```
import numpy as np
import pandas as pd
import matplotlib as mpl
import matplotlib.pyplot as plt
%matrplotlib inline
rcParams['figure.figure.dpi'] = 100
rcParams['axes.grid'] = True
```

```
data = pd.read_csv('loan_approval.csv')
data.shape
```

```
(614, 13)
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status (Approved)
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	Y
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	Rural	N
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	Urban	Y
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	Y
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0	Urban	Y

```
data.tail(5)
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status (Approved)
609	LP001974	Female	No	0	Graduate	No	2900	0.0	71.0	360.0	1.0	Rural	Y
610	LP001976	Male	Yes	3+	Graduate	No	4106	0.0	40.0	180.0	1.0	Rural	Y
611	LP001978	Male	Yes	1	Graduate	No	8072	240.0	253.0	360.0	1.0	Urban	Y
612	LP001979	Male	Yes	0	Graduate	No	7583	0.0	187.0	360.0	1.0	Urban	Y
613	LP001980	Female	No	0	Graduate	Yes	4583	0.0	133.0	360.0	0.0	Semiurban	N

```
data.info()
```

```
RangeIndex: 614 entries, 0 to 613
Data columns (total 13 columns):
```

The process of cleaning data

```
# Count missing values in each column
data.isnull().sum()
```

```
Loan_ID                0
Gender                13
Married                3
Dependents            15
Education              0
Self_Employed         32
ApplicantIncome        0
CoapplicantIncome      0
LoanAmount            22
Loan_Amount_Term      14
Credit_History        50
Property_Area          0
Loan_Status (Approved) 0
dtype: int64
```

```
# Check for duplicate rows
data.duplicated().sum()
```

```
0
```

```
# Drop duplicate rows
data.drop_duplicates()
```

```
# Check data types
data.dtypes
```

```
Loan_ID                object
Gender                 object
Married                object
Dependents             object
Education              object
Self_Employed          object
ApplicantIncome         int64
CoapplicantIncome       float64
LoanAmount              float64
Loan_Amount_Term        float64
Credit_History          float64
Property_Area          object
Loan_Status (Approved) object
dtype: int64
```

```
# Fill numerical columns with mean
data['LoanAmount'].fillna(data['LoanAmount'].mean(), inplace=True)
data['Loan_Amount_Term'].fillna(data['Loan_Amount_Term'].mean(), inplace=True)
data['Credit_History'].fillna(data['Credit_History'].mean(), inplace=True)

# Fill categorical columns with mode (most frequent value)
data['Gender'].fillna(data['Gender'].mode[0], inplace=True)
data['Married'].fillna(data['Married'].mode[0], inplace=True)
data['Dependents'].fillna(data['Dependents'].mode[0], inplace=True)
data['Self_Employed'].fillna(data['Self_Employed'].mode[0], inplace=True)
```

```
data.isnull().sum()
```

```
0
```

```
# Check if there are any missing values left
data.isnull().sum().sum
```

```
0
```

```
# Check for duplicate rows
data.duplicated().sum()
```

```
0
```

```
# Remove duplicate rows
data.drop_duplicates()
```

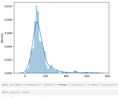
```
# Check data types after conversion
data.dtypes
```

```
Loan_ID                object
Gender                 object
Married                object
Dependents             object
Education              object
Self_Employed          object
ApplicantIncome         int64
CoapplicantIncome       float64
LoanAmount              float64
Loan_Amount_Term        float64
Credit_History          float64
Property_Area          object
Loan_Status (Approved) object
dtype: int64
```

```
# Create a copy of the dataset
df = data[['ApplicantIncome', 'CoapplicantIncome', 'LoanAmount']]
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
plt.tight_layout()
```

```
<ipykr_output_1>
```




```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
plt.tight_layout()
```

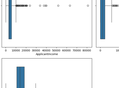
```
<ipykr_output_2>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
plt.tight_layout()
```

```
<ipykr_output_3>
```

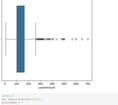
```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
plt.tight_layout()
```

```
<ipykr_output_4>
```



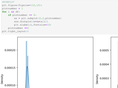

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks([])
plt.tight_layout()
```

```
<ipykr_output_5>
```



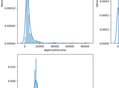

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks([])
    axes[i].set_xlabel('')
plt.tight_layout()
```

```
<ipykr_output_6>
```



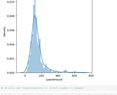

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks([])
    axes[i].set_xlabel('')
    axes[i].grid(False)
plt.tight_layout()
```

```
<ipykr_output_7>
```




```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks([])
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
plt.tight_layout()
```

```
<ipykr_output_8>
```




```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks([])
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
plt.tight_layout()
```

```
<ipykr_output_9>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks([])
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
plt.tight_layout()
```

```
<ipykr_output_10>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks([])
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
plt.tight_layout()
```

```
<ipykr_output_11>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
plt.tight_layout()
```

```
<ipykr_output_12>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
plt.tight_layout()
```

```
<ipykr_output_13>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
plt.tight_layout()
```

```
<ipykr_output_14>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
plt.tight_layout()
```

```
<ipykr_output_15>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
plt.tight_layout()
```

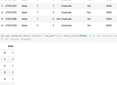
```
<ipykr_output_16>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
    axes[i].spines['left'].set_ticks[]
plt.tight_layout()
```

```
<ipykr_output_17>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
    axes[i].spines['left'].set_ticks[]
    axes[i].spines['left'].set_ticklabels[]
plt.tight_layout()
```

```
<ipykr_output_18>
```




```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
    axes[i].spines['left'].set_ticks[]
    axes[i].spines['left'].set_ticklabels[]
    axes[i].spines['left'].set_offset(0)
plt.tight_layout()
```

```
<ipykr_output_19>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
    axes[i].spines['left'].set_ticks[]
    axes[i].spines['left'].set_ticklabels[]
    axes[i].spines['left'].set_offset(0)
    axes[i].spines['left'].set_visible(False)
plt.tight_layout()
```

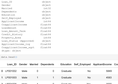
```
<ipykr_output_20>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
    axes[i].spines['left'].set_ticks[]
    axes[i].spines['left'].set_ticklabels[]
    axes[i].spines['left'].set_offset(0)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_ticks(False)
plt.tight_layout()
```

```
<ipykr_output_21>
```

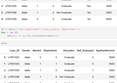
```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
    axes[i].spines['left'].set_ticks[]
    axes[i].spines['left'].set_ticklabels[]
    axes[i].spines['left'].set_offset(0)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_ticks(False)
    axes[i].spines['left'].set_ticklabels(False)
plt.tight_layout()
```

```
<ipykr_output_22>
```




```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
    axes[i].spines['left'].set_ticks[]
    axes[i].spines['left'].set_ticklabels[]
    axes[i].spines['left'].set_offset(0)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_ticks(False)
    axes[i].spines['left'].set_ticklabels(False)
    axes[i].spines['left'].set_offset(False)
plt.tight_layout()
```

```
<ipykr_output_23>
```




```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
    axes[i].spines['left'].set_ticks[]
    axes[i].spines['left'].set_ticklabels[]
    axes[i].spines['left'].set_offset(0)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_ticks(False)
    axes[i].spines['left'].set_ticklabels(False)
    axes[i].spines['left'].set_offset(False)
    axes[i].spines['left'].set_visible(False)
plt.tight_layout()
```

```
<ipykr_output_24>
```

```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible(False)
    axes[i].spines['right'].set_visible(False)
    axes[i].spines['bottom'].set_visible(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_linewidth(0.5)
    axes[i].spines['left'].set_color('black')
    axes[i].spines['left'].set_zorder(10)
    axes[i].spines['left'].set_clip_on(False)
    axes[i].spines['left'].set_position(('outward', 10))
    axes[i].spines['left'].set_visible(True)
    axes[i].spines['left'].set_ticks[]
    axes[i].spines['left'].set_ticklabels[]
    axes[i].spines['left'].set_offset(0)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_ticks(False)
    axes[i].spines['left'].set_ticklabels(False)
    axes[i].spines['left'].set_offset(False)
    axes[i].spines['left'].set_visible(False)
    axes[i].spines['left'].set_ticks(False)
plt.tight_layout()
```

```
<ipykr_output_25>
```



```
# Set up subplots for boxplots
fig, axes = plt.subplots(1, 3, figure=(15, 5))
for i, col in enumerate(df.columns):
    sns.boxplot(y=df[col], ax=axes[i])
    axes[i].set_title(f'Boxplot of {col}')
    axes[i].ticklabel_format(style='plain', axis='y')
    axes[i].set_ylim(0, 2*df[col].mean())
    axes[i].set_xlim(-0.5, 0.5)
    axes[i].set_xticks[]
    axes[i].set_xlabel('')
    axes[i].grid(False)
    axes[i].spines['top'].set_visible
```