

Python Commands

Action/Result

# Workflow Chart and Process Design

OPEN JUPYTER NOTEBOOK THROUGH ANACONDA



```
import pandas as pd
Data1='C:/Users/Dell/Desktop/Data Files/IPEDS Data.xlsx'
Data2='C:/Users/Dell/Desktop/Data Files/Poverty Rate by Zipcode.xlsx'
Data3='C:/Users/Dell/Desktop/Data Files/Educational Attainment.csv'
IPEDSData=pd.read_excel(Data1)
PovData=pd.read_excel(Data2)
EduData=pd.read_csv(Data3)
```

Importing research-specific data files



```
from uszipcode import SearchEngine

def retzip(lat, lng, rad):
    search = SearchEngine()
    result = search.by_coordinates(lat, lng, radius=rad, returns=30)

    ls_zip = []
    for zip in result:
        ls_zip.append(zip.values()[0])

    return ls_zip
```

Calling the “uszipcode” programmable database which includes up-to-date census and geometry information, specifically the latitudes and longitudes of zip codes within the U.S.



```
uni_rad = pd.DataFrame()

uni_rad['uni'] = IPEDSData['Institution']
uni_rad['lat'] = IPEDSData['Latitude']
uni_rad['lng'] = IPEDSData['Longitude']
uni_rad['zips'] = pd.Series()

for i in range(0, len(uni_rad)):
    res = retzip(uni_rad.loc[i]['lat'], uni_rad.loc[i]['lng'], 6)
    uni_rad['zips'][i] = res

uni_rad
```

Creating a loop to generate a list of zip codes which fall within 6 miles of Texas higher education institutions' latitudes and longitudes.



```
PovData = PovData.replace("-", 0)
PovData = PovData.replace(np.nan, 0)
uni_rad['avg_povrate'] = pd.Series()

for i in range(0, len(uni_rad)):
    ls = uni_rad['zips'][i]
    num = 0
    for zip in ls:
        rate1 = float(PovData[PovData['ZIP_NEW']==int(zip)].PovRate12.item())
        rate2 = float(PovData[PovData['ZIP_NEW']==int(zip)].PovRate13.item())
        rate3 = float(PovData[PovData['ZIP_NEW']==int(zip)].PovRate14.item())
        rate4 = float(PovData[PovData['ZIP_NEW']==int(zip)].PovRate15.item())
        rate5 = float(PovData[PovData['ZIP_NEW']==int(zip)].PovRate16.item())
        rate6 = float(PovData[PovData['ZIP_NEW']==int(zip)].PovRate17.item())
        totalrate=rate1 + rate2 + rate3 + rate4 + rate5 + rate6
        num = num + totalrate

    if len(ls)>0:
        avg = num/(6*len(ls))
    else:
        avg = np.nan
    uni_rad['avg_povrate'][i] = avg
    print(ls, avg, totalrate)

uni_rad
```

Creating a loop to generate the average poverty rates of zip codes that fall within 6 miles of a higher education institution in Texas.



```
import statsmodels.api as sm
model = sm.OLS(y4, x4, missing='drop').fit()
reg = model.predict(X) # make the predictions by the model
model.summary()
x = uni_rad["avg_povrate"]
y = IPEDSDData["Avg_enroll_12-17"]
model = sm.OLS(y, x, missing='drop').fit()
reg = model.predict(X) # make the predictions by the model
model.summary()
```

Running an ordinary least squares regression to analyze the effect of the average poverty rate of zip codes that fall within 6 miles of a higher education institution on the average enrollment of students at a Texas higher education institutions between 2012 and 2017. The output showed an R-squared value of 0.884 and a p-value of 0.000 which shows statistical significance.



CLOSE JUPYTER NOTEBOOK THROUGH ANACONDA