Python Codes for Statistical Analysis

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

ds = pd.read\_excel (r'C:\Users\estru\OneDrive\Documents\BB ICP-MS Results Analysis.xlsx')

fig, ax = plt.subplots()

nds = ds.fillna(0)

sns.set(font\_scale=1.2)

sns.set\_style("whitegrid")

sns.barplot(x='Method',y='Yield', data=ds, capsize = .05, hue='Loading', errwidth = 2, palette = "Blues").set(xlabel = None)

handles, labels = ax.get\_legend\_handles\_labels()

ax.legend(handles=handles[0:], labels=['1%', '5%', '10%'])

ax.set(ylabel = 'REE Yield %')

plt.show()

fig, ax = plt.subplots()

nds = ds.fillna(0)

sns.set(font\_scale=1.2)

sns.barplot(x='Method',y='Rate', data=ds, capsize = .05, hue='Loading', errwidth = 2, palette = "Reds").set(xlabel = None)

handles, labels = ax.get\_legend\_handles\_labels()

ax.legend(handles=handles[0:], labels=['1%', '5%', '10%'])

ax.set(ylabel = 'Leaching Rate (ug/hr)')

plt.show()

sns.set(font\_scale=1.2)

cords= nds.drop(columns=['Sample', 'Time','La','Ce','Nd','Rate'])

cor=cords.corr()

plt.rcParams["figure.figsize"] = (8,5.5)

sns.heatmap(cor, annot=True, cmap="Greens", fmt='.2f')

fig, ax = plt.subplots(figsize =(6, 3))

cords= ds.drop(columns=['Sample', 'Time', 'Yield', 'Nd', 'Ce', 'Method','Rate'])

cor=cords.corr()

cols = ['Loading','Temperature']

a = abs(cor['La'])

aa = (a[:2])

aa = aa.append(a[3:])

cords= nds.drop(columns=['Sample', 'Time', 'Yield', 'Nd', 'La', 'Method','Rate'])

cor=cords.corr()

a1 = abs(cor['Ce'])

aa1 = (a1[:2])

aa1 = aa1.append(a1[3:])

cords= nds.drop(columns=['Sample', 'Time', 'Yield', 'La', 'Ce', 'Method','Rate'])

cor=cords.corr()

a2 = abs(cor['Nd'])

aa2 = (a2[:2])

aa2 = aa2.append(a2[3:])

ax.barh(cols, aa[0:], height=-0.3, align='edge', color = 'dimgrey', linewidth = 1)

ax.barh(cols, aa2[0:], height=0.3, align='edge', color = 'lightgrey')

ax.barh(cols, aa1[0:], height=0.2, align='center', color = 'darkgrey')

ax.set\_title('Absolute Correlation to Yield%')

for s in ['top', 'bottom', 'left', 'right']:

ax.spines[s].set\_visible(False)

ax.xaxis.set\_ticks\_position('none')

ax.yaxis.set\_ticks\_position('none')

ax.xaxis.set\_tick\_params(pad = 5)

ax.yaxis.set\_tick\_params(pad = 10)

ax.grid(visible = True, color ='grey',

linestyle ='-.', linewidth = 0.2,

alpha = 0.2)

ax.invert\_yaxis()

ax.legend(['La','Ce', 'Nd'], loc = 'lower right', fontsize='small')

ax.set\_facecolor('xkcd:white')

plt.show()