Geolocation

November 22, 2024

[1]:
```
import requests
from bs4 import BeautifulSoup
import pandas as pd

# define url
url = "http://www.geonames.org/postal-codes/"

# create []
data = []

# web scrabing
def fetch_postal_codes(region_url):
    response = requests.get(region_url)
    soup = BeautifulSoup(response.text, "html.parser")

    # put latitude and longitude into table
    table = soup.find("table", {"class": "restable"})
    if table:
        rows = table.find_all("tr")
        for row in rows[1:]:
            cols = row.find_all("td")
            if len(cols) >= 4:
                postal_code = cols[1].text.strip()
                latitude = cols[2].text.strip()
                longitude = cols[3].text.strip()
                data.append([postal_code, latitude, longitude])

# specify postal codes
fetch_postal_codes(url + "CA/Toronto")

# save
df = pd.DataFrame(data, columns=["postal_code", "Latitude", "Longitude"])

# output
df.to_csv("canada_postal_codes.csv", index=False)
print("It's done,save as canada_postal_codes.csv")
```

```
It's done,save as canada_postal_codes.csv
```

[2]:
```
# reading
data_path = "z_score_cleaned_data_rounded.csv"
scores_data = pd.read_csv(data_path)

# read latitude and longitude
geolocation_path = "canada_postal_codes.csv"
geo_data = pd.read_csv(geolocation_path)

# keep the format consisdency
scores_data["postal_code"] = scores_data["postal_code"].str.strip().str.upper()
geo_data["postal_code"] = geo_data["postal_code"].str.strip().str.upper()

# merge
merged_data = pd.merge(scores_data, geo_data, on="postal_code", how="left")

# checking
unmatched = merged_data[merged_data["Latitude"].isna()]
print(f"the number of unmatached {len(unmatched)}")
print(unmatched["postal_code"])

# save new csv
output_path = "z_score_with_geolocation.csv"
merged_data.to_csv(output_path, index=False)
print(f"It's done, save as {output_path}")
```

```
the number of unmatached 96
0     M5V
1     M2N
2     M6P
3     M6S
4     M4S
     …
91    M3K
92    M1X
93    M3L
94    M4H
95    M9L
Name: postal_code, Length: 96, dtype: object
It's done, save as z_score_with_geolocation.csv
```

[ ]: