Exercise 3.2: Tree Maps, Area and Stacked Area Charts

Scott Breitbach

DSC640 - 01/13/2022

Plots Using **Python**

Load Data

In [1]:
```
# Load libraries
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
# Load data
unempDF = pd.read_csv('unemployement-rate-1948-2010.csv')
expenDF = pd.read_csv('expenditures.csv')
```

Tree Map

In [3]:
```
# Load libraries
import squarify
```

In [4]:
```
# Group expenses by category
expenCat = expenDF.groupby('category').sum()

# Set up values for chart
size = expenCat.expenditure
labels = expenCat.index
```

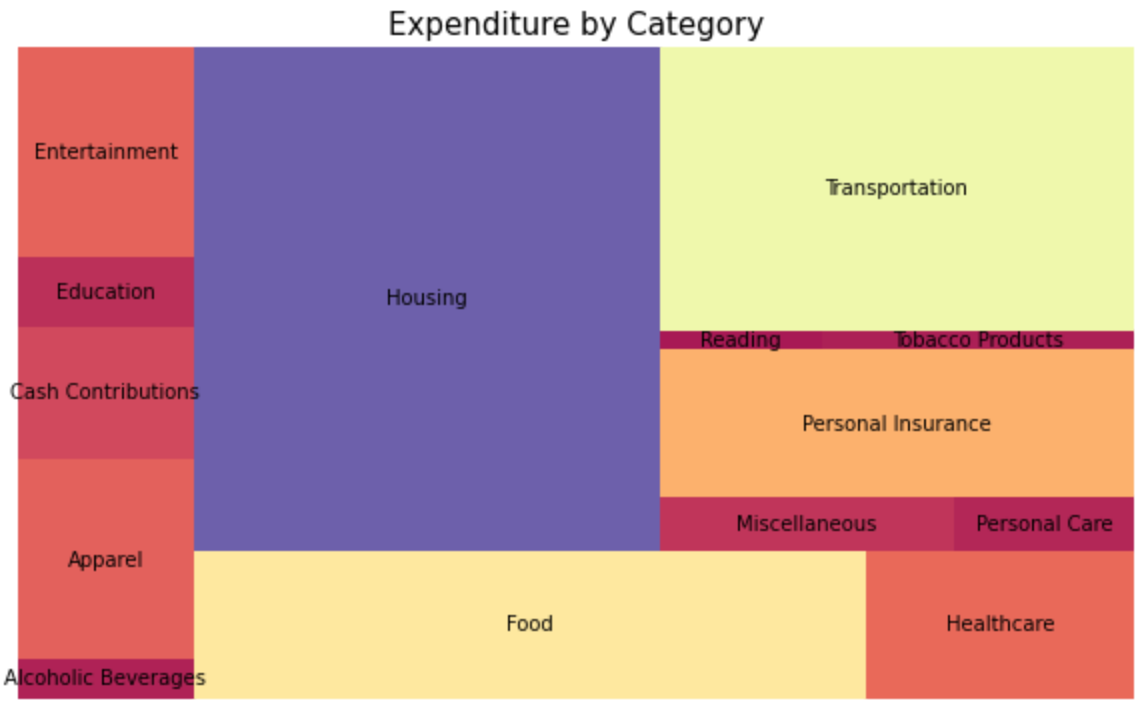
In [5]:
```
# Set up color palette, mapped to values
cmap = matplotlib.cm.Spectral # Blues is pretty good too
mini = min(size)
maxi = max(size)
norm = matplotlib.colors.Normalize(vmin=mini, vmax=maxi)
colors = [cmap(norm(value)) for value in size]
```

In [6]:
```
# Initialize the matplotlib figure
f, ax = plt.subplots(figsize=(10, 6))

# Plot tree map using squarify and matplotlib
squarify.plot(sizes = size,
              label = labels,
              alpha = 0.9,
              color = colors)
plt.axis('off')

# Add chart title
plt.title("Expenditure by Category", fontsize = 15)

plt.show()
```



Area Chart

In [7]:
```
# Convert Month numbers to Month names
monthDict = {"M01": "Jan", "M02": "Feb", "M03": "Mar",
             "M04": "Apr", "M05": "May", "M06": "Jun",
             "M07": "Jul", "M08": "Aug", "M09": "Sep",
             "M10": "Oct", "M11": "Nov", "M12": "Dec"}
unempDF.replace({"Period": monthDict}, inplace=True)

# Combine Month and Year columns to a single datetime column
unempDF["MoYr"] = pd.to_datetime(unempDF.Period + ' ' + unempDF.Year.astype(str))
```

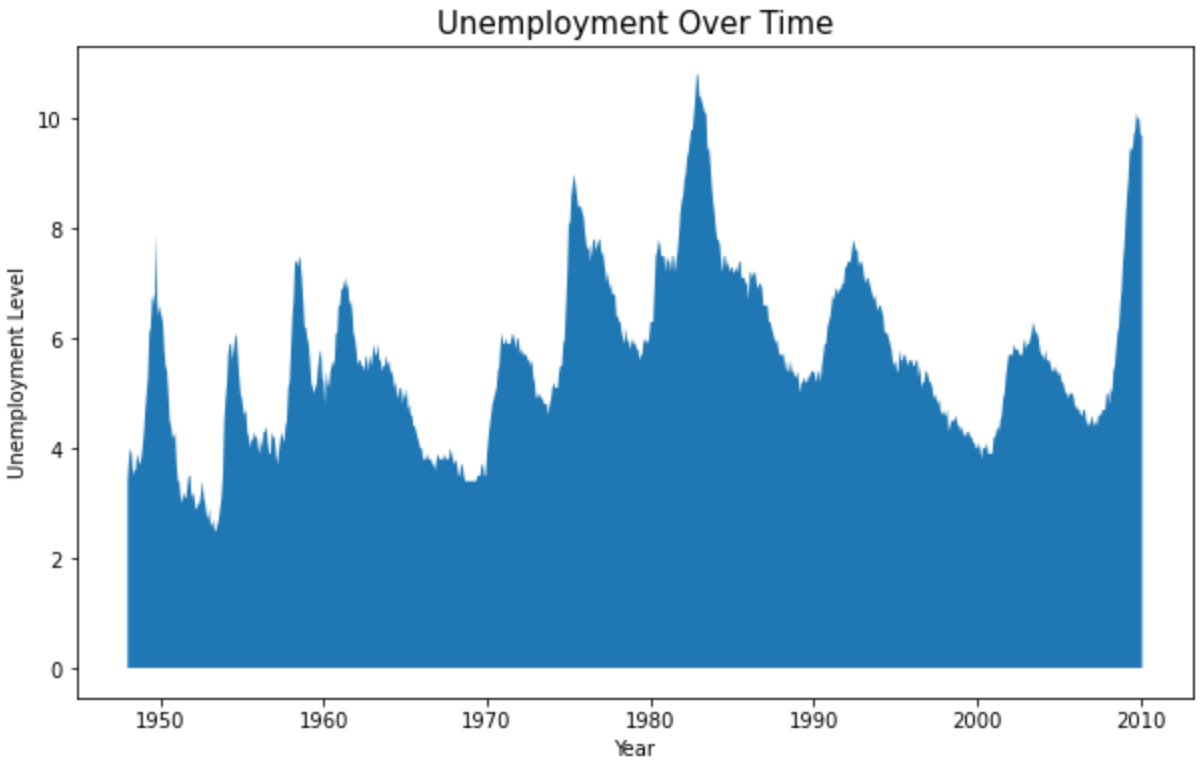
In [8]:
```
# Select data
x = unempDF.MoYr
y = unempDF.Value
```

In [9]:
```
# Initialize the matplotlib figure
f, ax = plt.subplots(figsize=(10, 6))

# Plot Area Chart
plt.fill_between(x, y)

# Add chart title and labels
plt.title("Unemployment Over Time", fontsize = 15)
plt.xlabel("Year")
plt.ylabel("Unemployment Level")

plt.show()
```



Stacked Area Chart

In [10]:
```
# Pivot dataframe into category expenditures by year
pivotDF = expenDF.pivot(index='year', columns='category')['expenditure']
```

In [11]:
```
# Get list of top 5 categories
sortedAvg = pivotDF.mean().sort_values(ascending=False)
topFiveCat = sortedAvg[:5].index.tolist()
```

In [12]:
```
# Select Data for chart
x = pivotDF.index
y0 = pivotDF[topFiveCat[0]]
y1 = pivotDF[topFiveCat[1]]
y2 = pivotDF[topFiveCat[2]]
y3 = pivotDF[topFiveCat[3]]
y4 = pivotDF[topFiveCat[4]]
```

In [13]:
```
# Initialize the matplotlib figure
f, ax = plt.subplots(figsize=(10, 6))

# Plot Stacked Area Chart
plt.stackplot(x, y0, y1, y2, y3, y4, labels=topFiveCat)
plt.legend(loc='upper left')

# Add chart title and labels
plt.title("Top 5 Expenditure Categories Over Time", fontsize = 15)
plt.xlabel("Year")
plt.ylabel("Expenditure ($USD)")

plt.show()
```