

MI_subs

April 25, 2022

0.1 Analyze data from the Michigan Department of Health and Human Services

```
[1]: import pandas as pd
```

```
[2]: pd.set_option("display.max_row", 300)
pd.set_option("display.max_columns", 300)
pd.set_option("display.max_seq_items", 300)
```

```
[3]: # Load full data
df = (
    pd
    .read_csv(
        "../../../data/states/michigan/mi_full.csv",
        skiprows = 0,
        parse_dates = ["complaint_date"],
        dtype={"ONGOING_CSE_ID": "str"}
    )
    .assign(
        risk = lambda x: x["Risk Level Reworked"].str.strip(),
        year = lambda x: x["complaint_date"].dt.year
    )
    .drop("Risk Level Reworked", axis = 1)
)

df.columns = [ each.strip() for each in df.columns ]

#df.head()
```

```
[4]: # Load county census demos
county = (
    pd
    .read_csv(
        "../../../data/national/county_B03002.csv",
        dtype = {
            "fips": object,
        }
    )
    .loc[ lambda x: x["state"] == "Michigan" ]
```

```
)

county.head(3)
```

```
[4]:
```

	fips	name	total	non_hispanic	white	\
1231	0500000US26001	Alcona County, Michigan	10353	10198	9916	
1232	0500000US26003	Alger County, Michigan	9151	9007	7650	
1233	0500000US26005	Allegan County, Michigan	116143	107611	102709	

	black	native	asian	pac	other	two	hispanic	year	state	\
1231	41	60	20	7	4	150	155	2019	Michigan	
1232	702	334	13	0	0	308	144	2019	Michigan	
1233	1426	364	802	0	132	2178	8532	2019	Michigan	


```

county
1231  Alcona
1232  Alger
1233  Allegan
```

```
[5]: # Load state census demos
state = (
    pd
    .read_csv(
        "../data/national/state_B03002.csv"
    )
    .loc[ lambda x: x["state"] == "Michigan" ]
)

state.head()
```

```
[5]:
```

	total	non_hispanic	white	black	native	asian	pac	other	\
22	9900571	9433550	7513622	1366632	47055	267671	1823	12083	
74	9920621	9488578	7609623	1381897	48838	243249	2389	12104	
126	9952687	9529275	7654981	1389178	48753	241673	2378	13427	
178	9973907	9452704	7428622	1342592	42931	314736	2675	27247	
230	9889024	9431915	7526388	1368159	48437	257464	1757	11526	

	two	hispanic	fips	state	year
22	224664	467021	0400000US26	Michigan	2015
74	190478	432043	0400000US26	Michigan	2011
126	178885	423412	0400000US26	Michigan	2010
178	293901	521203	0400000US26	Michigan	2020
230	218184	457109	0400000US26	Michigan	2014

```
[6]: # yearly substantiations
yearly = (
    df
```

```

.loc[ lambda x: x["year"] > 2009 ]
.groupby("year")["src_person_id"]
.nunique()
.to_frame("subs")
.reset_index()
)

yearly.to_csv("../outputs/MI_yearly_subs.csv", index = False)

yearly

```

```

[6]:
   year  subs
0  2010  4878
1  2011  5432
2  2012  6323
3  2013  7394
4  2014  7112
5  2015  8537
6  2016  9832
7  2017 11581
8  2018 13910
9  2019 17240
10 2020 16458
11 2021 12143

```

```

[7]: # Race by percent of registry
all_reg = df["src_person_id"].nunique()

byRace = (
    df
    .groupby("src_person_id")["MultiRaceDesc2"]
    .first()
    .reset_index()
    .groupby("MultiRaceDesc2")
    .count()
    .assign(
        percent_registry = lambda x: round((x["src_person_id"]/all_reg), 3)
    )
    .reset_index()
)

byRace

```

```

[7]:
   MultiRaceDesc2  src_person_id  percent_registry
0             Asian             349             0.003
1             Black          26784             0.222
2           Hispanic           5928             0.049

```

3	Multi Race	7477	0.062
4	Native American	441	0.004
5	Unknown	1652	0.014
6	White	78268	0.647

```
[8]: raceYear = (
    df
    # remove dupes
    .groupby(["year", "src_person_id", "MultiRaceDesc2"])
    .first()
    .reset_index()
    .groupby(["year", "MultiRaceDesc2"])
    .pipe( lambda grp: pd.DataFrame({
        "count": grp.size()
    }))
    .reset_index()
    .rename(columns = {"MultiRaceDesc2": "race"})
    .sort_values(
        "year",
        ascending = False
    )
)

#raceYear.head()
```

```
[9]: raceCountyYear = (
    df
    # remove dupes
    .groupby(["year", "countyname", "src_person_id", "MultiRaceDesc2"])
    .first()
    .reset_index()
    .groupby(["year", "countyname", "MultiRaceDesc2"])
    .pipe( lambda grp: pd.DataFrame({
        "count": grp.size()
    }))
    .reset_index()
    .rename(columns = {"MultiRaceDesc2": "race"})
    .sort_values(
        "year",
        ascending = False
    )
)

#raceCountyYear.head()
```

```
[10]: # helper to calculate percents across rows
def percent(row):
```

```
total = row.sum()
return row.apply( lambda x: x/total )
```

```
[11]: # race by percent
(
    raceYear
    .groupby(["year", "race"])
    ["count"]
    .sum()
    .unstack()
    .reset_index("year")
    .loc[ lambda x: x["year"] > 2009 ]
    .set_index("year")
    .apply( percent )
)
```

```
[11]: race      Asian      Black  Hispanic  Multi Race  Native American  Unknown  \
year
2010  0.054441  0.043301  0.037819    0.036120          0.027211  0.008475
2011  0.074499  0.045468  0.035455    0.045351          0.040816  0.013923
2012  0.065903  0.050587  0.050481    0.061538          0.043084  0.018765
2013  0.060172  0.057872  0.050819    0.073177          0.058957  0.018765
2014  0.060172  0.054248  0.052338    0.059799          0.038549  0.313559
2015  0.071633  0.061795  0.062637    0.069833          0.065760  0.378329
2016  0.100287  0.083464  0.086612    0.079732          0.074830  0.104722
2017  0.120344  0.099679  0.105521    0.086288          0.102041  0.039346
2018  0.094556  0.116566  0.120041    0.106622          0.117914  0.033293
2019  0.123209  0.142457  0.145703    0.143411          0.172336  0.030872
2020  0.094556  0.142457  0.145197    0.136054          0.163265  0.026029
2021  0.080229  0.102107  0.107378    0.102074          0.095238  0.013923

race      White
year
2010  0.040647
2011  0.046003
2012  0.052880
2013  0.062875
2014  0.055564
2015  0.067886
2016  0.079863
2017  0.095751
2018  0.116855
2019  0.144720
2020  0.135747
2021  0.101209
```

0.2 Risk Ratios

```
[12]: # by year and state

def rr(grp):
    year = int(grp.name)

    # state populations by race
    census = state.loc[ lambda x: x["year"] == year ]

    pop = census["total"].values[0]
    pop_black = census["black"].values[0]
    pop_white = census["white"].values[0]
    pop_hisp = census["hispanic"].values[0]

    # reg populations
    reg_total = grp["count"].sum()

    reg_black = grp.loc[
        lambda x: x["race"] == "Black"
    ]["count"].sum()

    reg_white = grp.loc[
        lambda x: x["race"] == "White"
    ]["count"].sum()

    reg_hisp = grp.loc[
        lambda x: x["race"] == "Hispanic"
    ]["count"].sum()

    # rates
    rate_white = reg_white/pop_white
    rate_black = reg_black/pop_black
    rate_hisp = reg_hisp/pop_hisp

    # risk ratios
    rrBlack = (rate_black/rate_white)
    rrHisp = rate_hisp/rate_white

    return pd.Series({
        "Risk Ratio, Black": rrBlack,
        "Risk Ratio, Hispanic": rrHisp
    })

# Risk ratios of substantiations and established
(
    raceYear
```

```

.loc[ lambda x: x["year"] < 2021]
.loc[ lambda x: x["year"] > 2009]
.groupby("year")
.apply(rr)
)

```

```

[12]:      Risk Ratio, Black  Risk Ratio, Hispanic
year
2010          2.008365          1.273509
2011          1.862071          1.027717
2012          1.800015          1.248556
2013          1.734306          1.031705
2014          1.837499          1.174191
2015          1.712210          1.123856
2016          1.969730          1.298639
2017          1.963527          1.290215
2018          1.878926          1.169861
2019          1.854315          1.123383
2020          1.986582          1.154185

```

```

[13]: # average risk ratios
(
    raceYear
    .loc[ lambda x: x["year"] < 2021 ]
    .loc[ lambda x: x["year"] > 2009 ]
    .groupby("year")
    .apply(rr)
    .mean()
    .to_frame("")
)

```

```

[13]:      Risk Ratio, Black      1.873413
      Risk Ratio, Hispanic  1.174165

```

```

[14]: # county data example
(
    county
    .loc[
        lambda x: (x["year"] == 2019) &
        (x["county"] == "Wayne")
    ]
)

```

```

[14]:      fips      name  total  non_hispanic  white \
1312  0500000US26163  Wayne County, Michigan  1757299      1652880  870241

```

	black	native	asian	pac	other	two	hispanic	year	state	\
1312	677346	4673	58726	426	4039	37429	104419	2019	Michigan	

	county
1312	Wayne

```
[15]: # risk ratios for grouping by year and county

# helper to avoid errors
def div(num, denom):
    if denom == 0:
        return 0
    return num/denom

def crr(grp):
    year = int(grp.name[0])
    countyname = grp.name[1]

    # if can't locate county, it's too small for census. Skip
    census = (
        county
        .loc[ lambda x: (x["year"] == year) &
              (x["county"] == countyname)
        ]
    )

    pop = census["total"].values[0]
    pop_black = census["black"].values[0]
    pop_white = census["white"].values[0]
    pop_hisp = census["hispanic"].values[0]

    # reg populations
    reg_total = grp["count"].sum()

    reg_black = grp.loc[
        lambda x: x["race"] == "Black"
    ]["count"].sum()

    reg_white = grp.loc[
        lambda x: x["race"] == "White"
    ]["count"].sum()

    reg_hisp = grp.loc[
        lambda x: x["race"] == "Hispanic"
    ]["count"].sum()
```



```

# rates
rate_white = div(reg_white,pop_white)
rate_black = div(reg_black,pop_black)
rate_hisp = div(reg_hisp, pop_hisp)

# risk ratios
rrBlack = div(rate_black,rate_white)
rrHisp = div(rate_hisp,rate_white)

return pd.Series({
    "Total population": pop,
    "Percent black": (pop_black/pop),
    "Percent registry black": (reg_black/reg_total),
    "Risk Ratio, Black": rrBlack,
    "Risk Ratio, Hispanic": rrHisp
})

# Risk ratios of substantiations and established
top20 = (
    raceCountyYear
    .loc[ lambda x: ~x["countyname"].str.contains(
        "\?|NO MATCH FOUND|Out-of-state")
        ]
    .loc[ lambda x: x["year"] == 2019 ]
    .groupby(["year", "countyname"])
    .apply(crr)
    .sort_values(
        ["Total population", "Risk Ratio, Black"],
        ascending = False
    ).iloc[0:20]
)

top20

```

```

[15]:

```

year	countyname	Total population	Percent black	Percent registry black \
2019	Wayne	1757299.0	0.385447	0.604328
	Oakland	1253185.0	0.135091	0.304590
	Macomb	870325.0	0.116289	0.262431
	Kent	648121.0	0.092261	0.273106
	Genesee	407875.0	0.195403	0.316441
	Washtenaw	367000.0	0.117779	0.463636
	Ingham	290587.0	0.111082	0.274966
	Ottawa	286558.0	0.014269	0.077748
	Kalamazoo	262745.0	0.105909	0.284038
	Saginaw	191821.0	0.182488	0.343931
	Livingston	189754.0	0.005839	0.022388

Muskegon	173297.0	0.135011	0.233119
St. Clair	159247.0	0.021840	0.078804
Jackson	158636.0	0.076931	0.123596
Berrien	154133.0	0.145121	0.250712
Monroe	149727.0	0.023469	0.116279
Calhoun	134212.0	0.103709	0.160784
Allegan	116143.0	0.012278	0.036932
Eaton	109456.0	0.065798	0.139665
Bay	104104.0	0.016743	0.092937

year	countyname	Risk Ratio, Black	Risk Ratio, Hispanic
2019	Wayne	2.488842	1.079946
	Oakland	2.779196	1.871688
	Macomb	2.645986	0.618700
	Kent	4.188180	1.582344
	Genesee	1.980490	1.041198
	Washtenaw	6.073230	1.188149
	Ingham	3.197290	1.245436
	Ottawa	6.553983	1.840860
	Kalamazoo	3.738357	1.358339
	Saginaw	2.774734	2.063436
	Livingston	3.908918	1.247251
	Muskegon	2.040262	1.006873
	St. Clair	3.901974	0.958058
	Jackson	1.785073	0.636008
	Berrien	1.957752	0.706770
	Monroe	5.894474	1.546048
	Calhoun	1.874455	0.633549
	Allegan	3.356048	1.596449
	Eaton	2.384439	1.280461
	Bay	6.254920	1.257658
