Basic CAR Model

Alvin Sheng

```
library(here)
```

```
## here() starts at /Users/Alvin/Documents/NCSU_Fall_2021/NIH_SIP/flood-risk-health-effects
```

```
library(coda)
library(CARBayes)
```

```
## Loading required package: MASS
```

```
## Loading required package: Rcpp
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
## v purrr   0.3.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x dplyr::select() masks MASS::select()
```

```
i_am("reports/basic_CAR_model_all_census_tract.Rmd")
```

```
## here() starts at /Users/Alvin/Documents/NCSU_Fall_2021/NIH_SIP/flood-risk-health-effects
```

```
fhs_model_df <- readRDS("intermediary_data/fhs_model_df_all_census_tract_pc.rds")
```

CAR model results, Coronary Heart Disease

Inference is based on 3 markov chains, each of which has been run for 110000 samples, the first 10000 of which has been removed for burn-in. The remaining 100000 samples are thinned by 2, resulting in 150000 samples for inference across the 3 Markov chains.

```
load(here("modeling_files/all_census_tract_intrinsic.RData"))
```

Model Diagnostics

Beta samples

```
beta_samples <- mcmc.list(chain1$samples$beta, chain2$samples$beta,
                          chain3$samples$beta)
```

```
effectiveSize(beta_samples)
```

```
##        var1        var2        var3        var4        var5        var6 
## 122395.4428  14834.0232  14514.9449  16624.4712  13932.7613  41445.1098 
##        var7        var8        var9       var10       var11       var12 
##  72533.6250  29406.4903  40877.7858  33876.1884  49370.1521  48196.8360 
##       var13       var14       var15       var16       var17       var18 
##  73092.0796  23545.2961  35603.4982  46107.9744  28291.1472  53502.3093 
##       var19       var20       var21       var22       var23       var24 
##  28947.9971  75385.3056  40866.2096   8759.4964   5230.2563    316.8711 
##       var25       var26       var27       var28       var29       var30 
##   3185.0408   1672.8907   2714.2912    945.3198    187.3544    703.9630 
##       var31       var32 
##   1158.1092  18763.5043
```

```
plot(beta_samples)
```

```
gelman.diag(beta_samples)
```

```
## Potential scale reduction factors:
## 
##       Point est. Upper C.I.
##  [1,]       1.00       1.00
##  [2,]       1.00       1.00
##  [3,]       1.00       1.00
##  [4,]       1.00       1.00
##  [5,]       1.00       1.00
##  [6,]       1.00       1.00
##  [7,]       1.00       1.00
##  [8,]       1.00       1.00
##  [9,]       1.00       1.00
## [10,]       1.00       1.00
## [11,]       1.00       1.00
## [12,]       1.00       1.00
## [13,]       1.00       1.00
## [14,]       1.00       1.00
## [15,]       1.00       1.00
## [16,]       1.00       1.00
## [17,]       1.00       1.00
## [18,]       1.00       1.00
## [19,]       1.00       1.00
## [20,]       1.00       1.00
## [21,]       1.00       1.00
## [22,]       1.00       1.00
## [23,]       1.00       1.00
## [24,]       1.02       1.06
## [25,]       1.01       1.02
## [26,]       1.00       1.01
## [27,]       1.00       1.01
## [28,]       1.06       1.17
## [29,]       1.14       1.43
## [30,]       1.01       1.03
## [31,]       1.00       1.01
## [32,]       1.00       1.00
##
## Multivariate psrf
##
## 1.08
```

Examining sigma2, nu2, rho

```
sigma2_samples <- mcmc.list(chain1$samples$sigma2, chain2$samples$sigma2,
                            chain3$samples$sigma2)

nu2_samples <- mcmc.list(chain1$samples$nu2, chain2$samples$nu2,
                         chain3$samples$nu2)
```

```
plot(sigma2_samples)

plot(nu2_samples)
```

```
gelman.diag(sigma2_samples)
```

```
## Potential scale reduction factors:
##
##      Point est. Upper C.I.
## [1,]          1          1
```

```
gelman.diag(nu2_samples)
```

```
## Potential scale reduction factors:
##
##      Point est. Upper C.I.
## [1,]          1          1
```

Examining a sample of the 3108 phi parameters

```
phi_samples <- mcmc.list(chain1$samples$phi, chain2$samples$phi, chain3$samples$phi)
```

```
set.seed(1157, kind = "Mersenne-Twister", normal.kind = "Inversion", sample.kind = "Rejection")

phi_subset_idx <- sample(1:ncol(phi_samples[[1]]), size = 10)

phi_samples_subset <- phi_samples[, phi_subset_idx]
```

```
plot(phi_samples_subset)
```

```
gelman.diag(phi_samples_subset)
```

```
## Potential scale reduction factors:
##
##       Point est. Upper C.I.
##  [1,]       1.00       1.00
##  [2,]       1.01       1.04
##  [3,]       1.00       1.01
##  [4,]       1.00       1.01
##  [5,]       1.00       1.00
##  [6,]       1.00       1.00
##  [7,]       1.00       1.00
##  [8,]       1.00       1.00
##  [9,]       1.00       1.00
## [10,]       1.01       1.05
##
## Multivariate psrf
##
## 1.02
```

Inference

```
beta_samples_matrix <- rbind(chain1$samples$beta, chain2$samples$beta, chain3$samples$beta)

colnames(beta_samples_matrix) <- c("Intercept", names(fhs_model_df[, 14:(ncol(fhs_model_df) - 4)]))
```

```
(beta_inference <- round(t(apply(beta_samples_matrix, 2, quantile, c(0.5, 0.025, 0.975))),5))
```

```
##                            50%     2.5%    97.5%
## Intercept            6.66082  6.65654  6.66509
## flood_risk_pc1      -0.03796 -0.04939 -0.02670
## flood_risk_pc2       0.00311 -0.00976  0.01592
## flood_risk_pc3      -0.00011 -0.00952  0.00920
## flood_risk_pc4       0.00796 -0.00205  0.01809
## EP_POV               0.31386  0.30164  0.32608
## EP_UNEMP             0.02996  0.02201  0.03795
## EP_PCI              -0.03638 -0.04817 -0.02446
## EP_NOHSDP            0.19421  0.17865  0.20972
## EP_AGE65             1.38014  1.37003  1.39019
## EP_AGE17             0.27891  0.26833  0.28953
## EP_DISABL            0.27029  0.26027  0.28038
## EP_SNGPNT           -0.06524 -0.07440 -0.05606
## EP_MINRTY           -0.03910 -0.05446 -0.02378
## EP_LIMENG           -0.06155 -0.07536 -0.04761
## EP_MUNIT            -0.05717 -0.06613 -0.04824
## EP_MOBILE            0.08001  0.07175  0.08817
## EP_CROWD            -0.04625 -0.05670 -0.03576
## EP_NOVEH             0.12710  0.11361  0.14054
## EP_GROUPQ           -0.09404 -0.10105 -0.08701
## EP_UNINSUR           0.10428  0.09387  0.11470
## co                  -0.14573 -0.18216 -0.10871
## no2                 -0.06302 -0.11019 -0.01500
## o3                  -0.13046 -0.20641 -0.05363
## pm10                -0.16501 -0.19714 -0.13252
## pm25                 0.46428  0.41576  0.51208
## so2                  0.03227  0.00044  0.06431
## summer_tmmx          0.06927  0.01626  0.12225
## winter_tmmx         -0.22816 -0.36825 -0.09249
## summer_rmax         -0.04795 -0.11669  0.02046
## winter_rmax          0.04809 -0.00107  0.09999
## Data_Value_CSMOKING  0.78516  0.76514  0.80537
```

List of significant beta coefficients:

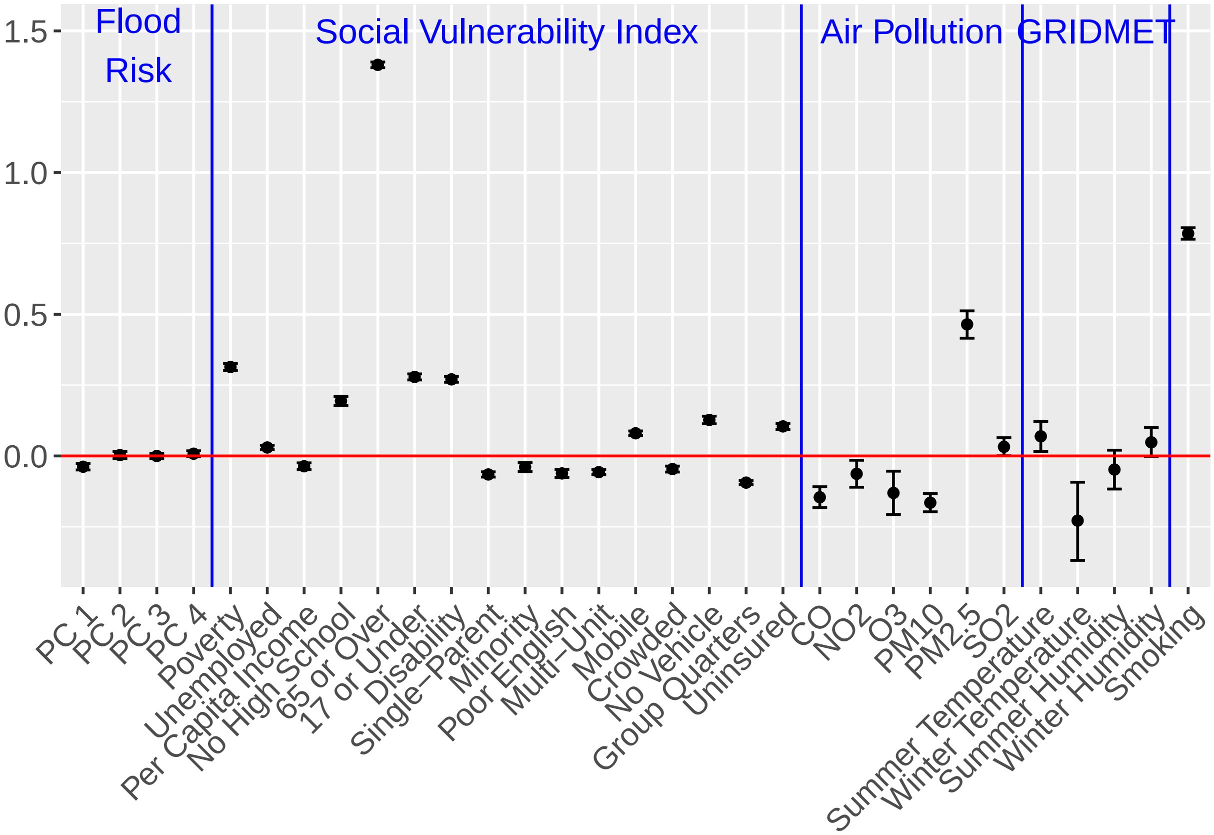
```
colnames(beta_samples_matrix)[sign(beta_inference[, 2]) == sign(beta_inference[, 3])]
```

```
##  [1] "Intercept"           "flood_risk_pc1"      "EP_POV"
##  [4] "EP_UNEMP"            "EP_PCI"              "EP_NOHSDP"
##  [7] "EP_AGE65"            "EP_AGE17"            "EP_DISABL"
## [10] "EP_SNGPNT"           "EP_MINRTY"           "EP_LIMENG"
## [13] "EP_MUNIT"            "EP_MOBILE"           "EP_CROWD"
## [16] "EP_NOVEH"            "EP_GROUPQ"           "EP_UNINSUR"
## [19] "co"                  "no2"                 "o3"
## [22] "pm10"                "pm25"                "so2"
## [25] "summer_tmmx"         "winter_tmmx"         "Data_Value_CSMOKING"
```

Credible Interval plots for the coefficients, in ggplot

```
# first, process the beta_inference matrix in a form ggplot can understand

beta_inference_df <- as.data.frame(beta_inference)

beta_inference_df <- mutate(beta_inference_df, var_name = row.names(beta_inference_df))

beta_inference_df <- rename(beta_inference_df,
                            post_median = `50%`,
                            post_2.5 = `2.5%`,
                            post_97.5 = `97.5%`)

beta_inference_df$var_name <- factor(beta_inference_df$var_name, levels = beta_inference_df$var_name)
```

```
ggplot(beta_inference_df[-1, ], aes(x = var_name, y = post_median)) +
  geom_point() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1), axis.title.x = element_blank(), axi
        axis.text=element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 10)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_vline(xintercept = c(4.5, 20.5, 26.5, 30.5), col = "blue") +
  geom_hline(yintercept = 0, col = "red") +
  annotate(geom = "text", x = 2.5, y = 1.45, label = "Flood\nRisk",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 12.5, y = 1.5, label = "Social Vulnerability Index",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 23.5, y = 1.5, label = "Air Pollution",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 28.5, y = 1.5, label = "GRIDMET",
           col = "blue", size = 4.5) +
  scale_x_discrete(labels = c("PC 1", "PC 2", "PC 3", "PC 4",
                              "Poverty", "Unemployed", "Per Capita Income", "No High School",
                              "65 or Over", "17 or Under", "Disability",
                              "Single-Parent", "Minority", "Poor English",
                              "Multi-Unit", "Mobile", "Crowded",
                              "No Vehicle", "Group Quarters", "Uninsured",
                              "CO", "NO2", "O3", "PM10", "PM2.5", "SO2",
                              "Summer Temperature", "Winter Temperature", "Summer Humidity", "Winter Hu
                              "Smoking"))
```



CAR model results, High Blood Pressure

Inference is based on 3 markov chains, each of which has been run for 110000 samples, the first 10000 of which has been removed for burn-in. The remaining 100000 samples are thinned by 2, resulting in 150000 samples for inference across the 3 Markov chains.

```
load(here("modeling_files/all_census_tract_BPHIGH.RData"))
```

Model Diagnostics

Beta samples

```
beta_samples <- mcmc.list(chain1$samples$beta, chain2$samples$beta,
                          chain3$samples$beta)
```

```
effectiveSize(beta_samples)
```

```
##       var1       var2       var3       var4       var5       var6       var7 
## 90874.2889 11916.3260 11242.0264 13681.4463 11197.4613 33048.6366 55944.0464
##       var8       var9      var10      var11      var12      var13      var14
## 24652.9603 33339.5924 27756.7699 39253.2624 41924.3639 56571.4180 19019.6674
##      var15      var16      var17      var18      var19      var20      var21
## 28746.5291 36569.7752 25248.2078 42232.9286 25073.7672 55251.9513 34329.3392
##      var22      var23      var24      var25      var26      var27      var28
##  7559.2706  4276.6832   252.3899  2712.1838  1630.8185  2214.1442   747.5273
##      var29      var30      var31      var32
##   149.2598   607.6737   928.1827 16333.5196
```

```
plot(beta_samples)
```

```
gelman.diag(beta_samples)
```

```
## Potential scale reduction factors:
## 
##       Point est. Upper C.I.
##  [1,]       1.00       1.00
##  [2,]       1.00       1.01
##  [3,]       1.00       1.00
##  [4,]       1.00       1.00
##  [5,]       1.00       1.00
##  [6,]       1.00       1.00
##  [7,]       1.00       1.00
##  [8,]       1.00       1.00
##  [9,]       1.00       1.00
## [10,]       1.00       1.00
## [11,]       1.00       1.00
## [12,]       1.00       1.00
## [13,]       1.00       1.00
## [14,]       1.00       1.00
## [15,]       1.00       1.00
## [16,]       1.00       1.00
## [17,]       1.00       1.00
## [18,]       1.00       1.00
## [19,]       1.00       1.00
## [20,]       1.00       1.00
## [21,]       1.00       1.00
## [22,]       1.00       1.00
## [23,]       1.00       1.00
## [24,]       1.02       1.07
## [25,]       1.01       1.03
## [26,]       1.00       1.01
## [27,]       1.00       1.01
## [28,]       1.08       1.24
## [29,]       1.17       1.53
## [30,]       1.01       1.03
## [31,]       1.01       1.02
## [32,]       1.00       1.00
## 
## Multivariate psrf
## 
## 1.11
```

Examining sigma2, nu2, rho

```
sigma2_samples <- mcmc.list(chain1$samples$sigma2, chain2$samples$sigma2,
                            chain3$samples$sigma2)

nu2_samples <- mcmc.list(chain1$samples$nu2, chain2$samples$nu2,
                         chain3$samples$nu2)
```

```
plot(sigma2_samples)

plot(nu2_samples)
```

```
gelman.diag(sigma2_samples)
```

```
## Potential scale reduction factors:
##
##      Point est. Upper C.I.
## [1,]          1          1
```

```
gelman.diag(nu2_samples)
```

```
## Potential scale reduction factors:
##
##      Point est. Upper C.I.
## [1,]          1          1
```

Examining a sample of the 3108 phi parameters

```
phi_samples <- mcmc.list(chain1$samples$phi, chain2$samples$phi, chain3$samples$phi)
```

```
set.seed(1157, kind = "Mersenne-Twister", normal.kind = "Inversion", sample.kind = "Rejection")

phi_subset_idx <- sample(1:ncol(phi_samples[[1]]), size = 10)

phi_samples_subset <- phi_samples[, phi_subset_idx]
```

```
plot(phi_samples_subset)
```

```
gelman.diag(phi_samples_subset)
```

```
## Potential scale reduction factors:
##
##       Point est. Upper C.I.
##  [1,]       1.00       1.00
##  [2,]       1.02       1.06
##  [3,]       1.00       1.01
##  [4,]       1.01       1.02
##  [5,]       1.00       1.00
##  [6,]       1.00       1.00
##  [7,]       1.00       1.01
##  [8,]       1.00       1.00
##  [9,]       1.00       1.00
## [10,]       1.02       1.06
##
## Multivariate psrf
```

```
##
## 1.03
```

Inference

```
beta_samples_matrix <- rbind(chain1$samples$beta, chain2$samples$beta, chain3$samples$beta)

colnames(beta_samples_matrix) <- c("Intercept", names(fhs_model_df[, 14:(ncol(fhs_model_df) - 4)]))
```

```
(beta_inference <- round(t(apply(beta_samples_matrix, 2, quantile, c(0.5, 0.025, 0.975))),5))
```

```
##                          50%     2.5%    97.5%
## Intercept           32.32342 32.31205 32.33477
## flood_risk_pc1      -0.07680 -0.11546 -0.03870
## flood_risk_pc2       0.04663  0.00330  0.08970
## flood_risk_pc3       0.00411 -0.02739  0.03534
## flood_risk_pc4       0.04094  0.00694  0.07534
## EP_POV               0.03648 -0.00285  0.07556
## EP_UNEMP             0.12803  0.10291  0.15324
## EP_PCI               0.25355  0.21521  0.29205
## EP_NOHSDP           -0.00312 -0.05265  0.04649
## EP_AGE65             4.06404  4.03162  4.09634
## EP_AGE17             0.60874  0.57506  0.64259
## EP_DISABL            0.74081  0.70905  0.77269
## EP_SNGPNT           -0.05062 -0.07956 -0.02165
## EP_MINRTY            2.64160  2.59091  2.69217
## EP_LIMENG           -0.88599 -0.93042 -0.84117
## EP_MUNIT            -0.60704 -0.63569 -0.57853
## EP_MOBILE            0.12289  0.09645  0.14889
## EP_CROWD            -0.12313 -0.15629 -0.08988
## EP_NOVEH             0.61286  0.56973  0.65575
## EP_GROUPQ           -0.58794 -0.61017 -0.56571
## EP_UNINSUR           0.23415  0.20104  0.26738
## co                  -0.72013 -0.84205 -0.59647
## no2                 -0.67157 -0.83123 -0.50705
## o3                  -0.44066 -0.71679 -0.15979
## pm10                -0.52379 -0.63431 -0.41206
## pm25                 1.17066  1.00529  1.33484
## so2                  0.03595 -0.07470  0.14774
## summer_tmmx          0.06112 -0.13118  0.25235
## winter_tmmx         -0.54855 -1.06733 -0.06381
## summer_rmax         -0.29192 -0.53561 -0.04895
## winter_rmax          0.17798  0.00289  0.36431
## Data_Value_CSMOKING  2.52585  2.46083  2.59144
```

List of significant beta coefficients:

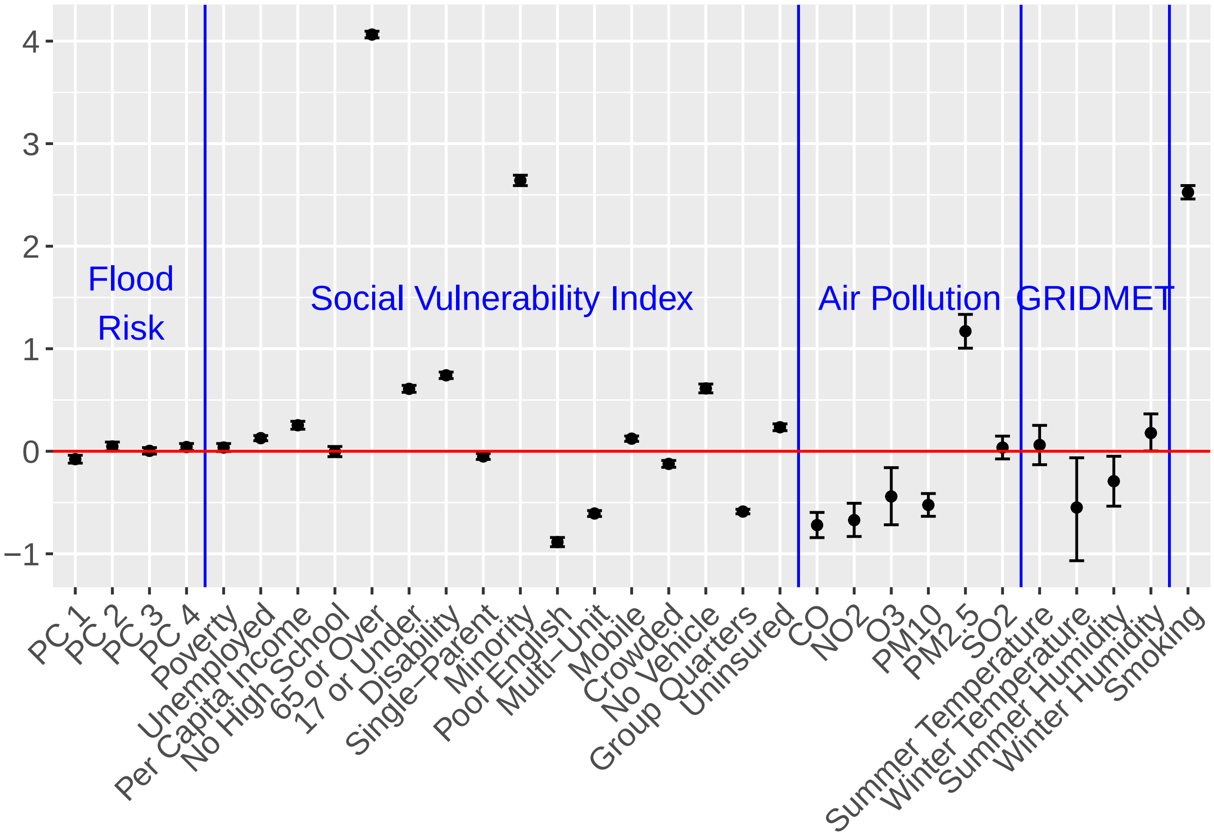
```
colnames(beta_samples_matrix)[sign(beta_inference[, 2]) == sign(beta_inference[, 3])]
```

```
##  [1] "Intercept"           "flood_risk_pc1"      "flood_risk_pc2"
##  [4] "flood_risk_pc4"      "EP_UNEMP"            "EP_PCI"
##  [7] "EP_AGE65"            "EP_AGE17"            "EP_DISABL"
## [10] "EP_SNGPNT"           "EP_MINRTY"           "EP_LIMENG"
## [13] "EP_MUNIT"            "EP_MOBILE"           "EP_CROWD"
## [16] "EP_NOVEH"            "EP_GROUPQ"           "EP_UNINSUR"
## [19] "co"                  "no2"                 "o3"
## [22] "pm10"                "pm25"                "winter_tmmx"
## [25] "summer_rmax"         "winter_rmax"         "Data_Value_CSMOKING"
```

Credible Interval plots for the coefficients, in ggplot

```
# first, process the beta_inference matrix in a form ggplot can understand

beta_inference_df <- as.data.frame(beta_inference)

beta_inference_df <- mutate(beta_inference_df, var_name = row.names(beta_inference_df))

beta_inference_df <- rename(beta_inference_df,
                            post_median = `50%`,
                            post_2.5 = `2.5%`,
                            post_97.5 = `97.5%`)

beta_inference_df$var_name <- factor(beta_inference_df$var_name, levels = beta_inference_df$var_name)
```

```
ggplot(beta_inference_df[-1, ], aes(x = var_name, y = post_median)) +
  geom_point() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1), axis.title.x = element_blank(), axi
        axis.text=element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 10)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_vline(xintercept = c(4.5, 20.5, 26.5, 30.5), col = "blue") +
  geom_hline(yintercept = 0, col = "red") +
  annotate(geom = "text", x = 2.5, y = 1.45, label = "Flood\nRisk",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 12.5, y = 1.5, label = "Social Vulnerability Index",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 23.5, y = 1.5, label = "Air Pollution",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 28.5, y = 1.5, label = "GRIDMET",
           col = "blue", size = 4.5) +
  scale_x_discrete(labels = c("PC 1", "PC 2", "PC 3", "PC 4",
                              "Poverty", "Unemployed", "Per Capita Income", "No High School",
                              "65 or Over", "17 or Under", "Disability",
                              "Single-Parent", "Minority", "Poor English",
                              "Multi-Unit", "Mobile", "Crowded",
                              "No Vehicle", "Group Quarters", "Uninsured",
                              "CO", "NO2", "O3", "PM10", "PM2.5", "SO2",
                              "Summer Temperature", "Winter Temperature", "Summer Humidity", "Winter Hu
                              "Smoking"))
```



CAR model results, Asthma

Inference is based on 3 markov chains, each of which has been run for 110000 samples, the first 10000 of which has been removed for burn-in. The remaining 100000 samples are thinned by 2, resulting in 150000 samples for inference across the 3 Markov chains.

```
load(here("modeling_files/all_census_tract_CASTHMA.RData"))
```

Model Diagnostics

Beta samples

```
beta_samples <- mcmc.list(chain1$samples$beta, chain2$samples$beta,
                          chain3$samples$beta)
```

```
effectiveSize(beta_samples)
```

```
##       var1       var2       var3       var4       var5       var6       var7
## 51337.7947  8205.8700  7710.4057  9736.1827  7725.8839 23151.5188 36343.0519
##       var8       var9      var10      var11      var12      var13      var14
## 17265.6247 22883.0159 21516.9361 26211.9705 28512.0875 36943.5300 13279.9552
##      var15      var16      var17      var18      var19      var20      var21
## 19453.9034 24953.7603 19304.7118 28385.7025 17865.6638 35989.6374 24257.5261
##      var22      var23      var24      var25      var26      var27      var28
##  5504.3918  3057.1886   178.8545  2077.6263  1259.9858  1555.1677   565.7546
##      var29      var30      var31      var32
##   106.9605   451.8046   663.8760 11761.9584
```

```
plot(beta_samples)
```

```
gelman.diag(beta_samples)
```

```
## Potential scale reduction factors:
##
##       Point est. Upper C.I.
##  [1,]       1.00       1.00
##  [2,]       1.00       1.01
##  [3,]       1.00       1.00
##  [4,]       1.00       1.00
##  [5,]       1.00       1.00
##  [6,]       1.00       1.00
##  [7,]       1.00       1.00
##  [8,]       1.00       1.00
##  [9,]       1.00       1.00
## [10,]       1.00       1.00
## [11,]       1.00       1.00
## [12,]       1.00       1.00
## [13,]       1.00       1.00
## [14,]       1.00       1.00
## [15,]       1.00       1.00
## [16,]       1.00       1.00
## [17,]       1.00       1.00
## [18,]       1.00       1.00
## [19,]       1.00       1.00
## [20,]       1.00       1.00
## [21,]       1.00       1.00
## [22,]       1.00       1.00
## [23,]       1.00       1.01
## [24,]       1.03       1.10
## [25,]       1.01       1.05
## [26,]       1.01       1.01
## [27,]       1.00       1.01
## [28,]       1.12       1.36
## [29,]       1.25       1.79
## [30,]       1.01       1.04
## [31,]       1.01       1.02
## [32,]       1.00       1.00
##
## Multivariate psrf
##
## 1.17
```

Examining sigma2, nu2, rho

```
sigma2_samples <- mcmc.list(chain1$samples$sigma2, chain2$samples$sigma2,
                            chain3$samples$sigma2)
```

```
nu2_samples <- mcmc.list(chain1$samples$nu2, chain2$samples$nu2,
                         chain3$samples$nu2)
```

```
plot(sigma2_samples)

plot(nu2_samples)
```

```
gelman.diag(sigma2_samples)
```

```
## Potential scale reduction factors:
##
##      Point est. Upper C.I.
## [1,]          1          1
```

```
gelman.diag(nu2_samples)
```

```
## Potential scale reduction factors:
##
##      Point est. Upper C.I.
## [1,]          1          1
```

Examining a sample of the 3108 phi parameters

```
phi_samples <- mcmc.list(chain1$samples$phi, chain2$samples$phi, chain3$samples$phi)
```

```
set.seed(1157, kind = "Mersenne-Twister", normal.kind = "Inversion", sample.kind = "Rejection")

phi_subset_idx <- sample(1:ncol(phi_samples[[1]]), size = 10)

phi_samples_subset <- phi_samples[, phi_subset_idx]
```

```
plot(phi_samples_subset)
```

```
gelman.diag(phi_samples_subset)
```

```
## Potential scale reduction factors:
##
##       Point est. Upper C.I.
##  [1,]       1.00       1.01
##  [2,]       1.03       1.10
##  [3,]       1.01       1.03
##  [4,]       1.01       1.05
##  [5,]       1.00       1.00
##  [6,]       1.00       1.01
##  [7,]       1.00       1.01
##  [8,]       1.00       1.01
##  [9,]       1.00       1.00
## [10,]       1.03       1.11
##
## Multivariate psrf
##
## 1.06
```

Inference

```
beta_samples_matrix <- rbind(chain1$samples$beta, chain2$samples$beta, chain3$samples$beta)

colnames(beta_samples_matrix) <- c("Intercept", names(fhs_model_df[, 14:(ncol(fhs_model_df) - 4)]))
```

```
(beta_inference <- round(t(apply(beta_samples_matrix, 2, quantile, c(0.5, 0.025, 0.975))),5))
```

```
##                           50%     2.5%    97.5%
## Intercept            9.89830  9.89675  9.89984
## flood_risk_pc1      -0.00034 -0.00714  0.00633
## flood_risk_pc2       0.00638 -0.00123  0.01394
## flood_risk_pc3       0.00584  0.00031  0.01127
## flood_risk_pc4       0.00038 -0.00560  0.00643
## EP_POV               0.28297  0.27628  0.28959
## EP_UNEMP             0.07059  0.06637  0.07484
## EP_PCI              -0.06377 -0.07035 -0.05718
## EP_NOHSDP            0.07442  0.06601  0.08284
## EP_AGE65             0.09787  0.09243  0.10329
## EP_AGE17             0.00997  0.00426  0.01569
## EP_DISABL           -0.06099 -0.06635 -0.05562
## EP_SNGPNT            0.03876  0.03390  0.04365
## EP_MINRTY            0.32470  0.31601  0.33345
## EP_LIMENG           -0.25482 -0.26242 -0.24717
## EP_MUNIT            -0.02239 -0.02725 -0.01755
## EP_MOBILE           -0.01565 -0.02012 -0.01125
## EP_CROWD            -0.02069 -0.02630 -0.01509
## EP_NOVEH             0.15010  0.14280  0.15738
## EP_GROUPQ           -0.04574 -0.04949 -0.04202
## EP_UNINSUR          -0.02832 -0.03392 -0.02269
## co                  -0.06060 -0.08177 -0.03903
## no2                 -0.16904 -0.19713 -0.13989
## o3                   0.02082 -0.03077  0.07297
## pm10                -0.16112 -0.18080 -0.14117
## pm25                 0.29473  0.26537  0.32387
## so2                 -0.02484 -0.04437 -0.00474
## summer_tmmx          0.02626 -0.00894  0.06162
## winter_tmmx         -0.05733 -0.15320  0.03227
## summer_rmax         -0.00022 -0.04329  0.04341
## winter_rmax         -0.05571 -0.08722 -0.02189
## Data_Value_CSMOKING  0.85440  0.84330  0.86558
```

List of significant beta coefficients:

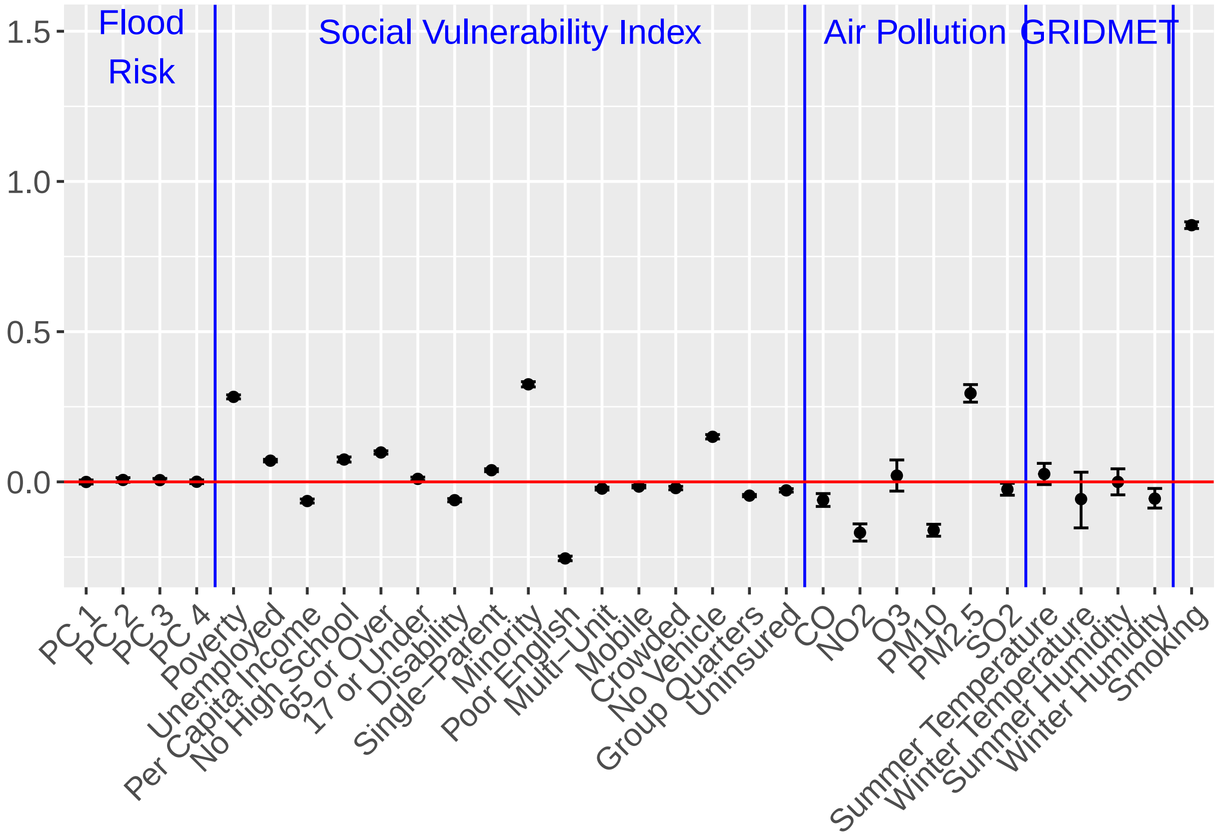
```
colnames(beta_samples_matrix)[sign(beta_inference[, 2]) == sign(beta_inference[, 3])]
```

```
##  [1] "Intercept"           "flood_risk_pc3"      "EP_POV"
##  [4] "EP_UNEMP"            "EP_PCI"              "EP_NOHSDP"
##  [7] "EP_AGE65"            "EP_AGE17"            "EP_DISABL"
## [10] "EP_SNGPNT"           "EP_MINRTY"           "EP_LIMENG"
## [13] "EP_MUNIT"            "EP_MOBILE"           "EP_CROWD"
## [16] "EP_NOVEH"            "EP_GROUPQ"           "EP_UNINSUR"
## [19] "co"                  "no2"                 "pm10"
## [22] "pm25"                "so2"                 "winter_rmax"
## [25] "Data_Value_CSMOKING"
```

Credible Interval plots for the coefficients, in ggplot

```r
# first, process the beta_inference matrix in a form ggplot can understand

beta_inference_df <- as.data.frame(beta_inference)

beta_inference_df <- mutate(beta_inference_df, var_name = row.names(beta_inference_df))

beta_inference_df <- rename(beta_inference_df,
                            post_median = `50%`,
                            post_2.5 = `2.5%`,
                            post_97.5 = `97.5%`)

beta_inference_df$var_name <- factor(beta_inference_df$var_name, levels = beta_inference_df$var_name)
```

```r
ggplot(beta_inference_df[-1, ], aes(x = var_name, y = post_median)) +
  geom_point() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1), axis.title.x = element_blank(), axi
        axis.text=element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 10)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_vline(xintercept = c(4.5, 20.5, 26.5, 30.5), col = "blue") +
  geom_hline(yintercept = 0, col = "red") +
  annotate(geom = "text", x = 2.5, y = 1.45, label = "Flood\nRisk",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 12.5, y = 1.5, label = "Social Vulnerability Index",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 23.5, y = 1.5, label = "Air Pollution",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 28.5, y = 1.5, label = "GRIDMET",
           col = "blue", size = 4.5) +
  scale_x_discrete(labels = c("PC 1", "PC 2", "PC 3", "PC 4",
                              "Poverty", "Unemployed", "Per Capita Income", "No High School",
                              "65 or Over", "17 or Under", "Disability",
                              "Single-Parent", "Minority", "Poor English",
                              "Multi-Unit", "Mobile", "Crowded",
                              "No Vehicle", "Group Quarters", "Uninsured",
                              "CO", "NO2", "O3", "PM10", "PM2.5", "SO2",
                              "Summer Temperature", "Winter Temperature", "Summer Humidity", "Winter Hu
                              "Smoking"))
```



CAR model results, Poor Mental Health

Inference is based on 3 markov chains, each of which has been run for 110000 samples, the first 10000 of which has been removed for burn-in. The remaining 100000 samples are thinned by 2, resulting in 150000 samples for inference across the 3 Markov chains.

```
load(here("modeling_files/all_census_tract_MHLTH.RData"))
```

Model Diagnostics

Beta samples

```
beta_samples <- mcmc.list(chain1$samples$beta, chain2$samples$beta,
                          chain3$samples$beta)
```

```
effectiveSize(beta_samples)
```

```
##       var1       var2       var3       var4       var5       var6       var7 
## 69933.5337 10357.1215  9827.3033 11998.5431  9770.6720 24646.5451 47945.4763 
##       var8       var9      var10      var11      var12      var13      var14 
## 21744.6033 28898.6024 26389.5570 33760.9092 36755.5802 48488.7028 16875.7884 
##      var15      var16      var17      var18      var19      var20      var21 
## 25027.4331 32115.5052 22703.8680 36457.1170 22238.2707 46882.2417 30415.5745 
##      var22      var23      var24      var25      var26      var27      var28
##  6745.1788  3795.2968   221.8825  2486.4742  1480.6492  1952.5681   671.4877
##      var29      var30      var31      var32
##   131.6527   543.4624   821.2714 14157.6810
```

```
plot(beta_samples)
```

```
gelman.diag(beta_samples)
```

```
## Potential scale reduction factors:
##
##       Point est. Upper C.I.
##  [1,]       1.00       1.00
##  [2,]       1.00       1.01
##  [3,]       1.00       1.00
##  [4,]       1.00       1.00
##  [5,]       1.00       1.00
##  [6,]       1.00       1.00
##  [7,]       1.00       1.00
##  [8,]       1.00       1.00
##  [9,]       1.00       1.00
## [10,]       1.00       1.00
## [11,]       1.00       1.00
## [12,]       1.00       1.00
## [13,]       1.00       1.00
## [14,]       1.00       1.00
## [15,]       1.00       1.00
## [16,]       1.00       1.00
## [17,]       1.00       1.00
## [18,]       1.00       1.00
## [19,]       1.00       1.00
## [20,]       1.00       1.00
## [21,]       1.00       1.00
## [22,]       1.00       1.00
## [23,]       1.00       1.00
## [24,]       1.02       1.08
## [25,]       1.01       1.04
## [26,]       1.00       1.01
## [27,]       1.00       1.01
## [28,]       1.09       1.28
## [29,]       1.19       1.61
## [30,]       1.01       1.03
## [31,]       1.01       1.02
## [32,]       1.00       1.00
##
## Multivariate psrf
##
## 1.12
```

Examining sigma2, nu2, rho

```
sigma2_samples <- mcmc.list(chain1$samples$sigma2, chain2$samples$sigma2,
                            chain3$samples$sigma2)
```

```
nu2_samples <- mcmc.list(chain1$samples$nu2, chain2$samples$nu2,
                         chain3$samples$nu2)
```

```
plot(sigma2_samples)

plot(nu2_samples)
```

```
gelman.diag(sigma2_samples)
```

```
## Potential scale reduction factors:
##
##      Point est. Upper C.I.
## [1,]          1          1
```

```
gelman.diag(nu2_samples)
```

```
## Potential scale reduction factors:
##
##      Point est. Upper C.I.
## [1,]          1          1
```

Examining a sample of the 3108 phi parameters

```
phi_samples <- mcmc.list(chain1$samples$phi, chain2$samples$phi, chain3$samples$phi)
```

```
set.seed(1157, kind = "Mersenne-Twister", normal.kind = "Inversion", sample.kind = "Rejection")

phi_subset_idx <- sample(1:ncol(phi_samples[[1]]), size = 10)

phi_samples_subset <- phi_samples[, phi_subset_idx]
```

```
plot(phi_samples_subset)
```

```
gelman.diag(phi_samples_subset)
```

```
## Potential scale reduction factors:
##
##       Point est. Upper C.I.
##  [1,]       1.00       1.01
##  [2,]       1.02       1.07
##  [3,]       1.00       1.02
##  [4,]       1.01       1.03
##  [5,]       1.00       1.00
##  [6,]       1.00       1.01
##  [7,]       1.00       1.01
##  [8,]       1.00       1.01
##  [9,]       1.00       1.00
## [10,]       1.02       1.08
##
## Multivariate psrf
##
## 1.04
```

Inference

```
beta_samples_matrix <- rbind(chain1$samples$beta, chain2$samples$beta, chain3$samples$beta)

colnames(beta_samples_matrix) <- c("Intercept", names(fhs_model_df[, 14:(ncol(fhs_model_df) - 4)]))
```

```
(beta_inference <- round(t(apply(beta_samples_matrix, 2, quantile, c(0.5, 0.025, 0.975))),5))
```

```
##                          50%     2.5%    97.5%
## Intercept           14.26594 14.26303 14.26885
## flood_risk_pc1       0.01224  0.00128  0.02302
## flood_risk_pc2       0.00331 -0.00897  0.01549
## flood_risk_pc3      -0.00111 -0.01003  0.00770
## flood_risk_pc4       0.00014 -0.00950  0.00989
## EP_POV               0.81870  0.80752  0.82977
## EP_UNEMP             0.07114  0.06417  0.07815
## EP_PCI              -0.32941 -0.34016 -0.31862
## EP_NOHSDP            0.25417  0.24035  0.26801
## EP_AGE65            -0.33592 -0.34490 -0.32699
## EP_AGE17            -0.10574 -0.11514 -0.09632
## EP_DISABL           -0.17502 -0.18385 -0.16616
## EP_SNGPNT            0.07617  0.06813  0.08422
## EP_MINRTY           -0.17898 -0.19315 -0.16476
## EP_LIMENG           -0.03129 -0.04371 -0.01877
## EP_MUNIT             0.07996  0.07198  0.08790
## EP_MOBILE           -0.01466 -0.02202 -0.00743
## EP_CROWD             0.05835  0.04912  0.06756
## EP_NOVEH             0.07672  0.06468  0.08867
## EP_GROUPQ            0.16874  0.16258  0.17491
## EP_UNINSUR          -0.02160 -0.03083 -0.01234
## co                   0.08515  0.05080  0.12011
## no2                 -0.05365 -0.09901 -0.00693
## o3                   0.03459 -0.04603  0.11616
## pm10                -0.13757 -0.16903 -0.10564
## pm25                 0.30362  0.25670  0.35029
## so2                  0.00554 -0.02594  0.03755
## summer_tmmx          0.07171  0.01604  0.12697
## winter_tmmx          0.05459 -0.09524  0.19475
## summer_rmax          0.04154 -0.02786  0.11114
## winter_rmax         -0.02884 -0.07901  0.02469
## Data_Value_CSMOKING  2.14225  2.12396  2.16071
```

List of significant beta coefficients:

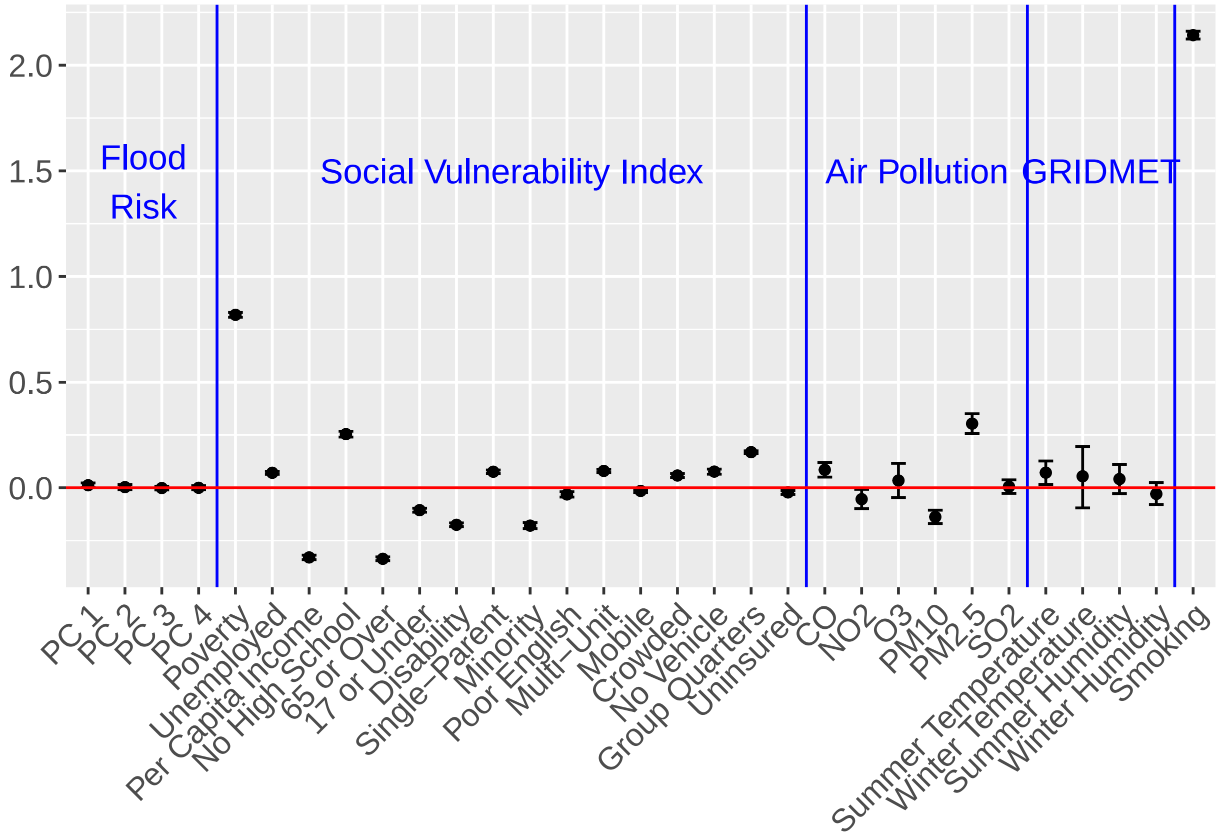
```
colnames(beta_samples_matrix)[sign(beta_inference[, 2]) == sign(beta_inference[, 3])]
```

```
##  [1] "Intercept"           "flood_risk_pc1"      "EP_POV"
##  [4] "EP_UNEMP"            "EP_PCI"              "EP_NOHSDP"
##  [7] "EP_AGE65"            "EP_AGE17"            "EP_DISABL"
## [10] "EP_SNGPNT"           "EP_MINRTY"           "EP_LIMENG"
## [13] "EP_MUNIT"            "EP_MOBILE"           "EP_CROWD"
## [16] "EP_NOVEH"            "EP_GROUPQ"           "EP_UNINSUR"
## [19] "co"                  "no2"                 "pm10"
## [22] "pm25"                "summer_tmmx"         "Data_Value_CSMOKING"
```

Credible Interval plots for the coefficients, in ggplot

```r
# first, process the beta_inference matrix in a form ggplot can understand

beta_inference_df <- as.data.frame(beta_inference)

beta_inference_df <- mutate(beta_inference_df, var_name = row.names(beta_inference_df))

beta_inference_df <- rename(beta_inference_df,
                            post_median = `50%`,
                            post_2.5 = `2.5%`,
                            post_97.5 = `97.5%`)

beta_inference_df$var_name <- factor(beta_inference_df$var_name, levels = beta_inference_df$var_name)
```

```r
ggplot(beta_inference_df[-1, ], aes(x = var_name, y = post_median)) +
  geom_point() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1), axis.title.x = element_blank(), axi
        axis.text=element_text(size=12),
        plot.margin = margin(5.5, 5.5, 5.5, 10)) +
  geom_errorbar(aes(ymin = post_2.5, ymax = post_97.5, width = 0.4)) +
  geom_vline(xintercept = c(4.5, 20.5, 26.5, 30.5), col = "blue") +
  geom_hline(yintercept = 0, col = "red") +
  annotate(geom = "text", x = 2.5, y = 1.45, label = "Flood\nRisk",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 12.5, y = 1.5, label = "Social Vulnerability Index",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 23.5, y = 1.5, label = "Air Pollution",
           col = "blue", size = 4.5) +
  annotate(geom = "text", x = 28.5, y = 1.5, label = "GRIDMET",
           col = "blue", size = 4.5) +
  scale_x_discrete(labels = c("PC 1", "PC 2", "PC 3", "PC 4",
                              "Poverty", "Unemployed", "Per Capita Income", "No High School",
                              "65 or Over", "17 or Under", "Disability",
                              "Single-Parent", "Minority", "Poor English",
                              "Multi-Unit", "Mobile", "Crowded",
                              "No Vehicle", "Group Quarters", "Uninsured",
                              "CO", "NO2", "O3", "PM10", "PM2.5", "SO2",
                              "Summer Temperature", "Winter Temperature", "Summer Humidity", "Winter Hu
                              "Smoking"))
```


2.0
1.5
1.0
0.5
0.0
Flood Risk
Social Vulnerability Index
Air Pollution
GRIDMET
PC 1
PC 2
PC 3
PC 4
Poverty
Unemployed
Per Capita Income
No High School
65 or Over
17 or Under
Disability
Single−Parent
Minority
Poor English
Multi−Unit
Mobile
Crowded
No Vehicle
Group Quarters
Uninsured
CO
NO2
O3
PM10
PM2.5
SO2
Summer Temperature
Winter Temperature
Summer Humidity
Winter Humidity
Smoking