

```
In [3]: pip install folium
```

```
Collecting folium
  Downloading folium-0.19.5-py2.py3-none-any.whl.metadata (4.1 kB)
Collecting branca>=0.6.0 (from folium)
  Downloading branca-0.8.1-py3-none-any.whl.metadata (1.5 kB)
Requirement already satisfied: jinja2>=2.9 in c:\users\amarachi\anaconda3\lib\site-packages (from folium) (3.1.4)
Requirement already satisfied: numpy in c:\users\amarachi\anaconda3\lib\site-packages (from folium) (1.26.4)
Requirement already satisfied: requests in c:\users\amarachi\anaconda3\lib\site-packages (from folium) (2.32.3)
Requirement already satisfied: xyzservices in c:\users\amarachi\anaconda3\lib\site-packages (from folium) (2022.9.0)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\amarachi\anaconda3\lib\site-packages (from jinja2>=2.9->folium) (2.1.3)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\amarachi\anaconda3\lib\site-packages (from requests->folium) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\users\amarachi\anaconda3\lib\site-packages (from requests->folium) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\amarachi\anaconda3\lib\site-packages (from requests->folium) (2.2.3)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\amarachi\anaconda3\lib\site-packages (from requests->folium) (2025.1.31)
  Downloading folium-0.19.5-py2.py3-none-any.whl (110 kB)
  Downloading branca-0.8.1-py3-none-any.whl (26 kB)
Installing collected packages: branca, folium
Successfully installed branca-0.8.1 folium-0.19.5
Note: you may need to restart the kernel to use updated packages.
```

```
In [7]: pip install geopy
```

```
Collecting geopy
  Downloading geopy-2.4.1-py3-none-any.whl.metadata (6.8 kB)
Collecting geographiclib<3,>=1.52 (from geopy)
  Downloading geographiclib-2.0-py3-none-any.whl.metadata (1.4 kB)
  Downloading geopy-2.4.1-py3-none-any.whl (125 kB)
  Downloading geographiclib-2.0-py3-none-any.whl (40 kB)
Installing collected packages: geographiclib, geopy
Successfully installed geographiclib-2.0 geopy-2.4.1
Note: you may need to restart the kernel to use updated packages.
```

```
In [11]: pip install osmnx
```

```
Collecting osmnx
  Downloading osmnx-2.0.2-py3-none-any.whl.metadata (4.9 kB)
Collecting geopandas>=1.0 (from osmnx)
  Downloading geopandas-1.0.1-py3-none-any.whl.metadata (2.2 kB)
Requirement already satisfied: networkx>=2.5 in c:\users\amarachi\anaconda3\lib\site-packages (from osmnx) (3.3)
Requirement already satisfied: numpy>=1.22 in c:\users\amarachi\anaconda3\lib\site-packages (from osmnx) (1.26.4)
Requirement already satisfied: pandas>=1.4 in c:\users\amarachi\anaconda3\lib\site-packages (from osmnx) (2.2.2)
Requirement already satisfied: requests>=2.27 in c:\users\amarachi\anaconda3\lib\site-packages (from osmnx) (2.32.3)
Collecting shapely>=2.0 (from osmnx)
  Downloading shapely-2.0.7-cp312-cp312-win_amd64.whl.metadata (7.1 kB)
Collecting pyogrio>=0.7.2 (from geopandas>=1.0->osmnx)
  Downloading pyogrio-0.10.0-cp312-cp312-win_amd64.whl.metadata (5.6 kB)
Requirement already satisfied: packaging in c:\users\amarachi\anaconda3\lib\site-packages (from geopandas>=1.0->osmnx) (24.1)
Collecting pyproj>=3.3.0 (from geopandas>=1.0->osmnx)
  Downloading pyproj-3.7.1-cp312-cp312-win_amd64.whl.metadata (31 kB)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\amarachi\anaconda3\lib\site-packages (from pandas>=1.4->osmnx) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\amarachi\anaconda3\lib\site-packages (from pandas>=1.4->osmnx) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\amarachi\anaconda3\lib\site-packages (from pandas>=1.4->osmnx) (2023.3)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->osmnx) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->osmnx) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->osmnx) (2.2.3)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->osmnx) (2025.1.31)
Requirement already satisfied: six>=1.5 in c:\users\amarachi\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas>=1.4->osmnx) (1.16.0)
Downloading osmnx-2.0.2-py3-none-any.whl (99 kB)
Downloading geopandas-1.0.1-py3-none-any.whl (323 kB)
Downloading shapely-2.0.7-cp312-cp312-win_amd64.whl (1.4 MB)
----- 0.0/1.4 MB ? eta -:---:-
----- 0.5/1.4 MB 3.3 MB/s eta 0:00:01
----- 1.4/1.4 MB 4.4 MB/s eta 0:00:00
Downloading pyogrio-0.10.0-cp312-cp312-win_amd64.whl (16.2 MB)
----- 0.0/16.2 MB ? eta -:---:-
----- 3.4/16.2 MB 16.8 MB/s eta 0:00:01
----- 5.2/16.2 MB 16.0 MB/s eta 0:00:01
----- 5.2/16.2 MB 16.0 MB/s eta 0:00:01
----- 5.2/16.2 MB 16.0 MB/s eta 0:00:01
----- 9.4/16.2 MB 8.6 MB/s eta 0:00:01
----- 12.1/16.2 MB 9.7 MB/s eta 0:00:01
----- 14.2/16.2 MB 9.3 MB/s eta 0:00:01
----- 16.2/16.2 MB 10.0 MB/s eta 0:00:00
Downloading pyproj-3.7.1-cp312-cp312-win_amd64.whl (6.3 MB)
----- 0.0/6.3 MB ? eta -:---:-
----- 2.6/6.3 MB 12.6 MB/s eta 0:00:01
----- 2.9/6.3 MB 12.0 MB/s eta 0:00:01
```

```
----- 6.3/6.3 MB 12.0 MB/s eta 0:00:00
Installing collected packages: shapely, pyproj, pyogrio, geopandas, osmnx
Successfully installed geopandas-1.0.1 osmnx-2.0.2 pyogrio-0.10.0 pyproj-3.7.1 shape
ly-2.0.7
Note: you may need to restart the kernel to use updated packages.
```

```
In [13]: import numpy as np # for numerical operations
import folium # for map plotting
from folium.plugins import MarkerCluster
import geopy # for geocoding
import matplotlib.pyplot as plt # for plotting
import seaborn as sns # for plotting
import time # for time operations to avoid Api limit issues for the geocoding
import requests
import pandas as pd
from tqdm import tqdm
import osmnx as ox
import geopandas as gpd
```

```
In [15]: df = pd.read_csv("ABIA_crosschecked.csv")
```

```
In [17]: df.head(10) # Display first 10 rows
```

Out[17]:

	State	LGA	Ward	PU-Code	PU-Name	Accredited_Voters	Registered_Voters
0	ABIA	ABA NORTH	EZIAMA	01-01-01-001	RAILWAY QUARTERS - RAILWAY QUARTERS I	85	968
1	ABIA	ABA NORTH	EZIAMA	01-01-01-002	RAILWAY QUARTERS - RAILWAY QUARTERS II	90	750
2	ABIA	ABA NORTH	EZIAMA	01-01-01-003	RAILWAY QUARTERS - RAILWAY QUARTERS III	105	750
3	ABIA	ABA NORTH	EZIAMA	01-01-01-005	ABIA POLY - ABIA POLY I	138	750
4	ABIA	ABA NORTH	INDUSTRIAL AREA	01-01-02-012	LEVER BROTHERS GATE- LEVER BROTHERS GATE IV	71	774
5	ABIA	ABA NORTH	OSUSU I	01-01-03-002	OSUSU RD.PRI.SCHOOL- SCHOOL PREMISES II	156	883
6	ABIA	ABA NORTH	OSUSU II	01-01-04-004	OSUSU SECONDARY SCHOOL - SCHOOL PREMISES IV	297	1678
7	ABIA	ABA NORTH	OSUSU II	01-01-04-014	BRIGHT WAY NURS. SCHOOL - SCHOOL PREMISES VII	144	934
8	ABIA	ABA NORTH	OSUSU II	01-01-04-016	POPE JOHN PAUL COLLEGE- SCHOOL PREMISES I	173	877
9	ABIA	ABA NORTH	OSUSU II	01-01-04-018	POPE JOHN PAUL COLLEGE- SCHOOL PREMISES III	165	874



```
In [19]: df.shape
```

```
Out[19]: (2492, 19)
```

```
In [21]: df.info
```

			State	LGA	Ward	PU-Co	
0	ABIA	ABA NORTH	EZIAMA	01-01-01-001			
1	ABIA	ABA NORTH	EZIAMA	01-01-01-002			
2	ABIA	ABA NORTH	EZIAMA	01-01-01-003			
3	ABIA	ABA NORTH	EZIAMA	01-01-01-005			
4	ABIA	ABA NORTH	INDUSTRIAL AREA	01-01-02-012			
...		
2487	ABIA	ABA SOUTH	EZIUKWU	01-02-01-054			
2488	ABIA	OHAFIA	NDI AGBO NKPORO	01-10-11-019			
2489	ABIA	OHAFIA	NDI AGBO NKPORO	01-10-11-020			
2490	ABIA	UKWA EAST	IKWURIATOR WEST	01-13-02-005			
2491	ABIA	IKWUANO	OLOKO I	01-05-01-018			
			PU-Name	Accredited_Voters		\	
0		RAILWAY QUARTERS - RAILWAY QUARTERS I		85			
1		RAILWAY QUARTERS - RAILWAY QUARTERS II		90			
2		RAILWAY QUARTERS - RAILWAY QUARTERS III		105			
3		ABIA POLY - ABIA POLY I		138			
4		LEVER BROTHERS GATE- LEVER BROTHERS GATE IV		71			
...			
2487		ASA-OKPUAJA HALL IV		25			
2488		AGBAJA PRIMARY SCHOOL		23			
2489		OGBARALI FARM SETTLEMENT		0			
2490		COMMUNITY PRIMARY SCHOOL, OBUOZU		54			
2491		OGWUMABIRI MARKET SQUARE AMAPU II		0			
	Registered_Voters	Results_Found	Transcription_Count			\	
0	968	True		-1			
1	750	True		-1			
2	750	True		-1			
3	750	True		-1			
4	774	True		-1			
...		
2487	57	True		-1			
2488	136	True		-1			
2489	50	True		-1			
2490	96	True		-1			
2491	0	True		-1			
	Result_Sheet_Stamped	Result_Sheet_Corrected	Result_Sheet_Invalid			\	
0	True	True		False			
1	False	False		False			
2	False	False		False			
3	False	False		False			
4	False	False		False			
...		
2487	False	False		False			
2488	False	False		True			
2489	False	False		False			
2490	False	False		False			
2491	True	False		True			
	Result_Sheet_Unclear	Result_Sheet_Unsigned	APC	LP	PDP	NNPP	\
0	False	UNKNOWN	7	56	25	1	
1	False	UNKNOWN	0	0	0	0	

2	False	UNKNOWN	0	0	0	0
3	False	UNKNOWN	0	0	0	0
4	False	UNKNOWN	0	71	0	0
...
2487	False	UNKNOWN	0	24	0	0
2488	False	UNKNOWN	0	21	0	0
2489	False	UNKNOWN	0	0	0	0
2490	False	UNKNOWN	2	25	13	0
2491	False	UNKNOWN	0	0	0	0

Results_File						
0	https://docs.inecelectionresults.net/elections...					
1	https://docs.inecelectionresults.net/elections...					
2	https://docs.inecelectionresults.net/elections...					
3	https://docs.inecelectionresults.net/elections...					
4	https://docs.inecelectionresults.net/elections...					
...	...					
2487	https://inec-cvr-cache.s3.eu-west-1.amazonaws...					
2488	https://docs.inecelectionresults.net/elections...					
2489	https://docs.inecelectionresults.net/elections...					
2490	https://docs.inecelectionresults.net/elections...					
2491	https://docs.inecelectionresults.net/elections...					

[2492 rows x 19 columns]>

In [23]: `df.describe()`

	Accredited_Voters	Registered_Voters	Transcription_Count	APC	LP
count	2492.000000	2492.000000	2492.0	2492.000000	2492.000000
mean	96.849920	475.422552	-1.0	2.926565	78.174960
std	76.049937	434.374296	0.0	10.733881	69.568717
min	0.000000	0.000000	-1.0	0.000000	0.000000
25%	39.000000	111.000000	-1.0	0.000000	26.000000
50%	84.000000	413.000000	-1.0	1.000000	63.000000
75%	138.000000	743.000000	-1.0	3.000000	111.000000
max	571.000000	4747.000000	-1.0	350.000000	506.000000



In [25]: `df.isnull().sum()`

```
Out[25]: State          0  
LGA           0  
Ward          0  
PU-Code       0  
PU-Name       0  
Accredited_Voters  0  
Registered_Voters 0  
Results_Found    0  
Transcription_Count 0  
Result_Sheet_Stamped 0  
Result_Sheet_Corrected 0  
Result_Sheet_Invalid   0  
Result_Sheet_Unclear    0  
Result_Sheet_Unsigned   0  
APC            0  
LP             0  
PDP            0  
NNPP           0  
Results_File     0  
dtype: int64
```

```
In [27]: df.duplicated().sum()
```

```
Out[27]: 0
```

```
In [29]: categorical_cols = df.select_dtypes(include=['object']).columns  
for col in categorical_cols:  
    print(f"Unique values in {col}: {df[col].nunique()}")
```

```
Unique values in State: 1  
Unique values in LGA: 17  
Unique values in Ward: 178  
Unique values in PU-Code: 2492  
Unique values in PU-Name: 2356  
Unique values in Result_Sheet_Unsigned: 1  
Unique values in Results_File: 2370
```

```
In [31]: for col in categorical_cols:  
    print(f"Column: {col}")  
    print(df[col].unique()) # Show first 10 unique values  
    print("-" * 50)
```

Column: State

['ABIA']

Column: LGA

['ABA NORTH' 'ABA SOUTH' 'AROCHUKWU' 'BENDE' 'IKWUANO' 'ISIALA NGWA NORTH'
 'ISIALA NGWA SOUTH' 'ISUIKWUATO' 'OBINGWA' 'OHAFIA' 'OSISIOMA'
 'UGWUNAGBO' 'UKWA EAST' 'UKWA WEST' 'UMUAHIA NORTH' 'UMUAHIA SOUTH'
 'UMU - NNEOCHI']

Column: Ward

['EZIAMA' 'INDUSTRIAL AREA' 'OSUSU I' 'OSUSU II'
 'ST.EUGENES BY OKIGWE RD.' 'URATTA' 'OLD ABA GRA' 'UMUOLA'
 'ARIARIA MARKET' 'UMUOGOR' 'EZIUKWU' 'ASA' 'ENYIMBA' 'NGWA' 'OHAZU I'
 'EKEOHA' 'GLOUCESTER' 'MOSQUE' 'ABA RIVER' 'ABA TOWN HALL' 'OVUKWU'
 'OHAKE' 'AROCHUKWU I' 'IKWUN IHECHIOWA' 'UTUTU' 'ISU'
 'AMANKALU/AKOLIUFU' 'BENDE' 'UGWUEKE/EZEUKWU' "IGBERE 'A'" "IGBERE 'B'"
 'ITEM A' 'ITEM B' 'ITEM C' 'ITUMBAUZO' 'OZUITEM' 'UMUHU / EZECHI'
 'UMU - IMENYI' 'UZUAKOLI' 'OLOKO I' 'OLOKO II' 'IBERE I' 'IBERE II'
 'OBORO II' 'OBORO III' 'OBORO IV' 'ARIAM' 'USAKA' 'AMASAA NSULU'
 'UMUNNA NSULU' 'ISIALA NSULU' 'NGWA UKWU I' 'NGWA UKWU II' 'IHIE'
 'AMASAA NTIGHA' 'AMAPU NTIGHA' 'UMUOHA' 'MBAWSI / UMUOMAINTA'
 'AMAISE / AMAISE ANABA' 'NGWAOBI' 'MBUTU UKWU' 'MBUTU NGWA'
 'EHINA GURU OSOKWA' 'AKUNEKPU EZIAMA NA OBUBA' 'OMOBA' 'OVUNGWU'
 'OVUOKWU' 'OKPORO AHABA' 'IMENYI' 'EZERE' 'ISIALA AMAWU' 'ISU AMAWU'
 'OGUNDUASA' 'UMUNNEKWU' 'ACHARA / MGBUGWU' 'IKEAGHA I' 'IKEAGHA II'
 'UMUANYI / ABSU' 'ABAYI I' 'ABAYI II' 'MGBOKO UMUANUNU'
 'MGBOKO ITUNGWA' 'AHIABA' 'MABOKO AMAIRI' 'ALAUKWU OHANZE' 'AKUMAIMO'
 'NDIAKATA / AMAIRINABUA' 'NTIGHAUZO AMAIRI' 'IBEME' 'EBEM OHAFIA'
 'NDI ELU NKPORO' 'NDI ETITI NKPORO' 'AMAEKE ABIRIBA' 'AMAOGUDU ABIRIBA'
 'OHAFOR OHAFIA' 'ANIA OHAFIA' 'NDI AGBO NKPORO' 'AMAVO'
 'AMAITOLU MBUTU UMUOJIMA' 'AMASATOR' 'ARO - NGWA' 'AMA - ASAA'
 'OSO - OKWA' 'URTTA' 'AMATOR' 'UMUNNEISE' 'OKPOR - UMUOBO' 'WARD ONE'
 'WARD TWO' 'WARD THREE' 'WARD FOUR' 'WARD FIVE' 'WARD SIX' 'WARD SEVEN'
 'WARD EIGHT' 'WARD NINE' 'WARD TEN' 'IKWURIATOR EAST' 'IKWURIATOR WEST'
 'AZUMINI' 'UMUIGUBE ACHARA' 'AKWETE' 'OBOHIA' 'IKWUEKE EAST'
 'IKWUEKE WEST' 'NKPOROBE/OHURU' 'IKWUORIE' 'ASA NORTH' 'OGWE'
 'ASA SOUTH' 'OBUZOR' 'IPU WEST' 'IPU EAST' 'IPU SOUTH' 'OZAA UKWU'
 'OZAA WEST' 'IBEKU EAST I' 'IBEKU EAST II' 'NDUME' 'IBEKU WEST'
 'UMUAHIA URBAN I' 'UMUAHIA URBAN II' 'UMUAHIA URBAN III' 'NKWOACHARA'
 'NKWOEGWU' 'AFUGIRI' 'UMUHU' 'ISINGWU' 'EZELEKE/OGBODIUKWU' 'OMAEGWU'
 'OHIAOCHA' 'AHIAUKWU I' 'AHIAUKWU II' 'OLD UMUAHIA' 'AMAKAMA'
 'UBAKALA A' 'NSIRIMO' 'UBAKALA B' 'AMUDA' 'UMUAKU' 'MBALA/ACHARA'
 'EZINGODO' 'NDIAWA/UMUELEM/I' 'EZIAMA - UGWU' 'EZIAMA - AGBO'
 'UBAHU/AKAWA/AROKPA' 'UMUCHIEZE I' 'UMUCHIEZE II' 'UMUCHIEZE III'
 'OBINOLU/OBIAGU/LA' 'IGWEBUIKE' 'OHAZU II' 'AROCHUKWU II' 'AROCHUKWU III'
 'ELEOHA IHECHIOWA' 'OHAFOR I' 'OHAFOR II' 'OBORO I']

Column: PU-Code

['01-01-01-001' '01-01-01-002' '01-01-01-003' ... '01-10-11-020'
 '01-13-02-005' '01-05-01-018']

Column: PU-Name

['RAILWAY QUARTERS - RAILWAY QUARTERS I'
 'RAILWAY QUARTERS - RAILWAY QUARTERS II'
 'RAILWAY QUARTERS - RAILWAY QUARTERS III' ... 'OGBARALI FARM SETTLEMENT'
 'COMMUNITY PRIMARY SCHOOL, OBUOZU' 'OGWUMABIRI MARKET SQUARE AMAPU II']

```
-----
Column: Result_Sheet_Unsigned
['UNKNOWN']

-----
Column: Results_File
['https://docs.inecelectionresults.net/elections_prod/1292/state/01/lga/01/ward/01/pu/001/001-1677420870.pdf'
 'https://docs.inecelectionresults.net/elections_prod/1292/state/1/lga/3103/ward/17526/pu/8/8-1677512048.pdf'
 'https://docs.inecelectionresults.net/elections_prod/1292/state/1/lga/3103/ward/17526/pu/9/9-1677512335.pdf'
 ...
 'https://docs.inecelectionresults.net/elections_prod/1292/state/1/lga/3112/ward/17635/pu/127368/127368-1677532728.pdf'
 'https://docs.inecelectionresults.net/elections_prod/1292/state/1/lga/3115/ward/17657/pu/127611/127611-1677579387.pdf'
 'https://docs.inecelectionresults.net/elections_prod/1292/state/1/lga/3107/ward/17574/pu/127034/127034-1677757414.pdf']
-----
```

This code counts how many results files appear more than once, flagging potential election irregularities. Duplicate files could indicate fraud or errors in reporting. It helps identify suspicious patterns for further investigation.

```
In [33]: # Count occurrences of each Results_File
file_counts = df['Results_File'].value_counts()

# Identify duplicate files (appearing more than once)
duplicate_files = file_counts[file_counts > 1].index

# Count the number of flagged duplicates
num_duplicates = df['Results_File'].isin(duplicate_files).sum()

# Display the count of duplicate entries
print(f"Total duplicate entries: {num_duplicates}")
```

Total duplicate entries: 176

```
In [35]: invalid_cases = df[df["Accredited_Voters"] > df["Registered_Voters"]]
print("Total invalid cases:", len(invalid_cases))
```

Total invalid cases: 0

```
In [37]: # Drop the specified columns
df = df.drop(columns=["Result_Sheet_Unsigned", "State"])
```

```
In [82]: # # Your Google Maps API Key
# API_KEY = "AIzaSyAn8XXMEOJrqo5Hk5UdnfgvFM4-L5_0NQ8"

# # Function to get Latitude and Longitude
# def get_lat_lon(address):
#     base_url = "https://maps.googleapis.com/maps/api/geocode/json"
#     params = {
#         "address": address,
#         "key": API_KEY
#     }
```

```

#     response = requests.get(base_url, params=params)
#     result = response.json()

#     if result["status"] == "OK":
#         location = result["results"][0]["geometry"]["Location"]
#         return location["lat"], location["lng"]
#     else:
#         return None, None

# # Create new columns for Latitude and Longitude
# df["Latitude"] = None
# df["Longitude"] = None

# # Iterate over polling units and get coordinates
# for index, row in tqdm(df.iterrows(), total=len(df)):
#     address = f"{row['PU-Name']}, {row['Ward']}, {row['LGA']}, {row['State']}, Ni
#     lat, lon = get_lat_lon(address)
#     df.at[index, "Latitude"] = lat
#     df.at[index, "Longitude"] = lon

# # Save the updated dataset
# df.to_csv("geocoded_dataset.csv", index=False)
# print("Geocoding complete. Saved as geocoded_dataset.csv")

```

In [28]: `#pip install haversine`

In [39]: `# Define the place name for Abia State`
`place_name = "Abia, Nigeria"`

`# Download the administrative boundary from OSM`
`abia_boundary = ox.geocode_to_gdf(place_name)`

`# Save as GeoJSON file`
`abia_boundary.to_file("abia_state_boundary.geojson", driver="GeoJSON")`

`print("✓ Abia State boundary saved as abia_state_boundary.geojson")`

✓ Abia State boundary saved as abia_state_boundary.geojson

In [41]: `import folium`
`import pandas as pd`
`import geopandas as gpd`
`from shapely.geometry import Point`

`# Load polling unit data`
`df = pd.read_csv("geocoded_dataset.csv")`

`# Load Abia State boundary (GeoJSON file)`
`abia_boundary = gpd.read_file("abia_state_boundary.geojson")`

`# Convert polling unit coordinates into geometry points`
`df["geometry"] = df.apply(lambda row: Point(row["Longitude"], row["Latitude"]), axis=1)`

`# Convert polling unit dataframe to GeoDataFrame`
`gdf = gpd.GeoDataFrame(df, geometry="geometry", crs="EPSG:4326")`

```

# Check if polling units are inside Abia's boundary
gdf["inside_abia"] = gdf.geometry.within(abia_boundary.union_all())

# Filter out polling units outside Abia
gdf = gdf[gdf["inside_abia"] == True]

# Save the filtered data to a new CSV file
gdf.drop(columns=["geometry", "inside_abia"]).to_csv("filtered_polling_units.csv",

# Create the map centered around Abia
abia_center = [5.532, 7.486] # Central coordinates of Abia
m = folium.Map(location=abia_center, zoom_start=9)

# Plot only polling units inside Abia (Green dots)
for _, row in gdf.iterrows():
    folium.CircleMarker(
        location=[row["Latitude"], row["Longitude"]],
        radius=3,
        color="green",
        fill=True,
        fill_color="green",
        fill_opacity=0.7,
        popup=f"Polling Unit: {row['PU-Name']}",
    ).add_to(m)

# Save the map as an HTML file
m.save("abia_polling_units_map.html")

# Display the map
m

```



In [46]: pip install haversine

Requirement already satisfied: haversine in c:\users\amarachi\anaconda3\lib\site-packages (2.9.0)

Note: you may need to restart the kernel to use updated packages.

```
In [48]: import pandas as pd
import numpy as np
from sklearn.cluster import DBSCAN
from haversine import haversine

# Load geocoded dataset
df = pd.read_csv("filtered_polling_units.csv")

# Convert Latitude and Longitude to tuples
df["coordinates"] = list(zip(df["Latitude"], df["Longitude"]))

# Function to apply DBSCAN clustering with Haversine distance
def apply_dbSCAN(df, radius_meters, min_samples=2):
    radius_km = radius_meters / 1000 # Convert meters to kilometers
    coords = np.radians(df["coordinates"].tolist()) # Convert to radians
    clustering = DBSCAN(eps=radius_km / 6371, min_samples=min_samples, metric='haversine')
    return clustering.labels_

# Apply DBSCAN clustering for different radius values
df["Cluster_500m"] = apply_dbSCAN(df, 500)
df["Cluster_1km"] = apply_dbSCAN(df, 1000)
df["Cluster_2km"] = apply_dbSCAN(df, 2000)

# Save the clustered dataset
df.to_csv("clustered_abia.csv", index=False)
print("Clustering complete! Results saved in 'clustered_abia.csv'.")
```

Clustering complete! Results saved in 'clustered_abia.csv'.

```
In [52]: pip installesda
```

```
Collecting esda
```

```
  Downloading esda-2.7.0-py3-none-any.whl.metadata (2.0 kB)
Requirement already satisfied: geopandas>=0.12 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (1.0.1)
Collecting libpysal>=4.12 (from esda)
  Downloading libpysal-4.13.0-py3-none-any.whl.metadata (4.8 kB)
Requirement already satisfied: numpy>=1.24 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (1.26.4)
Requirement already satisfied: pandas>1.5 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (2.2.2)
Requirement already satisfied: scikit-learn>=1.2 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (1.5.1)
Requirement already satisfied: scipy>=1.9 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (1.13.1)
Requirement already satisfied: shapely>=2.0 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (2.0.7)
Requirement already satisfied: pyogrio>=0.7.2 in c:\users\amarachi\anaconda3\lib\site-packages (from geopandas>=0.12->esda) (0.10.0)
Requirement already satisfied: packaging in c:\users\amarachi\anaconda3\lib\site-packages (from geopandas>=0.12->esda) (24.1)
Requirement already satisfied: pyproj>=3.3.0 in c:\users\amarachi\anaconda3\lib\site-packages (from geopandas>=0.12->esda) (3.7.1)
Requirement already satisfied: beautifulsoup4>=4.10 in c:\users\amarachi\anaconda3\lib\site-packages (from libpysal>=4.12->esda) (4.12.3)
Requirement already satisfied: platformdirs>=2.0.2 in c:\users\amarachi\anaconda3\lib\site-packages (from libpysal>=4.12->esda) (3.10.0)
Requirement already satisfied: requests>=2.27 in c:\users\amarachi\anaconda3\lib\site-packages (from libpysal>=4.12->esda) (2.32.3)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\amarachi\anaconda3\lib\site-packages (from pandas>1.5->esda) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\amarachi\anaconda3\lib\site-packages (from pandas>1.5->esda) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\amarachi\anaconda3\lib\site-packages (from pandas>1.5->esda) (2023.3)
Requirement already satisfied: joblib>=1.2.0 in c:\users\amarachi\anaconda3\lib\site-packages (from scikit-learn>=1.2->esda) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in c:\users\amarachi\anaconda3\lib\site-packages (from scikit-learn>=1.2->esda) (3.5.0)
Requirement already satisfied: soupsieve>1.2 in c:\users\amarachi\anaconda3\lib\site-packages (from beautifulsoup4>=4.10->libpysal>=4.12->esda) (2.5)
Requirement already satisfied: certifi in c:\users\amarachi\anaconda3\lib\site-packages (from pyogrio>=0.7.2->geopandas>=0.12->esda) (2025.1.31)
Requirement already satisfied: six>=1.5 in c:\users\amarachi\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas>1.5->esda) (1.16.0)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->libpysal>=4.12->esda) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->libpysal>=4.12->esda) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->libpysal>=4.12->esda) (2.2.3)
  Downloading esda-2.7.0-py3-none-any.whl (142 kB)
  Downloading libpysal-4.13.0-py3-none-any.whl (2.8 MB)
----- 0.0/2.8 MB ? eta -:---:
----- 0.3/2.8 MB ? eta -:---:
----- 1.3/2.8 MB 3.9 MB/s eta 0:00:01
----- 2.8/2.8 MB 5.9 MB/s eta 0:00:00
```

```
Installing collected packages: libpysal,esda
Successfully installed esda-2.7.0 libpysal-4.13.0
Note: you may need to restart the kernel to use updated packages.
```

```
In [64]: pip install --upgradeesda libpysal
```

Requirement already satisfied:esda in c:\users\amarachi\anaconda3\lib\site-packages (2.7.0)
Note: you may need to restart the kernel to use updated packages.

Requirement already satisfied: libpysal in c:\users\amarachi\anaconda3\lib\site-packages (4.13.0)

Requirement already satisfied: geopandas>=0.12 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (1.0.1)

Requirement already satisfied: numpy>=1.24 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (1.26.4)

Requirement already satisfied: pandas>1.5 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (2.2.2)

Requirement already satisfied: scikit-learn>=1.2 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (1.5.1)

Requirement already satisfied: scipy>=1.9 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (1.13.1)

Requirement already satisfied: shapely>=2.0 in c:\users\amarachi\anaconda3\lib\site-packages (from esda) (2.0.7)

Requirement already satisfied: beautifulsoup4>=4.10 in c:\users\amarachi\anaconda3\lib\site-packages (from libpysal) (4.12.3)

Requirement already satisfied: packaging>=22 in c:\users\amarachi\anaconda3\lib\site-packages (from libpysal) (24.1)

Requirement already satisfied: platformdirs>=2.0.2 in c:\users\amarachi\anaconda3\lib\site-packages (from libpysal) (3.10.0)

Requirement already satisfied: requests>=2.27 in c:\users\amarachi\anaconda3\lib\site-packages (from libpysal) (2.32.3)

Requirement already satisfied: soupsieve>1.2 in c:\users\amarachi\anaconda3\lib\site-packages (from beautifulsoup4>=4.10->libpysal) (2.5)

Requirement already satisfied: pyogrio>=0.7.2 in c:\users\amarachi\anaconda3\lib\site-packages (from geopandas>=0.12->esda) (0.10.0)

Requirement already satisfied: pyproj>=3.3.0 in c:\users\amarachi\anaconda3\lib\site-packages (from geopandas>=0.12->esda) (3.7.1)

Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\amarachi\anaconda3\lib\site-packages (from pandas>1.5->esda) (2.9.0.post0)

Requirement already satisfied: pytz>=2020.1 in c:\users\amarachi\anaconda3\lib\site-packages (from pandas>1.5->esda) (2024.1)

Requirement already satisfied: tzdata>=2022.7 in c:\users\amarachi\anaconda3\lib\site-packages (from pandas>1.5->esda) (2023.3)

Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->libpysal) (3.3.2)

Requirement already satisfied: idna<4,>=2.5 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->libpysal) (3.7)

Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->libpysal) (2.2.3)

Requirement already satisfied: certifi>=2017.4.17 in c:\users\amarachi\anaconda3\lib\site-packages (from requests>=2.27->libpysal) (2025.1.31)

Requirement already satisfied: joblib>=1.2.0 in c:\users\amarachi\anaconda3\lib\site-packages (from scikit-learn>=1.2->esda) (1.4.2)

Requirement already satisfied: threadpoolctl>=3.1.0 in c:\users\amarachi\anaconda3\lib\site-packages (from scikit-learn>=1.2->esda) (3.5.0)

Requirement already satisfied: six>=1.5 in c:\users\amarachi\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas>1.5->esda) (1.16.0)

In [80]:

```
import pandas as pd
import numpy as np
from sklearn.ensemble import IsolationForest
from esda.moran import Moran_Local
```

```

fromesda.getisord import G_Local # Use G_Local instead of G
from libpsyal.weights import DistanceBand
from scipy.stats import zscore

# Load the dataset
df = pd.read_csv("clustered_abia.csv")

# Ensure Latitude and Longitude exist
if "Latitude" not in df.columns or "Longitude" not in df.columns:
    raise ValueError("Latitude and Longitude columns are required!")

# Ensure total_votes column exists
vote_columns = ["APC", "LP", "PDP", "NNPP"]
df["total_votes"] = df[vote_columns].sum(axis=1)

# Create spatial weight matrix (Increase threshold to avoid zero neighbours)
coords = df[["Latitude", "Longitude"]].values
w = DistanceBand(coords, threshold=5, silence_warnings=True) # Increased to 5km

# 1. Local Moran's I Score Calculation (Handle NaN values)
moran_local = Moran_Local(df["total_votes"], w)
df["Moran_I_Score"] = zscore(np.nan_to_num(moran_local.Is)) # Replace NaNs with 0

# 2. Getis-Ord Gi* Score Calculation
df["total_votes"] = df["total_votes"].astype(float) # Ensure it's float
df["total_votes"] = df["total_votes"].fillna(0) # Handle NaNs

getis = G_Local(df["total_votes"], w, n_jobs=1) # Use G_Local without parallel pro
df["Getis_Ord_Gi_Score"] = zscore(np.nan_to_num(getis.Zs)) # Handle NaNs

# 3. Isolation Forest Outlier Score Calculation
iso_forest = IsolationForest(contamination=0.05, random_state=42)
df["Isolation_Forest_Score"] = iso_forest.fit(df[["Latitude", "Longitude", "total_v
    df[["Latitude", "Longitude", "total_votes"]]
)

# 4. Final Weighted Outlier Score
df["Final_Outlier_Score"] = (
    ((df["Moran_I_Score"] * 0.4) + (df["Getis_Ord_Gi_Score"] * 0.4)) *
    (1 + (df["Isolation_Forest_Score"] * 0.5)) # Boosts flagged outliers
)

# Save the updated dataset
df.to_excel("spatial_abia.xlsx", index=False)
print("Outlier score calculation complete! Results saved to spatial_abia.csv")

```

Outlier score calculation complete! Results saved to spatial_abia.csv

In [70]:

```

import pandas as pd
import matplotlib.pyplot as plt
file_path = "Historical Data.xlsx"
df = pd.read_excel(file_path)
df['Year'] = df['Year'].astype(str)

```

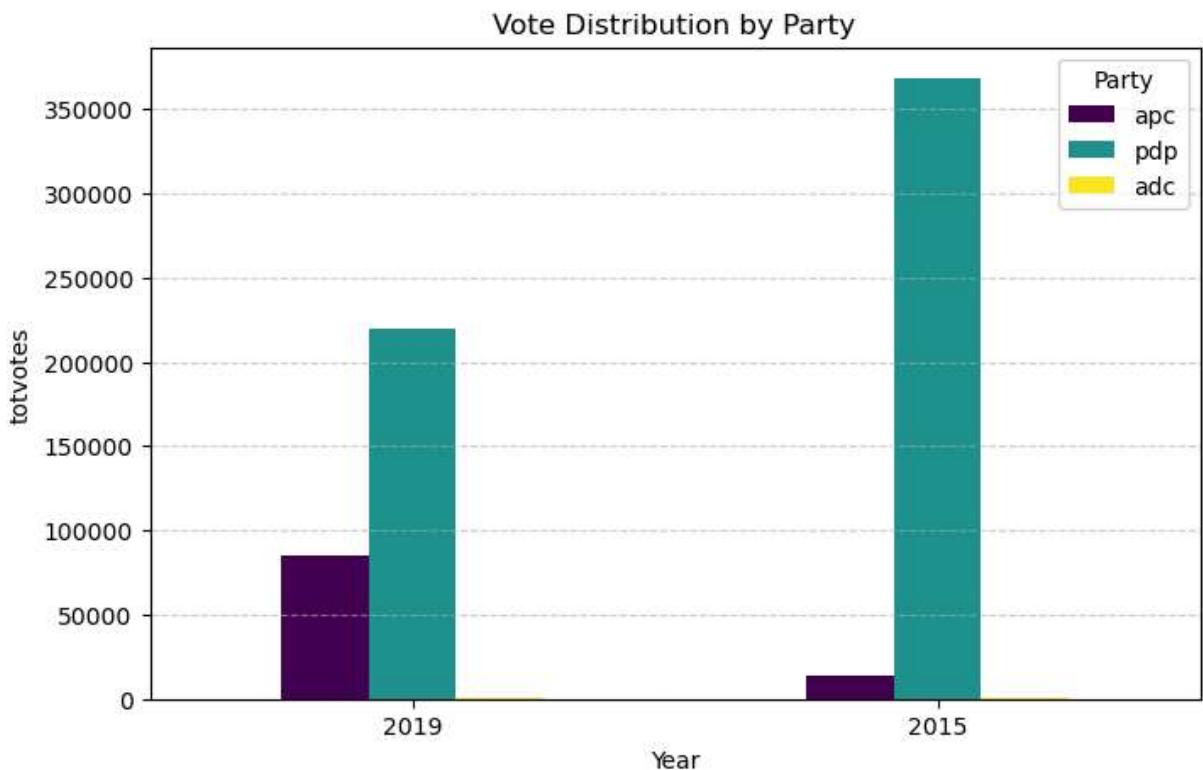
In [72]:

```

# 🎨 Bar Chart - Vote Distribution
df.set_index("Year")[["apc", "pdp", "adc"]].plot(kind="bar", figsize=(8, 5), colorm

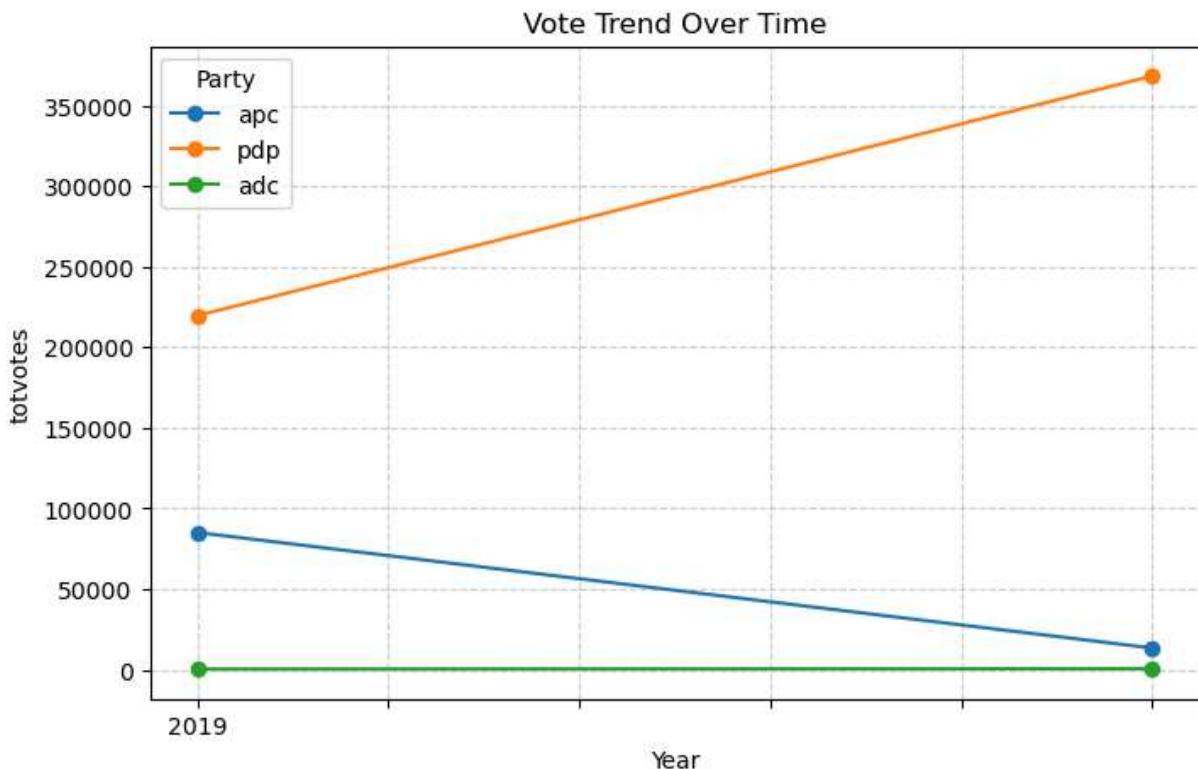
```

```
plt.title("Vote Distribution by Party")
plt.xlabel("Year")
plt.ylabel("totvotes")
plt.xticks(rotation=0)
plt.legend(title="Party")
plt.grid(axis="y", linestyle="--", alpha=0.6)
plt.show()
```



In [74]:

```
# ♦ Line Chart - Vote Trend Over Time
df.set_index("Year") [["apc", "pdp", "adc"]].plot(marker="o", figsize=(8, 5))
plt.title("Vote Trend Over Time")
plt.xlabel("Year")
plt.ylabel("totvotes")
plt.legend(title="Party")
plt.grid(axis="both", linestyle="--", alpha=0.6)
plt.show()
```



```
In [76]: import pandas as pd

# Load the datasets
election_df = pd.read_excel ("Historical data.xlsx") # Election results for past 2
population_df = pd.read_excel ("Abia population Census.xlsx") # Census data for LG
```

```
# Standardize LGA names (if needed)
population_df["LGA"] = population_df["LGA"].str.strip().str.lower()

print("\nDemographic Data:")
print(population_df.head())
```

Demographic Data:

	LGA	Population Census (1991)	Population Census (2006)	\
0	aba north	86331	106844	
1	aba south	413852	427421	
2	arochukwu	97800	169339	
3	bende	132271	192621	
4	ikwuano	52214	137897	

	Population Projection (2022)	Population Projection (2025)
0	155600	159842
1	622400	631478
2	246600	249763
3	280500	289527
4	200800	204938

```
In [78]: # Preview datasets
print("Election Data:")
print(election_df.head())
```

Election Data:

	state	Year	apc	pdp	adc	regvoters	totvotes	winner
0	Abia	2019	85058	219698	336	1932892	323291	PDP
1	Abia	2015	13394	368303	569	391045	1349134	PDP