

Practical Experiment: Data Representation and Visualization

Objective:

To implement a Python script to perform data representation and visualization using various methods.

Steps:

1. **Read Data:**
 - Use the Pandas library to read a CSV file.
2. **Line Plots:**
 - Create line plots to visualize trends over time or sequential data.
3. **Bar Plots:**
 - Create bar plots to compare different categories or groups.
4. **Histograms and Density Plots:**
 - Create histograms and density plots to visualize the distribution of numeric data.
5. **Scatter Plots:**
 - Create scatter plots to observe relationships between pairs of numeric variables.

Sample Data:

Use the 'Adult' dataset from UCI Adult dataset.

Python Script:

python

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

# Step 1: Read Data
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.data'
columns = ['age', 'workclass', 'fnlwgt', 'education',
            'education_num', 'marital_status', 'occupation',
            'relationship', 'race', 'sex', 'capital_gain',
            'capital_loss', 'hours_per_week', 'native_country', 'income']
data = pd.read_csv(url, header=None, names=columns, na_values=' ?')

# Display the first few rows of the dataset
print("Original Data:\n", data.head())

# Step 2: Line Plots
# Create a line plot for age and hours per week
plt.figure(figsize=(10, 6))
plt.plot(data['age'], data['hours_per_week'], 'o', markersize=2)
```

```

plt.title('Line Plot: Age vs Hours per Week')
plt.xlabel('Age')
plt.ylabel('Hours per Week')
plt.grid(True)
plt.show()

# Step 3: Bar Plots
# Create a bar plot for the count of different education levels
plt.figure(figsize=(12, 6))
sns.countplot(x='education', data=data,
order=data['education'].value_counts().index)
plt.title('Bar Plot: Education Levels')
plt.xlabel('Education')
plt.ylabel('Count')
plt.xticks(rotation=45)
plt.show()

# Step 4: Histograms and Density Plots
# Create a histogram and density plot for age
plt.figure(figsize=(10, 6))
sns.histplot(data['age'], kde=True)
plt.title('Histogram and Density Plot: Age')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()

# Step 5: Scatter Plots
# Create a scatter plot for age and capital gain
plt.figure(figsize=(10, 6))
sns.scatterplot(x='age', y='capital_gain', data=data)
plt.title('Scatter Plot: Age vs Capital Gain')
plt.xlabel('Age')
plt.ylabel('Capital Gain')
plt.grid(True)
plt.show()

# Save the figures to files
plt.savefig('line_plot.png')
plt.savefig('bar_plot.png')
plt.savefig('histogram_density_plot.png')
plt.savefig('scatter_plot.png')

```

Explanation:

1. **Read Data:** The script reads the Adult dataset from a URL and loads it into a Pandas DataFrame.
2. **Line Plots:** A line plot is created to visualize the relationship between age and hours worked per week.
3. **Bar Plots:** A bar plot is created to show the count of different education levels in the dataset.
4. **Histograms and Density Plots:** A histogram and density plot are created to visualize the distribution of ages in the dataset.

5. **Scatter Plots:** A scatter plot is created to observe the relationship between age and capital gain.