# Matplotlib Notes

---

**Pyplot Interface**

import matplotlib.pyplot as plt

fig, ax = plt.subplots( )

plt.show( )

---

**Adding Data to Axes**

ax.plot(< DATAFRAME >[“< COLUMN >”] , < DATAFRAME >[“< COLUMN >”]

x

y

---

**Adding Markers**

ax.plot(< DATAFRAME >[“< COLUMN >”] , < DATAFRAME >[“< COLUMN >”] , marker = “o”)

---

**Setting Linestyle**

ax.plot(< DATAFRAME >[“< COLUMN >”] , < DATAFRAME >[“< COLUMN >”] , linestyle = “- -”)

---

**Eliminating Liines with Linestyle**

ax.plot(< DATAFRAME >[“< COLUMN >”] , < DATAFRAME >[“< COLUMN >”] , linestyle =“None”)

---

**Choosing Colour**

ax.plot(< DATAFRAME >[“< COLUMN >”] , < DATAFRAME >[“< COLUMN >”] , color= = “r”)

---

**Customizing Axes Labels**

ax.set_xlabel(“< LABEL >)

ax.set_ylabel(“< LABEL >)

---

**Adding Title**

ax.set_title(“< TITLE >)

---

**Small Multiples with plt.subplots**

fig, ax = plt.subplots( <#> , <#> )

Row

Column

**Sharing the y-Axis range**

fig, ax = plt.subplots( <#> ,  <#> , sharey=True)

**DateTimeIndex**

< DATAFRAME >.index

**Plotting Time-Series Data**

ax.plot(< DATAFRAME >.index , < DATAFRAME >[“< COLUMN >”])

**Plotting Two Time-Series Together**

< DATAFRAME > = pd.read_csv(“< DATAFRAME >.csv”,

parse_dates = [“date”],

index_col = “date)

**Using Twin Axes**

ax.plot(< DATAFRAME >.index , < DATAFRAME >[“< COLUMN 1 >”])

Ax2 = ax.twinx( )

ax2.plot(< DATAFRAME >.index , < DATAFRAME >[“< COLUMN 2 >”])

**Separating Variables by Color**

ax.plot(< DATAFRAME >.index , < DATAFRAME >[“< COLUMN 1 >”] , color= “blue”)

ax.set_xlabel(“< LABEL >)

ax.set_ylabel(“< LABEL > , color= “blue”)

Ax2 = ax.twinx( )

ax2.plot(< DATAFRAME >.index , < DATAFRAME >[“< COLUMN 2 >”] , color =“red”)

ax2.set_ylabel(“< LABEL > , color= “red”)

**Colouring the Ticks**

ax.tick_parmas(“y”, color=“blue”)

ax.tick_params(“y”, color= “red”)

**A Function That Plots Time-Series**

Def plot_timeseries(axes, x, y, color, xlabel, ylabel)

**Using Function**

plot_timeseries(ax, < DATAFRAME >.index, < DATAFRAME >[<COLUMN>], “blue”,

< LABEL >, < LABEL >)

**Bar Chart**

ax.bar(< DATAFRAME >.index , < DATAFRAME >[“< COLUMN >”])

**Rotate Charts**

ax.set_xticklabels (< DATAFRAME >.index, rotation= 90)

**Stacked Bar Chart (2)**

ax.bar(< DATAFRAME >.index , < DATAFRAME >[“< COLUMN 1 >”])

ax.bar(< DATAFRAME >.index , < DATAFRAME >[“< COLUMN 2 >”], bottom=

< DATAFRAME >[“< COLUMN 1 >”])

**Stacked Bar Chart (3+)**

ax.bar(< DATAFRAME >.index , < DATAFRAME >[“< COLUMN 3 >”], bottom=

< DATAFRAME >[“< COLUMN 1 >”] + < DATAFRAME >[“< COLUMN 2 >”])

**Adding a Legend**

ax.legend( )

**Histogram**

ax.hist(< DATAFRAME >[“< COLUMN >”])

**Histogram Labels**

ax.hist(< DATAFRAME >[“< COLUMN >”], label= “< LABEL >”)

**Setting Number of Bins**

ax.hist(< DATAFRAME >[“< COLUMN >”], label= “< LABEL >”, bin= < # >)

**Setting Bin Boundaries**

ax.hist(< DATAFRAME >[“< COLUMN >”], label= “< LABEL >”, bin=[150, 160, 170, 180, 190, 200, 210])

**Setting Transparency**

ax.hist(< DATAFRAME >[“< COLUMN >”], label= “< LABEL >”, histtype = “step”)

**Adding Error Bars to Bar Charts**

ax.bar(“< LABEL >”, < DATAFRAME >[“< COLUMN >”].mean( ),

yerr= < DATAFRAME >[“< COLUMN >”].std( ))

**Adding Error Bars to Plots**

ax.errorbar(< DATAFRAME >[“< MEAN COLUMN >”],

yerr= < DATAFRAME >[“< STD COLUMN >”])

**Boxplots**

ax.boxplot(< DATAFRAME >[“< COLUMN >”])

**Scatterplots**

ax.scatter(< DATAFRAME >[“< COLUMN 1>”],< DATAFRAME >[“< COLUMN 2 >”])

**Choosing a Style**

plt.style.use(< STYLE >)

Back to default >>> plt.style.use(default)

**Saving Figure to File**

fig.savefig(“< FIG.NAME >.png”)

fig.savefig(“< FIG.NAME >.jpg”)

fig.savefig(“< FIG.NAME >.svg”)

**Figure Resolution**

fig.savefig(“< FIG.NAME >.png”, dpi= 300)

**Figure Size**

fig.set_size_inches([< # > ,< # >])

Width

Height