Lab07 Hofstede's Cultural Dimensions (2015)

Learning Goals

By the end of this lab, you will be able to:

1. Load and explore data using Pandas
2. Clean and prepare a dataset for analysis
3. Use visualizations (Matplotlib, Seaborn, Plotly) to explore cultural differences
4. Apply statistical summaries and similarity measures between countries
5. Create and use a Python class to structure your analysis

1. Setup

Install the required libraries (if not already installed):

```
!pip install pandas matplotlib seaborn plotly scikit-learn --quiet
```

Then, import them:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import plotly.graph_objects as go
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
```

2. Data Collection

Load Hofstede's 2015 cultural dimensions dataset directly from GitHub:

```
url = "https://raw.githubusercontent.com/masterfloss/data/refs/heads/main/culture2015.csv"
df = pd.read_csv(url, sep=";")
df.head()
```

3. Understanding the Data

Inspect the structure and summary statistics:

```
df.info()
df.describe()
```

Student Task

- Display missing values per column.

- Compute the mean and standard deviation for each dimension.

Example:

```
print("Missing values per column:")
print(df.isna().sum())
```

4. Data Cleaning

Clean column names and handle missing entries:

```
df.columns = df.columns.str.strip().str.lower()
df = df.dropna()
df.head()
```

Student Task

Verify that all dimension columns are numeric. If not, convert them.

```
for col in df.columns[1:]:
    df[col] = pd.to_numeric(df[col], errors='coerce')
df.info()
```

Set index and prepare a cleaned dataset:

```
df1 = df.drop(columns='country').dropna()
df1.set_index('ctr', inplace=True)
```

5. Data Exploration

Explore statistical summaries and correlations:

```
df1.describe().T

plt.figure(figsize=(8,6))
sns.heatmap(df1.corr(), annot=True, cmap="coolwarm")
plt.title("Correlation Matrix of Hofstede Dimensions")
plt.show()
```

Student Discussion

Which dimensions seem positively or negatively correlated?

6. Visualization of Cultural Differences

Example: Comparing Portugal and the USA

```
df1.reset_index(inplace=True)
country_subset = df1[df1['ctr'].isin(['POR', 'USA'])].set_index('ctr')
country_subset.T.plot(kind='bar', figsize=(10,6))
plt.title('Cultural Profile Comparison: Portugal vs USA')
plt.ylabel('Score')
plt.show()
```

Radar Chart Example

```
countries = ['POR', 'USA']
fig = go.Figure()

for c in countries:
    values = df1[df1['ctr'] == c][['pdi', 'idv', 'mas', 'uai', 'ltowvs', 'ivr']].iloc[0].values
    fig.add_trace(go.Scatterpolar(r=values, theta=['pdi', 'idv', 'mas', 'uai', 'ltowvs', 'ivr'],
fill='toself', name=c))

fig.update_layout(title='Cultural Profiles (Radar Chart)')
fig.show()
```

Scatter Matrix

```
fig = px.scatter_matrix(df1, dimensions=df1.columns[1:], color="ctr")
fig.update_layout(title="Scatter Matrix of Hofstede Cultural Dimensions")
fig.show()
```

Student Task

Create your own comparison chart for three countries of your choice.

7. Cultural Clustering (Optional Extension)

Cluster countries based on cultural similarity:

```
X = df1.drop(columns=['ctr'])
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

kmeans = KMeans(n_clusters=4, random_state=42, n_init=10)
df1['cluster'] = kmeans.fit_predict(X_scaled)

px.scatter_3d(df1, x='pdi', y='idv', z='mas', color='cluster',
hover_name='ctr',
              title="Cultural Clusters").show()
```

Student Discussion

- Which countries are culturally similar?
- Are there regional patterns?

8. Object-Oriented Programming: Cultural Analyzer Class

Build a reusable Python class for cultural data exploration:

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

class CultureAnalyzer:
    def __init__(self, url):
        self.df = pd.read_csv(url, sep=";")
        self.clean_data()
    def clean_data(self):
        self.df.columns = self.df.columns.str.strip().str.lower()
        self.df = self.df.dropna(how="all")
        for col in self.df.columns:
            if col != 'country':
                self.df[col] = pd.to_numeric(self.df[col], errors='coerce')
        self.df = self.df.dropna(subset=['country'])
        self.df = self.df.dropna(how="all",
subset=self.df.columns.difference(['country']))
        self.df['country'] = self.df['country'].str.strip()
    def summary(self):
        return self.df.describe()
    def correlation_heatmap(self):
        numeric_df = self.df.select_dtypes(include='number')
        if numeric_df.empty:
            print("No numeric data available for correlation heatmap.")
            return
        plt.figure(figsize=(8, 6))
        sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm',
fmt=".2f")
        plt.title('Cultural Dimension Correlations')
        plt.tight_layout()
        plt.show()
    def compare_countries(self, countries):
        subset =
self.df[self.df['country'].isin(countries)].set_index('country')
        subset = subset.select_dtypes(include='number')
        if subset.empty:
            print("No numeric data to compare for the selected countries.")
            return
        subset.T.plot(kind='bar', figsize=(10, 6))
        plt.title(f'Cultural Comparison: {" vs ".join(countries)}')
        plt.ylabel('Score')
        plt.tight_layout()
        plt.show()
```

Example Usage

```
analyzer =
CultureAnalyzer("https://raw.githubusercontent.com/masterfloss/data/refs/head
s/main/culture2015.csv")
print(analyzer.summary())
analyzer.correlation_heatmap()
analyzer.compare_countries(["Portugal", "United States", "Japan"])
```

Student Tasks

1. Add a `.radar_chart(countries)` method using Plotly to compare countries.
2. Add a `.cluster_countries(k)` method to perform K-Means clustering and visualize results.
3. Add a `.export_clean_data(filename)` method to save the cleaned dataset.

9. Reflection

Answer briefly:

1. Which cultural dimensions appear most strongly correlated?
2. Do countries form identifiable cultural clusters?
3. How could these metrics inform international business or policy decisions?

Summary of Skills Practiced

- Data loading and cleaning with Pandas
- Visual exploration with Matplotlib, Seaborn, and Plotly
- Application of K-Means clustering
- Object-oriented design for data analysis
- Interpretation of cultural data patterns