***Data Plotting Using matplotlib.pyplot as plt***

*Using plot container (Single Plot):*

First-Way(Empty Plot):

|  |
| --- |
| import matplotlib.pyplot as plt  plt.plot()  plt.show() |

Second-Way (Multiple Values, and adding xticks, Title, xlabel, ylabel):

|  |
| --- |
| plt.plot(x\_values, y\_values)  plt.xticks(rotation=90)  plt.xlabel('Month')  plt.ylabel('Unemployment Rate')  plt.title('Monthly Unemployment Trends, 1948') |

Third-Way (With figsize, color, for loop, label and legend) (Single Plot):

|  |
| --- |
| import matplotlib.pyplot as plt  fig = plt.figure(figsize=(12,12))    for i in range(5):  start\_index = i\*12 # For starting index of the columns to be plotted  end\_index = (i+1)\*12 # For starting index of the columns to be plotted  subset = unrate[start\_index:end\_index] # Here unrate is a Dataframe and  plt.plot(subset['DATE'], subset['VALUE'], c=colors[i]) # DATE & VALUE are it’s columns  label = str(1948 + i) # Adds label for line in the plot    plt.legend(loc='upper left') # At this locations  plt.show() |

*Using axes container (Multiple Plots):*

First-Way:

|  |
| --- |
| import matplotlib.pyplot as plt  fig = plt.figure()  x\_values = [0.0, 0.5, 1.0]  y\_values = [10, 20, 40]  ax1 = fig.add\_subplot(2, 1, 1)  ax1.plot(x\_values, y\_values)  ax2 = fig.add\_subplot(2, 1, 2)  ax2.plot([0], [0])  plt.show() |

Second-Way (With figsize, color and using for loop) (Multiple Plots):

|  |
| --- |
| import matplotlib.pyplot as plt  fig = plt.figure(figsize=(12,12))  colors = ['red', 'blue', 'green', 'orange', 'black']    for i in range(5):  ax = fig.add\_subplot(5,1,i+1) # For positioning Axes Objects  start\_index = i\*12 # For starting index of the columns to be plotted  end\_index = (i+1)\*12 # For starting index of the columns to be plotted  subset = unrate[start\_index:end\_index] # Here unrate is a Dataframe and  ax.plot(subset['DATE'], subset['VALUE'], c=colors[i]) # DATE & VALUE are it’s columns    plt.show() |

Adding More Options (Labels, Legend, Title, xlabel, ylabel) (Multiple Plots):

|  |
| --- |
| import matplotlib.pyplot as plt  fig = plt.figure(figsize=(12,12))  colors = ['red', 'blue', 'green', 'orange', 'black']    for i in range(5):  ax = fig.add\_subplot(5,1,i+1) # For positioning Axes Objects  start\_index = i\*12 # For starting index of the columns to be plotted  end\_index = (i+1)\*12 # For starting index of the columns to be plotted  subset = unrate[start\_index:end\_index] # Here unrate is a Dataframe and  ax.plot(subset['DATE'], subset['VALUE'], c=colors[i]) # DATE & VALUE are it’s columns  ax.legend([str(1948 + i)], loc='upper left')  ax.set\_xlabel('Year - ' + unrate.loc[i\*12, 'DATE'].strftime("%Y")) # Choose either xlabel or title  ax.set\_ylabel("Unemployment Rate, Percent")  ax.set\_title(“Monthly Unemployment Trends, “ + unrate.loc[i\*12, 'DATE'].strftime("%Y") )    plt.show() |

***Data Plotting for BarPlots, ScatterPlots, Histograms and BoxPlots***

*BarPlots:*

First-Way:

|  |
| --- |
| fig, ax = plt.subplots() # Quick way of creating a plot with single axes object    from numpy import arange # arange gives array of numbers in a sequence and given range  bar\_positions = arange(5) + 0.75 # left starting positions  num\_cols = ['RT\_user\_norm', 'Metacritic\_user\_nom', 'IMDB\_norm', 'Fandango\_Ratingvalue', 'Fandango\_Stars'] # xticklabels    bar\_heights = norm\_reviews[num\_cols].iloc[0].values # height    ax.bar(bar\_positions, bar\_heights)  # ax.bar(left position, height, width)  # Default width is 0.8  plt.show() |

Adding More Options (xlabel, ylabel, xticks, xticklabels, title):

|  |
| --- |
| fig, ax = plt.subplots()    bar\_position = arange(5) + 0.75  bar\_height = norm\_reviews[num\_cols].iloc[0].values    ax.bar(bar\_position, bar\_height, 0.5)  tick\_positions = range(1, 6)  ax.set\_xticks(tick\_positions) # Setting tick positions at x-axis    ax.set\_xticklabels(num\_cols, rotation=90) # Giving names/labels to the x-axis ticks    ax.set\_xlabel("Rating Source") # X-axis label  ax.set\_ylabel("Average Rating") # Y-axis label  ax.set\_title("Average User Rating For Avengers: Age of Ultron (2015)") # Title of the axes Object  plt.show()  ----------------------------------------------------------------------------------------------------------------------------------------  # Incase of horizontal Bar Plot:  # fig, ax = plt.subplots()    # bar\_position = arange(5) + 0.75  # bar\_height = norm\_reviews[num\_cols].iloc[0].values    # ax.barh(bar\_position, bar\_height, 0.5, align="edge")    # ax.set\_yticklabels(num\_cols)  # ax.set\_yticks(range(1, 6))    # plt.ylabel("Rating Source")  # plt.xlabel("Average Rating")    # plt.title("Average User Rating For Avengers: Age of Ultron (2015)")  # plt.show() |

*ScatterPlots:*

First-Way:

|  |
| --- |
| fig, ax = plt.subplots()    ax.scatter(norm\_reviews['Fandango\_Ratingvalue'],  norm\_reviews['RT\_user\_norm'])    ax.set\_xlabel("Fandango")  ax.set\_ylabel("Rotten Tomatoes")  ax.set\_xlim(0, 5)  ax.set\_ylim(0, 5)  ax.set\_title('f vs. r') # Use Either xlabel or title in case of multiple plots    plt.title("Fandango VS Rotten Tomatoes") # Title to be set at Either axes level or plot level  plt.show() |

*Histograms:*

First-Way:

|  |
| --- |
| fig, ax = plt.subplots()    ax.hist(norm\_reviews['Fandango\_Ratingvalue'], bins=20, range=(0,5))  ax.set\_ylim(0,50)  plt.title("Fandango\_Ratingvalue Histogram")  plt.show() |

*BoxPlots:*

First-Way:

|  |
| --- |
| fig, ax = plt.subplots()    ax.boxplot(norm\_reviews['RT\_user\_norm'])  ax.set\_ylim(0, 5)  ax.set\_xticklabels(['Rotten Tomatoes'])  plt.show() |

Second-Way(Multiple Plots):

|  |
| --- |
| fig, ax = plt.subplots()    num\_cols = ['RT\_user\_norm', 'Metacritic\_user\_nom', 'IMDB\_norm', 'Fandango\_Ratingvalue', 'Fandango\_Stars']  ax.boxplot(norm\_reviews[num\_cols].values)  ax.set\_ylim(0, 5)  ax.set\_xticklabels(num\_cols, rotation=90)  plt.show() |

***Data Plotting Using Pandas (Additional to others: Scatter Matrix)***

First-Way

|  |
| --- |
| df.plot(x='Sample\_size', y='Employed', kind='scatter')  # More Options  df.plot(x='Sample\_size', y='Employed', kind='scatter', title='Employed vs. Sample\_size', figsize=(5,10)) |

Second-Way (Using Axes Object Container)

|  |
| --- |
| ax = df.plot(x='Sample\_size', y='Employed', kind='scatter')  ax.set\_title('Employed vs. Sample\_size') |

For Histograms

|  |
| --- |
| df['Sample\_size'].plot(kind='hist')  *# With Bins:*  df['Sample\_size'].hist(bins=25, range=(0,5000)) |

For Scattter Matrix:

|  |
| --- |
| scatter\_matrix(df[['Women', 'Men']], figsize=(10,10)) |

For Bars:

|  |
| --- |
| df[:5]['Women'].plot(kind='bar')  # If X-Labels needs to be specified or else index is used  df[:5].plot.bar(x='Major', y='Women') |