DSO 545: Statistical Computing and Data Visualization

Abbass Al Sharif

Fall 2019

Lab 9: Data Visualization Using Matplotlib (Part3)

```
#import necessary packages
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings;
```

```
#supress warnings
warnings.filterwarnings(action='once')
```

```
# Set Parameters for plots in matplotlib

large = 22; med = 16; small = 12

params = {"figure.figsize": (16,10),
          "figure.dpi": 200,
          "figure.titlesize": large,
          "axes.titlesize": med,
          "axes.labelsize": med,
          "legend.fontsize":med,
          "xtick.labelsize": med,
          "ytick.labelsize":med
          }

plt.rcParams.update(params)
plt.style.use("seaborn-whitegrid")
sns.set_style("white")
```

Density Plots

1. Create the following density plot to describe how the distribution of city mileage varies with respect the number of cylinders.

```
import seaborn as sns

data = pd.read_csv('mpg.csv')

plt.figure()

sns.kdeplot(data.loc[data.cyl == 4, 'cty'],
            shade = True,
            color = 'green',
            label = "Cyl = 4",
            alpha = 0.4)
```

```
sns.kdeplot(data.loc[data.cyl == 5, 'cty'],
            shade = True,
            color = 'deeppink',
            label = "Cyl = 5",
            alpha = 0.4)
```

```
## <matplotlib.axes._subplots.AxesSubplot object at 0x11f55dc90>
```

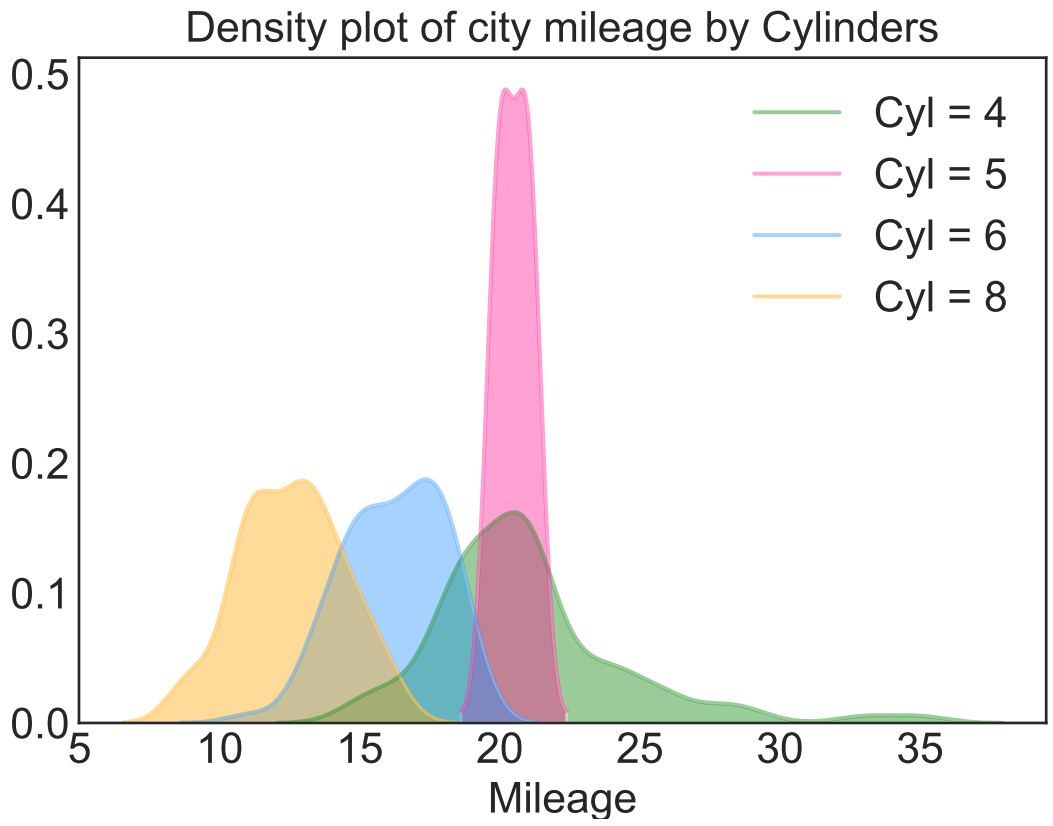
```
sns.kdeplot(data.loc[data.cyl == 6, 'cty'],
            shade = True,
            color = 'dodgerblue',
            label = "Cyl = 6",
            alpha = 0.4)
```

```
## <matplotlib.axes._subplots.AxesSubplot object at 0x11f55dc90>
```

```
sns.kdeplot(data.loc[data.cyl == 8, 'cty'],
            shade = True,
            color = 'orange',
            label = "Cyl = 8",
            alpha = 0.4)
```

```
## <matplotlib.axes._subplots.AxesSubplot object at 0x11f55dc90>
```

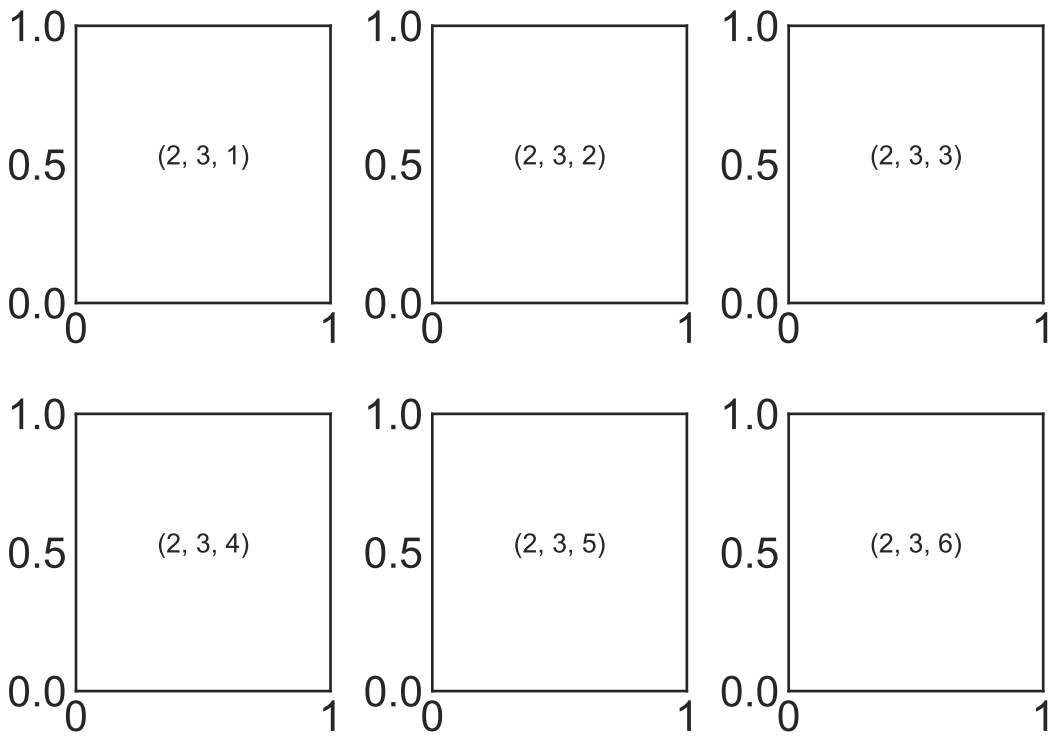
```
plt.xlabel('Mileage')
plt.title("Density plot of city mileage by Cylinders")
plt.show()
```



Multiple Subplots

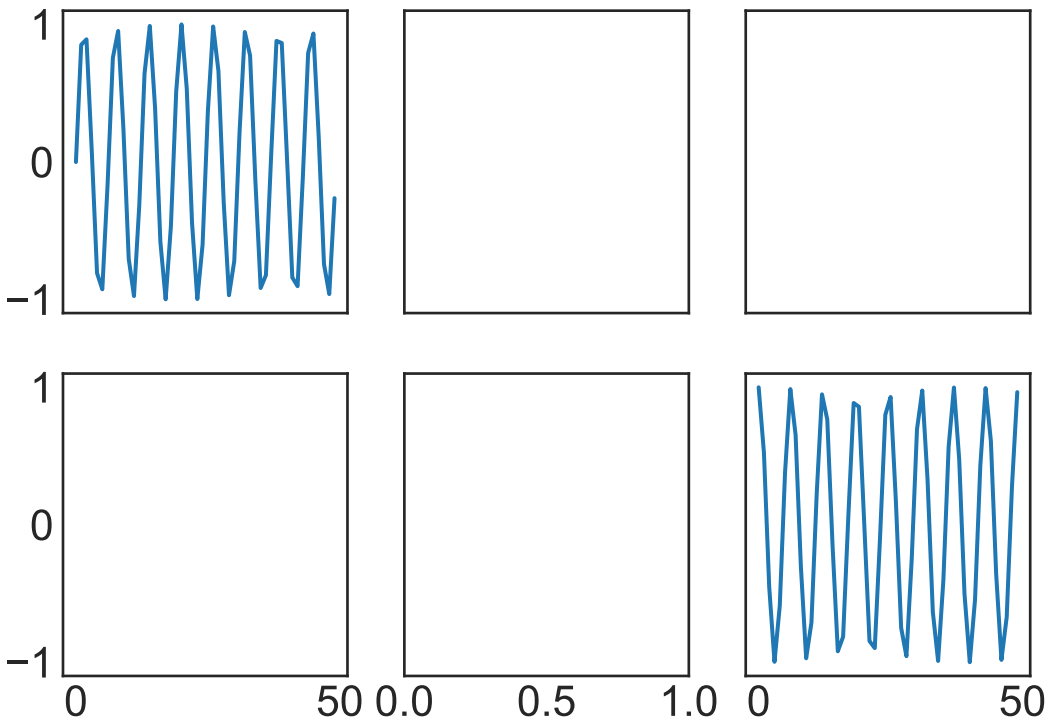
2. Create a 2x3 grid of plots as follows:

```
plt.figure()
plt.subplots_adjust(hspace = 0.4, wspace = 0.4)

for i in range(1,7):
  plt.subplot(2,3,i)
  plt.text(0.5,0.5, str((2,3,i)), ha = 'center')

plt.show()
```



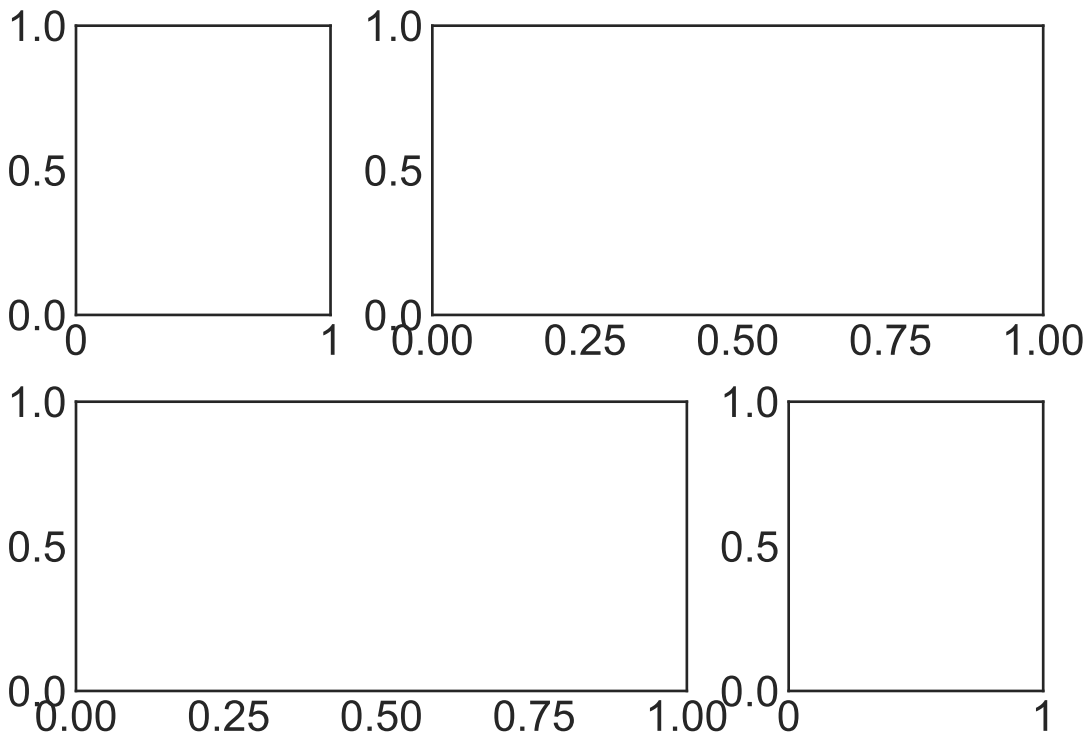
3. Create a 2x3 grid of plots as follows:

```
plt.figure()

fig, ax = plt.subplots(2,3, sharex = "col", sharey= "row")

x = np.linspace(0, 50)
ax[0,0].plot(np.sin(x))
ax[1,2].plot(np.cos(x))

plt.show()
```



4. Create the following plot:

```
fig = plt.figure()

grid = plt.GridSpec(2,3,wspace = 0.4, hspace = 0.3)

fig.add_subplot(grid[0,0])
fig.add_subplot(grid[0,1:3])
fig.add_subplot(grid[1,0:2])
fig.add_subplot(grid[1,2])
plt.show()
```



Marginal Boxplots

5. Use the `mpg.csv` dataset to create the following plot that shows the relationship between the variables `displ` and `hwy`. In addition, it shows the distribution of both using boxplots.

```
df = pd.read_csv("mpg.csv")

fig = plt.figure()
grid = plt.GridSpec(4,4,wspace = 0.5, hspace = 0.5)

#define axes

ax_main = fig.add_subplot(grid[:-1,:-1])
ax_right = fig.add_subplot(grid[:-1, -1], xticklabels = [], yticklabels = [])
ax_bottom = fig.add_subplot(grid[-1, :-1], xticklabels = [], yticklabels = [])

#scatter plot (main plot)
ax_main.scatter('displ', 'hwy', data = df)

#right boxplot
sns.boxplot(df.hwy, ax = ax_right, orient="v")
ax_right.set(ylabel = "")
ax_right.set(xticks = [], yticks = [])
```

```
#bottom boxplot
```

```
## [[], []]
```

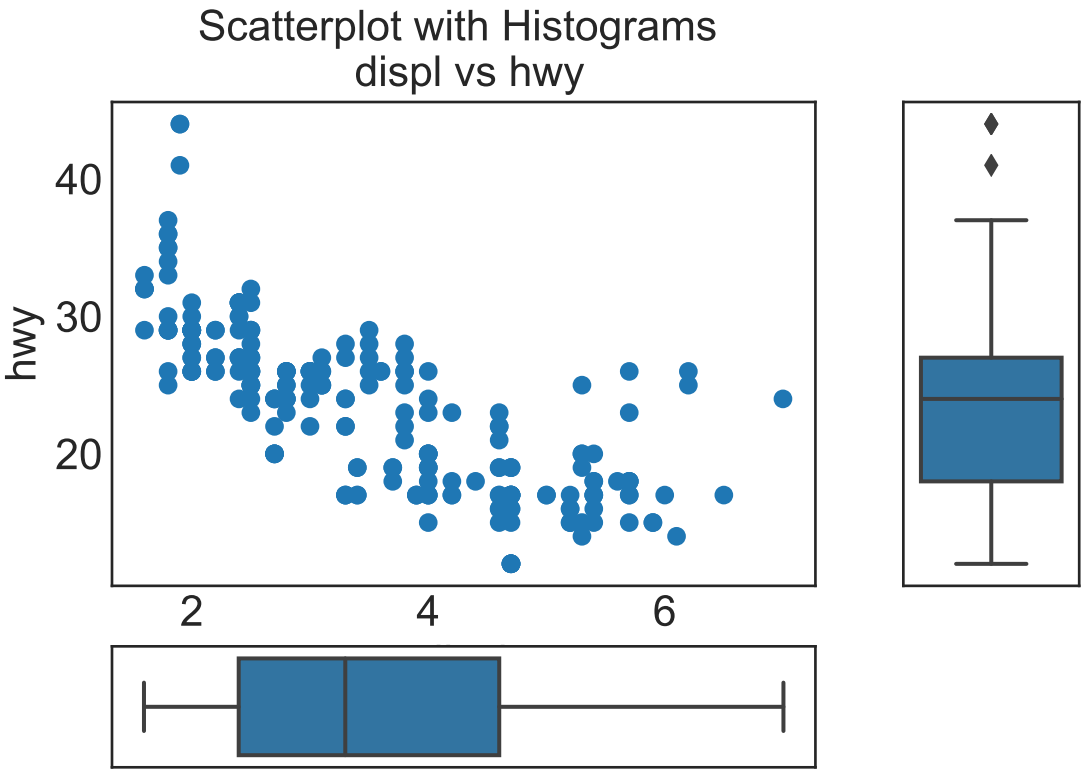
```
sns.boxplot(df.displ, ax = ax_bottom, orient="h")
ax_bottom.set(xlabel = "")
ax_bottom.set(xticks = [], yticks = [])

# Main Title, Xlabel and YLabel
```

```
## [[], []]
```

```
ax_main.set(title='Scatterplot with Histograms \n displ vs hwy',
            xlabel='displ', ylabel='hwy')
```

```
## [Text(0, 0.5, 'hwy'), Text(0.5, 0, 'displ'), Text(0.5, 1.0, 'Scatterplot with Histograms \n displ vs
```

```
plt.show()
```



Plotting with different scales using secondary Y axis

6. Use the `economics.csv` dataset to create the following plot which shows the relationship between personal savings rate and # of unemployed from 1967 to 2012.

```
data = pd.read_csv("economics.csv")
data.head()
```

```
##          date    pce     pop  psavert  uempmed  unemploy
## 0  1967-07-01  507.4  198712     12.5      4.5      2944
## 1  1967-08-01  510.5  198911     12.5      4.7      2945
## 2  1967-09-01  516.3  199113     11.7      4.6      2958
## 3  1967-10-01  512.9  199311     12.5      4.9      3143
## 4  1967-11-01  518.1  199498     12.5      4.7      3066
```

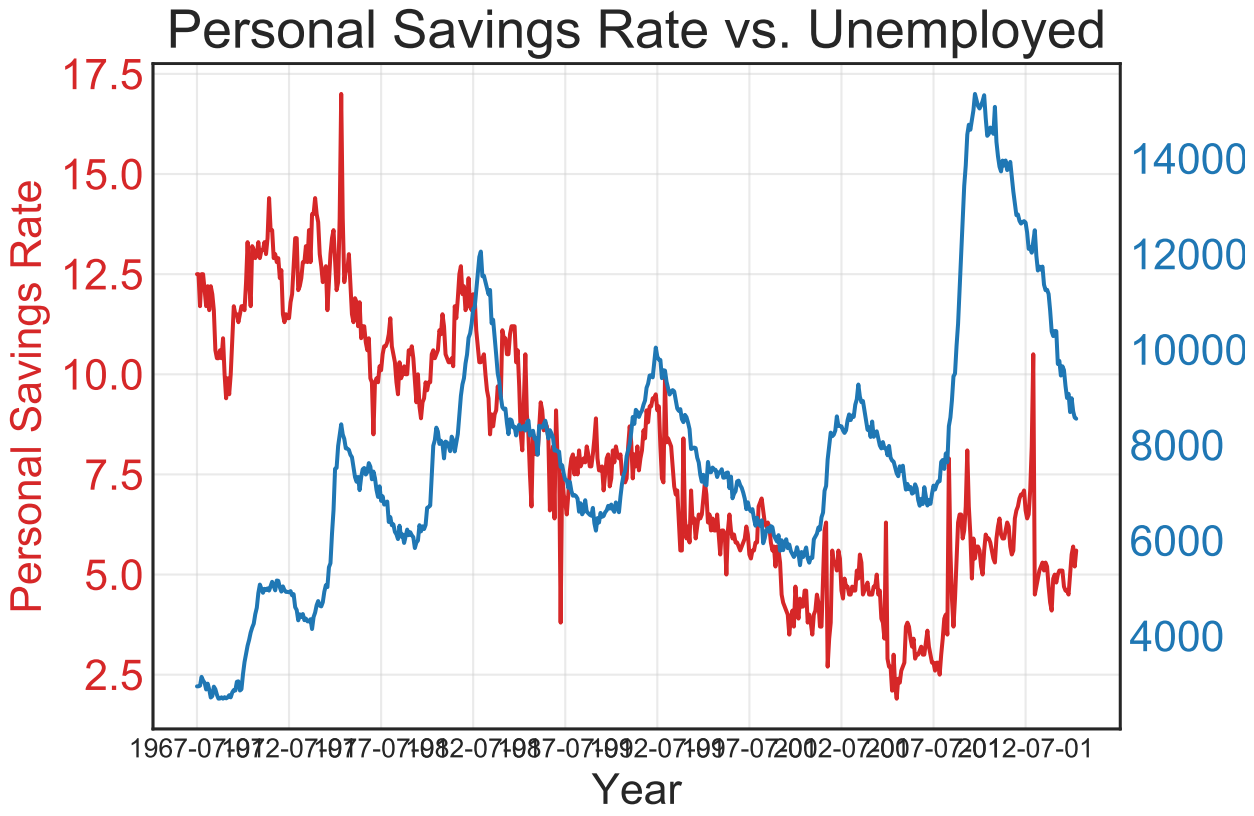
```
fig = plt.figure()

plt.plot(data.date, data.psavert, color = "tab:red")

ax = plt.gca()

#set the main y axis (left)
ax.tick_params(axis = 'y', labelcolor = "tab:red")
ax.set_ylabel("Personal Savings Rate", color = "tab:red", fontsize = 16)
ax.grid(alpha = 0.4)

#set the secondary y axis (right)

ax_right = ax.twinx()
ax_right.plot(data.date, data.unemploy, color = 'tab:blue')
ax_right.tick_params(axis = 'y', labelcolor = "tab:blue")
ax_right.set_ylabel("# Unemployed (1000's)", color = "tab:blue", fontsize = 16)

#set x-axis
ax.set_xlabel("Year", fontsize = 16)
ax.set_xticks(np.arange(0, len(data.date), 60))
```

```
## [<matplotlib.axis.XTick object at 0x122cab690>, <matplotlib.axis.XTick object at 0x122cdb450>, <matpl
```

```
ax.set_xticklabels(data.date[::60], rotation = 0, fontsize = 10)

#set title
```

```
## [Text(0, 0, '1967-07-01'), Text(0, 0, '1972-07-01'), Text(0, 0, '1977-07-01'), Text(0, 0, '1982-07-0
```

```
plt.title("Personal Savings Rate vs. Unemployed", fontsize = 20)

plt.show()
```


Personal Savings Rate vs. Unemployed
Personal Savings Rate
17.5
15.0
12.5
10.0
7.5
5.0
2.5
14000
12000
10000
8000
6000
4000
1967-07-01
1972-07-01
1977-07-01
1982-07-01
1987-07-01
1992-07-01
1997-07-01
2002-07-01
2007-07-01
2012-07-01
Year