DataFrames

CSE416-Section 1

This slide set contains slides selected from lecture material developed for ISE369 – Introduction to Political Informatics

1

© Robert F. Kelly, 2021-2026

Example - Download Data

- Create a DataFrame from a download file
- Access the data for all candidates in the 2021-2022 Congressional election period to determine the leading candidate fund-raisers
- Data is available at OpenFEC for download at https://www.fec.gov/data/browse-data/?tab=bulk-data
- We only use the first 20 rows for this example

2

© Robert F. Kelly, 2021-2026

Example - Data

- The file we downloaded from OpenFEC contains contribution data for all the candidates in the 2021-2022 election cycle

H2AK00200|CONSTANT, CHRISTOPHER|C|1|DEM|164637.9|0|164637.9|0|0|0|614.85|0|0|0|0|143180.09|158023.05|AK|00|||||1000|5000|12/31/2022|8300|0
H2AK01158|PELTOLA, MARY|I|1|DEM|7751293.39|186868.19|7060033.09|0|0|691260.3|25|0|0|0|0|0|7149826.02|AK|00|||||384020.59|10000|12/31/2022|136657.7|3912.66
H2AK01240|WOOL, ADAM L|O|1|DEM|16217.07|0|16217.07|0|0|0|1100|0|0|0|0|0|15117|AK|00|||||0|0|07/15/2022|0|0
H2AK00218|REVAK, JOSHUA CARL|O|2|REP|121841|0|121841|0|0|0|0|0|0|0|0|0|116666|AK|00|||||5000|0|09/16/2022|14600|0
H2AK00226|PALIN, SARAH|O|2|REP|1971160.93|112963.43|1924781.35|0|0|46379.58|0|0|0|0|0|2525.05|1770697.9|AK|00|||||81305|0|12/31/2022|43128.37|1000

Notice that data fields are separated by a "|"

Lines above are wrapped, but each record begins on a new line

3

© Robert F. Kelly, 2021-2026

Access an OpenFEC File

Widely used alias for a pandas DataFrame

```
import pandas as pd
import os
os.chdir('G:\\My Drive\\Courses\\ISE369\\Data\\DataFrameLesson')
df = pd.read_csv('weball22-first20.txt', sep='|', header=None)
print(df)
```

Uses the pandas read_csv function to read the file into a pandas DataFrame

Specifies the separator in the csv file

Indicates that the data file did not have headers

None is a literal in Python indicating nothing is assigned

	0	1	...	28	29
0	H2AK00200	CONSTANT, CHRISTOPHER	...	8300.00	0.00
1	H2AK01158	PELTOLA, MARY	...	136657.70	3912.66
2	H2AK01240	WOOL, ADAM L	...	0.00	0.00
3	H2AK00218	REVAK, JOSHUA CARL	...	14600.00	0.00
4	H2AK00226	PALIN, SARAH	...	43128.37	1000.00

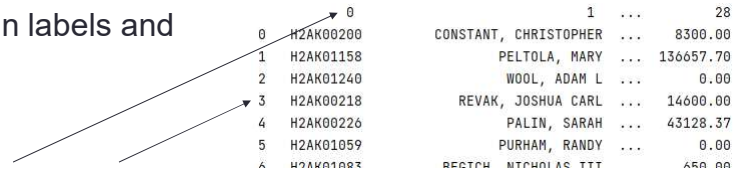
Notice that 1) the data contains more columns than we care about and 2) the columns and row have number headers

4

© Robert F. Kelly, 2021-2026

DataFrame

- A 2-dimensional data structure
- Main data structure in the pandas library
- Think of it as a spreadsheet that you can easily manipulate in Python
- Contains
  - Row and column labels and
  - Data content



	0	1	...	28
0	H2AK00200	CONSTANT, CHRISTOPHER	...	8300.00
1	H2AK01158	PELTOLA, MARY	...	136657.70
2	H2AK01240	WOOL, ADAM L	...	0.00
3	H2AK00218	REVAK, JOSHUA CARL	...	14600.00
4	H2AK00226	PALIN, SARAH	...	43128.37
5	H2AK01059	PURHAM, RANDY	...	0.00

The data file had no headers, so pandas just assigns row and column numbers to the values

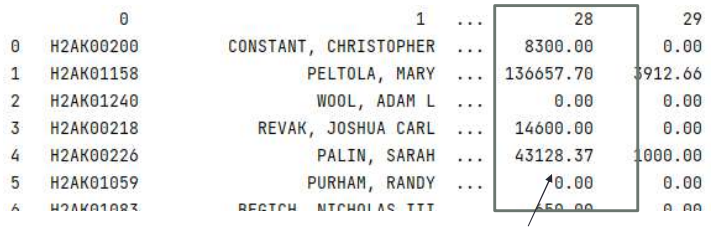
5

© Robert F. Kelly, 2021-2026

Python Library - Pandas

- The DataFrame object contains many Series objects
- A Series object contains all or part of a row or column, along with the heading for the elements of the row or column

Knowing the object structure helps you to understand error messages



	0	1	...	28	29
0	H2AK00200	CONSTANT, CHRISTOPHER	...	8300.00	0.00
1	H2AK01158	PELTOLA, MARY	...	136657.70	3912.66
2	H2AK01240	WOOL, ADAM L	...	0.00	0.00
3	H2AK00218	REVAK, JOSHUA CARL	...	14600.00	0.00
4	H2AK00226	PALIN, SARAH	...	43128.37	1000.00
5	H2AK01059	PURHAM, RANDY	...	0.00	0.00

Series object

6

© Robert F. Kelly, 2021-2026

File Structure

- Layout of the All Candidates data is available at https://www.fec.gov/campaign-finance-data/all-candidates-file-description/

All candidates file format

Column name	Field name	Position	Null	Data type	Description
CAND_ID	Candidate identification	1	N	VARCHAR2 (9)	
CAND_NAME	Candidate name	2	Y	VARCHAR2(200)	
CAND_ICI	Incumbent challenger status	3	Y	VARCHAR2(1)	
PTY_CD	Party code	4	Y	VARCHAR2(1)	

7

© Robert F. Kelly, 2021-2026

Let's Explore the DataFrame We Created

- We can look for DataFrame attributes at https://pandas.pydata.org/docs/reference/frame.html

```
print('df index \n', df.index)
print('DataFrame size = ', df.size)
print(df.iat[3,1])
```

These are properties of the DataFrame object that we access with df.*propertyName*

Line feed

Indexing operator

```
df index
 RangeIndex(start=0, stop=20, step=1)
DataFrame size =  600
REVAK, JOSHUA CARL
```

20 rows and 30 elements per row

Element at [3,1]

8

© Robert F. Kelly, 2021-2026

Python

- Bracket operator (subscript operator)
- Used to access the elements in a string, list, etc.
- Indexed left to right, starting at 0

```
print(df.iat[3,1])
```

More on iat in a little while

	0	1	2
0	H2AK00200	CONSTANT, CHRISTOPHER	C
1	H2AK01158	PELTOLA, MARY	I
2	H2AK01240	WOOL, ADAM L	O
3	H2AK00218	REVAK, JOSHUA CARL	O
4	H2AK00226	PALIN, SARAH	O
5	H2AK01059	BURHAM, RANDY	C

9

© Robert F. Kelly, 2021-2026

Remove Columns from a DataFrame

- Usually, data needs to be cleaned up before processing
- Two clean-up tasks we do with the candidate contribution data are
  - Remove any columns that are not needed
  - Relabel the remaining columns
- For now, we only need
  - Candidate id
  - Candidate name
  - Incumbent / challenger status
  - Party code
  - Party affiliation
  - Total receipts
  - Candidate state
  - Primary election status
  - General election status

10

© Robert F. Kelly, 2021-2026

Remove Unneeded Columns

- We can drop columns we don't need, either in-place or by assigning the DataFrame to a new DataFrame

```
df1=df.drop(columns=[3,6,7,8,9,10,11,12,13,14,15,16,17,19,20,21,22,24,25,26,27,28,29])
```

Creates a new DataFrame, keeping the original one intact

	0	1	2	4	5	18	23
0	H2AK00200	CONSTANT, CHRISTOPHER	C	DEM	164637.90	AK	NaN
1	H2AK01158	PELTOLA, MARY	I	DEM	7751293.39	AK	NaN
2	H2AK01240	WOOL, ADAM L	0	DEM	16217.07	AK	NaN
3	H2AK00218	REVAK, JOSHUA CARL	0	REP	121841.00	AK	NaN
4	H2AK00226	PALIN, SARAH	0	REP	1971160.93	AK	NaN
5	H2AK01059	DURHAM, RANDY	C	REP	1548.51	AK	NaN

11

© Robert F. Kelly, 2021-2026

Print All the Columns

- If we saw the truncate view but would like to see all the columns, we can allow a row to print on multiple lines

```
pd.set_option('max_columns', None)
print(df)
```

```
 0          1 2  3   4       5      6 \
0 H8AK00132     SHEIN, DIMITRI  C  1 DEM      0.00     0.00
1 H6AK00045     YOUNG, DONALD E I  2 REP  1950289.86 138304.94
2 H8AK01031 NELSON, THOMAS JOHN  C  2 REP       0.00      0.00
3 H8AK00140      GALVIN, ALYSE  C  3 IND  5253251.54  60024.76

       7  8       9       10     11 12 13     14 15 16 \
0    367.52  0    367.52     0.00    0.00  0  0 367.52  0  0
1 1817836.79  0 116720.12 249173.19    0.00  0  0   0.00  0  0
2    466.51  0    466.51     0.00    0.00  0  0   0.00  0  0
3 5162902.93  0   6245.09  96593.70 4371.82  0  0   0.00  0  0

       17  18  19  20  21  22  23  24       25    26       27 \
0      0.00  AK   0 NaN NaN NaN NaN NaN      0.00     0 09/30/2019
1  963416.04  AK   0 NaN NaN NaN NaN NaN  839094.63   500 12/31/2020
2      0.00  AK   0 NaN NaN NaN NaN NaN      0.00     0 03/31/2019
3 4796137.47  AK   0 NaN NaN NaN NaN NaN  340486.99 10025 12/31/2020
```

In truncate view, pandas will detect the width of the terminal and print a center truncated object that fits the screen width

12

© Robert F. Kelly, 2021-2026

Python

- Comments on the Python code on a previous slide

```
df1=df.drop(columns=[3,6,7,8,9,10,11,12,13,14,15,16,17,19,20,21,22,24,25,26,27,28,29])
```

- A Python list
  - Can contain items of a different data type
  - Contains values, separated by commas (,) and enclosed within square brackets ([])
  - Is changable

13

© Robert F. Kelly, 2021-2026

Remove Unneeded Columns

- We can also drop columns by selecting the columns we would like to retain and assigning the columns to a different DataFrame

```
df2=df.loc[:,[0,1,2,4,5,18,23]]
print(df2)
```

	0	1	2	4	5	18
0	H8AK00132	SHEIN, DIMITRI	C	DEM	0.00	AK
1	H6AK00045	YOUNG, DONALD E	I	REP	1950289.86	AK
2	H8AK01031	NELSON, THOMAS JOHN	C	REP	0.00	AK
3	H8AK00140	GALVIN, ALYSE	C	IND	5253251.54	AK

14

© Robert F. Kelly, 2021-2026

Python

Selects all the rows

- Consider the Python statement in a previous slide

```
df2=df.loc[:,[0,1,2,4,5,18,23]]
```

- Python slice operation
  - Operates on strings, lists, and tuples
  - Selects a portion of the string or list
  - Indices, start, stop, and step (stops one before stop index)
  - Syntax – [*start* : *stop* : *step*]
  - Must include at least one colon (:)

15

© Robert F. Kelly, 2021-2026

Pandas loc

- Consider the Python statement in a previous slide

```
df2=df.loc[:,[0,1,2,4,5,18,23]]
```

All rows

List of column labels

- Allows you to access a group of rows and columns from a DataFrame using either access labels (loc) or integers (iloc)
- Selection can use a list of labels, a slice, or a Boolean array

16

© Robert F. Kelly, 2021-2026

Accessing Data in a DataFrame

Multiple approaches differ in what you access and what the property returns

- iloc
  - Integer based indexing for selection by position
  - Allowed inputs include an integer, a list or array of integers, and a slice object
- loc
  - Access a group of rows and columns by labels
  - Allowed inputs include a single label, a list or array of labels, and a slice object with labels
- iat
  - accesses a single value for a row/column by integer position
  - Use iat if you only need to get or set a single value
- at
  - Similar to iat, but access a single value by label pair

17

© Robert F. Kelly, 2021-2026

Rename the Columns

- We can rename the columns in the DataFrame

```
df1 = df1.rename(columns = {0:"ID"})
df1 = df1.rename(columns = {1:"Name"})
df1 = df1.rename(columns = {2:"I/C"})
df1 = df1.rename(columns = {4:"Pty"})
df1 = df1.rename(columns = {5:"$$"})
df1 = df1.rename(columns = {18:"State"})
df1 = df1.rename(columns = {23:"Status"})
print(df1)
```

Python dictionary syntax

NaN stands for Not a Number in Python. Often used to represent missing values

	ID	Name	I/C	Pty	$$	State	Status
0	H2AK00200	CONSTANT, CHRISTOPHER	C	DEM	164637.90	AK	NaN
1	H2AK01158	PELTOLA, MARY	I	DEM	7751293.39	AK	NaN
2	H2AK01240	WOOL, ADAM L	O	DEM	16217.07	AK	NaN
3	H2AK00218	REVAK, JOSHUA CARL	O	REP	121841.00	AK	NaN
4	H2AK00226	PALIN, SARAH	O	REP	1971160.93	AK	NaN
5	H2AK01059	DUBHAM, RANDY	C	REP	1548.51	AK	NaN

18

© Robert F. Kelly, 2021-2026

Sort the DataFrame

- We can sort the rows in the DataFrame

```
dfsort = df1.sort_values('$$', ascending=False)
print(dfsort)
```

	ID	Name	I/C	Pty	$$	State	Status
1	H2AK01158	PELTOLA, MARY	I	DEM	7751293.39	AK	NaN
4	H2AK00226	PALIN, SARAH	O	REP	1971160.93	AK	NaN
6	H2AK01083	BEGICH, NICHOLAS III	C	REP	1640060.27	AK	NaN
17	H0AL01055	CARL, JERRY LEE, JR	I	REP	1313718.57	AL	NaN

But numbers in the $$ display are difficult to read

19

© Robert F. Kelly, 2021-2026

Format

- We would like to
  - Change the format of the numbers in the $$ column and
  - Print the top 25 rows (if our input file contained more than 25 rows)

Format method of the string object

```
def format(x):
    return "${:.2f}M".format(x/1000000)
dfsort['$$'] = dfsort['$$'].apply(format)
print("Top 25")
print(dfsort.head(25))
```

Positional argument

Replacement fields are enclosed in curly braces. Text outside the curly braces are included, as is. Replacement fields are substituted for by the arguments to the format() method

Format specifier
(https://www.python.org/dev/peps/pep-3101/)

20

© Robert F. Kelly, 2021-2026

Example

- For the downloaded full list of candidate's campaign contributions for 2021-2022
- Display the top 25 donations in excess of $2M

```
Top 25
            ID                    Name  I/C  Pty       $$ State  Status
3389  P40004541          MERCER JR, LEE  NaN  DEM  $384.00M    00     NaN
3635  S0GA00559        WARNOCK, RAPHAEL    I  DEM  $206.59M    GA     NaN
3392  P40007296             MERCER, LEE  NaN  DEM  $128.00M    00     NaN
3526  S0AZ00350             KELLY, MARK    I  DEM   $92.77M    AZ     NaN
682   H2FL08063   DEMINGS, VALDEZ 'VAL'    I  DEM   $81.09M    FL     NaN
3608  S2FL00631            DEMINGS, VAL    C  DEM   $81.09M    FL     NaN
4015  S6PA00274     FETTERMAN, JOHN KARL    O  DEM   $76.34M    PA     NaN
3648  S2GA00225     WALKER, HERSCHEL MR     C  REP   $73.75M    GA     NaN
```

21

© Robert F. Kelly, 2021-2026

Example - Solution

- DataFrame df1 contains the downloaded candidate data with the columns renamed (as in previous slides)

```
dfsort = df1.sort_values('$$', ascending=False)
def format(x):
    return "${:.2f}M".format(x/1000000)
dfsort['$$'] = dfsort['$$'].apply(format)
print("Top 25")
print(dfsort.head(25))
```

22

© Robert F. Kelly, 2021-2026

Reading

- Wiki – Shapefile
  en.wikipedia.org/wiki/Shapefile
- GeoPandas data structures
  geopandas.org/en/stable/docs/user_guide/data_structures.html
- Geographic coordinate systems
  en.wikipedia.org/wiki/Geographic_coordinate_system
- Pyplot tutorial
  matplotlib.org/stable/tutorials/introductory/pyplot.html
- Data selection in a DataFrame
  https://www.shanelynn.ie/pandas-iloc-loc-select-rows-and-columns-dataframe/
- Adding columns to a DataFrame
  https://www.interviewkickstart.com/learn/adding-new-column-to-existing-dataframe-in-pandas

23

© Robert F. Kelly, 2021-2026

Reference

- Geopandas documentation
  https://geopandas.org/en/stable/docs.html
- Redistricting Data Hub
  https://redistrictingdatahub.org/data/
  (register for your account)
- Shapely User Manual
  https://shapely.readthedocs.io/en/stable/manual.html
- World Geodetic System
  https://en.wikipedia.org/wiki/World_Geodetic_System
- Installing packages
  https://www.jetbrains.com/help/pycharm/installing-uninstalling-and-upgrading-packages.html
- Matplotlib
  https://matplotlib.org/stable/api/index.html
- Color maps
  https://matplotlib.org/stable/tutorials/colors/colormaps.html

24

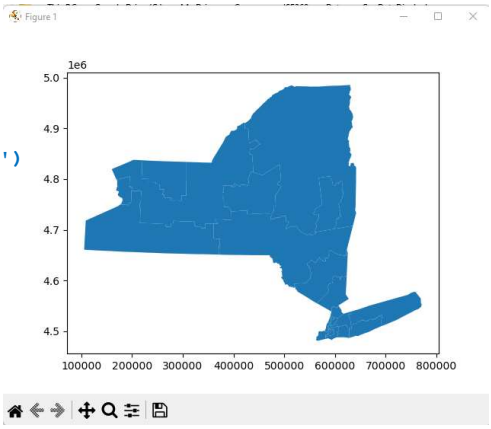
© Robert F. Kelly, 2021-2026

Access the NY Congressional Shapefile

- File contains the boundaries of the 2022 New York Congressional districts as a Shapefile

```
import geopandas
import os
import matplotlib.pyplot as plt
os.chdir('G:\\My Drive\\Courses\\ISE369\\Data'
         +'\\GeoDataDisplayLesson')
ny_cd = geopandas.read_file('ny_cong_adopted_2022.zip')
ny_cd.plot()
plt.show()
```

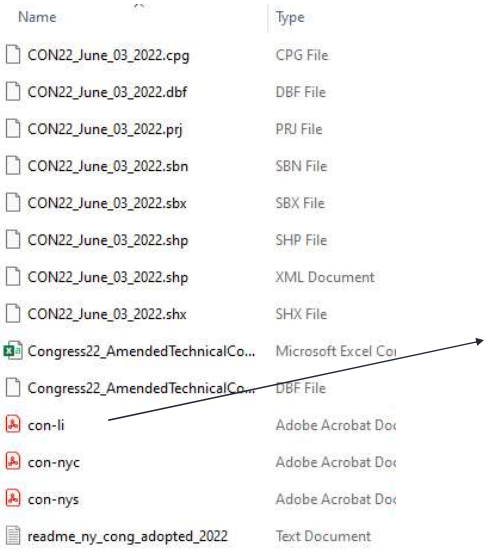
There's a lot going on in this code block, so let's review it

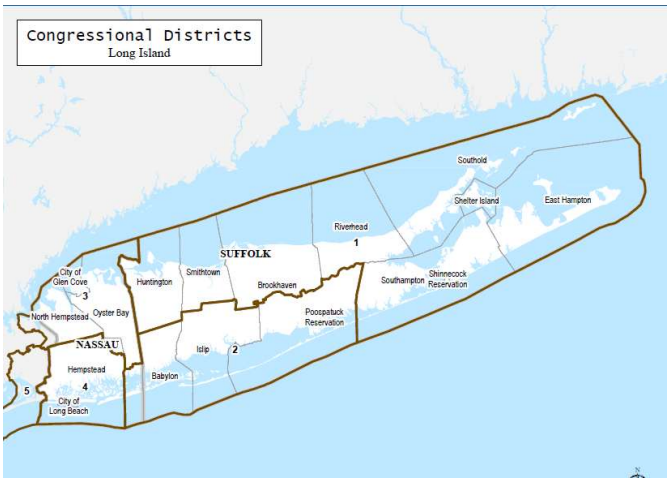


25

© Robert F. Kelly, 2021-2026

What's in the Zipped Shapefile File?



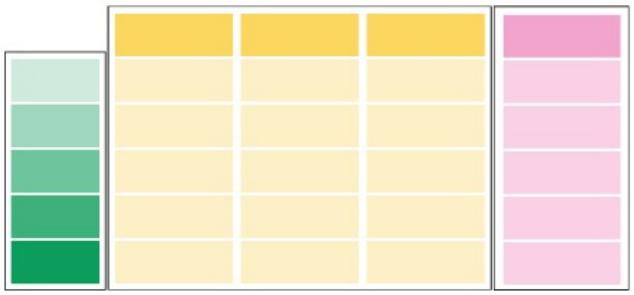


26

© Robert F. Kelly, 2021-2026

Geopandas

- Open-source project that adds support to pandas for geographic data objects
- Extends the pandas DataFrame to add geometric operations
- Built on top of the Shapely geometric library for geometric operations



GeoDataFrame (Source: geopandas.org)

A GeoDataFrame is a DataFrame with an added geometry column

© Robert F. Kelly, 2021-2026

Create a GeoDataFrame

```
...
ny_cd =
geopandas.read_file('ny_cong_adopted_2022.zip')
print(ny_cd)
```

read_file method has other parameters

Directory containing a Shapefile

FIPS Code

State FIPS Codes

Name	Postal Code	FIPS
Alabama	AL	01
Alaska	AK	02
Arizona	AZ	04
Arkansas	AR	05
California	CA	06
Colorado	CO	08

```
22        36  ...  POLYGON ((-73.97112 40.81631, -73.97069 40.816...
23        36  ...  POLYGON ((-79.07537 43.08135, -79.07400 43.083...
24        36  ...  POLYGON ((-76.47265 42.00007, -76.47213 42.000...
25        36  ...  POLYGON ((-73.98784 40.65982, -73.98750 40.660...
26        36  ...  POLYGON ((-75.55548 42.12170, -75.55268 42.121...

[27 rows x 338 columns]
```

ny_cd is a GeoDataFrame

Row #

FIPS codes are listed in https://en.wikipedia.org/wiki/Federal_Information_Processing_Standard_state_code

© Robert F. Kelly, 2021-2026

GeoDataFrame

- Part of the geopandas library
- A pandas.DataFrame that has a column with geometry
- A GeoDataFrame always has one GeoSeries column (geometry) that holds a special status
- Spatial methods applied to a GeoDataFrame will operate on the "geometry" column
- Geometry column accessed through gdf.geometry
- The name of the geometry column accessed through *gdf*.geometry.name

Geometry objects are Shapely objects

29

© Robert F. Kelly, 2021-2026

GeoDataFrame

- A GeoDataFrame extends a pandas DataFrame, so you need to import pandas to use DataFrame options

```
import geopandas
import os
import pandas as pd
os.chdir(...)
ny_cd =
geopandas.read_file('ny_cong_adopted_2022.zip')
pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)
print(ny_cd)
```

	BASENAME	NAME	FUNCSTAT	POP100
0	1	Congressional District 1	N	740319
1	7	Congressional District 7	N	762833
2	27	Congressional District 27	N	720092
3	17	Congressional District 17	N	763751
4	14	Congressional District 14	N	750025
5	6	Congressional District 6	N	769247
6	5	Congressional District 5	N	778780
7	24	Congressional District 24	N	717307
8	16	Congressional District 16	N	770401
9	11	Congressional District 11	N	766236

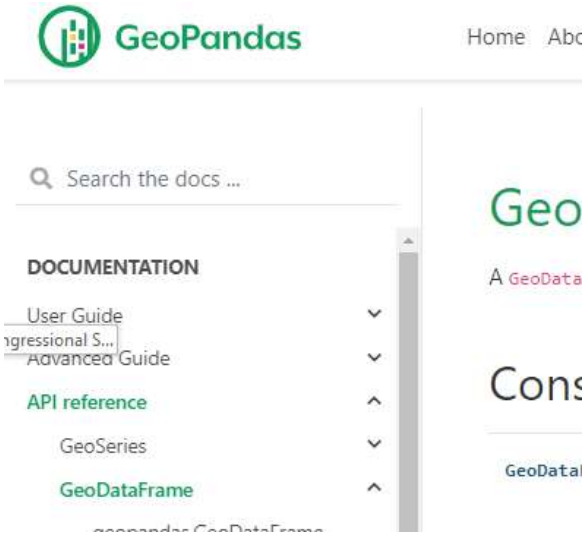
Sets the DataFrame options

30

© Robert F. Kelly, 2021-2026

How Do I Find GeoPandas Properties/Methods?

- API Reference geopandas.org/en/stable/docs/reference.html
- Methods (e.g., GeoDataFrame) geopandas.org/en/stable/docs/reference/geodataframe.html



31

© Robert F. Kelly, 2021-2026

What are the Columns in this GeoDataFrame?

```
...
for col_name in ny_cd.columns:
    print(col_name)
```

```
OBJECTID
Shape_Leng
Shape_Area
DISTRICT
geometry
```

Let's look at this column more carefully

32

© Robert F. Kelly, 2021-2026

Geometry Column in a GeoDataFrame

```
...
geom_cd = ny_cd.iat[0,4]
print(geom_cd)
```

iat accesses a single DataFrame element by integer position, so this accesses the geometry of Congressional District # 1

```
POLYGON ((742169.5257000001 4532898.369100001, 736334.2504000003 4529914.095799999, 73
```

Note that this GeoDataFrame cell contains a polygon

33

© Robert F. Kelly, 2021-2026

GeoPandas Data Structures

- GeoPandas implements 2 main data structures
  - GeoSeries – subclass of pandas.Series
  - GeoDataFrame – subclass of pandas.DataFrame
- GeoPandas shapes, which are Shapely objects
  - Points / Multi-Points
  - Lines / Multi-Lines
  - Polygons / Multi-Polygons

Each entry in a GeoSeries vector is a set of shapes

islands

For example, a political district might be defined as a multi-polygon or multiple polygons if there are islands in the district

Shapely operations assume a cartesian coordinate system

One NY district (10) contains multiple polygons

34

© Robert F. Kelly, 2021-2026

Shapely

Geometric operations might be helpful when you need to verify state geometry (e.g., towns)

- A Python package for computational geometry
- Fundamental geometric objects
  - Point – e.g., Point class
  - Curve – e.g., LineString class
  - Surface – e.g., Polygon class

Shapely assumes all features exist in the same Cartesian plane

- Geometric relationships
  - Contains
  - Intersects
  - Overlaps
  - Touches
- Geometric operations
  - Buffer
  - Convex hull
  - intersection

35

© Robert F. Kelly, 2021-2026

Shapely Attributes

- Area
- Bounds
- Length
- Minimum_clearance

36

© Robert F. Kelly, 2021-2026

District Geometry

```
...
geom_cd = ny_cd.iat[0,4]
print(geom_cd)
```

```
POLYGON ((742169.5257000001 4532898.369100001, 736334.2504000003 4529914.095799999, 73
```

A polygon is specified as a series of points

What are these coordinates?

37

© Robert F. Kelly, 2021-2026

How Do We Add a Column to a DataFrame?

- Frequently, data you need is contained in different data sources, for example
  - Geometry
  - Provisional voting
  - Income
  - Demographic data
- Approach options
  - Declaring a new list as a column – insert data as the last column
  - Using DataFrame.insert() – insert a column anywhere

38

© Robert F. Kelly, 2021-2026

Example …

columns returns the column labels

```
…
ny_cd = gpd.read_file('ny_cong_adopted_2022.zip')
print('\n' + 'DataFrame column names')
print(ny_cd.columns.tolist())
ny_cd['values'] = range(28)
print('\n' + 'DataFrame column names after insert')
print(ny_cd.columns.tolist())
```

Generates 28 integers, from 0 to 27

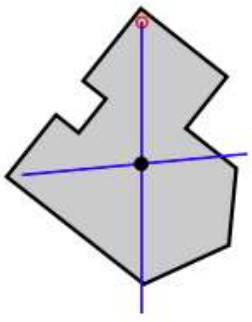
```
DataFrame column names
['OBJECTID', 'Shape_Leng', 'Shape_Area', 'DISTRICT', 'geometry']

DataFrame column names after insert
['OBJECTID', 'Shape_Leng', 'Shape_Area', 'DISTRICT', 'geometry', 'values']
```

39

© Robert F. Kelly, 2021-2026

Useful GeoDataFrame Geometric Operations

- Area
- Polygon boundary
- Centroid
- Distance
- Convex hull
- Simplify
- Union
- Buffer



Polygon Centroids (source: Wikipedia)

40

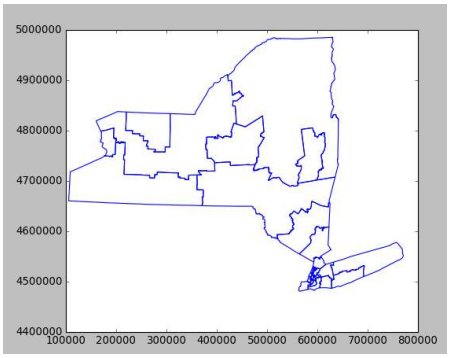
© Robert F. Kelly, 2021-2026

Show Boundaries

- But in some of the styles, the image does not clearly show the district boundaries since the boundary blends with the image background
- We can show just the boundaries with the following statements

```
...
ny_cd =
geopandas.read_file('ny_cong_adopted_2022.zip')
plt.style.use('classic')
ny_cd.boundary.plot()
plt.show()
```

The boundary property returns a GeoSeries, which can then be plotted



41