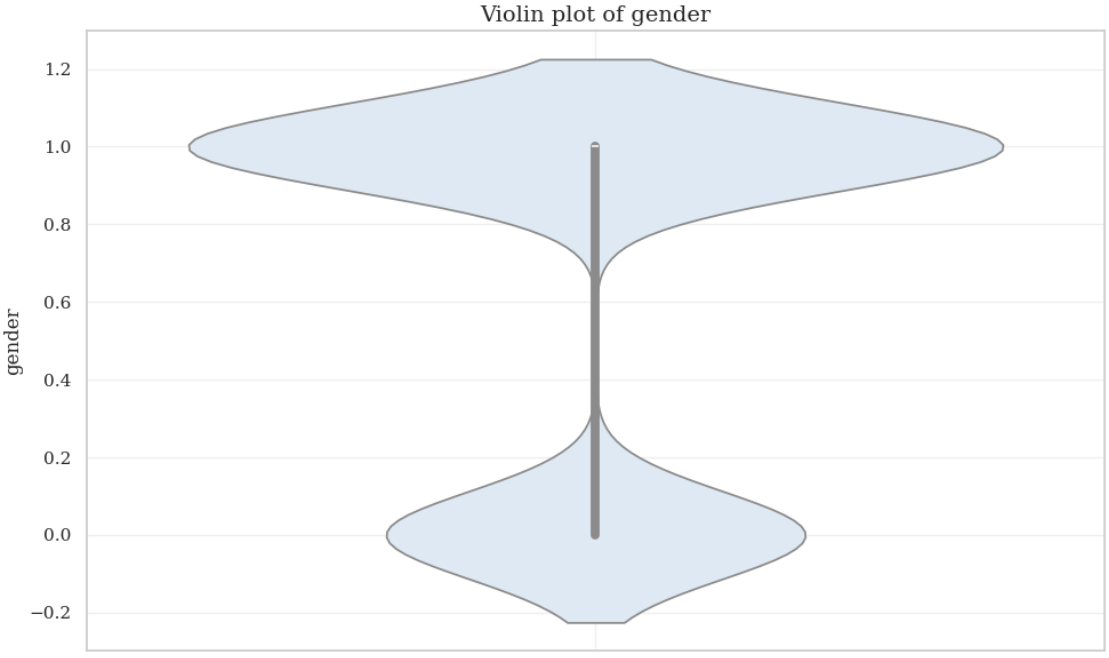

Boxplot of bs
bs
500
400
300
200
100

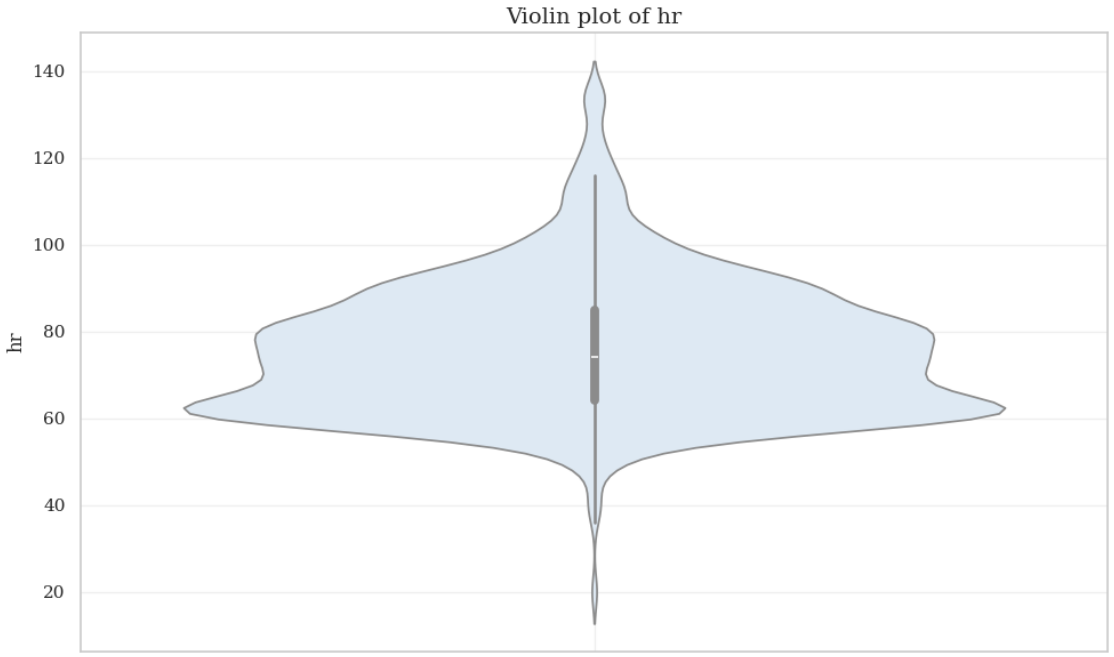

Boxplot of ckmb
ckmb
300
250
200
150
100
50
0

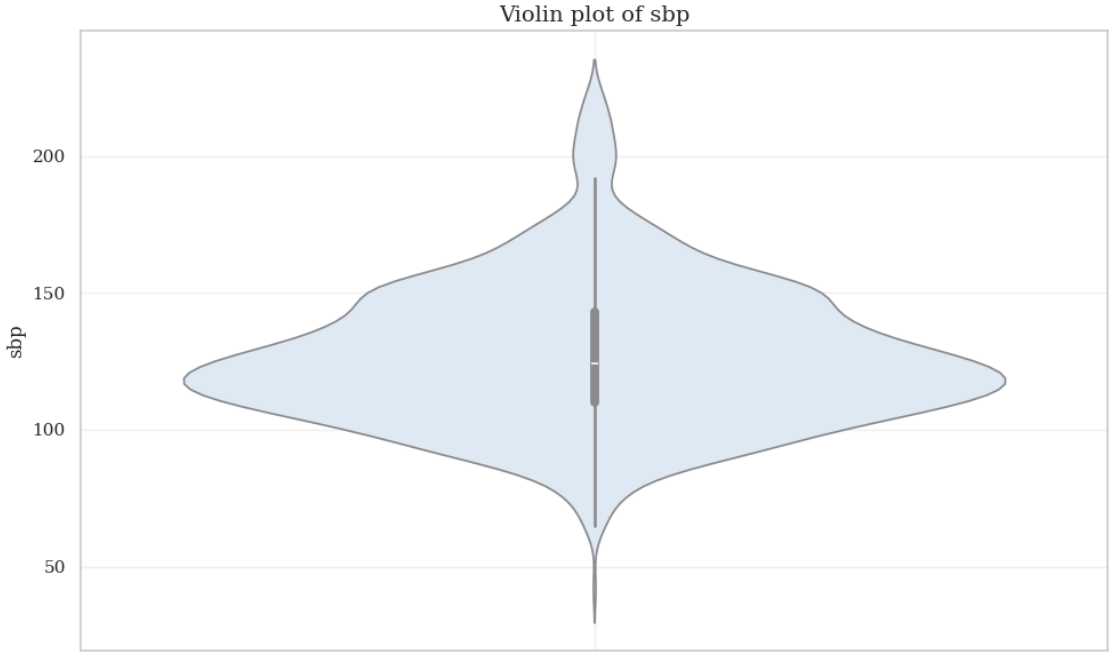


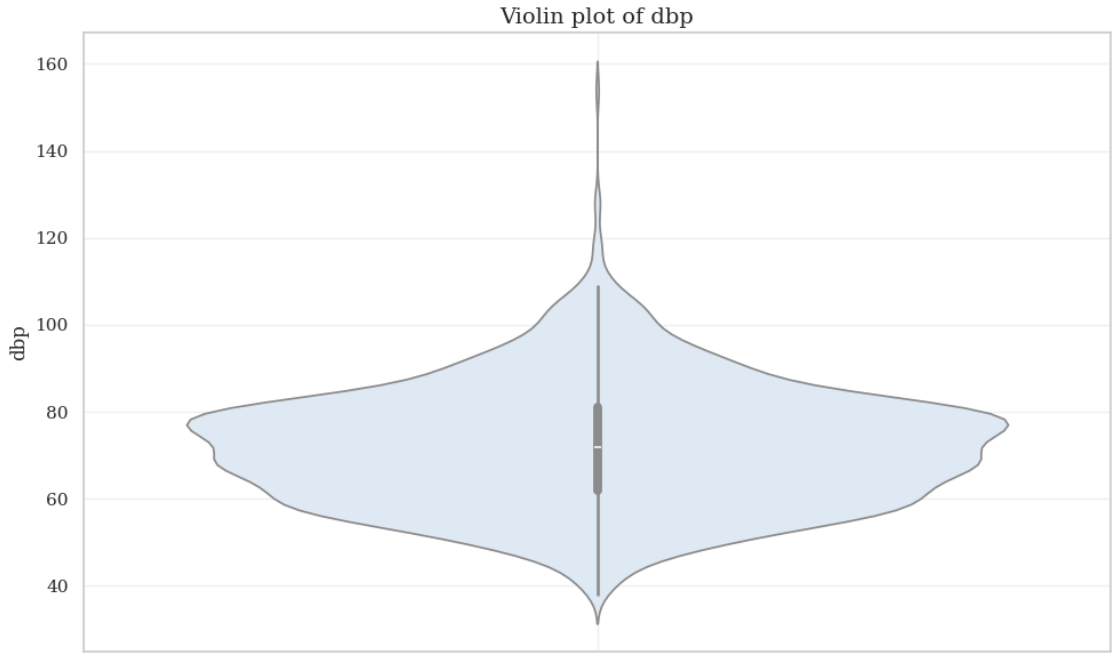
[17]:
```
numeric_cols = df.select_dtypes(include='number').columns

for i, col in enumerate(numeric_cols):
    plt.figure(figsize=(10, 6))
    sns.violinplot(
        data=df,
        y=col,
        inner='box'
    )
    plt.title(f'Violin plot of {col}', fontsize=14)
    plt.ylabel(col, fontsize=12)
    plt.grid(True, alpha=0.3)
    plt.tight_layout()
    plt.show()
```

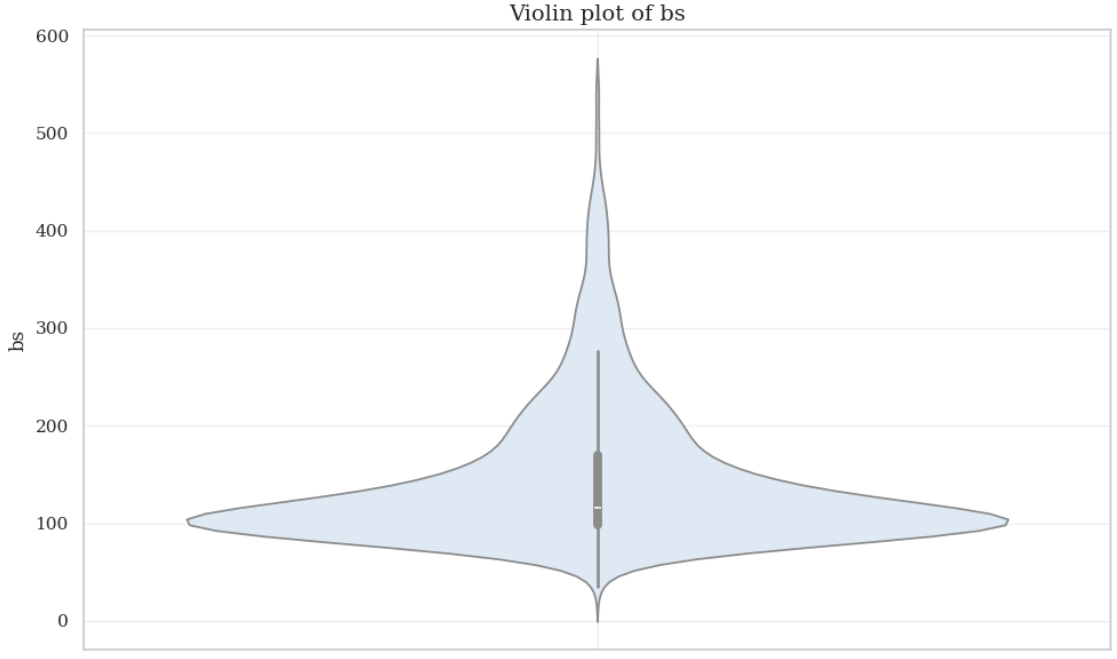

Violin plot of age
age
100
80
60
40
20

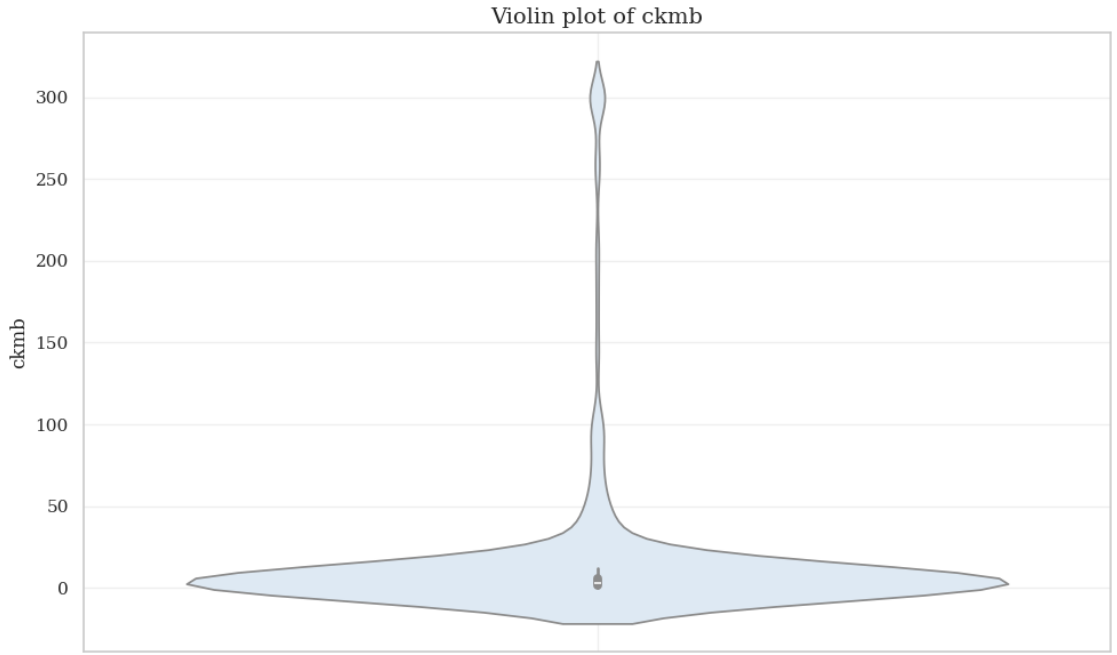

Violin plot of gender
gender
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−0.2

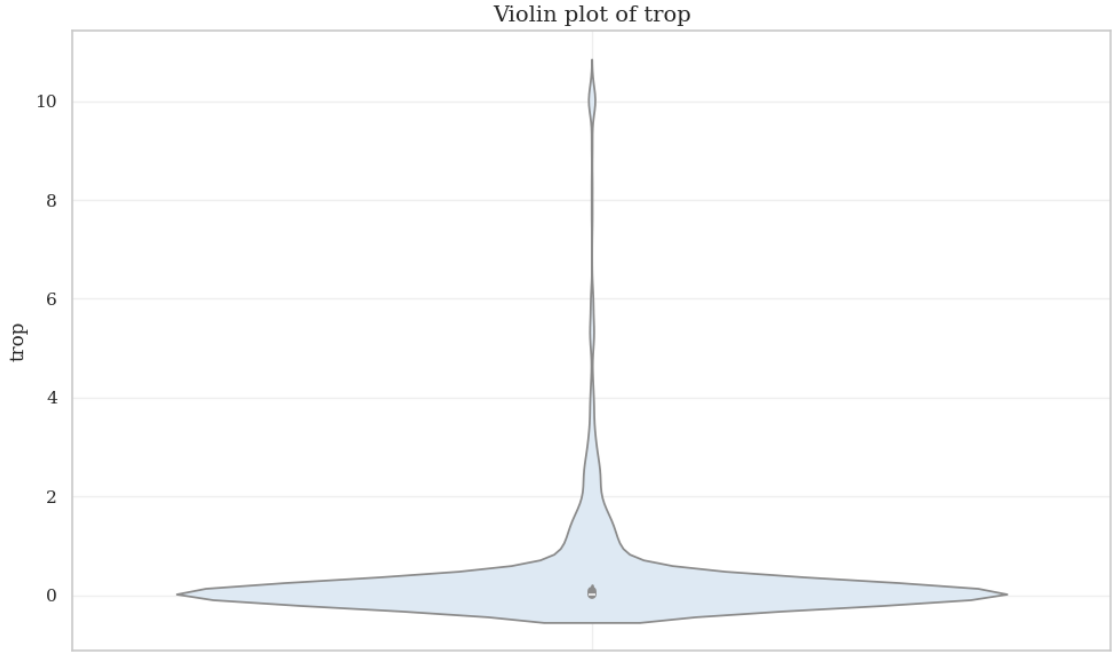

Violin plot of hr
hr
140
120
100
80
60
40
20


Violin plot of sbp
sbp
200
150
100
50


Violin plot of dbp
dbp
160
140
120
100
80
60
40


Violin plot of bs
bs
600
500
400
300
200
100
0





3.5 Convertir 'res' a números:

Se decide convertir el campo resultados de categorías positivo y negativo a valores numéricoas 1 y 0 respectivamente, esto posibilitaría la mejor integración con el entrenamiento del modelo.

[18]:
```
df_numeric = df.copy()
df_numeric['res'] = df_numeric['res'].map({'positive': 1, 'negative': 0})
```

3.6 Exportar base de datos preparada para entrenamiento

Se exporta la base de datos con todas las modificaciones realizadas, preparada para el entrenamiento del modelo Random Forest

[19]:
```
df_numeric.to_csv('../data/processed/medicaldataset.csv', index=False)
```