**#Area Graph**

import datetime

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

turnover = [2, 7, 14, 17, 20, 27, 30, 38, 25, 18, 6, 1]

plt.fill\_between(np.arange(12), turnover,

                 color="skyblue", alpha=0.4)

plt.plot(np.arange(12), turnover, color="Slateblue",

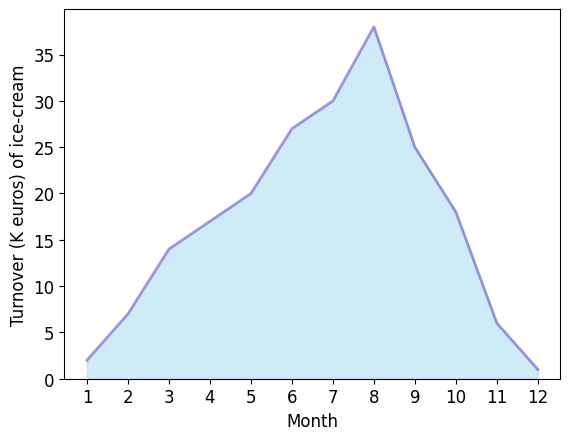
         alpha=0.6, linewidth=2)

plt.tick\_params(labelsize=12)

plt.xticks(np.arange(12), np.arange(1,13))

plt.xlabel('Month', size=12)

plt.ylabel('Turnover (K euros) of ice-cream', size=12)

plt.ylim(bottom=0)

plt.show()

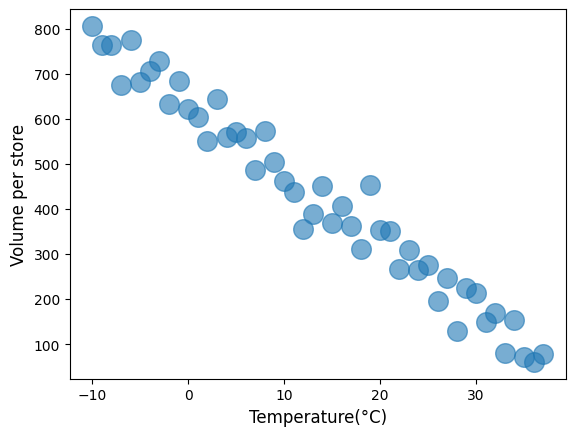
**#Scatter plots**

import numpy as np

import matplotlib.pyplot as plt

plt.scatter(x=range(-10, 38, 1),

y=range(770, 60, -15)-np.random.randn(48)\*40,

            s=200,

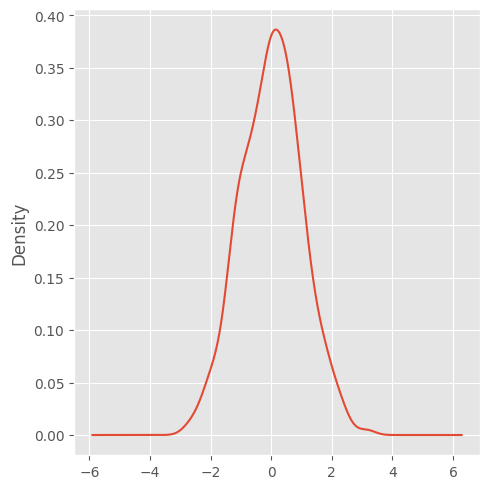
            alpha=0.6)

plt.xlabel('Temperature(°C)', size=12)

plt.ylabel('Volume per store', size=12)

plt.show()

**#Kernel Density Estimate**

import sys

import os

import pandas as pd

import matplotlib as ml

import numpy as np

from matplotlib import pyplot as plt

ml.style.use('ggplot')

figl=plt.figure(figsize=(5, 5))

ser = pd.Series (np.random.randn(1000))

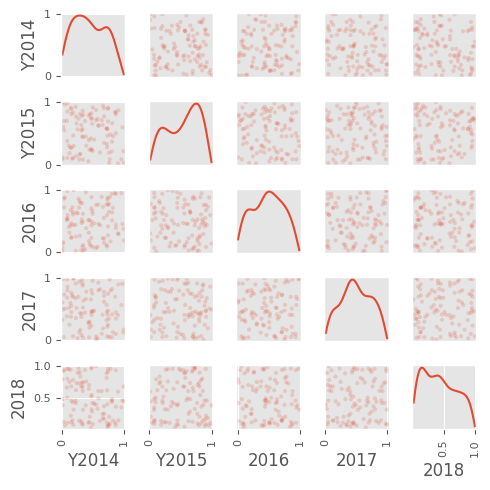
ser.plot(figsize=(5, 5), kind='kde')

sPicNameOut1='/content/kde.png'

plt.savefig(sPicNameOut1,dpi=600)

plt.tight\_layout()

plt.show()

**#Scatter Plot Matrix**

import sys

import os

import pandas as pd

import matplotlib as ml

import numpy as np

from matplotlib import pyplot as plt

fig2=plt.figure(figsize=(5, 5))

from pandas.plotting import scatter\_matrix

df = pd.DataFrame(np.random.rand(100, 5),

columns=['Y2014', 'Y2015','2016','2017','2018'])

scatter\_matrix(df, alpha=0.2,

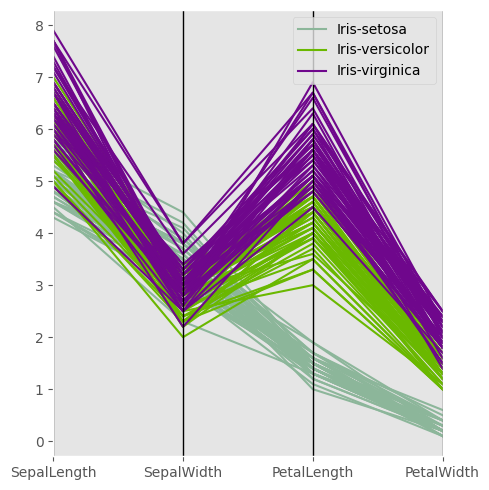
figsize=(5,5), diagonal='kde')

sPicNameOut2='/content/scatter\_matrix.png'

plt.savefig(sPicNameOut2, dpi=600)

plt.tight\_layout()

plt.show()

**#Parallel Coordinates**

import sys

import os

import pandas as pd

import matplotlib as ml

import numpy as np

from matplotlib import pyplot as plt

from pandas.plotting import parallel\_coordinates

plt.figure(figsize=(5,5))

sDataFile='/content/iris1.csv'

data = pd.read\_csv(sDataFile)

parallel\_coordinates (data, 'Name')

sPicNameOut2='/content/parallel\_coordinates.png'

plt.savefig(spicNameOut2, dpi=600)

plt.tight\_layout()

plt.show()

**#Autocorrelation Plot**

import sys

import os

import pandas as pd

import matplotlib as ml

import numpy as np

from matplotlib import pyplot as plt

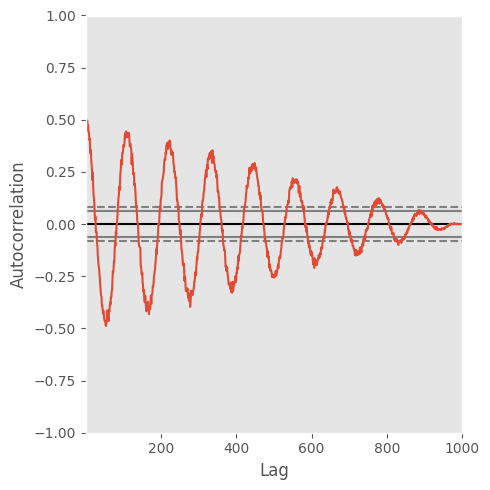
from pandas.plotting import autocorrelation\_plot

plt.figure(figsize=(5,5))

data = pd.Series (0.7\* np.random.rand(1000) + \

0.3\* np.sin(np.linspace(-9\* np.pi, 9 \* np.pi, num=1000)))

autocorrelation\_plot(data)

****spicNameOut2='/content/autocorrelation\_plot.png'

plt.savefig(sPicNameOut2,dpi=600)

plt.tight\_layout()

plt.show()