Share of Environmental Exposure Burden Across Population Groups

Fan Wang

Contents

1 Location, Population, and Pollution

Go to the **RMD**, **R**, **PDF**, or **HTML** version of this file. Go back to fan's REconTools Package, R Code Examples Repository (bookdown site), or Intro Stats with R Repository (bookdown site).

1.1 Location, Population, and Environmental Exposures (Pollution)

1.1.1 Local Public Exposures $Z_{l,y}$

Environmental exposure, specifically PM 2.5, is a local public good (bad). Prior research has shown that due to the nature of particular matter formation in the air, while there is particular matter variation at the city level, there is not significant variation in particular matter pollution at the neighborhood level He et al. (2019) and Liu et al. (2022). Additionally, there is a difference between ambient environmental exposures and the amount of particular matter inhaled by individual residents: the former is a common level shared by all residents, but the latter can differ depending on individuals' physical and socio-economic attributes.

There are M different locations (cities/counties/townships), indexed from $l = 1$ to $l = M$. In a particular time-period, we assume that the potential pollution exposure for all residents in the same location. Suppose additionally that we compute statistics at the interval of years y, and each year includes daily pollution measure for each day of the year t, indexed from $t = 1$ to $t = T_y$, where T_y is the total number of days in a year.

Let $\mathrm{Z}_{l,y}$ be the total pollution exposure by a resident in location l. This is equal to the sum of pollution exposure during the course of a year:

$$Z_{l,y} = \sum_{t=1}^{T_y} Z_{l,y,t}$$

1.1.2 Group Exposures $\mathcal{Z}_{i,y}$

Let there be N population groups, indexed from $i = 1$ to $i = N$. The N population groups reside in the M locations. the share, let $P_{l,i}$ denote share of population belong to group i that resides in location l:

$$1 = \sum_{l=1}^{M} \sum_{i=1}^{N} P_{l,i}$$

The share of population belong to population group i is:

$$P_i = \sum_{l=1}^{M} P_{l,i}$$

$\mathcal{Z}_{i,y}$, which is the average pollution exposure facing an individual belonging to group i in year y, is determined by how individuals from population group i are distributed across the M locations:

$$\mathcal{Z}_{i,y} = \sum_{l=1}^{M} \left(\frac{P_{l,i}}{P_i} \times Z_{l,y} \right)$$

Additionally, $\mathcal{Z}_y$ is the average pollution exposure facing an individual, regardless of population group, in year y:

$$\mathcal{Z}_y = \sum_{i=1}^{N} \left(\sum_{l=1}^{M} \left(\frac{P_{l,i}}{P_i} \times Z_{l,y} \right) \times P_i \right) = \sum_{i=1}^{N} \sum_{l=1}^{M} \left(P_{l,i} \times Z_{l,y} \right)$$

1.1.3 Excess Environmental Exposure Burden to Population

We have $\mathcal{Z}_{i,y}$, the average pollution burden facing an individual in a particular population group. We also have P_i, the share of population belong to population group i.

How does the share of pollution burden facing a population group relate to the share of population this group has in the overall population?

We define $\mathcal{E}_{i,y}$ as the share of pollution burden for population group i that is in excess of its population share as:

$$\mathcal{E}_{i,y} = \left(\left(\frac{\mathcal{Z}_{i,y} \times P_i}{\sum_{\widehat{i}}^{N} \left(\mathcal{Z}_{\widehat{i},y} \times P_{\widehat{i}} \right)} \right) \times \frac{1}{P_i} \right) - 1 = \frac{\mathcal{Z}_{i,y}}{\mathcal{Z}_y} - 1$$

Note that $\mathcal{E}_{i,y}$ is simply the ratio between the average pollution exposure for an individual in group i in year y and the overall average pollution exposure for an individual in year y, minus 1.

1.2 Simulate Population Distribution over Location and Demographics

Use the binomial distribution to generate heterogenous demographic break-down by location. There are N demographic cells, and the binomial distribution provides the probability mass in each of the N cell. Different bernoulli "win" chance for each location. There is also probability distribution over population in each location.

First, construct empty population share dataframe:

```
# 7 different age groups and 12 different locationso
it_N_pop_groups <- 100
it_M_location <- 20
# Matrix of demographics by location
mt_pop_data_frac <- matrix(data=NA, nrow=it_M_location, ncol=it_N_pop_groups)
colnames(mt_pop_data_frac) <- paste0('popgrp', seq(1,it_N_pop_groups))
rownames(mt_pop_data_frac) <- paste0('location', seq(1,it_M_location))

# For succinct visualization select subset of population groups to display
it_popgrp_disp <- 7
ar_it_popgrp_disp <- seq(1, it_N_pop_groups, length.out=it_popgrp_disp)
ar_it_popgrp_disp <- round(ar_it_popgrp_disp)
st_popgrp_disp <- paste0('(', it_popgrp_disp, ' of ', it_N_pop_groups, ' pop-groups shown)')
it_loc_disp <- 10
ar_it_loc_disp <- seq(1, it_M_location, length.out=it_loc_disp)
ar_it_loc_disp <- round(ar_it_loc_disp)
st_loc_disp <- paste0('(', it_loc_disp, ' of ', it_M_location, ' locations shown)')

# Display
st_caption = paste('Location and demographic cell',
  st_popgrp_disp, st_loc_disp, sep=" ")
mt_pop_data_frac[ar_it_loc_disp, ar_it_popgrp_disp] %>%
  kable(caption = st_caption) %>%
  kable_styling_fc()
```

Location and demographic cell (7 of 100 pop-groups shown) (10 of 20 locations shown)

	popgrp1	popgrp18	popgrp34	popgrp50	popgrp67	popgrp84	popgrp100
location1	NA	NA	NA	NA	NA	NA	NA
location3	NA	NA	NA	NA	NA	NA	NA
location5	NA	NA	NA	NA	NA	NA	NA
location7	NA	NA	NA	NA	NA	NA	NA
location9	NA	NA	NA	NA	NA	NA	NA
location12	NA	NA	NA	NA	NA	NA	NA
location14	NA	NA	NA	NA	NA	NA	NA
location16	NA	NA	NA	NA	NA	NA	NA
location18	NA	NA	NA	NA	NA	NA	NA
location20	NA	NA	NA	NA	NA	NA	NA

Second, generate conditional population distribution for each location, and then multiply by the share of population in each locality:

```
# Share of population per location
set.seed(123)
ar_p_loc <- dbinom(0:(3*it_M_location-1), 3*it_M_location-1, 0.5)
it_start <- length(ar_p_loc)/2-it_M_location/2
ar_p_loc <- ar_p_loc[it_start:(it_start+it_M_location-1)]
ar_p_loc <- ar_p_loc/sum(ar_p_loc)

# Different bernoulli "win" probability for each location
set.seed(234)
# ar_fl_unif_prob <- sort(runif(it_M_location)*(0.25)+0.4)
ar_fl_unif_prob <- sort(runif(it_M_location))

# Generate population proportion by locality
for (it_loc in 1:it_M_location ) {
  ar_p_pop_condi_loc <- dbinom(0:(it_N_pop_groups-1), it_N_pop_groups-1, ar_fl_unif_prob[it_loc])
  mt_pop_data_frac[it_loc,] <- ar_p_pop_condi_loc*ar_p_loc[it_loc]
}

# Sum of cells, should equal to 1
print(paste0('pop frac sum = ', sum(mt_pop_data_frac)))
```

```
## [1] "pop frac sum = 1"
```

```
# Display
st_caption = paste('Share of population in each location and demographic cell',
  st_popgrp_disp, st_loc_disp, sep=" ")
round((mt_pop_data_frac[ar_it_loc_disp, ar_it_popgrp_disp])*100, 3) %>%
  kable(caption=st_caption) %>%
  kable_styling_fc()
```

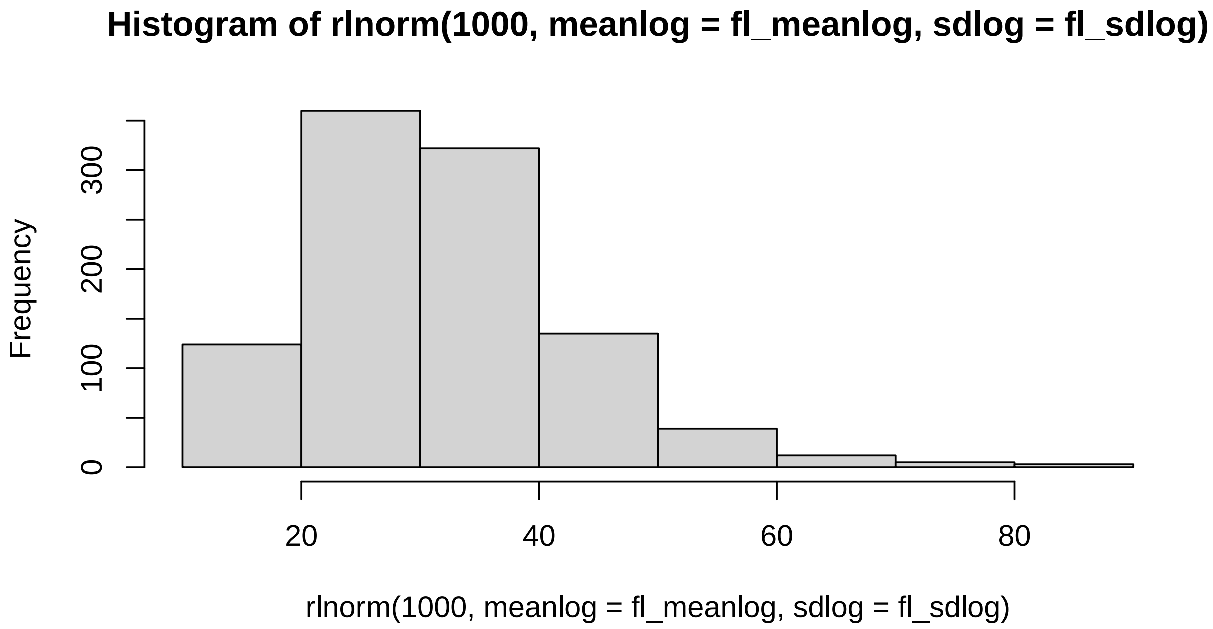
Share of population in each location and demographic cell (7 of 100 pop-groups shown) (10 of 20 locations shown)

	popgrp1	popgrp18	popgrp34	popgrp50	popgrp67	popgrp84	popgrp100
location1	0.218	0.000	0.000	0.000	0.000	0.000	0.000
location3	0.001	0.000	0.000	0.000	0.000	0.000	0.000
location5	0.000	0.009	0.122	0.000	0.000	0.000	0.000
location7	0.000	0.000	0.041	0.238	0.000	0.000	0.000
location9	0.000	0.000	0.000	0.325	0.057	0.000	0.000
location12	0.000	0.000	0.000	0.008	0.792	0.000	0.000
location14	0.000	0.000	0.000	0.000	0.194	0.059	0.000
location16	0.000	0.000	0.000	0.000	0.020	0.175	0.000
location18	0.000	0.000	0.000	0.000	0.000	0.173	0.000
location20	0.000	0.000	0.000	0.000	0.000	0.001	0.001

1.3 Simulate Environmental Exposure

Use log-normal distribution to describe average daily PM10 exposures distribution by locality:

```
fl_meanlog <- 3.4
fl_sdlog <- 0.35
hist(rlnorm(1000, meanlog = fl_meanlog, sdlog = fl_sdlog))
```



First, draw pollution measure for each locality:

```r
# draw
set.seed(123)
ar_pollution_loc <- rlnorm(it_M_location, meanlog = fl_meanlog, sdlog = fl_sdlog)
# pollution dataframe
# 5 by 3 matrix

# Column Names
ar_st_varnames <- c('location','avgdailypm10')

# Combine to tibble, add name col1, col2, etc.
tb_loc_pollution <- as_tibble(ar_pollution_loc) %>%
  rowid_to_column(var = "id") %>%
  rename_all(~c(ar_st_varnames)) %>%
  mutate(location = paste0('location', location))

# Display
st_caption = paste('PM10 Exposure across locations', st_loc_disp, sep=" ")
tb_loc_pollution[ar_it_loc_disp,] %>%
  kable(caption = st_caption) %>% kable_styling_fc()
```

Second, reshape population data:

```r
# Reshape population data, so each observation is location/demo
df_pop_data_frac_long <- as_tibble(mt_pop_data_frac, rownames='location') %>%
  pivot_longer(cols = starts_with('popgrp'),
               names_to = c('popgrp'),
               names_pattern = paste0("popgrp(.*)"),
               values_to = "pop_frac")
```

Third, join with pollution data:

PM10 Exposure across locations (10 of 20 locations shown)

location	avgdailypm10
location1	24.62676
location3	51.70466
location5	31.35114
location7	35.20967
location9	23.56121
location12	33.98553
location14	31.14765
location16	56.00380
location18	15.05461
location20	25.39426

```r
# Reshape population data, so each observation is location/demo
df_pop_pollution_long <- df_pop_data_frac_long %>%
  left_join(tb_loc_pollution, by='location')

# display
st_caption = paste('Population x Location Long Frame (15 rows shown)', sep=" ")
df_pop_pollution_long[
  round(seq(1, dim(df_pop_pollution_long)[1], length.out=15)),] %>%
  kable(caption = st_caption) %>% kable_styling_fc()
```

Population x Location Long Frame (15 rows shown)

location	popgrp	pop_frac	avgdailypm10
location1	1	0.0021767	24.62676
location2	44	0.0000000	27.64481
location3	87	0.0000000	51.70466
location5	29	0.0022435	31.35114
location6	72	0.0000000	54.61304
location8	15	0.0000000	19.24456
location9	58	0.0063374	23.56121
location10	100	0.0000000	25.63653
location12	43	0.0000004	33.98553
location13	86	0.0000427	34.47623
location15	29	0.0000000	24.66674
location16	72	0.0018508	56.00380
location18	14	0.0000000	15.05461
location19	57	0.0000000	38.30094
location20	100	0.0000065	25.39426

1.4 Compute Demographic Group Specific Exposure Distributions

What is the p10, median, p90 and mean pollution exposure for each demographic group?

1. group by population group
2. sort by pollution exposure within group
3. generate population group specific conditional population weights
4. generate population CDF for each population group (sorted by pollution)

```r
# Follow four steps above
df_pop_pollution_by_popgrp_cdf <- df_pop_pollution_long %>%
  arrange(popgrp, avgdailypm10) %>%
  group_by(popgrp) %>%
  mutate(cdf_pop_condi_popgrp_sortpm10 = cumsum(pop_frac/sum(pop_frac)),
         pmf_pop_condi_popgrp_sortpm10 = (pop_frac/sum(pop_frac)))

# Display
st_caption = paste('Distribution within groups, sorted CDFs (15 rows shown)', sep=" ")
df_pop_pollution_by_popgrp_cdf[
  round(seq(1, dim(df_pop_pollution_by_popgrp_cdf)[1], length.out=15)),] %>%
  kable(caption = st_caption) %>% kable_styling_fc_wide()
```

Distribution within groups, sorted CDFs (15 rows shown)

location	popgrp	pop_frac	avgdailypm10	cdf_pop_condi_popgrp_sortpm10	pmf_pop_condi_popgrp_sortpm10
location18	1	0.0000000	15.05461	0.0000000	0.0000000
location1	15	0.0000000	24.62676	0.0000000	0.0000000
location10	21	0.0000000	25.63653	0.0000000	0.0000000
location4	28	0.0000031	30.71275	0.0006858	0.0006854
location12	34	0.0000000	33.98553	0.2666791	0.0000000
location17	40	0.0000000	35.66778	0.8019330	0.0000000
location3	47	0.0000000	51.70466	0.9971283	0.0000000
location16	53	0.0000000	56.00380	1.0000000	0.0000001
location9	60	0.0049897	23.56121	0.2860777	0.1673161
location20	67	0.0000000	25.39426	0.1045665	0.0000000
location4	73	0.0000000	30.71275	0.2102459	0.0000000
location12	8	0.0000000	33.98553	0.1625804	0.0000000
location7	86	0.0000000	35.20967	0.6805744	0.0000000
location11	92	0.0000000	45.99021	0.9988046	0.0000000
location16	99	0.0000000	56.00380	1.0000000	0.0000002

1.5 Various Relative Burden Statistics

What to compute?

1. Excess pollution burden: Share of pollution burden by population group and overall population share, this is simply the ratio of population group mean and the overall weighted mean.
2. What is the fraction of the people in each population group with below and above overall average?
3. Merge results for different quantiles together.

1.5.1 Group-specific Means and Excess pollution burden

We compute within group means, $\mathcal{Z}_{i,y}$. We compute excess population burden, $\mathcal{E}_{i,y}$, *pm10_grp_exc_burden*. 0.10 means 10 percent in excess, this means the pollution burden share is 10 percent in excess of the population share. -0.10 means 10 percent less than what population share is.

Additionally, we compute the share of people within group above the overall mean: *pm10_grp_shr_exc*. This shows the share of people having excess burden. This complements the first number. Because the 10 percent excess could be due to very high exposure to a very small number of people within a population group, or it could be that most people in the group are in "excess".

```r
# Stats 1: excess pollution burden
df_excess_pollution_burden <- df_pop_pollution_by_popgrp_cdf %>%
  ungroup() %>%
  mutate(pm10_overall_mean = weighted.mean(avgdailypm10, pop_frac)) %>%
  group_by(popgrp) %>%
  mutate(
    popgrp_mass = sum(pop_frac), # The share of population for this group
    pm10_grp_mean = weighted.mean(avgdailypm10, pop_frac) # Pop-group mean
  ) %>%
  slice(1) %>%
  mutate(pm10_grp_exc_burden = pm10_grp_mean/pm10_overall_mean - 1) %>%
  select(popgrp, popgrp_mass,
         pm10_grp_mean, pm10_overall_mean, pm10_grp_exc_burden)
fl_pm10_overall_mean <- mean(df_excess_pollution_burden %>% pull(pm10_overall_mean))

# Stats 2: share of people within group below or above overall mean
df_share_below_or_excess <- df_pop_pollution_by_popgrp_cdf %>%
  arrange(popgrp, avgdailypm10) %>%
  filter(avgdailypm10 < fl_pm10_overall_mean) %>%
  slice_tail() %>%
  mutate(pm10_grp_shr_exc = 1 - cdf_pop_condi_popgrp_sortpm10) %>%
  select(popgrp, pm10_grp_shr_exc)
# merge stats 2 with stats 1
df_excess_pollution_burden <- df_excess_pollution_burden %>%
  left_join(df_share_below_or_excess, by="popgrp")

# display
st_caption = paste('Mean and Excess Burden by Population Groups',
  st_popgrp_disp, sep=" ")
df_excess_pollution_burden[ar_it_popgrp_disp,] %>%
  kable(caption = st_caption) %>%
  kable_styling_fc_wide()
```

Mean and Excess Burden by Population Groups (7 of 100 pop-groups shown)

popgrp	popgrp_mass	pm10_grp_mean	pm10_overall_mean	pm10_grp_exc_burden	pm10_grp_shr_exc
1	0.0028472	25.41850	33.25198	-0.2355794	0.0033669
24	0.0020059	39.83004	33.25198	0.1978245	0.3655482
39	0.0035380	40.43425	33.25198	0.2159951	0.9256668
53	0.0203775	28.38652	33.25198	-0.1463210	0.2803180
69	0.0218601	33.18168	33.25198	-0.0021141	0.6637586
84	0.0064339	34.11731	33.25198	0.0260235	0.5404550
99	0.0001213	33.03851	33.25198	-0.0064196	0.5953337

1.5.2 Within Group Percentiles

Compute within group percentiles for each population groups. Use the list of percentiles below to specify which percentiles should be computed.

```
# Stats 3: percentiles and ratios
ar_fl_percentiles <- c(0.1, 0.2, 0.8, 0.9)
# Stats 3a: generate key within group percentiles
# 1. 20th and 80th percentiles
# 2. 10th and 90th percentiles
# 3. 50th percentile
# Generate pollution quantiles by population groups
for (it_percentile_ctr in seq(1, length(ar_fl_percentiles))) {

  # Current within group percentile to compute
  fl_percentile <- ar_fl_percentiles[it_percentile_ctr]
  svr_percentile <- paste0('pm10_p', round(fl_percentile*100))

  # Frame with specific percentile
  df_within_percentiles_cur <- df_pop_pollution_by_popgrp_cdf %>%
    group_by(popgrp) %>%
    filter(cdf_pop_condi_popgrp_sortpm10 >= fl_percentile) %>%
    slice(1) %>%
    mutate(!!sym(svr_percentile) := avgdailypm10) %>%
    select(popgrp, one_of(svr_percentile))

  # Merge percentile frames together
  if (it_percentile_ctr > 1) {
    df_within_percentiles <- df_within_percentiles %>%
      left_join(df_within_percentiles_cur, by='popgrp')
  } else {
    df_within_percentiles <- df_within_percentiles_cur
  }
}

# display
st_caption = paste('PM10 Exposure Distribution by Population Groups',
  st_popgrp_disp, sep=" ")
df_within_percentiles[ar_it_popgrp_disp,] %>%
  kable(caption = st_caption) %>%
  kable_styling_fc()
```

PM10 Exposure Distribution by Population Groups (7 of 100 pop-groups shown)

popgrp	pm10_p10	pm10_p20	pm10_p80	pm10_p90
1	24.62676	24.62676	27.64481	27.64481
24	31.35114	31.35114	54.61304	54.61304
39	35.20967	35.20967	54.61304	54.61304
53	19.24456	19.24456	45.99021	45.99021
69	24.66674	31.14765	34.47623	34.47623
84	15.05461	15.05461	56.00380	56.00380
99	25.39426	25.39426	38.30094	38.30094

1.5.3 Percentiles As Excess Burden

The 80th percentile for a population group, how is this exposed relative to the mean? We simply divide the within group pollution percentiles by the overall mean across all groups. All are properly weighted.

This is relating within group percentiles to the overall mean. These can be interpreted as excess burdens at specific percentiles. Individuals at the 80th percentile of a particular population group, how does their pollution burden compare to their population share?

```
# merge stats 3 with stats 1 and 2
df_excess_pollution_burden <- df_excess_pollution_burden %>%
  left_join(df_within_percentiles, by="popgrp")

# Stats 3b: Percentiles to Relative Burdens
# Convert percentiles to be relative of overall means
for (it_percentile_ctr in seq(1, length(ar_fl_percentiles))) {
  # Current within group percentile to compute
  fl_percentile <- ar_fl_percentiles[it_percentile_ctr]
  svr_percentile <- paste0('pm10_p', round(fl_percentile*100))
  svr_perc_exc_burden <- paste0('pm10_grp_excbrd_p', round(fl_percentile*100))

  # Percentiles to excess percentiles
  df_excess_pollution_burden  <- df_excess_pollution_burden  %>%
    mutate(!!sym(svr_perc_exc_burden) := (!!sym(svr_percentile)/pm10_overall_mean) - 1)
}

# display
st_caption = paste('PM10 Within Population Group Percentiles and Excess Burden',
  st_popgrp_disp, sep=" ")
df_excess_pollution_burden[ar_it_popgrp_disp,] %>%
  select(-pm10_overall_mean,
         -starts_with('pm10_p')) %>%
  kable(caption = st_caption) %>%
  kable_styling_fc_wide()
```

PM10 Within Population Group Percentiles and Excess Burden (7 of 100 pop-groups shown)

popgrp	popgrp_mass	pm10_grp_mean	pm10_grp_exc_burden	pm10_grp_shr_exc	pm10_grp_excbrd_p10	pm10_grp_excbrd_p20	pm10_grp_excbrd_p80	pm10_grp_excbrd_p90
1	0.0028472	25.41850	-0.2355794	0.0033669	-0.2593898	-0.2593898	-0.1686267	-0.1686267
24	0.0020059	39.83004	0.1978245	0.3655482	-0.0571646	-0.0571646	0.6423998	0.6423998
39	0.0035380	40.43425	0.2159951	0.9256668	0.0588743	0.0588743	0.6423998	0.6423998
53	0.0203775	28.38652	-0.1463210	0.2803180	-0.4212506	-0.4212506	0.3830820	0.3830820
69	0.0218601	33.18168	-0.0021141	0.6637586	-0.2581875	-0.0632842	0.0368173	0.0368173
84	0.0064339	34.11731	0.0260235	0.5404550	-0.5472568	-0.5472568	0.6842246	0.6842246
99	0.0001213	33.03851	-0.0064196	0.5953337	-0.2363084	-0.2363084	0.1518395	0.1518395

1.5.4 Within Group Relative Exposure Ratios Across Percentiles

We now compute within group relative ratios of interest. This is purely within group inequality.

```
# lower and upper bound or relative within group ratios
# can only use values appearing in the percentiles list prior
ar_fl_ratio_upper <- c(0.8, 0.9)
ar_fl_ratio_lower <- c(0.2, 0.1)
# Stats 4c: Ratios
# Generate P80 to P20 ratio, and P90 to P10 standard inequality ratios
for (it_ratio_ctr in seq(1, length(ar_fl_ratio_upper))) {

  # Upper and lower percentile bounds
  fl_ratio_upper <- ar_fl_ratio_upper[it_ratio_ctr]
  fl_ratio_lower <- ar_fl_ratio_lower[it_ratio_ctr]
  svr_ratio_upper_perc <- paste0('pm10_p', round(fl_ratio_upper*100))
  svr_ratio_lower_perc <- paste0('pm10_p', round(fl_ratio_lower*100))

  # New relative within group ratio variable name
  svr_ratio <- paste0('pm10_rat_p', round(fl_ratio_upper*100), '_dvd_p', round(fl_ratio_lower*100))

  # Generate P80 to P20 ratio, etc.
  df_excess_pollution_burden  <- df_excess_pollution_burden  %>%
    mutate(!!sym(svr_ratio) := !!sym(svr_ratio_upper_perc)/!!sym(svr_ratio_lower_perc))
}

# display
st_caption = paste('PM10 Exposure within Population Group P80-P20 Inequality',
  st_popgrp_disp, sep=" ")
df_excess_pollution_burden[ar_it_popgrp_disp,] %>%
  select(-pm10_overall_mean,
         -starts_with('pm10_grp_excbrd_p'),
         -starts_with('pm10_p')) %>%
  kable(caption = st_caption) %>%
  kable_styling_fc_wide()
```

PM10 Exposure within Population Group P80-P20 Inequality (7 of 100 pop-groups shown)

popgrp	popgrp_mass	pm10_grp_mean	pm10_grp_exc_burden	pm10_grp_shr_exc	pm10_rat_p80_dvd_p20	pm10_rat_p90_dvd_p10
1	0.0028472	25.41850	-0.2355794	0.0033669	1.122552	1.122552
24	0.0020059	39.83004	0.1978245	0.3655482	1.741979	1.741979
39	0.0035380	40.43425	0.2159951	0.9256668	1.551081	1.551081
53	0.0203775	28.38652	-0.1463210	0.2803180	2.389777	2.389777
69	0.0218601	33.18168	-0.0021141	0.6637586	1.106864	1.397681
84	0.0064339	34.11731	0.0260235	0.5404550	3.720044	3.720044
99	0.0001213	33.03851	-0.0064196	0.5953337	1.508252	1.508252

1.6 Summary Statistics for Inequality across Groups

Whether we are looking at inequality within group, or inequality across groups, we can compute various aggregate inequality related statistics, including:

1. Gini
2. Atkinson
3. Standard deviation
4. P80 to P20 ratio

The literature has focused on Atkinson statistics computed over mean pollution exposures across groups. When we focus on variations across groups, we compute these based on group means. When we focus on variations within groups, we compute group-specific GINI, Atkinson, etc based on within group data.

Earlier, we have already computed the P80 to P20 ratio for each population group.

1.6.1 Gini, Atkinson, and S.D. Functions

For variations across groups, note that the GINI and ATKINSON require positive values for inputs, so they can be used with $\mathcal{Z}_{i,y}$, but they can not be applied to the $\mathcal{E}_{i,y}$, which is positive or negative. For variations across groups, we compute these statistics given the distribution of P_i.

First, we generate an internal function for GINI:

$$\text{GINI Index} = 1 - \frac{2}{N+1} \cdot \left(\sum_{i=1}^{N} \sum_{j=1}^{i} x_j \right) \cdot \left(\sum_{i=1}^{N} x_i \right)^{-1}$$

We use the Gini index which is documented in on the fs_gini_disc page.

```
ffi_dist_gini_random_var_pos_test <- function(ar_data, ar_prob_data) {
  #' @param ar_data array sorted array values low to high
  #' @param ar_prob_data array probability mass for each element along `ar_data`, sums to 1

  fl_mean <- sum(ar_data*ar_prob_data);
  ar_mean_cumsum <- cumsum(ar_data*ar_prob_data);
  ar_height <- ar_mean_cumsum/fl_mean;
  fl_area_drm <- sum(ar_prob_data*ar_height);
  fl_area_below45 <- sum(ar_prob_data*(cumsum(ar_prob_data)/sum(ar_prob_data)))
  fl_gini_index <- (fl_area_below45-fl_area_drm)/fl_area_below45
```

```r
  return(fl_gini_index)
}
```

Second, we create an internal function for Atkinson statistics, from Atkinson (JET, 1970). The formula is:

$$\text{Atkinson Index} = A\left(\left\{Y_i\right\}_{i=1}^{N}, \lambda\right) = 1 - \left(\sum_{i=1}^{N}\frac{1}{N}\left(\frac{Y_i}{\sum_{j=1}^{N}\left(\frac{Y_j}{N}\right)}\right)^{\lambda}\right)^{\frac{1}{\lambda}} \in [0, 1]$$

Note that the Atkinson statistics differ by planner preference, and is arbitrary. We use the Atkinson index formula which is documented on the fs_atkinson_ces page.

```r
# Formula
ffi_atkinson_random_var_ineq <- function(ar_data, ar_prob_data, fl_rho) {
  #' @param ar_data array sorted array values
  #' @param ar_prob_data array probability mass for each element along `ar_data`, sums to 1
  #' @param fl_rho float inequality aversion parameter fl_rho = 1 for planner
  #' without inequality aversion. fl_rho = -infinity for fully inequality averse.

  fl_mean <- sum(ar_data*ar_prob_data);
  fl_atkinson <- 1 - (sum(ar_prob_data*(ar_data^{fl_rho}))^(1/fl_rho))/fl_mean
  return(fl_atkinson)
}
```

Third, we also create an internal function for standard deviation for discrete random variable.

```r
# Formula
ffi_std_drv <- function(ar_data, ar_prob_data) {
  #' @param ar_data array array values
  #' @param ar_prob_data array probability mass for each element along `ar_data`, sums to 1

  fl_mean <- sum(ar_data*ar_prob_data)
  fl_std <- sqrt(sum(ar_prob_data*(ar_data - fl_mean)^2))
  return(fl_std)
}
```

1.6.2 Inequality in Group Means

We compute a number of aggregate statistics over the means of the subgroups. We can only compute 1 gini and 1 standard deviation, but we have a spectrum of Atkinson index values, depending in the inequality aversion value picked.

First, we get the data inputs we need.

```r
# 1. SORT FIRST!
df_excess_pollution_burden_sorted <- df_excess_pollution_burden %>% arrange(pm10_grp_mean)
# 2. Obtain the means across groups, and also excess burden across groups
ar_data_grp_means <- df_excess_pollution_burden_sorted %>% pull(pm10_grp_mean)
ar_data_grp_exc_burden <- df_excess_pollution_burden_sorted %>% pull(pm10_grp_exc_burden)
# 3. Obtain the probability mass for each group
ar_data_grp_shares <- df_excess_pollution_burden_sorted %>% pull(popgrp_mass)
```

Second, we compute the GINI and STD over group-specific means, with group-specific population shares.

```r
# compute gini over group means, and standard deviations
fl_grp_means_gini <- ffi_dist_gini_random_var_pos_test(ar_data_grp_means, ar_data_grp_shares)
# STD
fl_grp_means_std <- ffi_std_drv(ar_data_grp_means, ar_data_grp_shares)
```

Third, compute an array of Atkinson index with differing inequality aversions.

```
# Log10 scaled Inequality Measures
ar_rho <- 1 - (10^(c(seq(-2,2, length.out=30))))
tb_rho <- as_tibble(unique(ar_rho))
# Array of atkinson values
ar_grp_means_atkinson <- apply(tb_rho, 1, function(row){
  ffi_atkinson_random_var_ineq(ar_data_grp_means, ar_data_grp_shares, row[1])})
# atkinson results table
ar_st_varnames <- c('id','rho','atkinson_index')
tb_atkinson <- as_tibble(cbind(ar_rho, ar_grp_means_atkinson)) %>%
  rowid_to_column(var = "id") %>%
  rename_all(~c(ar_st_varnames)) %>%
  mutate(one_minus_rho = 1 - rho)
# Max Atkinson
fl_atkinson_max <- max(tb_atkinson %>% pull(atkinson_index))
# display
it_rows_shown <- 10
st_caption <- paste0('Atkinson Inequality Index',
  '(', it_rows_shown, ' of ', length(ar_rho), ' inequality preferences shown)')
tb_atkinson[round(seq(1, length(ar_rho), length.out = it_rows_shown)),] %>%
  kable(caption = st_caption) %>%
  kable_styling_fc()
```

Atkinson Inequality Index(10 of 30 inequality preferences shown)

id	rho	atkinson_index	one_minus_rho
1	0.9900000	0.0000826	0.0100000
4	0.9740706	0.0002141	0.0259294
7	0.9327664	0.0005545	0.0672336
11	0.7604973	0.0019672	0.2395027
14	0.3789831	0.0050568	0.6210169
17	-0.6102620	0.0128582	1.6102620
20	-3.1753189	0.0324195	4.1753189
24	-13.8735211	0.1336964	14.8735211
27	-37.5662042	0.2671924	38.5662042
30	-99.0000000	0.3236560	100.0000000

Fourth, compute the standard deviation of excess burden. Note that excess burden is both positive and negative, so we can not compute GINI or Atkinson statistics from it.

```
fl_grp_exc_burden_std <- ffi_std_drv(ar_data_grp_exc_burden, ar_data_grp_shares)
```

1.6.3 Visualize Inequality in Group Means (GINI and ATKINSON)

Now we can present what the existing literature focuses on, which is to compute Atkinson index over group means. As can be seen in the figure below, this index value is arbitrary depending on inequality aversion. Additionally, this value shows in this testing example, very low inequality in pollution exposure. Additionally, we also have the GINI index, which can be thought of as mapping to a particular level of inequality aversion.

Potentially, an issue with using these types of measures is that they are hard to interpret, potentially arbitrary, and potentially generates misleading impressions. In the test examples here, there is almost equality using GINI and Atkinson measures.

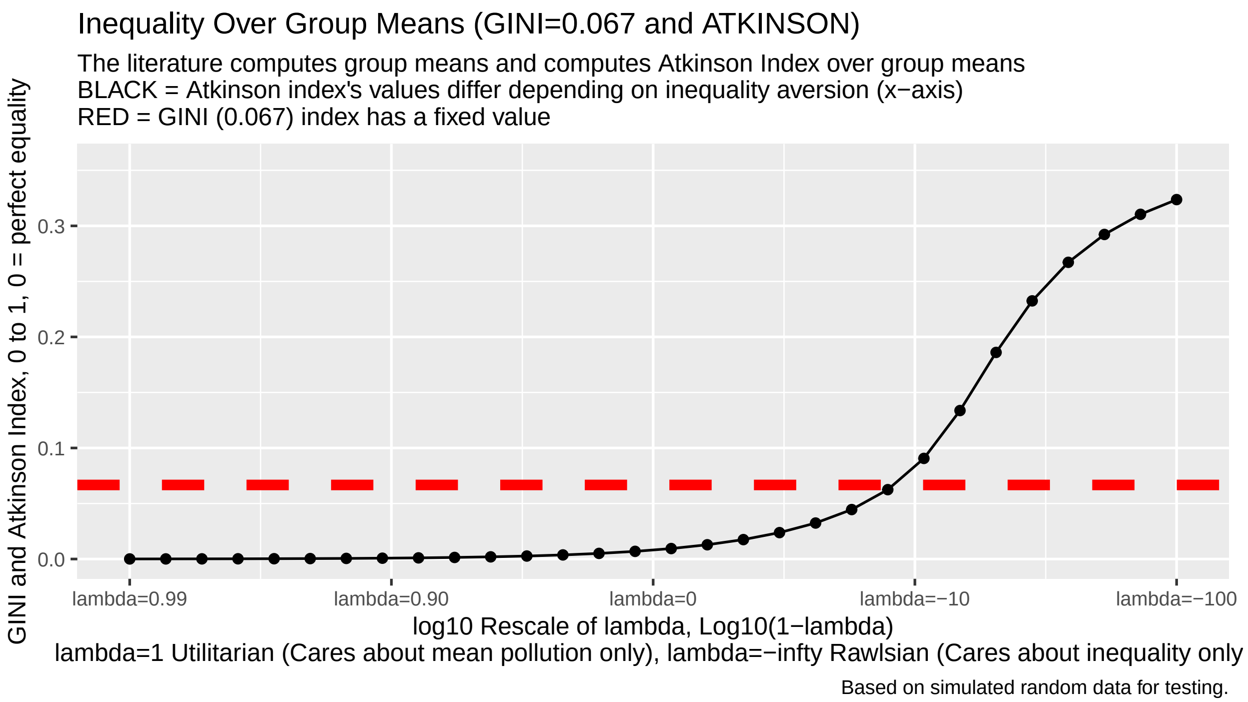
Our excess pollution burden statistics, however, provides a more clearly interpretable lense for looking at across group inequality. Additionally, we also look at within group inequality.

```
# x-labels
x.labels <- c('lambda=0.99', 'lambda=0.90', 'lambda=0', 'lambda=-10', 'lambda=-100')
x.breaks <- c(0.01, 0.10, 1, 10, 100)

# title line 2
fl_grp_means_gini_fmt <- round(fl_grp_means_gini, 3)
st_title <- paste0("Inequality Over Group Means (GINI=", fl_grp_means_gini_fmt," and ATKINSON)")
title_line1 <- paste0("The literature computes group means and computes Atkinson Index over group means
title_line2 <- paste0("BLACK = Atkinson index's values differ depending on inequality aversion (x-axis)
title_line3 <- paste0("RED = GINI (", fl_grp_means_gini_fmt,") index has a fixed value")

# Graph Results--Draw
pl_gini_atkinson <- tb_atkinson %>%
  ggplot(aes(x=one_minus_rho, y=atkinson_index)) +
  geom_line() +
  geom_point() +
  geom_hline(yintercept=fl_grp_means_gini, linetype='dashed', color='red', size=2) +
  # geom_vline(xintercept=c(1), linetype="dotted") +
  labs(title = st_title,
       subtitle = paste0(title_line1, '\n', title_line2, '\n', title_line3),
       x = 'log10 Rescale of lambda, Log10(1-lambda)\nlambda=1 Utilitarian (Cares about mean pollution
       y = paste0('GINI and Atkinson Index, 0 to 1, 0 = perfect equality'),
       caption = 'Based on simulated random data for testing.') +
  scale_x_continuous(trans='log10', labels = x.labels, breaks = x.breaks) +
  ylim(0, max(min(fl_atkinson_max*1.1, 1), fl_grp_means_gini)) +
  theme(text = element_text(size = 10),
        legend.position="right")

# Print
print(pl_gini_atkinson)
```

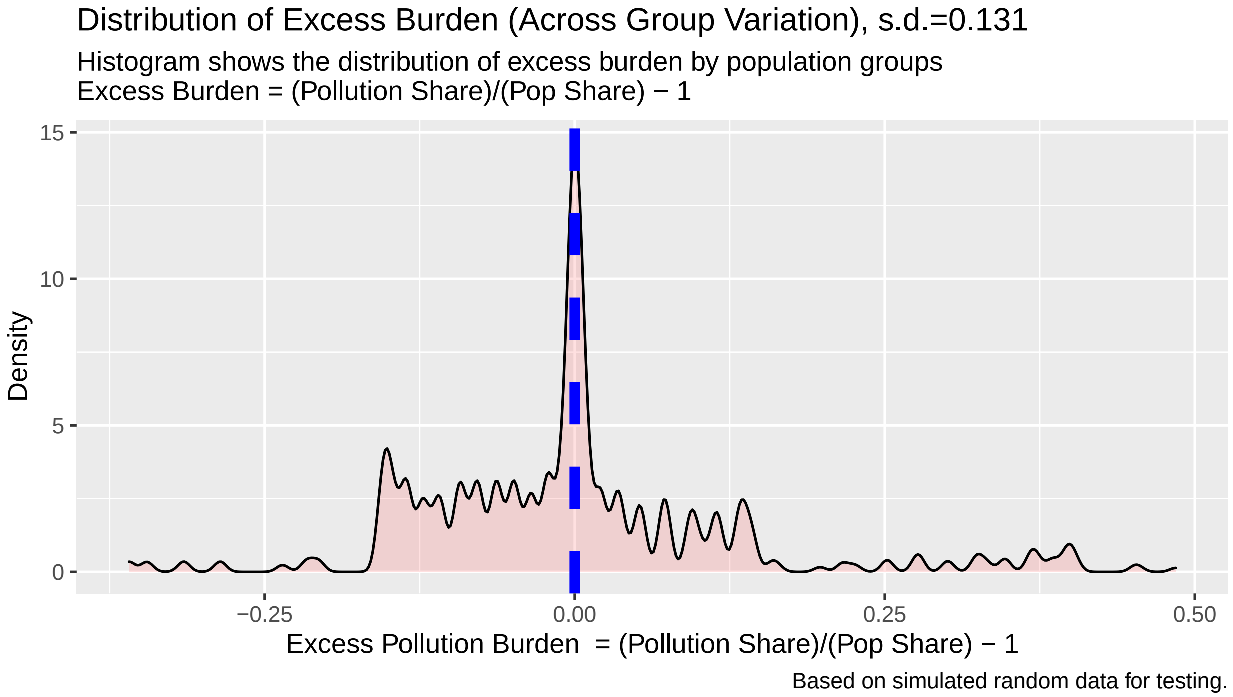


1.6.4 Visualize Inequality in Group Means (Excess Burden)

Now we visualize excess pollution burden as a distribution, this is our preferred statistics to look at variations in means across groups.

```
# Rounding excess burden s.d.
fl_grp_exc_burden_std_fmt <- round(fl_grp_exc_burden_std, 3)
st_title <- paste0("Distribution of Excess Burden (Across Group Variation), s.d.=", fl_grp_exc_burden_s
title_line1 <- paste0("Histogram shows the distribution of excess burden by population groups")
title_line2 <- paste0("Excess Burden = (Pollution Share)/(Pop Share) - 1")

# Generate a Data Sample by Drawing from the Distribution
it_sample_draws <- 1e6
ar_it_draws <- sample(1:it_N_pop_groups, it_sample_draws, replace=TRUE, prob=ar_data_grp_shares)
ar_sample_draws <- ar_data_grp_exc_burden[ar_it_draws]
# Draw histogram
pl_excess_burden <- as_tibble(ar_sample_draws) %>%
  ggplot(aes(x=value)) +
  # geom_histogram(aes(y=..density..),
  #                colour="darkblue", fill="lightblue")+
  geom_density(alpha=.2, fill="#FF6666") +
  geom_vline(aes(xintercept=0),
             color="blue", linetype="dashed", size=2) +
  labs(title = st_title,
       subtitle = paste0(title_line1, '\n', title_line2),
       x = 'Excess Pollution Burden  = (Pollution Share)/(Pop Share) - 1',
       y = 'Density',
       caption = 'Based on simulated random data for testing.')
# Print
print(pl_excess_burden)
```



Distribution of Excess Burden (Across Group Variation), s.d.=0.131

Histogram shows the distribution of excess burden by population groups
Excess Burden = (Pollution Share)/(Pop Share) − 1

Based on simulated random data for testing.