Week 4 Exploratory Lab

February 1, 2021

Exploring COVID Case Data and County GEOJSON Data

In this file I'm going to do a little more exploring of my datasets and try and get a better sense of how I want to vistually portray all my different data sources.

First thing to do is load the libraries…

0.0.1 Load Libraries

I think I'm going to load pandas, geopandas, matplot.lib, osmnx, and contextily, which is all the libraries we've worked with so far, but I want to have some flexibility in how I conduct my analysis moving forward, so I want to have all the tools we've learned thus far in the class - mapping, data visualization, creating isochrones if needed, etc.

[1]:
```
import pandas as pd
import geopandas as gpd
import matplotlib.pyplot as plt
import osmnx as ox
import contextily as ctx
```

Now that my libraries are loaded, I'm going to upload my 3 data sets: water debt, shapefile of CA counties, and COVID cases by county.

0.0.2 Upload Data Sets

Now, I will read each of my datasets: COVID data, county boundaries, and water debt.

[2]:
```
countycovid = pd.read_csv('Data/us_county_confirmed_cases.csv')

countyshape = gpd.read_file('Data/
 ↪united_states_california_administrative_boundaries_level6_counties_polygon.
 ↪geojson')

low_memory=False
```

```
/opt/conda/lib/python3.8/site-packages/IPython/core/interactiveshell.py:3071:
DtypeWarning: Columns (376) have mixed types.Specify dtype option on import or
set low_memory=False.
  has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
```

After a bit of trouble trying to get the shape file uploaded, my 3 datasets have been added! In addition, I spent a good portion of time trying to find a shapefile that also had county names (and not just the geometry) because I needed a way to match the county names with the polygons themselves. I do not love this shapefile (for reasons described below), but its the closest thing I could find to what I need, so it is a temporary fix.

0.1 Explore & Clean COVID Data

Before I try and map the COVID data, I am going to get a sense of what it looks like using .head

[3]:
```
countycovid.head()
```

[3]:
```
   COUNTY            NAME     County Name State  stateFIPS  POP70  HHD70  \
0    1001  Autauga County  Autauga County    AL          1  24457   6792
1    1003  Baldwin County  Baldwin County    AL          1  59132  17641
2    1005  Barbour County  Barbour County    AL          1  22484   6796
3    1007     Bibb County     Bibb County    AL          1  13812   4015
4    1009   Blount County   Blount County    AL          1  26844   8431

   POP80  HHD80  POP90  …  1/8/21  1/9/21  1/10/21  1/11/21  1/12/21  \
0  32266  10199  34236  …    4770    4847     4879     4902     4970
1  78213  26641  98277  …   15052   15202    15327    15417    15572
2  24685   8352  25418  …    1634    1648     1658     1663     1679
3  15680   5153  16589  …    2015    2038     2051     2060     2090
4  36456  12679  39247  …    5018    5047     5066     5080     5134

   1/13/21  1/14/21  1/15/21  1/16/21  1/17/21
0     4998     5075     5103     5154     5184
1    15701    15841    16002    16176    16251
2     1685     1696     1712     1723     1729
3     2109     2113     2130     2144     2151
4     5170     5219     5264     5292     5304

[5 rows x 377 columns]
```

Before I move on, I want to override the display settings so I can see all of the columns

[4]:
```
pd.set_option('display.max_columns', None)
countycovid.head()
```

[4]:
```
   COUNTY            NAME     County Name State  stateFIPS  POP70  HHD70  \
0    1001  Autauga County  Autauga County    AL          1  24457   6792
1    1003  Baldwin County  Baldwin County    AL          1  59132  17641
2    1005  Barbour County  Barbour County    AL          1  22484   6796
3    1007     Bibb County     Bibb County    AL          1  13812   4015
4    1009   Blount County   Blount County    AL          1  26844   8431

   POP80  HHD80  POP90  HHD90   POP00  HHD00   POP10  HHD10  1/22/20  1/23/20  \
```

0	32266	10199	34236	11830	43685	16007	54571	20221	0	0
1	78213	26641	98277	37041	140406	55330	182265	73180	0	0
2	24685	8352	25418	9217	29037	10409	27457	9820	0	0
3	15680	5153	16589	5750	20827	7421	22915	7953	0	0
4	36456	12679	39247	14644	51020	19264	57322	21578	0	0

	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20	1/29/20	1/30/20	1/31/20	\
0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	

	2/1/20	2/2/20	2/3/20	2/4/20	2/5/20	2/6/20	2/7/20	2/8/20	2/9/20	\
0	0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	

	2/10/20	2/11/20	2/12/20	2/13/20	2/14/20	2/15/20	2/16/20	2/17/20	\
0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	

	2/18/20	2/19/20	2/20/20	2/21/20	2/22/20	2/23/20	2/24/20	2/25/20	\
0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	

	2/26/20	2/27/20	2/28/20	2/29/20	3/1/20	3/2/20	3/3/20	3/4/20	3/5/20	\
0	0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	

	3/6/20	3/7/20	3/8/20	3/9/20	3/10/20	3/11/20	3/12/20	3/13/20	\
0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	

	3/14/20	3/15/20	3/16/20	3/17/20	3/18/20	3/19/20	3/20/20	3/21/20	\
0	0	0	0	0	0	0	0	0	
1	1	1	1	1	1	1	2	2	
2	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	

	3/22/20	3/23/20	3/24/20	3/25/20	3/26/20	3/27/20	3/28/20	3/29/20	\
0	0	0	1	4	6	6	6	6	
1	3	3	4	4	5	5	10	15	
2	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	
4	0	0	0	1	2	5	5	5	

	3/30/20	3/31/20	4/1/20	4/2/20	4/3/20	4/4/20	4/5/20	4/6/20	4/7/20	\
0	7	7	10	10	12	12	12	12	12	
1	18	19	23	25	28	29	34	38	42	
2	0	0	0	0	1	2	2	3	3	
3	2	3	3	4	4	4	7	7	8	
4	5	5	5	6	9	10	10	10	10	

	4/8/20	4/9/20	4/10/20	4/11/20	4/12/20	4/13/20	4/14/20	4/15/20	\
0	12	17	17	19	19	19	23	25	
1	49	59	59	66	71	78	87	98	
2	3	7	9	10	10	10	11	13	
3	9	11	11	13	16	17	17	19	
4	10	11	12	12	13	15	16	17	

	4/16/20	4/17/20	4/18/20	4/19/20	4/20/20	4/21/20	4/22/20	4/23/20	\
0	25	25	25	27	28	30	32	33	
1	102	103	109	114	117	123	132	143	
2	14	15	18	20	22	28	29	30	
3	23	23	26	28	32	32	33	33	
4	18	20	20	21	22	26	29	31	

	4/24/20	4/25/20	4/26/20	4/27/20	4/28/20	4/29/20	4/30/20	5/1/20	\
0	36	37	37	39	40	42	42	42	
1	147	154	161	168	171	173	174	175	
2	32	33	33	35	37	37	39	42	
3	34	35	38	42	42	42	42	42	
4	31	31	34	34	34	36	37	39	

	5/2/20	5/3/20	5/4/20	5/5/20	5/6/20	5/7/20	5/8/20	5/9/20	5/10/20	\
0	45	48	53	53	58	61	67	68	74	
1	181	187	188	189	196	205	208	216	222	
2	43	45	45	47	47	51	53	58	59	

3	42	42	42	43	43	44	44	45	46
4	40	40	40	40	42	44	44	44	44

	5/11/20	5/12/20	5/13/20	5/14/20	5/15/20	5/16/20	5/17/20	5/18/20	\
0	84	91	93	103	103	110	110	120	
1	224	227	231	243	244	254	254	260	
2	61	67	69	74	79	79	81	85	
3	46	46	46	46	49	50	50	50	
4	45	45	45	45	45	45	46	47	

	5/19/20	5/20/20	5/21/20	5/22/20	5/23/20	5/24/20	5/25/20	5/26/20	\
0	127	136	147	149	155	159	173	189	
1	262	270	270	271	273	274	276	277	
2	90	96	100	104	105	110	116	122	
3	51	52	52	55	58	59	62	66	
4	47	47	48	49	49	49	49	51	

	5/27/20	5/28/20	5/29/20	5/30/20	5/31/20	6/1/20	6/2/20	6/3/20	\
0	192	205	212	216	220	233	238	239	
1	281	281	282	283	288	292	292	292	
2	130	132	147	150	164	172	175	177	
3	71	71	71	72	75	76	76	76	
4	53	58	60	61	62	63	63	63	

	6/4/20	6/5/20	6/6/20	6/7/20	6/8/20	6/9/20	6/10/20	6/11/20	6/12/20	\
0	241	248	259	265	272	282	295	312	323	
1	293	296	304	313	320	325	331	343	353	
2	177	183	190	193	197	199	208	214	221	
3	76	76	77	77	79	85	89	93	97	
4	63	64	70	72	73	75	79	87	95	

	6/13/20	6/14/20	6/15/20	6/16/20	6/17/20	6/18/20	6/19/20	6/20/20	\
0	331	357	368	373	375	400	411	431	
1	361	364	383	389	392	401	413	420	
2	226	234	238	245	251	263	266	272	
3	100	104	111	116	118	121	126	126	
4	102	110	116	121	123	130	139	143	

	6/21/20	6/22/20	6/23/20	6/24/20	6/25/20	6/26/20	6/27/20	6/28/20	\
0	434	442	453	469	479	488	498	503	
1	430	437	450	464	477	515	555	575	
2	272	277	280	288	305	312	317	317	
3	127	129	135	141	149	153	161	162	
4	149	153	159	168	176	184	188	189	

	6/29/20	6/30/20	7/1/20	7/2/20	7/3/20	7/4/20	7/5/20	7/6/20	7/7/20	\
0	527	537	553	561	568	591	615	618	644	

1	643	680	703	751	845	863	881	911	997
2	322	325	326	335	348	350	352	356	360
3	165	170	174	179	189	190	193	197	199
4	199	208	218	222	230	234	239	247	255

	7/8/20	7/9/20	7/10/20	7/11/20	7/12/20	7/13/20	7/14/20	7/15/20	\
0	651	661	670	684	706	728	746	756	
1	1056	1131	1187	1224	1294	1359	1414	1518	
2	366	371	381	398	403	413	428	441	
3	201	211	218	224	228	231	236	242	
4	262	282	292	307	331	350	366	389	

	7/16/20	7/17/20	7/18/20	7/19/20	7/20/20	7/21/20	7/22/20	7/23/20	\
0	780	789	827	842	857	865	886	905	
1	1599	1689	1819	1937	2013	2102	2196	2461	
2	459	463	483	495	503	514	518	534	
3	247	255	264	269	279	283	287	289	
4	424	440	458	482	507	524	547	585	

	7/24/20	7/25/20	7/26/20	7/27/20	7/28/20	7/29/20	7/30/20	7/31/20	\
0	921	932	942	965	974	974	1002	1015	
1	2513	2662	2708	2770	2835	2835	3028	3101	
2	539	552	562	569	575	575	585	598	
3	303	318	324	334	337	338	352	363	
4	615	637	646	669	675	675	731	767	

	8/1/20	8/2/20	8/3/20	8/4/20	8/5/20	8/6/20	8/7/20	8/8/20	8/9/20	\
0	1030	1052	1066	1073	1073	1096	1113	1134	1215	
1	3142	3223	3265	3320	3380	3438	3504	3564	3606	
2	602	610	612	614	615	619	624	628	630	
3	368	372	382	389	392	421	424	434	446	
4	792	813	830	836	839	874	909	923	934	

	8/10/20	8/11/20	8/12/20	8/13/20	8/14/20	8/15/20	8/16/20	8/17/20	\
0	1215	1215	1241	1250	1252	1262	1273	1274	
1	3714	3736	3776	3813	3860	3909	3948	3960	
2	631	643	646	651	656	663	671	672	
3	450	455	464	469	477	483	483	488	
4	947	958	967	977	989	996	1005	1008	

	8/18/20	8/19/20	8/20/20	8/21/20	8/22/20	8/23/20	8/24/20	8/25/20	\
0	1291	1293	1293	1293	1322	1324	1351	1355	
1	3977	4002	4035	4054	4115	4147	4167	4190	
2	674	683	690	690	699	702	720	724	
3	490	503	507	509	516	523	526	527	
4	1034	1049	1077	1083	1096	1099	1135	1160	

	8/26/20	8/27/20	8/28/20	8/29/20	8/30/20	8/31/20	9/1/20	9/2/20	\
0	1366	1377	1389	1400	1438	1442	1452	1452	
1	4265	4311	4347	4424	4525	4545	4568	4583	
2	732	739	745	753	757	757	764	768	
3	530	533	535	540	550	554	558	562	
4	1195	1213	1219	1248	1277	1287	1303	1308	

	9/3/20	9/4/20	9/5/20	9/6/20	9/7/20	9/8/20	9/9/20	9/10/20	9/11/20	\
0	1466	1475	1492	1498	1504	1508	1522	1544	1551	
1	4628	4654	4686	4713	4730	4757	4787	4833	4886	
2	771	776	776	777	778	778	778	785	786	
3	564	570	576	581	583	589	591	594	602	
4	1336	1361	1376	1379	1384	1390	1401	1430	1441	

	9/12/20	9/13/20	9/14/20	9/15/20	9/16/20	9/17/20	9/18/20	9/19/20	\
0	1565	1576	1585	1601	1619	1624	1664	1673	
1	4922	4959	4978	4992	5003	5021	5033	5047	
2	792	794	801	806	809	809	824	830	
3	604	607	610	611	612	617	619	628	
4	1446	1453	1464	1475	1487	1504	1527	1542	

	9/20/20	9/21/20	9/22/20	9/23/20	9/24/20	9/25/20	9/26/20	9/27/20	\
0	1690	1691	1714	1715	1715	1757	1764	1773	
1	5061	5087	5124	5141	5141	5456	5477	5526	
2	835	838	848	851	851	873	882	885	
3	632	635	635	638	638	652	654	656	
4	1551	1560	1573	1580	1580	1608	1611	1617	

	9/28/20	9/29/20	9/30/20	10/1/20	10/2/20	10/3/20	10/4/20	10/5/20	\
0	1785	1787	1791	1798	1805	1818	1828	1831	
1	5588	5606	5640	5997	6024	6048	6073	6085	
2	886	886	896	898	902	921	921	921	
3	657	658	664	672	675	678	686	687	
4	1618	1621	1629	1634	1642	1655	1656	1662	

	10/6/20	10/7/20	10/8/20	10/9/20	10/10/20	10/11/20	10/12/20	10/13/20	\
0	1839	1852	1863	1882	1898	1905	1911	1924	
1	6116	6134	6141	6172	6190	6203	6220	6248	
2	923	927	927	939	942	942	944	950	
3	691	703	708	719	726	736	738	744	
4	1665	1673	1681	1689	1704	1713	1722	1742	

	10/14/20	10/15/20	10/16/20	10/17/20	10/18/20	10/19/20	10/20/20	\
0	1928	1949	1966	1983	1989	1999	2010	
1	6270	6285	6333	6350	6369	6375	6405	
2	950	965	968	977	981	981	988	
3	744	761	771	775	785	789	791	

```
4      1750      1768      1783      1807      1827      1838      1848

   10/21/20  10/22/20  10/23/20  10/24/20  10/25/20  10/26/20  10/27/20  \
0      2021      2023      2030      2048      2059      2074      2082
1      6443      6475      6615      6637      6658      6694      6712
2       996       997      1012      1031      1033      1033      1042
3       801       811       825       828       840       843       850
4      1873      1893      1911      1925      1932      1942      1972

   10/28/20  10/29/20  10/30/20  10/31/20  11/1/20  11/2/20  11/3/20  11/4/20  \
0      2103      2126      2141      2159     2173     2186     2197     2212
1      6743      6768      6888      6940     6966     6985     6995     7061
2      1045      1055      1056      1060     1061     1065     1074     1079
3       856       861       866       873      878      883      890      897
4      1988      2009      2039      2074     2095     2108     2162     2188

   11/5/20  11/6/20  11/7/20  11/8/20  11/9/20  11/10/20  11/11/20  11/12/20  \
0     2230     2242     2267     2283     2304      2328      2351      2385
1     7097     7134     7188     7226     7263      7348      7409      7454
2     1080     1090     1092     1095     1098      1107      1112      1113
3      907      917      924      926      932       948       961       966
4     2222     2253     2286     2297     2335      2378      2400      2429

   11/13/20  11/14/20  11/15/20  11/16/20  11/17/20  11/18/20  11/19/20  \
0      2417      2435      2456      2481      2506      2529      2554
1      7523      7596      7646      7696      7772      7849      7933
2      1117      1123      1128      1130      1134      1137      1145
3       973       978       986       993      1004      1008      1011
4      2488      2518      2549      2574      2594      2648      2683

   11/20/20  11/21/20  11/22/20  11/23/20  11/24/20  11/25/20  11/26/20  \
0      2580      2597      2617      2634      2661      2686      2704
1      8038      8131      8199      8269      8376      8473      8576
2      1151      1157      1160      1161      1167      1170      1170
3      1024      1036      1136      1142      1157      1162      1170
4      2704      2735      2754      2763      2822      2855      2879

   11/27/20  11/28/20  11/29/20  11/30/20  12/1/20  12/2/20  12/3/20  12/4/20  \
0      2716      2735      2751      2780     2818     2873     2893     2945
1      8603      8733      8820      8890     9051     9163     9341     9501
2      1171      1173      1175      1178     1189     1206     1214     1217
3      1173      1179      1188      1196     1204     1239     1252     1270
4      2888      2922      2946      2997     3061     3100     3158     3231

   12/5/20  12/6/20  12/7/20  12/8/20  12/9/20  12/10/20  12/11/20  12/12/20  \
0     2979     3005     3043     3087     3117      3186      3233      3233
1     9626     9728     9821     9974    10087     10288     10489     10489
```

```
2     1219     1223     1224     1240     1245     1258     1264     1264
3     1283     1293     1299     1317     1322     1359     1398     1398
4     3281     3299     3324     3426     3496     3600     3663     3663

   12/13/20  12/14/20  12/15/20  12/16/20  12/17/20  12/18/20  12/19/20  \
0      3233      3329      3426      3510      3570      3647      3698
1     10489     10898     11061     11212     11364     11556     11722
2      1264      1275      1292      1296      1309      1318      1330
3      1398      1455      1504      1520      1548      1577      1601
4      3663      3803      3881      3950      4036      4118      4191

   12/20/20  12/21/20  12/22/20  12/23/20  12/24/20  12/25/20  12/26/20  \
0      3741      3780      3841      3889      3942      3990      3999
1     11827     11952     12155     12321     12521     12666     12708
2      1336      1336      1363      1383      1390      1396      1398
3      1613      1628      1660      1683      1711      1725      1739
4      4218      4234      4313      4367      4405      4441      4446

   12/27/20  12/28/20  12/29/20  12/30/20  12/31/20  1/1/21  1/2/21  1/3/21  \
0      4029      4065      4105      4164      4190    4239    4268    4305
1     12825     12962     13172     13392     13601   13823   13955   14064
2      1406      1417      1462      1492      1514    1517    1528    1530
3      1746      1762      1792      1817      1834    1854    1863    1882
4      4465      4483      4535      4584      4641    4693    4729    4746

   1/4/21  1/5/21  1/6/21  1/7/21  1/8/21  1/9/21  1/10/21  1/11/21  1/12/21  \
0    4336    4546    4645    4705    4770    4847     4879     4902     4970
1   14187   14440   14656   14845   15052   15202    15327    15417    15572
2    1533    1575    1597    1614    1634    1648     1658     1663     1679
3    1885    1923    1944    1981    2015    2038     2051     2060     2090
4    4771    4849    4898    4957    5018    5047     5066     5080     5134

   1/13/21  1/14/21  1/15/21  1/16/21  1/17/21
0     4998     5075     5103     5154     5184
1    15701    15841    16002    16176    16251
2     1685     1696     1712     1723     1729
3     2109     2113     2130     2144     2151
4     5170     5219     5264     5292     5304
```

I already can tell I'll need to clean the data a bit. First, I am only going to keep the columns I need for my analysis. Since I know the debt data for my 2nd dataset was collected at the end of November, I want that to be in line with my COVID data, so I am only going to keep the column from 11/30/20. I know I can keep this column because the COVID data itself is cumulative, so it will show how many cases that particular county has had from the beginning of the pandemic up until the end of November.

[5]:
```
countycovid1 = countycovid[['COUNTY', 'NAME', 'County Name', 'State',␣
↪'stateFIPS', 'POP70', 'HHD70' , 'POP80',
                            'HHD80', 'POP90', 'HHD90', 'POP00' , 'HHD00' ,␣
↪'POP10' , 'HHD10' , '11/30/20']]

countycovid1.head()
```

[5]:

	COUNTY	NAME	County Name	State	stateFIPS	POP70	HHD70
0	1001	Autauga County	Autauga County	AL	1	24457	6792
1	1003	Baldwin County	Baldwin County	AL	1	59132	17641
2	1005	Barbour County	Barbour County	AL	1	22484	6796
3	1007	Bibb County	Bibb County	AL	1	13812	4015
4	1009	Blount County	Blount County	AL	1	26844	8431

	POP80	HHD80	POP90	HHD90	POP00	HHD00	POP10	HHD10	11/30/20
0	32266	10199	34236	11830	43685	16007	54571	20221	2780
1	78213	26641	98277	37041	140406	55330	182265	73180	8890
2	24685	8352	25418	9217	29037	10409	27457	9820	1178
3	15680	5153	16589	5750	20827	7421	22915	7953	1196
4	36456	12679	39247	14644	51020	19264	57322	21578	2997

Now that I have my columns of interest, I need to remove all states that are not California from the State column.

[6]:
```
countycovid1 = countycovid1.loc[countycovid1['State'] == 'CA']
```

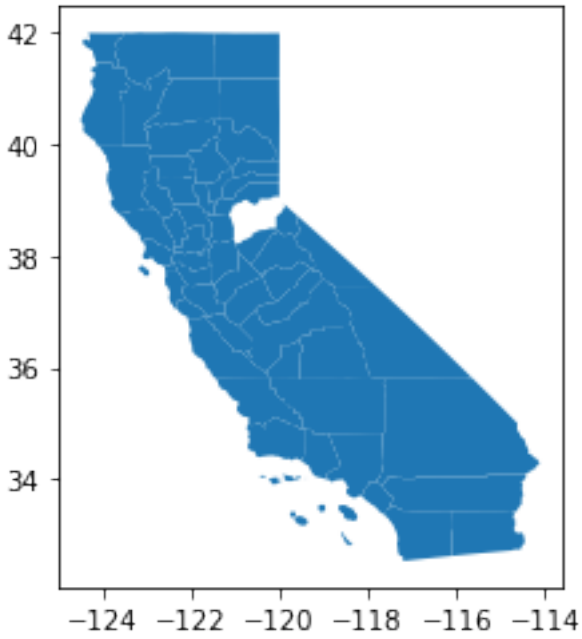
Great! Now I only have values for California and COVID infection rates from the 30th of November.

0.2 Exploring and Cleaning County GEOJSON File

Now, it's time to examine my shapefile. I'm going to plot it to see if everything looks okay.

[7]:
```
countyshape.plot()
```

[7]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f60ed3e97f0>
```



Yepp! That's a map of California! The only issue is the small county missing in the middle of the state? Let's try and figure out why that's happening…First, let's take a look at the top 5 rows of data.

[8]:
```
countyshape.head()
```

[8]:

	gid	admin_level	area	boundary	name	place	\
0	-396505	6	None	administrative	Ventura County	None	
1	40501106	6	yes	administrative	None	island	
2	40501107	6	yes	administrative	None	island	
3	40501108	6	yes	administrative	None	island	
4	-396479	6	None	administrative	Los Angeles County	None	

	population	z_order	way_area	tid	territory_name	\
0	850536	0	1.548730e+08	None	None	
1	None	0	2.226610e+03	None	None	
2	None	0	3.716340e+03	None	None	
3	None	0	5.450680e+03	None	None	
4	None	0	2.456990e+08	None	None	

	geometry
0	POLYGON ((-119.75770 33.36296, -119.75715 33.3…
1	POLYGON ((-118.50165 32.85270, -118.50161 32.8…
2	POLYGON ((-118.53169 32.89987, -118.53167 32.8…
3	POLYGON ((-118.53422 32.90499, -118.53420 32.9…
4	POLYGON ((-118.60965 33.01726, -118.60643 33.0…

This didn't really tell met much. Let's get a better sense of missing values and datatype with .info

[9]:
```
countyshape.info()
```

```
<class 'geopandas.geodataframe.GeoDataFrame'>
RangeIndex: 116 entries, 0 to 115
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   gid             116 non-null    int64
 1   admin_level     116 non-null    object
 2   area            46 non-null     object
 3   boundary        116 non-null    object
 4   name            71 non-null     object
 5   place           47 non-null     object
 6   population      52 non-null     object
 7   z_order         116 non-null    int64
 8   way_area        116 non-null    float64
 9   tid             0 non-null      object
 10  territory_name  0 non-null      object
 11  geometry        116 non-null    geometry
dtypes: float64(1), geometry(1), int64(2), object(8)
memory usage: 11.0+ KB
```

Okay, we see there are 58 rows of data - overall it is consistent across all columns, which seems promising. How about we list the county names to see if anything looks weird there?

[10]:
```
countyshape['name'].value_counts()
```

[10]:
```
Ventura County                  7
Santa Barbara County            5
Los Angeles County              3
San Francisco City and County   2
Nevada County                   1
San Joaquin County              1
Alameda County                  1
Yuba County                     1
Fresno County                   1
Calaveras County                1
Kern County                     1
Mendocino County                1
Butte County                    1
Goat Island                     1
Mariposa County                 1
Glenn County                    1
Madera County                   1
Orange County                   1
Sutter County                   1
```

```
Colusa County                   1
Sacramento County               1
Trinity County                  1
San Luis Obispo County          1
Placer County                   1
Monterey County                 1
Lassen County                   1
Merced County                   1
San Diego County                1
Lake County                     1
Sierra County                   1
Inyo County                     1
Riverside County                1
Modoc County                    1
San Benito County               1
Stanislaus County               1
Santa Cruz County               1
Yolo County                     1
Humboldt County                 1
Marin County                    1
Santa Clara County              1
Tulare County                   1
Plumas County                   1
San Mateo County                1
Imperial County                 1
Contra Costa County             1
Kings County                    1
Mono County                     1
Shasta County                   1
Tehama County                   1
Del Norte County                1
Napa County                     1
San Bernardino County           1
Solano County                   1
Siskiyou County                 1
Tuolumne County                 1
Sonoma County                   1
Bird Island                     1
Alpine County                   1
Name: name, dtype: int64
```

Ah, the problem is that El Dorado County is missing from the dataset. I'm going to continue my analysis with this in mind, but I might want to consider finding a complete dataset if I want to include these COVID numbers in my analysis. This particular dataset was from a website called IGISMap.com.

I had a difficult time trying to find a dataset of county boundaries that could easily be merged with my COVID data, but I might need to continue looking.

For now, I am going to try and merge the existing datasets together though a common column.

0.3 Merging Datasets

The first thing I need to do is rename the column titles so 1 from each data set match.

[11]:
```
list(countyshape)
```

[11]:
```
['gid',
 'admin_level',
 'area',
 'boundary',
 'name',
 'place',
 'population',
 'z_order',
 'way_area',
 'tid',
 'territory_name',
 'geometry']
```

[12]:
```
countyshape.columns = ['gid',
 'admin_level',
 'area',
 'boundary',
 'County Name',
 'place',
 'population',
 'z_order',
 'way_area',
 'tid',
 'territory_name',
 'geometry']
```

You can see above that I changed the "Name" column in my Countyshape file so that it was named "County Name" and could be matched with my countycovid1 file.

Now, let's check to make sure it renamed.

[13]:
```
countyshape.head()
```

[13]:
```
        gid admin_level  area        boundary         County Name   place  \
0   -396505           6  None  administrative      Ventura County    None
1  40501106           6   yes  administrative                None  island
2  40501107           6   yes  administrative                None  island
3  40501108           6   yes  administrative                None  island
4   -396479           6  None  administrative  Los Angeles County    None

  population  z_order      way_area   tid territory_name  \
```

```
0     850536        0  1.548730e+08  None           None
1       None        0  2.226610e+03  None           None
2       None        0  3.716340e+03  None           None
3       None        0  5.450680e+03  None           None
4       None        0  2.456990e+08  None           None

                                              geometry
0  POLYGON ((-119.75770 33.36296, -119.75715 33.3…
1  POLYGON ((-118.50165 32.85270, -118.50161 32.8…
2  POLYGON ((-118.53169 32.89987, -118.53167 32.8…
3  POLYGON ((-118.53422 32.90499, -118.53420 32.9…
4  POLYGON ((-118.60965 33.01726, -118.60643 33.0…
```

It worked! Now that the column, "County Name" on my shape file matches my COVID data file. I am going to merge the files via this column.

[18]:
```
merged = countycovid1.merge(countyshape,
                            on='County Name')
```

Through a lot of Googling, I figured out the code for merging, and I did so by my renamed column.

Now, let's check to see if it worked…

[19]:
```
merged.head()
```

[19]:
```
   COUNTY              NAME       County Name State  stateFIPS    POP70  \
0    6001    Alameda County    Alameda County    CA          6  1066698
1    6003     Alpine County     Alpine County    CA          6      481
2    6007      Butte County      Butte County    CA          6   101959
3    6009  Calaveras County  Calaveras County    CA          6    13517
4    6011     Colusa County     Colusa County    CA          6    12420

    HHD70    POP80   HHD80    POP90   HHD90    POP00   HHD00    POP10   HHD10  \
0  365015  1101902  426043  1275749  478544  1443745  523359  1510271  545138
1     178     1092     384     1116     451     1208     483     1175     497
2   34896   143850   56906   182122   71662   203168   79566   220000   87618
3    4683    20639    7975    31996   12650    40553   16467    45578   18886
4    4132    12752    4676    16277    5614    18803    6098    21419    7056

   11/30/20     gid admin_level  area        boundary place population  \
0     29668 -396499           6  None  administrative  None    1638215
1        47 -396497           6  None  administrative  None       None
2      4131 -396508           6  None  administrative  None     225411
3       450 -396470           6  None  administrative  None      44828
4       737 -396476           6  None  administrative  None       None

   z_order      way_area   tid territory_name  \
0        0  3.393090e+09  None           None
```

```
1        0  3.149070e+09  None           None
2        0  7.339060e+09  None           None
3        0  4.351060e+09  None           None
4        0  4.992720e+09  None           None

                                              geometry
0  POLYGON ((-122.37384 37.88364, -122.37381 37.8…
1  POLYGON ((-120.07258 38.44718, -120.07221 38.4…
2  POLYGON ((-122.06926 39.84005, -122.06922 39.8…
3  POLYGON ((-120.99564 38.22533, -120.98791 38.2…
4  POLYGON ((-122.78509 39.38297, -122.78469 39.3…
```

It worked!!! Now, let's map COVID rates by county…

0.4 Data Normalization and Maps

Now I want to map the number of COVID infections by County. First, I need to standardize the data so I have a plot of COVID cases by population.

However, I realized my data for "11/30/20" (aka COVID cases) and "population" are both strings and not integers (though a lot of errors) I need to change them accordingly.

[56]:
```
merged = pd.DataFrame(merged)
merged['11/30/20'] = merged['11/30/20'].astype(int)
```

Though some Googling, I looked up how to change my data from string to integer, but I need to check to make sure it worked.

[61]:
```
print (merged)
print (merged.dtypes)
```

```
    COUNTY              NAME       County Name State  stateFIPS    POP70  \
0     6001    Alameda County    Alameda County    CA          6  1066698
1     6003     Alpine County     Alpine County    CA          6      481
2     6007      Butte County      Butte County    CA          6   101959
3     6009  Calaveras County  Calaveras County    CA          6    13517
4     6011     Colusa County     Colusa County    CA          6    12420
..     …               …              …   …        …       …
62    6111    Ventura County    Ventura County    CA          6   376420
63    6111    Ventura County    Ventura County    CA          6   376420
64    6111    Ventura County    Ventura County    CA          6   376420
65    6113       Yolo County       Yolo County    CA          6    91790
66    6115       Yuba County       Yuba County    CA          6    44739

     HHD70    POP80   HHD80    POP90   HHD90    POP00   HHD00    POP10  \
0   365015  1101902  426043  1275749  478544  1443745  523359  1510271
1      178     1092     384     1116     451     1208     483     1175
2    34896   143850   56906   182122   71662   203168   79566   220000
3     4683    20639    7975    31996   12650    40553   16467    45578
```

```
4     4132    12752    4676    16277    5614    18803    6098    21419
..     …       …      …       …       …       …      …       …
62  106492   528867  172824   669221  217386   753507  243340   823318
63  106492   528867  172824   669221  217386   753507  243340   823318
64  106492   528867  172824   669221  217386   753507  243340   823318
65   28323   113391   41305   141113   50981   168661   59376   200849
66   13075    49739   17507    58233   19778    60219   20534    72155

     HHD10  11/30/20     gid admin_level  area        boundary place  \
0   545138     29668 -396499           6  None  administrative  None
1      497        47 -396497           6  None  administrative  None
2    87618      4131 -396508           6  None  administrative  None
3    18886       450 -396470           6  None  administrative  None
4     7056       737 -396476           6  None  administrative  None
..     …       …      …          …   …             …   …
62  266920     20066 -396505           6  None  administrative  None
63  266920     20066 -396505           6  None  administrative  None
64  266920     20066 -396505           6  None  administrative  None
65   70872      4893 -396507           6  None  administrative  None
66   24307      2088 -396475           6  None  administrative  None

    population  z_order      way_area   tid territory_name  \
0    1638215.0        0  3.393090e+09  None           None
1          NaN        0  3.149070e+09  None           None
2     225411.0        0  7.339060e+09  None           None
3      44828.0        0  4.351060e+09  None           None
4          NaN        0  4.992720e+09  None           None
..         …      …           …   …            …
62    850536.0        0  2.985070e+03  None           None
63    850536.0        0  9.238970e+05  None           None
64    850536.0        0  7.624770e+09  None           None
65    213016.0        0  4.347000e+09  None           None
66     74492.0        0  2.785720e+09  None           None

                                              geometry
0   POLYGON ((-122.37384 37.88364, -122.37381 37.8…
1   POLYGON ((-120.07258 38.44718, -120.07221 38.4…
2   POLYGON ((-122.06926 39.84005, -122.06922 39.8…
3   POLYGON ((-120.99564 38.22533, -120.98791 38.2…
4   POLYGON ((-122.78509 39.38297, -122.78469 39.3…
..                                                …
62  POLYGON ((-119.40795 34.00598, -119.40794 34.0…
63  POLYGON ((-119.40761 34.00581, -119.40743 34.0…
64  POLYGON ((-119.50095 34.32692, -119.48422 34.3…
65  POLYGON ((-122.42293 38.90283, -122.42291 38.9…
66  POLYGON ((-121.63634 39.24632, -121.63634 39.2…

[67 rows x 27 columns]
```

```
COUNTY                int64
NAME                 object
County Name          object
State                object
stateFIPS             int64
POP70                 int64
HHD70                 int64
POP80                 int64
HHD80                 int64
POP90                 int64
HHD90                 int64
POP00                 int64
HHD00                 int64
POP10                 int64
HHD10                 int64
11/30/20              int64
gid                   int64
admin_level          object
area                 object
boundary             object
place                object
population          float64
z_order               int64
way_area            float64
tid                  object
territory_name       object
geometry           geometry
dtype: object
```

It seems that based on my conversion above, both 'population' and '11/30/20' have been converted from a string. Now I just need to see if I can divide them....

[63]:
```
merged['COVID Cases by Population'] = merged['11/30/20'] /␣
↪merged['population']*100
```

Now that I have divided COVID Cases by population, I need to check that a new column appeared with the proper calcaultions.

[64]:
```
merged.head()
```

[64]:

	COUNTY	NAME	County Name	State	stateFIPS	POP70	\
0	6001	Alameda County	Alameda County	CA	6	1066698	
1	6003	Alpine County	Alpine County	CA	6	481	
2	6007	Butte County	Butte County	CA	6	101959	
3	6009	Calaveras County	Calaveras County	CA	6	13517	
4	6011	Colusa County	Colusa County	CA	6	12420	

	HHD70	POP80	HHD80	POP90	HHD90	POP00	HHD00	POP10	HHD10	\

```
0  365015  1101902  426043  1275749  478544  1443745  523359  1510271  545138
1     178     1092     384     1116     451     1208     483     1175     497
2   34896   143850   56906   182122   71662   203168   79566   220000   87618
3    4683    20639    7975    31996   12650    40553   16467    45578   18886
4    4132    12752    4676    16277    5614    18803    6098    21419    7056

   11/30/20     gid admin_level  area        boundary place  population  \
0     29668 -396499           6  None  administrative  None   1638215.0
1        47 -396497           6  None  administrative  None         NaN
2      4131 -396508           6  None  administrative  None    225411.0
3       450 -396470           6  None  administrative  None     44828.0
4       737 -396476           6  None  administrative  None         NaN

   z_order      way_area   tid territory_name  \
0        0  3.393090e+09  None           None
1        0  3.149070e+09  None           None
2        0  7.339060e+09  None           None
3        0  4.351060e+09  None           None
4        0  4.992720e+09  None           None

                                            geometry  \
0  POLYGON ((-122.37384 37.88364, -122.37381 37.8…
1  POLYGON ((-120.07258 38.44718, -120.07221 38.4…
2  POLYGON ((-122.06926 39.84005, -122.06922 39.8…
3  POLYGON ((-120.99564 38.22533, -120.98791 38.2…
4  POLYGON ((-122.78509 39.38297, -122.78469 39.3…

   COVID Cases by Population
0                   1.810996
1                        NaN
2                   1.832652
3                   1.003837
4                        NaN
```

It worked! And the math checks out!

Another thing I realized (through a lot of errors) is that in order to map a dataset with geopandas, the dataset itself must be a GeoDataFrame… so, now that I have conducted the calculations I need, I can convert this dataset to a GeoDataFrame.

[67]:
```
merged = gpd.GeoDataFrame(merged)
```

Now to check if it worked…

[68]:
```
type(merged)
```

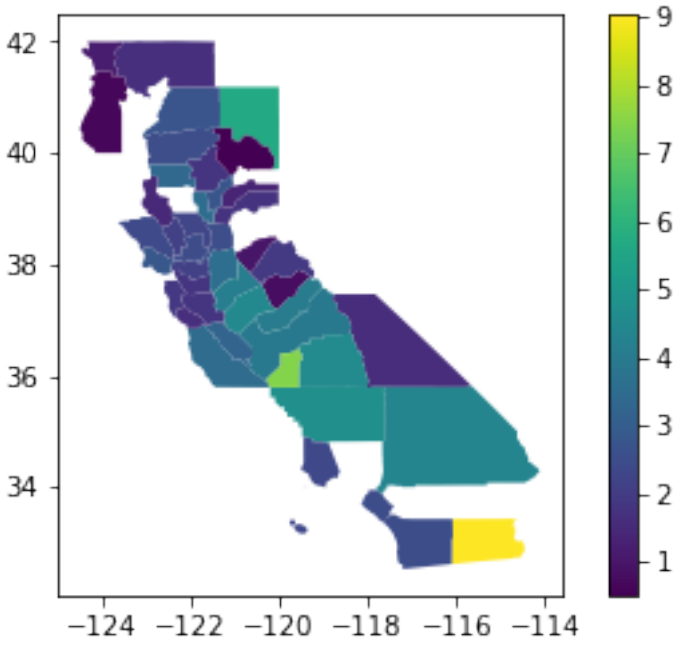
[68]:
```
geopandas.geodataframe.GeoDataFrame
```

It worked!

Last on the list is to map (finally!)

[66]:
```
merged.plot('COVID Cases by Population', legend=True)
```

[66]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f60e4b8d760>
```



I did it! Above you can see a nice map of COVID cases by Population in each county in California! The highest percentage seems to be in Imperial County (and by a long shot too). I think is it definitely worth looking into this trend. However, there are also some higer numbers in San Bernardino and Kern, and up through Kings and Fresno. These are some interesting trends and it might be nice to compare them to other factors (like my water bill debt and demographics data).

TBD on if I want to keep this as part of my final project - I think there are still some kinks to work out. But overall, I think this was a helpful exercise, and if I decide to include it in my final project, it will be a nice supplement to my existing data.