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

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 (), resulting in () 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)
```

```
saveRDS(beta_samples, file = here("modeling_files/all_census_tract_intrinsic_beta_samples.rds"))
```

```
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) - 1)]))
```

```
(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
```

Net Effect interpretation: what if each variable in a group (flood risk variables, SVIs, air pollution variables) increased by 1 standard deviation? What is the resulting change in the CHD prevalence?

List of significant beta coefficients:

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

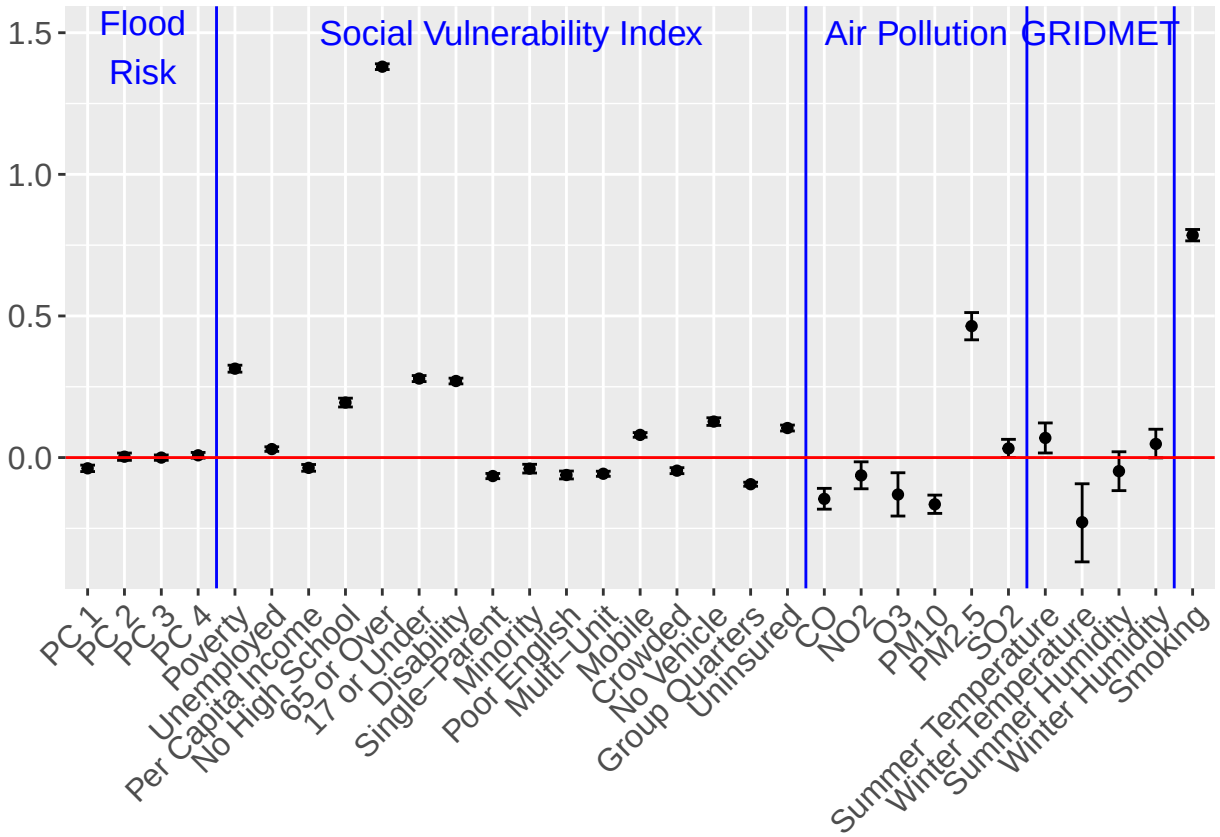
```
##  [1] "Intercept"           "flood_risk_pc1"      "EP_POV"
##  [4] "EP_UNEMP"            "EP_PCI"              "EP_NOHSDP"
##  [7] "EP_AGE65"            "EP_AGE17"            "EP_DISABL"
```

```
## [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(), axis
        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"))
```


Flood Risk
Social Vulnerability Index
Air Pollution
GRIDMET
1.5
1.0
0.5
0.0
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