Patient Appointment No-show Prediction

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the data
df_raw = pd.read_csv("KaggleV2-May-2016.csv")
df_clean = df_raw.copy()

# Before Cleaning: Target Distribution
plt.figure(figsize=(5, 4))
sns.countplot(x='No-show', data=df_raw)
plt.title("Original Target Distribution (Raw Data)")
plt.show()

# Preprocessing steps

# Convert No-show column
df_clean['No-show'] = df_clean['No-show'].map({'No': 0, 'Yes': 1})

# Convert dates
df_clean['ScheduledDay'] = pd.to_datetime(df_clean['ScheduledDay'])
df_clean['AppointmentDay'] = pd.to_datetime(df_clean['AppointmentDay'])

# Add AppointmentGap
df_clean['AppointmentGap'] = (df_clean['AppointmentDay'] - df_clean['ScheduledDay']).dt.days

# Drop unwanted columns
df_clean = df_clean.drop(['PatientId', 'AppointmentID', 'ScheduledDay', 'AppointmentDay'], axis=1)

# Save age distribution before filtering
age_before = df_raw['Age']
```

```
# Remove invalid ages
df_clean = df_clean[df_clean['Age'] >= 0]

# Plot Age Distributions
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
sns.histplot(age_before, bins=30, kde=True)
plt.title("Age Distribution (Before Cleaning)")

plt.subplot(1, 2, 2)
sns.histplot(df_clean['Age'], bins=30, kde=True, color="orange")
plt.title("Age Distribution (After Cleaning)")
plt.show()

# Plot Appointment Gap
plt.figure(figsize=(6, 4))
sns.histplot(df_clean['AppointmentGap'], bins=40, kde=True)
plt.title("Distribution of Days Between Scheduling and Appointment")
plt.xlabel("AppointmentGap (Days)")
plt.show()

# One-hot encode
df_encoded = pd.get_dummies(df_clean, columns=['Gender', 'Neighbourhood'], drop_first=True)

# Top 10 Neighborhoods (based on counts before encoding)
plt.figure(figsize=(10, 4))
df_raw['Neighbourhood'].value_counts().head(10).plot(kind='bar')
plt.title("Top 10 Neighborhoods by Appointment Count")
plt.xticks(rotation=45)
plt.show()

# Correlation Heatmap
plt.figure(figsize=(12, 8))
corr = df_encoded.corr()
```

```
sns.heatmap(corr[['No-show']].sort_values(by='No-show', ascending=False), cmap='coolwarm', annot=True)
plt.title("Feature Correlation with Target (No-show)")
plt.show()
```